When users create industry : the case of Web-based applications

Paris Chrysos

École doctorale n°396 : Économie, Organisations & Société

Doctorat européen ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

l'École nationale supérieure des mines de Paris

Spécialité "Sciences de Gestion"

présentée et soutenue publiquement par

Paris CHRYSOS

le 17 décembre, 2013

Quand les utilisateurs créent l'industrie
– le cas des applications Web–

~ ~ ~

When users create industries
– the case of Web-based applications –

Directeur de thèse: **Armand HATCHUEL**

Co-encadrement de la thèse: **Philippe LEFEBVRE**

Jury

M. Christophe BENAVENT,	Professeur, Université Paris Ouest Nanterre La Défense	Président du jury
M. Konstantinos CHATZIS,	HDR, CR1 (Chargé de recherche 1) à l'IFSTTAR, mis à la disposition de l'Ecole des Ponts ParisTech au LATTS	Examinateur
M. Gilles GAREL,	Professeur, Conservatoire National des Arts et Métiers (CNAM), Chaire de gestion de l'innovation	Rapporteur
M. Stefan HAEFLIGER,	Reader, Cass Business School, City University London Privatdozent (équivalent HDR), ETH Zurich	Rapporteur
M. Armand HATCHUEL,	Professeur, Centre de Gestion Scientifique, Mines ParisTech	Examinateur
M. Philippe LEFEBVRE,	Maître-assistant, Centre de Gestion Scientifique, Mines ParisTech	Examinateur

T H È S E

MINES ParisTech
Centre de Gestion Scientifique
60 Boulevard Saint Michel, 75 272 Paris Cedex 06, France

à Vaggo,
à Yota

Remerciements

On peut dire que le travail de thèse est un grand voyage, où l'on apprend souvent des choses sur soi-même, avant d'en apprendre sur son objet de recherche. Heureusement, que j'ai trouvé près de moi, tout le long du chemin et lors des pauses, des gens remarquables pour me conseiller, m'inspirer, pour discuter avec, échanger, partager. La beauté de ce voyage serait, sans doute, moins séduisante sans eux.

Tout d'abord, je souhaiterais exprimer ma plus profonde gratitude à l'égard d'Armand Hatchuel et Philippe Lefebvre. Les enseignements et les théories d'Armand Hatchuel furent pour moi une boussole tout au long du voyage et ont profondément influencé ma pensé. Philippe Lefebvre m'a appris à cartographier le territoire : par ses conseils et nos discussions toujours stimulantes il m'a accompagné avec patience dans l'élaboration de cette étude et m'a donné les clefs analytiques et synthétiques pour la construire. Sa contribution à cette thèse fut décisive.

J'ai pu bénéficier également de la considération de l'équipe du CGS et de l'équipe de la Chaire Théorie et Méthodes de la Conception Innovante de Mines ParisTech, que je tiens à remercier chaleureusement dans leur ensemble. Merci à Blanche Segrestin, qui a discrètement suivi tout le parcours de mon travail, à Akin Kazakçi, dont les remarques brillantes m'ont beaucoup aidé, à Pascal Le Masson et à Michel Nakhla, pour leurs conseils constructifs à l'occasion de l'exposition de mes travaux aux élèves de l'École, à Mathias Béjean, Sophie Hooge et Douglas Robinson, pour avoir partagé ma curiosité, et à Anne-Françoise Schmid, pour nos conversations toujours passionnantes. Il ne faudrait pas manquer de remercier également Stéphanie Brunet, Céline Bourdon et Martine Jouanon pour leur aide administrative tout au long de l'élaboration de ma thèse.

À diverses occasions et à différents niveaux d'avancement de mon étude, j'ai eu l'opportunité de profiter des commentaires et des réflexions des universitaires que je respecte. Je tiens à remercier Georg von Krogh, Stefan Haeflinger, Rémi Maniak, Barry Bozeman, et les organisateurs et participants des conférences et séminaires auxquels j'ai présenté des parties de mon travail, notamment le *MT Seminar (ETH)*, le *Seminaire du Jeudi (Télécom ParisTech)*, l'*Open and User Innovation Workshop (MIT)*, et les séminaires doctoraux du CGS.

Je souhaiterais également exprimer ma reconnaissance aux chercheurs qui m'ont aidé lors de mes tout premiers pas dans la recherche. Je tiens à remercier Yannis Maistros et Yannis Milios de l'École Polytechnique d'Athènes, qui ont suivi mon mémoire de diplôme, et m'ont encouragé de poursuivre mon chemin dans la recherche. Merci à Patrice Flichy, qui a dirigé mon mémoire de Master, Konstantinos Chatzis et les autres chercheurs du LATTS avec qui j'ai pu échanger lors de la réalisation de mon Master.

De même, je tiens à remercier Christophe Aguiton, Dominique Cardon, Christophe Prieur et Benoît Sibaud, qui m'ont permis de participer très tôt à la discussion chez Orange Labs sur le « Web 2.0 ». Merci aussi à tous ceux qui ont partagé leur réflexions avec moi, notamment Pamela Fox (Google), Pascal Finette (Mozilla) et à ceux qui, de plus, m'ont accompagné dans mes débuts dans l'enseignement, spécialement Domique Laousse (SNCF), Lomig Unger (Renault) et Nicolas Barcet (eNovance).

Je dois beaucoup à Éric Simon et à Andrès Atenza, ainsi qu'à tous les collègues de l'Institut Supérieur de Commerce, pour leur compréhension, leur support et leur confiance, aussi bien en ce qui concerne la rédaction de ma thèse que la réalisation de mes nouveaux devoirs de professeur au sein de l'école, durant ces deux dernières années. Je souhaite également remercier Fred Adam (UC Cork) et les autres collègues avec qui j'ai eu la chance de collaborer dans le cadre du MBS-IEB.

Enfin, merci à mes amis, Clément Mouhot, Grégory Bekhtari, Fanny Gallot, Dimitris Vardoulakis, Guillaume Communal, Tassos Grigorakis, Katerina Livydikou, Claire Quilichini, Laurent Jesover, Fivos Maniatakos, ßStavros Stromatas, Giorgos Velegrakis, Yannis Margaris et Petros Papadakis, pour avoir partagé mes moments de joie et mes soucis. Je tiens enfin à remercier mes parents, mes frères et mes tantes pour leur soutien.

Contents

List of Tables

List of Figures

Introduction

Introduction générale

Le problème posé dans cette étude

La présente étude explore la manière dont se développent les applications Web, en soulevant la question de la nature de leur développement, son *modus operandi*, ainsi que la question de la façon dont les entreprises peuvent l'exploiter.

L'importance du Web en général a été explorée par des études dans plusieurs disciplines scientifiques. La Gestion, comme les Sciences Sociales, ont étudié l'usage des services Web ainsi que les implications de cet usage, dans une pléthore de contextes (dans les entreprises et les organisations, mais aussi comme dispositifs marchants et dispositifs de communication). Cependant, il n'y a pas d'étude dans ces disciplines fournissant un cadre d'analyse et de déploiement des biens de ce type, au-delà de la discussion des aspects technique dans le domaine de l'ingénierie.

Les questions mentionnées ci-dessus seront étudiées en utilisant une stratégie de *« phenomenon - based research »* (von Krogh et al., 2012), qui vise à la *distinction* et l'*exploration* de la nature d'un phénomène, avant la proposition d'un *« research design»*, une conception de recherche particulière, susceptible de donner la possibilité au chercheur de mieux creuser les spécificités du phénomène en question. Ce travail se déroulera dans trois parties, en décomposant les questions de recherche et en menant une enquête de leurs différents aspects.

La nature du *modus operandi* du développement des applications Web sera alors recherchée dans une perspective managériale, en explorant les questions de *« qui »* le fait, « comment » cela se fait et *« pourquoi »* cela se fait, par analogie avec la distinction introduite par Raasch et von Hippel (2012) entre les approches d'innovation d'usager et d'industriel. L'exploration en parallèle des questions *qui*, *comment* et *pourquoi* n'est pas nouvelle pour autant dans la gestion. Comme Hatchuel et Weil (1992) l'ont montré, les entreprises elles-mêmes, lorsqu'elles font face à une épreuve de rationalisation, sont appelées à concevoir et à mettre en place un nouveau *schéma organisationnel,* une *philosophie managériale* et un *substrat technique*. L'ensemble de ces réponses constitue alors une nouvelle rationalisation. Néanmoins, en ce qui concerne notre propre recherche, nous ne nous attendons pas à identifier un terrain rationalisé. Cependant, cette recherche identifiera les *acteurs,* les *raisons* et les *moyens* spécifiques du développement des applications Web, tels qu'ils se dessinent en lien avec les modèles d'innovation usager et industriel (von Hippel et von Krogh, 2003; 2006, Raasch et von Hippel 2012). Par conséquent, la première partie de la présente étude exposera les grandes lignes du *modus operandi* rencontré sur le terrain, apparaissant comme original.

Une fois le problème de *« qu'est le développement d'applications Web »* exploré, en conduisant à une proposition d'un *modus operandi* spécifique, composé par des *acteurs* surprenants, ainsi que des *raisons et moyens d'action,* l'enjeu de l'étape suivante de cette recherche sera l'identification des conditions d'apparition d'un tel phénomène. À ce propos, nous utiliserons les leçons de la première partie comme une règle de lecture de l'histoire du développement d'autres configurations industrielles, dans la seconde partie. Les industries que nous étudierons ainsi seront celles de l'*ordinateur entreprise,* de l'*ordinateur personnel* et de la *radio,* afin de répondre à la question de savoir si le *modus operandi* initialement identifié est, quant à lui, une spécificité du Web ou, en revanche, s'il est rencontré également dans des secteurs industriels proches.

Enfin, la troisième partie reviendra dans le champ du développement des applications Web afin d'explorer les conditions d'exploitation du *modus* identifié précédemment. À ce propos, nous discuterons la littérature sur la gestion des dispositifs relationnels informels, les communautés ou les réseaux, qui ont un rôle important dans l'innovation ouverte et l'innovation par les usagers (von Krogh et al., 2003; Simard et West, 2006; von Hippel, 2007; West et Lakhani, 2008; Benkeltoum, 2009). Malgré des recherches riches en ce domaine tout au long de cette dernière décennie, la question de l'exploitation de ce genre de dispositifs par les entreprise reste encore ouverte. Cependant, notre exploration se déploiera sur la base de trois aspects de la question, en rapport avec les acteurs du développement des applications Web. Le premier aspect exploré, portera sur les conditions de possibilité d'émergence de tels dispositifs, en premier lieu. Puis, le deuxième aspect portera sur l'exploitation de tels dispositifs par une entreprise à des fins *d'exploration* du potentiel de son service. Enfin, le troisième aspect abordera la question de l'exploitation de tels dispositifs par une entreprise à des fins *d'exploitation* du potentiel de son service. Ces enquêtes seront soutenues par un cadre d'analyse qui situera les notions de *conversation* et d'*action collective* parmi celles d'une *communauté* et d'un *réseau.*

Méthodologie

L'étude actuelle fera usage d'une stratégie de *« phenomenon-based research »* (von Krogh et al., 2012) afin d'explorer le développement des applications Web. Von Krogh et al., en exposant les grandes lignes de cette approche selon les différents objectifs de recherche, décrivent les étapes suivantes :

1. *Étape de distinction*, durant laquelle la recherche vise à décrire les spécificités sous-jacentes du phénomène en référence à un corpus des savoirs existant, à décrire le contexte en termes généraux et culturels, et à identifier des concepts pertinents pour une étude plus approfondie du phénomène. À ce propos, les auteurs suggèrent l'usage des méthodes ethnographiques ou des narrations.

2. *Étape d'exploration*, durant laquelle la recherche vise à intensifier la collecte des données, à la fois en utilisant les concepts de référence et en allant au-delà de ces concepts, générant des concepts plus robustes pouvant servir de filtre pour une collecte de données encore plus approfondie. À ce propos, les auteurs suggèrent l'usage des méthodes statistiques, l'étude d'archives ou les sondages.

3. *Étape de conception*, durant laquelle la recherche vise à essayer des designs de recherche alternatifs. À ce propos, l'utilisation des concepts de recherche opportunistes est suggérée, afin d'approfondir ou mettre en question les concepts initiaux, afin de permettre la prise en compte de la dynamique du phénomène.

De plus, les chercheurs continuent leur exposition par la proposition des étapes suivantes, celles de la *théorisation* et de la *synthèse,* correspondant à la génération inductive d'une nouvelle théorie dans le premier cas, et l'établissement d'une vue d'ensemble du phénomène dans le deuxième. Au cours de notre étude, nous n'utiliserons que les trois premières étapes de la stratégie de recherche mentionnées ci-dessus. La méthodologie utilisée sera conforme à des exigences et des contraintes des questions de recherche posées, à la fois en ce qui concerne les dispositifs particuliers étudiés et la méthode d'exploration utilisée à chaque fois.

La Partie I fera usage d'une variété de méthodes permettant l'accès et l'étude des terrains différents, ayant comme objectif la distinction et l'exploration des acteurs, des raisons et des moyens d'action du développement d'applications Web. Tout d'abord, par le biais d'entretiens d'experts d'entreprises du secteur, nous allons procéder à une restitution du discours des fournisseurs de services, ce qui conduira à des premières indications sur l'existence d'un *modus operandi* au delà de celui de l'entreprise, ce dernier définissant le champ d'expertise des personnes interviewées. Par la suite, et en suivant ces indications, nous utiliserons une méthode de « participation observante » afin de rejoindre et observer l'action de ceux qui semblent être des nouveaux acteurs, même si leur identité reste encore mal définie. Notre but sera d'enquêter sur la raison et les moyens d'action de ces acteurs. Ensuite, nous examinerons des *« cookbooks »,* les *« livres de cuisine »* que ces acteurs étudient afin d'être en position d'utiliser les moyens en question pour développeur leurs propres applications. Ici, notre objectif sera d'identifier et d'explorer la nature de ces acteurs, par le biais de leurs raisons d'action liées aux hypothèses (implicites ou explicites) que font les auteurs de ces livres pour leur public. Enfin, nous utiliserons l'histoire d'une application, dont la raison de développement est connue, afin de mener une exploration plus approfondie du *modus operandi* dans son ensemble. Cette méthodologie sera analysée de manière plus approfondie dans la Section 2.2.

La Partie II comparera les résultats de l'exploration de la première partie à d'autres cadres industriels, afin d'explorer si les spécificités identifiées en comparaison avec la distinction des modèles d'innovation par l'usager et par le fabriquant, sont propres au développement des applications Web ou pas. Sur la base des spécifications de notre problème de recherche, nous étudierons l'histoire de l'*ordinateur entreprise,* celle de l'*ordinateur personnel* et celle de la *radio.* À ce propos, nous utiliserons les travaux d'historiens dans ces secteurs, et nous allons nous permettre de consulter d'autres sources originales dans des cas où certains aspects sont peu explorés par ces auteurs. Cette méthodologie sera discutée de manière plus approfondie dans la Section 9.3.

Enfin, la Partie III utilisera des méthodes d'accès au terrain propres à des dispositifs informels et éphémères. Tout d'abord, nous utiliserons la méthode de *« participation observante »* afin de rejoindre les *« Barcamps »,* des dispositifs conversationnels, ainsi qu'une *« analyse de monuments d'interaction »,* faisant usage de traces d'interaction liée aux dispositifs et qui sont disponibles en ligne, afin de compléter notre observation. Cette observation portera sur 16 *Barcamps* qui ont pris lieu à Paris lors d'une période de trois ans. Ensuite, nous utiliserons les mêmes méthodes pour étudier le *« Hackathon »,* un dispositif d'action éphémère et exploratoire, en étudiant le cas de celui qui a pris lieu aux locaux de *Google,* en Californie. Le dernier dispositif étudié sera celui des forums de soutien des développeurs, que nous étudierons en utilisant une analyse de monuments d'interaction, ayant comme objectif d'explorer la manière dont les fournisseurs de service soutiennent l'exploitation du potentiel de leur service par ces acteurs. Cette méthodologie sera plus approfondie dans la Section 14.3.

Contribution

Cette étude aboutit à la conclusion d'une configuration d'un *modus operandi* spécifique et distinct à la fois des modèles d'innovation usager et industriel (von Hippel et von Krogh, 2003; 2006, Raasch et von Hippel 2012), qui sera identifié et exploré dans la première partie. Curieusement, comme il sera discuté dans la seconde partie, ce *modus operandi* étrange n'est pas propre au développement d'applications Web : des acteurs similaires, ainsi que des moyens et des raisons d'action similaires, sont historiquement apparus dans chacun des les trois cas industriels étudiés, même si l'utilité de ce *modus* est liée à des phases assez spécifiques du développement industriel. Enfin, trois dispositifs d'interaction informelle seront identifiés, pouvant être utiles à des entreprises exploitant l'action des tiers développeurs.

Partie I : Un *modus operandi* étrange

La première partie identifie un étrange *modus operandi,* qui se positionne entre le modèle d'innovation par l'usager et celui par l'industriel Raasch and von Hippel (2012), à la fois en ce qui regarde ses acteurs et ses moyens et raisons d'action. Plus précisément, trois figures d'acteur seront identifiées, selon leur raisons d'action :

1. L'*Usager-Développeur (UD),* utilisant ses compétences de développement afin d'innover pour son propre usage, est semblable à un *« lead user » ;* ce dernier est décrit dans la littérature sur l'innovation des usagers (von Hippel et Katz, 2002; von Hippel, 2005; Franke et al., 2006), sauf qu'il dispose à la fois de la *« sticky information »* relative au contexte d'usage, attribuée par von Hippel (1990) à des usagers, et de la *« sticky information »* relative aux technologies, attribuée par von Hippel à des industriels.

2. L' *Usager-Développeur-Entrepreneur (UDE)*, qui, bien qu'il soit caractérisé par une démarche d'innovation similaire à celle de l'UD, ne révèle pas de façon libre sa création, comme le modèle *« privé-collective »* (von Hippel et von Krogh, 2003; 2006) le suggérait, mais décide de poursuivre sa commercialisation.

3. Le *Développeur-Entrepreneur (DE),* qui, contrairement aux deux figures précédentes, n'innove pas pour son propre usage, mais le fait pour les autres, tant en intégrant une démarche commerciale dès le départ de son processus de conception.

Toutes les trois figures utilisent pour autant les mêmes moyens, qui pourraient être décrits comme une *« palette à innover »,* et qui proviennent à la fois du modèle d'innovation de l'usager et de celui de l'industriel. D'un côté, ils utilisent un logiciel libre, qui fut l'objet de nombreuses recherches en Gestion durant la dernière décennie. De l'autre, ils utilisent des *« Interfaces de Programmation d'Applications » (Application Programming Interfaces, APIs)*, qui sont des dispositifs fournis par les entreprises afin que les développeurs puissent créer des application en utilisant leur technologie.

Partie II : les conditions d'apparition du *modus operandi* et ses effets sur le développement industriel

Dans la deuxième partie, en étudiant l'originalité du *modus operandi* identifié dans le cas du développement des applications Web et en se posant la question de son apparition dans d'autres cadres industriels, nous sommes conduits à une exploration plus approfondie de ce *modus* dans son ensemble, en rapport avec les différentes phases industrielles.

Le modèle résultant de l'identification de ces phases industriels, où les UDEs jouent un rôle important, est comparé *a posteriori* aux modèles connus de développement industriel les plus proches. Le résultat de cette comparaison suggère que la transition des premières *« lead user innovations »* (von Hippel, 1978b) à leur production et diffusion industrielles par des entreprises n'est pas linéaire : il y a une transformation progressive à la fois des acteurs, des raisons et des moyens d'action lors de l'exploration du potentiel sous-jacent d'un nouvel objet. De même, la perturbation d'une industrie par une innovation (*« disruption »*) décrite par Christensen (1997), durant laquelle les agents d'une nouvelle technologie arrivent à identifier un marché correspondant, n'est pas la seule manière dont un nouveau potentiel est déployé : il peut y avoir également des situations où à la fois un potentiel commercial et un potentiel technologique sont « visibles » aux acteurs, alors même qu'une exploration plus approfondie est exigée, engendrant souvent des risques considérables et nécessitant des révisions majeures des conceptions précédentes.

Les rôles des figures d'acteur, déjà configurées dans la première partie de notre étude, sont les suivants, selon les différentes phases du développement industriel identifiées :

- **Matérialisation Précoce.** Des usagers-développeurs (UDs), souvent ayant des rapports intimes avec des Universités, utilisent une nouvelle théorie afin de créer les premiers objets illustrant le potentiel d'usage de la théorie. Ces matérialisations peuvent être soit des « meilleures solutions » à des anciens concepts, servant des besoins connus, soit des « nouveaux rêves », c'est-à-dire des matérialisations illustrant la possibilité d'existence d'autres sortes d'objets, des objets inédits. Au sein de ces cercles initiaux de UDs, quelques uns vont poursuivre leurs efforts dans des tentatives de commercialisation des objets en questions, et devenir alors des usagers-développeurs-entrepreneurs (UDEs), tandis que d'autres deviendront des *« adopteurs précoces »* (*« early adopters »* selon Rogers (1962)), une première clientèle.

- **Émergence du Marché.** Les UDEs arrivent à trouver leurs premiers clients parmi les cercles des premiers UDs et développeurs des premiers produits. Souvent, ce fait peut déclencher l'intérêt des entreprises actives dans des secteurs industriels proches, qui rejoindront l'épreuve. Cependant, à cette aube d'une nouvelle industrie, le nouveau potentiel est largement inconnu dans son ensemble de tous les acteurs impliqués. La compétition alors démarre, et les UDEs doivent inclure des préoccupations marchandes dans leur raisonnement de développement, au delà de leurs propres préférences personnelles, afin d'avoir une chance de survie.

- **Compétition dans le Brouillard.** Les développeurs-entrepreneurs (DEs), souvent originaires des premiers cercles d'UDs, continuent l'exploration à la fois du potentiel de marché et du potentiel technologique du nouvel objet, en compétition avec les entreprises qui sont entrées dans l'exploration. Une pléthore d'objets devient disponible à des adopteurs précoces, même si le savoir disponible, provenant des phases précédentes ainsi que des rationalisations dans des secteurs industriels proches, s'avère insuffisant pour que les UDs puissent proposer des règles de conception (*« design rules »*, Baldwin et Clark, 2000) englobantes, pouvant conduire à une segmentation de marché.

- **Rationalisation Industrielle.** La phase précédente prend fin, lorsqu'une entreprise, utilisant le savoir produit jusqu'ici par les acteurs qui ont émergé, aussi bien que son propre savoir, crée une synthèse, en rationalisant la conception, la production et la commercialisation d'une ligne de produits unifiée. Dans cette phase, les DEs ne sont pas en mesure de tenir la compétition avec les entreprises, à moins qu'ils déclenchent un nouveau cercle, en se retournant vers les UDs et en ouvrant une nouvelle trajectoire, sur la base d'un concept qui n'est pas inclut dans la rationalisation en question.

Partie III : méthodes pour l'émergence et l'usage par les entreprises des dispositifs UDE

La dernière partie de notre étude contribue à l'exploration plus approfondie du *modus operandi* identifié par l'analyse de trois dispositifs différents. À l'aide d'un cadre d'analyse incluant l'action collective (Hatchuel, 2005a) et la conversation parmi les réseaux sociaux et les communautés, cette étude suggère des méthodes d'encadrement et d'exploitation de l'activité des UDEs.

Les *Barcamps* constituent un cas exemplaire des dispositifs éphémères de conversation, utilisés pour l'exploration de la possibilité d'émergence des nouvelles communautés et de réseaux UDEs. Leur conception permet la mise en réseau ainsi que l'exploration des technologies, des marchés et des usages émergeant, notamment par la conversation. Distincts des « anciens membres » des communautés et du « noyau » des réseaux, les « réguliers » de ses dispositifs conversationnels, même s'ils ne partagent pas nécessairement un terrain commun et ne sont pas connectés *a priori,* sont néanmoins en mesure de bénéficier de ces dispositifs dans leur ensemble, en prenant en compte ces conversations dans leur propre action et en se rapprochant des réseaux et des communautés émergeant.

Le dispositif de *Hackathon* permet une exploration ciblée du potentiel d'un service ou une technologie spécifique, par le biais de la constitution de groupes éphémères qui parcourent tous les niveaux de qualité du savoir tacite (Erden et al., 2008) en trois jours, en développant des applications ou des prototypes explorant le potentiel ciblé. Bien que la conception d'un *Hackathon* ait des attributs en commun avec celui d'un *Barcamp*, les apports du cas étudié suggèrent une dimension « personnelle » en ce qui regarde à la fois le processus d'innovation et son résultat, dans ce genre de dispositif.

Enfin, les *forums de support de développeurs,* malgré leur ressemblance avec des dispositifs de résolution des problèmes, servent à des fonctions au plus avancés que la résolution des problèmes en tant que telle. En faisant usage d'un cadre d'analyse des experts et des systèmes (Hatchuel et Weil, 1992), nous sommes conduits à la proposition d'un autre type d'expert entreprise, actif dans ce terrain. Le « curateur » n'a pas nécessairement besoin de disposer d'un « savoir faire », d'un « savoir comprendre » ou d'un « savoir planifier » (Hatchuel et Weil, 1992). En revanche, ses actions visent à « prendre soin » des UDEs, une activité qui exige des compétences telles que savoir identifier une nouveauté, structurer une conversation et développer une sorte d'intimité avec les interlocuteurs. La notion d'*« empathie entre entreprise et UDEs »* pourrait s'avérer utile à une exploration plus approfondie de cette fonction, au delà de la résolution des problèmes.

Synopsis de la thèse

Le Tableau 1.1 (page 17) résume la structure de la présente étude. La Partie I aborde le problème du *modus operandi* du développement des applications Web, en explorant la pertinence de la distinction entre les modèles d'innovation par l'usager et par fabriquant (Raasch et von Hippel, 2012) pour cette configuration. Nous étudierons ce problème à l'usage d'une stratégie de recherche de phénomène, visant à distinguer ce modus, en répondant aux questions de *qui* développe, *comment* développer et *pourquoi* développer ce type d'applications. Cette exploration nous conduira à la configuration de trois figures d'acteur, les UDs, les UDEs et les DEs. Leur action est entreprise à l'aide des dispositifs à la fois de logiciel libre et des *APIs*, du fait qu'ils exploitent leurs produits issus des communautés aussi bien que des entreprises, ces derniers prescrivant un usage spécifique, en laissant la possibilité en parallèle de leur exploitation comme moyens de conception (ce sont alors des « produits ouverts » (Chrysos et al., 2010)).

Vu que ni le modèle *« d'investissement privé »* ni le *« collectif-privatif »* (von Hippel et von Krogh, 2003; 2006) ne suffisent pas à décrire le *modus operandi* identifié dans son ensemble, la question qui se pose est de savoir si ce modus constitue une particularité du terrain du Web ou non. Nous abordons ce problème en étudiant trois cadres industriels proches mais différents, en particulier ceux des industries de l'ordinateur d'entreprise, de l'ordinateur personnel et de la radio. En identifiant des ressemblances, nous poursuivons notre exploration en nous intéressant aux conditions de possibilité de ce mode d'action, ainsi qu'à ses effets sur le monde des affaires. Nous concluons en proposant que les acteurs identifiés précédemment contribuent à l'exploration d'une multitude de nouveaux concepts durant leur diffusion à des adopteurs précoces, durant les phases de développement qui précèdent une rationalisation industrielle.

Enfin, la Partie III revient sur le terrain du Web et, en s'intéressant davantage aux acteurs, aborde le problème de savoir comment les entreprises peuvent exploiter l'activité des UDEs. Afin d'étudier cette question, nous explorons trois dispositifs différents. En posant les questions des conditions de possibilité d'émergence social des UDEs, ainsi que celles de l'exploitation de leur activité par les entreprises à des fins liées à l'exploration et l'exploitation du potentiel des services de ces dernières, nous proposons trois méthodes différentes (*Barcamps, Hackathons,* et *Forum de Support de Développeurs)*, qui rendent ces démarches possibles.

Chapter 1

General Introduction

Contents

1.1 The problem addressed in this study

The current study explores the way in which Web-based applications are developed, posing the question of *"what is a Web-based application development and how can enterprises harness its benefits?"*.

The importance of the Web in general has been explored by many different disciplines. Management Science, as well as the Social Sciences, have investigated the use of Web services and their implications on a plethora of use contexts (in enterprises and organisations, as a market setting, as a communication means). However, no study is available that provides a framework of analysis and deployment of such goods, beyond the technical discussions within the engineering community.

The above mentioned questions will be studied using a phenomenon-based strategy (von Krogh et al., 2012), aiming at *distinguishing* and *exploring* the nature of phenomenon, before providing a *research design* for both its harnessing and its further study. This work will be undertaken in three parts, breaking down the questions and investigating their different aspects in parallel.

The nature of Web-based application development will thus be investigated from a managerial perspective, by initially exploring the questions of *"who"* does it, *"how"* is it done and *"why"* it is done, as compared to the synthetic distinction of Raasch and von Hippel

(2012) between user and manufacturer innovation paradigms. The parallel exploration of who, why and how questions is not new to management. In fact, as Hatchuel and Weil (1992, 1995) have shown, enterprises themselves, when facing a rationalisation challenge, are led to the design and the implementation of a new *organisational schema*, a *managerial philosophy* and a *technical substrate*. These responses as a whole constitute thus a new rationalisation. Still, I do not expect to identify a rationalised field, since - as it will become clear in the chapters to follow - my research started when third party applications were still new to the field and ends before an industrial rationalisation has been proposed. However, my research will identify the specific *actors*, *reasons* and *means* of Web-based application development, as distinguished from the user and the manufacturer models (von Hippel and von Krogh, 2003, 2006; Raasch and von Hippel, 2012). Thence, the first part of the current study will make an outline of a *modus operandi* met in the field that appears to be original.

Once the problem of "what is Web-based application development" is explored and brings us to the proposal of a specific *modus operandi* composed by peculiar *actors*, *reasons* and *means*, the next phase of my research will be to identify the conditions of appearance of such a modus. For this, I will use this problem in Part II as a reading rule of the history of other industrial settings, namely the *enterprise computer*, the *personal computer* and the *radio* industries, to answer the question of whether or not this *modus operandi* is specific to Web development or, in the contrary, is common to other, close industries.

Finally, Part III will return to the field of Web-based application development to explore the conditions that are in place to allow for the harnessing of the modus previously explored. For that, I will discuss the literature on informal settings, communities or networks, which have been claimed to be important for Open and User innovation (von Krogh et al., 2003a; Simard and West, 2006; von Hippel, 2007; West and O'Mahony, 2008). Despite intensive research on these approaches during the last decade, the question of how enterprises can harness such settings for open and user innovation still remains open. Thus, my exploration will be based on three different aspects of the question, in relation to the actors of Web-based applications development. The first aspect regards the conditions of possibility for the emergence of such settings in the first place. The second aspect regards the harnessing of such settings for enterprise service potential *exploration*. Finally, the third aspect regards their harnessing for enterprise service potential *exploitation*. This investigation will be supported by an analytical framework that places the concepts of *conversation* and *collective action* in-between the communities and networks settings.

1.2 Methodology

This research will use a phenomenon-based strategy (von Krogh et al., 2012) to explore Web-based application development. Von Krogh et al. outline the following steps, according to their research goals:

1. *Distinction*, during which research goals include encountering bracket peculiarities against an existing body of knowledge, the description of the context in broad cultural terms and the identification of relevant concepts for study, through the use of ethnography or narratives.

2. *Exploration*, during which research goals include the intensification of data gathering inside and outside the focal concepts, the generation of more solid concepts that can

serve as filter for further data gathering, through the use of statistics, archival data or surveys.

3. *Design*, during which research goals include experimentation with alternative research designs and the employment of opportunistic research designs that expand or collapse concepts to take into account the dynamics of the phenomenon (von Krogh et al., 2012, p. 290).

Furthermore, von Krogh et al. (2012) outline the steps of *theorising* and *synthesising*, for the inductive generation of a new theory and the establishment of a phenomenon overview, respectively. During my study I will use the three first steps of this strategy. The methodology that will be used will comply with the requirements of the research questions, both in what regards the specific setting of the field that will be explored and the exploration method.

Part I will use a variety of methods to access and study different fields aiming at the *distinction* and the *exploration* of the actors and the means of and reasons for of Web-based application development. Initially, through interviews with service provider managers I will reproduce the discourse of service providers leading to some early indications on the existence of another *modus operandi*, beyond the enterprise one, in which my interlocutors are experts. Then, following those indications, I will use an *"observant participation"* method to join and observe the action of those appearing to be new actors, though still remaining poorly defined. My goal will be to investigate the reason for and the means of this actor's activity. Afterwards, I will examine the available "cookbooks" (books on Web-based application development) to identify and explore the actors using them and the corresponding reasons for their action, as assumed by the books' authors. Finally, I will use the story of an application of which the development reason is known to further explore the identified *modus operandi* as a whole. This methodology will be further analysed in Section 2.2.

Part II will compare the outcomes of *Part I* with other industrial settings, to explore whether or not the peculiarities identified are particular to Web-based application development or not. On the basis of the requirements of my problem, I will study the history of the *enterprise computer*, the *personal computer* and the *radio* industry. For this, I will use the works of historians in the corresponding fields, in the few cases where a topic in poorly explored by those authors, I will also use primary field texts. This methodology will be further analysed in Section 9.3.

Finally, *Part III* will use methods to access the field that are proper to informal and ephemeral settings. Firstly, I will use the method of '*"objervant participation"* to join *Barcamps*' conversational setting, and a "interaction monument analysis" method, using traces of interaction relating to the setting which are available on the Web to complete my observation. 16 Barcamps that took place in Paris over a period of three years will be studied and analysed. Subsequently, I will use the same methods to study the *Hackathon* exploratory setting, which took place in August 2010 in the headquarters of *Google*, California. Finally, I will use "interaction monument analysis" to explore the ways in which service providers support developers who use their technologies to innovate. This methodology will be further analysed in Section 14.3.

1.3 Contribution

The outcome of this study is the configuration of a specific *modus operandi*, distinct from the user and the manufacturer innovation paradigms (von Hippel and von Krogh, 2003, 2006; Raasch and von Hippel, 2012), which will be identified and explored in the first part. Surprisingly, as studied in the second part, this peculiar *modus operandi* is not proper to Web-based application development: similar actors, motives and means have appeared in all three industries studied, contributing to industrial development during special phases. Finally, three settings of informal interaction are identified that can be useful to enterprises harnessing the action of third party developers.

1.3.1 Part I: A peculiar *modus operandi*

More specifically, in the first part, I identify a peculiar *modus operandi*, which is placed in-between the user and the manufacturer paradigms (Raasch and von Hippel, 2012), in what regards its actors, reasons and means of action.

More specifically, three actor figures will be identified, according to their motives for action:

1. The User - Developer (UD), who uses his development skills to innovate for own use, much like an "lead user" as described in user innovation literature (von Hippel and Katz, 2002; von Hippel, 2005; Franke et al., 2006), though utilising both *"use-related sticky information"*, attributed by von Hippel (1990) to users, and *"technological sticky information"*, attributed to manufacturers.

2. The User - Developer - Entrepreneur (UDE), who, while having a similar starting point to the UD, does not "freely reveal" his creation, as the "private-collective" model suggests (von Hippel and von Krogh, 2003, 2006), but decides to go on, and try to commercialise it.

3. The Developer - Entrepreneur (DE), who, unlike the previous two figures, does not innovate for his own use, but does so for others, having a commercial goal at the outset of the design process.

Still, all three figures use the same means, which could be described as an "innovation palette", originating both from the user and the manufacturer paradigm. On the one hand, they use *free and open source software (FOSS)*, which has been the object of extensive management research during the last decade, on the other hand they use *Application Programming Interfaces (APIs)*, which have been given less attention by management scholars. Those *APIs* are settings provided by enterprises in order for developers to create applications using their technology.

1.3.2 Part II: the conditions of appearance of this *modus operandi* and its effects on industrial development

In the second part, while studying the question on whether or not the *modus operandi* met in the Web-based application development field is encountered in other industrial settings, I am led to further exploring it as a whole, in relation to the different industrial phases.

Retrospectively related to some of the most relevant models for industrial development, the outcome of my study will suggest that a transition from early "lead user innovation" (von Hippel, 1978b) to an industrial production is not linear: there is a progressive transformation of the actors, the means and the reasons of action during the exploration of the potential of a new object. In parallel, disruption (Christensen, 1997), during which a novel technology manages to identify a corresponding market, is not the only way in which a new potential is deployed: there can be situations where both a commercial and a technological potential are "visible" to the actors, though more exploration, often risky and implying revisions of previous concepts, is required for its to become true.

The roles of the actor figures identified in the first part according to the phases of industrial development will be the following:

- **Early Materialisation.** *User-Developers*, often having intimate ties with universities, use a new theory to create the first objects to illustrate the use potential of the theory. These materialisations can either be "better solutions" for old concepts, or "new dreams", materialisations that illustrate that other kinds of objects are possible. From these early UD circles, some will go on and attempt to commercialise this materialisation, thus becoming *User-Developer-Entrepreneurs (UDEs)*, while others will become their early adopters.

- **Market Emergence**. UDEs manage to find some early clients and produce an early product. Usually, this fact can trigger the interest of neighbouring business sectors actors, joining the challenge. Still, at this early level, the full potential of the new object remains largely unknown for all actors implied. Competition begins and UDEs have to reason commercially, beyond their own preferences, to have a chance to survive.

- **Foggy Competition**. *Developers-Entrepreneurs (DEs)*, often originating from early UD circles, further explore the marketing and technological potential, in parallel with the enterprises that have joined. A plethora of objects becomes available to early adopters, though the knowledge available, both from earlier phases and from other industrial rationalisations, is not sufficient to propose encompassing design rules (Baldwin and Clark, 2000) being able to lead in a market segmentation.

- **Industrial Rationalisation**. The previous phase ends, when an enterprise, using the knowledge produced so far as well as its own, creates a synthesis rationalising design, production and marketing of a unified product line. DEs cannot compete with enterprises, unless they trigger the beginning of a new cycle by returning to UDs and drawing a trajectory on a new concept, not encompassed by this rationalisation.

1.3.3 Part III: methods for the emergence and enterprise use of UDE settings

The last part contributes to the further exploration of this *modus operandi* by investigating three different settings. By constructing a framework of analysis that positions *collective action* (Hatchuel, 2005b) and *conversation* in-between social networks and communities, it suggests methods for the framing and the harnessing of UDE collectivities.

Barcamps constitute an exemplary case of ephemeral conversational settings that are used for the exploration of the emergence possibility of new UDE communities and networks. Their design also enables networking through a mainly conversational exploration of emerging technologies, markets and uses. In a different mode from communities' older members and networks core nodes, the "regulars" of these settings, while not necessarily sharing a common ground or being connected, are in position to benefit from the overall setting by taking into account those conversations and coming closer to the emerging networks and communities.

A *Hackathon* setting allows the focussed exploration of a specific service or technology potential, through the ad hoc constitution of groups that cover all levels of tacit group knowledge quality (Erden et al., 2008) in three days, by developing applications or prototypes on the basis of the desired potential. While their design shares attributes with other methods for creativity and knowledge-sharing, the outcomes of the *Hackathon* studied suggest that ephemeral settings may be particularly characterised by a "personal" dimension of both innovation process and its outcome.

Finally, *developer support forums*, despite their appearance as problem-solving settings, serve functions beyond problem solving itself. Using a system expert analytical framework (Hatchuel and Weil, 1992), I am led to the conclusion of a different kind of enterprise expertise, active in these settings. The "curator" is not necessarily required to have a *"doing", "understanding"* or *"planning know-how"* (Hatchuel and Weil, 1992). Instead, her actions aim at "taking care" of the UDEs, an activity that requires skills such as novelty identifying, conversation structuring and intimacy developing. The notion of "enterprise-UDE empathy" may be useful for the further exploration of this function, beyond the one of problem-solving.

1.3.4 Study overview

Table 1.1 outlines the structure of the current study. Part I addresses the problem of whether or not Web-based application development can be described according to the distinction of Raasch and von Hippel (2012) between user and enterprise innovation paradigms. I study this problem using a phenomenon-based research strategy (von Krogh et al., 2012) aiming in the distinction and the exploration of the Web-based application development *modus operandi*, exploring the questions of *who, how* and *why* develops such applications. This exploration will lead me to the identification of three actor figures, UDs, UDEs and DEs, of whose the action is based on use, use and profit or just profit reasons. Their action is undertaken by the use of both *FOSS* and *APIs*, as they exploit products, developed either by communities or enterprises, that prescribe a specific use, while allowing their utilisation as design instrument, too (such objects have also been described as "open products" (Chrysos et al., 2010)).

Since both the "private investment" and the "private-collective" models (von Hippel and von Krogh, 2003, 2006) are not sufficient to describe this *modus operandi*, the problem posed is whether or not it constitutes a peculiarity of the Web. Thus, I address this problem by examining different industrial settings (namely the enterprise computer, the personal computer and the radio ones). Finding similarities, I further explore when such *modi* appear and what their effects are for business, concluding with the proposal that they contribute to the exploration of a multitude of new concepts while they diffuse them to UDs or early adopters, during the industrial development phases that precede rationalisation.

Finally, Part III returns to the question of the Web and, focussing on the actors, addresses the problem of how enterprises can harness UDE activity. To investigate it, I study three different settings, being limited at the level of the actors. Posing the questions of the conditions of possibility for the social emergence of UDEs, as well as those of UDE activity harnessing by enterprises for potential exploration and exploitation ends, I propose three different methods (*Barcamps*, *Hackathons* and *Developers Support Forums*) rendering these conditions possible.

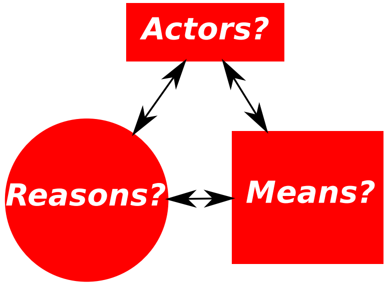
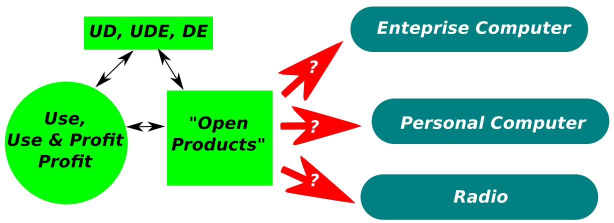
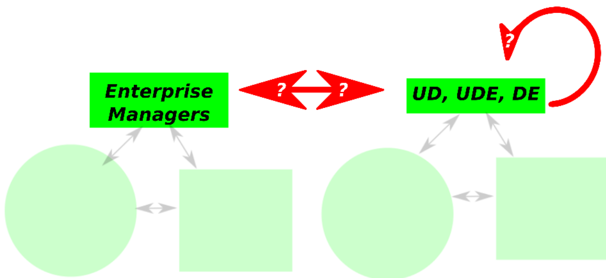
Part	Problem Schema *(in red)*	Problem	Research Questions	Outcome
I		Can Web-based application development be described by the user or the enterprise innovation paradigms, according to Raasch and von Hippel (2012)?	Who, how and why develops Web-based applications?	A *modus operandi*, described by: *Actors*: User-Developer (UD), User-Developer-Entrepreneur (UDE), Developer-Entrepreneur (DE). *Reasons*: use, use & profit, profit. *Means*: *FOSS* & *APIs* (*"open products"*).
II		Is this *modus operandi* particular to Web-based application development?	Does it appear to other industries, too? When and which are its effects?	Similar *modi* appear before industrial rationalisation and result to a the exploration of a multitude of new concepts during their diffusion to user-developers and early adopters.
III		How can enterprises harness UDE activity for their own ends?	How can these actors emerge socially? How can they be used for service potential exploration? How can they be used for service potential exploitation?	Three methods.

Table 1.1: Study overview.

Part I

The surprising characteristics of Web-based application development

Les caractéristiques étonnantes du développement des applications Web. Introduction à la Partie I

Cette partie aborde la question de « qu'est-ce le développement des applications Web ». Elle explore le problème de la nature de ce type de développement en comparaison aux paradigmes d'innovation par les usagers et par les industriels Raasch and von Hippel (2012) en étudiant ses spécificités. Cette exploration sera entreprise en utilisant une approche de *« phenomenon-based research »* von Krogh et al. (2012) qui fera usage des méthodes permettant de saisir les spécificités du terrain. Cette enquête suggère l'existence d'un *modus operandi* étrange, à la fois en ce qui regarde les acteurs, les moyens et les raisons de développement d'applications Web.

Le problème posé : Qu'est-ce le développement d'applications Web?

La littérature en Gestion, ainsi que celle en Sciences Sociales, a étudié le champ des services Web en adoptant une perspective d'usage et en explorant un éventail de contextes d'usage (usagers individuels, usagers en groupe, marchés et transactions, externalités de réseau). Cependant, cette littérature a ignoré le processus même de développement de ces biens, en le considérant de façon implicite comme une question qui ne regarde que la communauté d'ingénieurs.

Cette partie de notre recherche explore la nature du développement des applications Web comme un phénomène (von Krogh et al., 2012) et en comparaison aux processus de développement étudiés par la littérature de Gestion, ayant comme référence la distinction entre les modèles d'innovation par l'usager et par l'industriel (Raasch et von Hippel, 2012), illustrés dans la Figure 2.1 (page 28), prêtée par les auteurs. Dans cette première phase de notre recherche, nous sommes intéressés par la configuration de cette activité en rapport avec ces deux modèles, en ayant comme objectif d'inclure ce *modus operandi* dans un de ces modèles, ou, dans le cas inverse, de produire une description des spécificités du modus étudié.

Notre recherche sera alors limitée à la distinction et l'exploration (von Krogh et al., 2012) du *modus operandi* de développement des applications Web. À ce propos, nous divisons la question de la nature du développement d'applications Web aux questions suivantes:

1. *Qui* développe des applications Web? Cette question vise à identifier les *acteurs* spécifiques de ce processus, ainsi que des divergences éventuelles de ces acteurs en comparaison au modèle proposé par Raasch et von Hippel (2012).

2. *Comment* développent-ils des applications Web? Cette question vise à identifier les *moyens* spécifiques à ce processus, toujours en comparaison avec des modèles connus.

3. *Pourquoi* développent-ils ces applications? Cette question vise à identifier les *raisons* de cette activité, comme comparés à des motivations connues des deux modèles (von Hippel et von Krogh, 2003; 2006; Raasch et von Hippel, 2012).

L'exploration parallèle de ces trois questions n'est pas nouvelle en gestion. Comme Hatchuel et Weil (1992; 1995) le montrent dans leur travail séminal, *« L'expert et le système »,* les entreprises, lorsqu'elles se trouvent face à des nouvelles épreuves de rationalisation, doivent concevoir et mettre en place un nouveau *schéma organisationnel,* une *philosophie gestionnaire* et un *substrat technique*. Donc, les entreprises elles-mêmes sont appelées à répondre aux questions *« qui », « comment »* et *« pourquoi* », par la conception de ces trois éléments, établissant des nouveaux *acteurs, raisons* et *moyens d'action*, sur la base desquels prend lieu le changement organisationnel. En utilisant cette lecture des travaux de Hatchuel et Weil, nous transformons la question de la nature du développement des applications Web et son rapport avec les modèles connus, à un problème de configuration des trois éléments de son *modus operandi.*

La méthodologie utilisée : quatre axes d'exploration

La méthodologie qui sera utilisée dans la partie actuelle, est schématiquement illustrée dans la Figure 2.3 (page 29) et visera à distinguer et à explorer les acteurs, les raisons et les moyens étranges rencontrés sur le terrain. Cette exploration sera entreprise sous quatre axes :

1. *D'un modus operandi connu à un inconnu.* Par le biais d'entretiens d'experts de services Web bien connus (comme *Google* ou *Yahoo),* nous restituerons le discours des fournisseurs de services en ce qui concerne les originalités contemporaines dans leur développement. L'objectif sera d'identifier quelques premières indications d'un mode opératoire différent, au delà de celui dont nos interlocuteurs sont les experts, et qui leur semble être original. À ce niveau d'exploration, nous ignorons la manière dont les entreprises gèrent ce processus, mais nous savons que, dans la mesure où ce processus est géré, nos experts y sont impliqués.

2. *Identification des raisons et des moyens d'action, en considérant l'acteur connu*. En considérant l'acteur connu (les *« développeurs »*), nous utiliserons nos observations tirées de notre propre participation à leur action, afin de distinguer leurs moyens et leurs raisons d'action. À ce propos, nous utiliserons une méthode de *« participation observante »*, en prenant partie au développement d'un site Web.

3. *Identification des acteurs et de leurs raisons, en considérant leurs moyens connus.* En considérant leurs moyens connus (les *« Interfaces de Programmation d'Applications (APIs) »*), nous étudierons les *« livres de cuisine »* du développement des applications Web pour identifier les acteurs et leurs raisons d'action, telles que comprises par les auteurs de ces livres.

4. *Exploration des acteurs et leurs moyens, en considérant leurs raisons connues.* En considérant leurs raisons connues (la commercialisation d'une application initialement développée pour leur propre usage), nous approfondirons notre exploration du *modus operandi* dans son ensemble.

Résultats de la partie

La Figure 2.4 (page 31) illustre les résultats de cette partie. Tout d'abord, les moyens utilisés pour le développement des applications sont des *logiciels libres,* connus en gestion par un grand nombre d'études (Lakhani et von Hippel, 2003; von Krogh et al., 2003; Benkeltoum, 2009, et autres), ainsi que des nouveaux moyens, les *APIs,* fournis aux développeurs par des entreprises afin que les premiers puissent créer des applications en utilisant les technologies des seconds (Chrysos et al., 2010). Ensuite, en ce qui concerne les acteurs et leurs raisons d'action, nous distinguons trois figures d'acteur différents :

- Les usagers-Développeurs (UDs), exploitant leurs compétences de développement afin de créer des applications pour leur propre usage, de façon similaire à l'exemple d'innovation par des *« lead users » (von Hippel, 2005)*, sauf que les UDs ont des compétences à la fois liés à l'usage et à la technologie, qui les distinguent des autres usagers.
- Les usagers-Développeurs-Entrepreneurs (UDEs), exploitant leurs compétences de développement comme les UDs, sauf qu'ils poursuivent un effort de commercialisation de leurs applications, contrairement à la *« révélation libre »* des innovations rencontrées dans le modèle d'innovation par les usagers (von Hippel et von Krogh, 2003; 2006).
- Les Développeurs-Entrepreneurs (DEs), exploitant leurs compétences de développement afin de créer une application commerciale, potentiellement attirant une audience de marché ayant des préférences différentes de celles des DEs en question.

Donc, ces figures d'acteur sont positionnées entre les deux modèles décrits par Raasch et von Hippel, comme illustré dans la Figure 2.5 (page 31).

Présentation synthétique de la Partie I

Le Tableau 2.1 (page 33) présente une synopsis des chapitres à suivre. Tout d'abord, nous construirons un cadre d'analyse pour l'étude des innovations dans les services Web, en nous basant sur la littérature y relative. Ensuite, nous décrirons l'approche méthodologique qui sera utilisée sur les différents terrains de recherche à étudier. Les trois chapitres qui suivront, exploreront le phénomène en faisant usage des biais d'accès différents : premièrement, nous identifierons les étrangetés du champ de recherche, comme exprimées par le discours des fournisseurs de services. Dans la suite, nous approfondirons notre enquête en examinant les normes d'action de la figure du développeur, comme exercées lors du développement d'un nouveau service Web. Ces premiers résultats seront utilisés pour une enquête plus systématique sur les raisons d'action qui se basera à des *« livres de cuisine »* du développement des applications. Enfin, nous étudierons un cas spécifique de l'action d'un UDE, qui a développé une application pour les usagers-vendeurs du service *eBay.*

Les paragraphes suivants résumeront ces chapitres.

Concepts théoriques

Le Chapitre 3 est consacré à la revue de la littérature, où l'accent est mis sur les concepts à utiliser par la suite. En réexaminant l'expérience du phénomène des *« Dot.com »*, et la manière dont il a été abordé par les chercheurs de l'époque (Section 3.1), nous faisons remarquer le besoin d'une étude minutieuse des étrangetés du champ sous exploration (von Krogh et al., 2012) lorsqu'on examine l'éventualité d'un phénomène, pour contrebalancer les effets de la *« management fashion »* (Abrahamson et Fairchild, 1999).

Ensuite, la Section 3.2 examinera les approches différentes utilisées par diverses disciplines afin de décrire l'usage des services Web : trois conceptualisations proposées par des économistes, portant sur la valeur d'usage des services en ligne, ont eu une influence majeure sur les études en gestion. Donc, les notions de réduction des *« coûts de transaction »* et des *« coûts d'information »*, ainsi que des approximations sur la valeur d'un réseau pour ses usagers sont largement mobilisées dans la gestion de ce type de services. Néanmoins, quant à la pratique de ces service, d'autres valeurs sont révélées : de chercheurs en gestion ont mis l'accent sur le fait que, au delà de faciliter l'échange d'information, la mise en place dans les entreprises des dispositifs dits *« social software »,* conduit au développement des rapports peu conventionnels, qui incitent à des nouvelles pratiques de gestion. D'un point de vue différent, les sociologues, analysant l'usage de ce type de services, proposent que leur valeur distinctive soit associée au développement de l'identité personnelle des usagers. Cependant, toutes ces approches, explorent l'usage des dispositifs Web et ses implications, sans étudier les processus spécifiques de leur conception.

Par la suite, la Section 3.3 discutera la distinction entre les paradigmes d'innovation par l'usager et par le industriel (Raasch et von Hippel, 2012), qui servira de référence tout au long de cette partie pour le positionnement comparatif de l'activité sous exploration. Comme l'objet de notre investigation porte sur les services Web, le paragraphe 3.3.4 discutera les approches générales de gestion de services, du point de vue de l'innovation. À ce propos, un « paradoxe » apparaît lorsqu'on étudie le discours académique sur les services Web, lié d'une part à la tendance forte de transition des produits aux services (Cusumano, 2008), d'autre part à la nature « automatique » de ces services.

Approche méthodologique

Le Chapitre 4 discute les approches méthodologies mobilisées dans la Science de Gestion, en faisant remarquer les difficultés qu'impose un champ de recherche particulièrement fluide, susceptible de révéler un nouveau phénomène. À cause de ces limitations, nous construirons une posture méthodologique qui consiste à explorer en parallèle l'objet sous développement et le discours de ses développeurs, une posture qui présente des caractéristiques favorables à l'identification de phénomènes d'innovation potentiellement originaux, et plus particulièrement le cas du développement des applications Web.

Indications sur un nouveau *modus operandi* : le discours des fournisseurs de services

Le Chapitre 5 fera une restitution du discours des fournisseurs de services en ce qui concerne l'originalité du développement des applications Web, visant à identifier des indications conduisant potentiellement à la suggestion d'un nouveau mode opératoire, au delà de celui déjà connu aux entreprises.

Quelques premières indications seront donc identifiées, portant sur un acteur étrange, le « développeur » , qui semble explorer par son action le potentiel d'un service donné, en servant à ses propres fins. La création d'une « start-up » constitue une des possibilités de cet acteur, sans qu'elle soit pour autant un objectif toujours clair. De nouvelles technologies fournies par les entreprises, et plus précisément des interfaces conçues pour donner la possibilité à des tiers de créer leurs applications, visent un vaste public de développeurs.

Identification d'un *modus operandi,* se situant entre les modèles d'innovation par usager et par industriel

Le Chapitre 6 continue l'exploration de ce qui semble être un nouveau *modus operandi,* en étudiant les acteurs, les moyens et les raisons d'action qui peuvent le décrire. Cette configuration sera poursuivie sur la base d'une recherche de phénomène, qui se déroulera dans deux étapes : la première étape se base sur mes observations de ma propre participation au développement d'un site Web, la seconde se base sur l'étude systématique des « livres de cuisine » fournissant le savoir nécessaire pour cette activité.

Ces deux étapes viseront à l'identification des normes d'action (Argyris et Schon, 1978) des acteurs en question, en comparaison avec les modèles d'innovation *d'investissement privé* et *le collectif-privatif* (von Hippel et von Krogh, 2003; 2006). Les configurations en résultant (UDs, UDEs et DEs) émergeront comme une hybridation des normes d'action rencontrées dans les deux modèle, même si « l'investissement personnel » des développeurs semble jouer un rôle structurant dans cette activité.

Les figures d'acteur émergeant dans mon étude peuvent être décrites par la raison de leur action, dans les UDs, UDEs et DEs.

Exploration du *modus operandi* identifié

Enfin, le Chapitre 7 poursuivra l'exploration sur la manière dont les éléments différents de ce mode d'action sont mobilisés en pratique, par le biais du « récit raconté d'une histoire » d'innovation par un UDE.

En faisant usage d'une approche de narration, ce chapitre étudie un cas où une application développée pour un usage propre devient un bien commercial. N'étant pas un *« success story »*, ce cas illustre les dilemmes auxquels un UDE fait face.

Chapter 2

Introduction to Part I

Contents

This part addresses the question "what is Web-based application development". It explores this issue in comparison to the user and the manufacturer innovation paradigms (Raasch and von Hippel, 2012) through the study of its specificities. This exploration is undertaken by a phenomenon-based research strategy (von Krogh et al., 2012) and through the use of methods allowing one to seize the specific challenges of the field. It suggests the existence of a peculiar *modus operandi*, in regard to the actors, their means and their reasons for developing Web-based applications.

2.1 The problem addressed: what is Web-based application development?

Literature in Management, as well as in Social Sciences, has studied the field of Web service through a use perspective, exploring a wide range of use contexts (individual users, user groups, markets and transactions, network externalities) though ignoring the very process of development of these goods, implicitly considering it as an exclusively engineering issue.

This part explores the nature of Web-based application development through a phenomenon-based approach (von Krogh et al., 2012), in comparison to development processes studied by management literature, having as a reference the synthetic distinction proposed by Raasch and von Hippel (2012) between user and manufacturer innovation paradigms, quoted in Figure 2.1. In this early phase of my research, I am interested in configuring whether or not this activity enters any one of the two models proposed, or if, in the contrary, it constitutes a different model, distinct from both.

My research will be limited to distinguishing and exploring (von Krogh et al., 2012) the *modus operandi* of Web-based application development. For that, I break down the question "what is Web-based application development" into three questions:

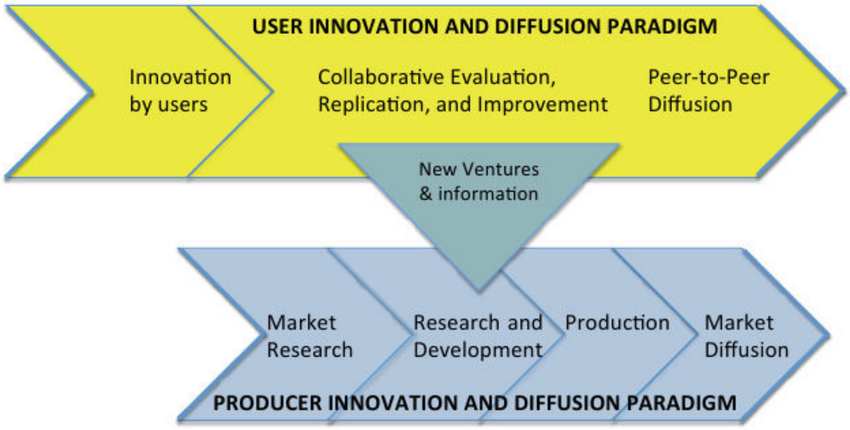


Figure 2.1: The synthetic distinction of innovation paradigms by Raasch and von Hippel (2012).

1. *Who* develops Web-based applications? - This question aims at the identification of the specific *actors* of this process, and their potential specificities in regards to the models proposed by Raasch and von Hippel (2012).

2. *How* do they develop Web-based applications? - This question aims at the identification of the specific *means* for this process, always in comparison to the known models.

3. *Why* do they do it? - This question addresses the *reasons* for this activity, as compared to the known motives of user and manufacturer innovation paradigms (von Hippel and von Krogh, 2003, 2006; Raasch and von Hippel, 2012).

Addressing these three questions in parallel is not unprecedented in management. As Hatchuel and Weil (1992, 1995) show in their seminal work *"Experts in Organizations: A Knowledge-Based Perspective on Organizational Change"*, enterprises, when faced by a new rationalisation challenge, have to conceive and implement a new *organisational schema*, a *managerial philosophy* and a *technical substrate*. Thus, in such situations, enterprises themselves are called to answer the questions "who", "why" and "how", by the design of these three elements, establishing new *actors*, *reasons* and *means* on the basis of which organisational change occurs. Using this reading of the work of Hatchuel and Weil, I formulate the question of what Web-based application development is and its distinction of known models into the problem of configuring its elements as a *modus operandi*, shown in the Figure 2.2.

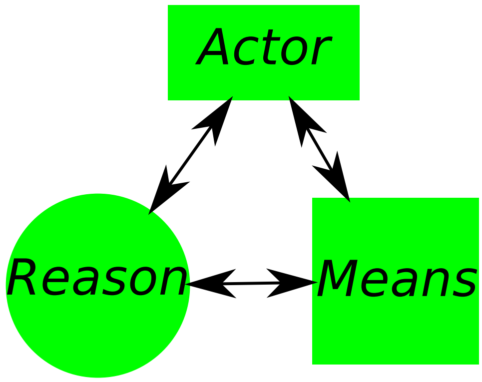


Figure 2.2: Elements of a *modus operandi*.

2.2 The methodology used: four angles of exploration

The methodology that will be used in the current part will aim at the distinction and exploration of the peculiar (von Krogh et al., 2012) actors, reasons and means of those developing Web-based applications. This exploration will be undertaken using four angles, as shown in Figure 2.3:

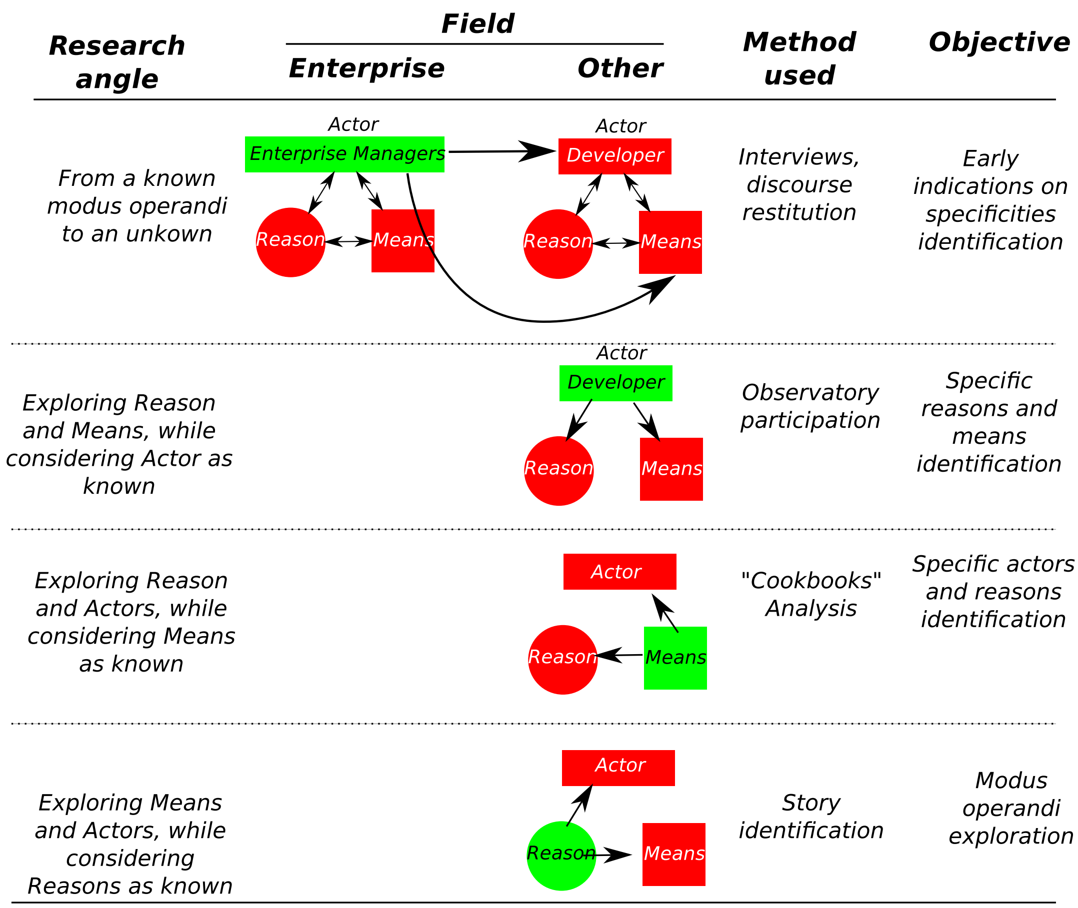


Figure 2.3: Part methodology.

1. *From a known modus operandi to an unknown.* Through interviews with managers of well-known Web services (such as *Google* or *Yahoo*), I will present their discourse

on what they think is original. The aim will be to identify some early indications of a different *modus operandi*, beyond the one in which they are already experts, where Web-based application development occurs and which seems to be, in their own knowledge, original. At this level, I do not know how enterprises manage this process, though I do know that - to the extent that it's the case - it is managed by the managers.

2. *Reasons and means identification, considering the actor known*. Considering the actors as known (*"developers"*), I will use my observations garnered from participating with their action in order to distinguish their reasons for and means of action. For this, I have used an *"observant participation"* method, taking part in the development of a Web site.

3. *Reasons and actors identification, considering means known*. Considering their peculiar means as known (*"Application Programming Interfaces"*), I will study the "cookbooks" of Web-based application development to identify the actors and their reasons for action, as assumed by the authors of these books themselves.

4. *Actor and means exploration, considering reasons known*. Considering the reasons of action known (*the commercialisation of an application developed for own use*), I will further explore the resulting *modus operandi* as a whole.

2.3 Part outcome

Figure 2.4 illustrates the results of the current part. Firstly, the means used for application development are both *open source*, those means having been studied by a great number of management scholars (Lakhani and Wolf, 2003; von Krogh et al., 2003b; Benkeltoum, 2008, and others), as well as a new means, *Application Programming Interfaces (APIs)*, furnished by enterprises to developers for them to create applications extending their service (Chrysos et al., 2010). Then, regarding the actors and their action reasons, I distinguish three different actors, all using the same means:

- The User-Developer (UD), using his developing skills to create an application for his own use, much like "lead users" do in the user innovation paradigm (von Hippel, 2005), though they differ from other users in the fact that they have specific skills.
- the User-Developer-Entrepreneur (UDE), using his developing skills to create an application for his own use, though later attempting to commercialise his creation, unlike the "free revealing" innovation model (von Hippel and von Krogh, 2003, 2006).
- the Developer-Entrepreneur (DE), using his developing skills to create a commercial application, that could be useful to a market audience having different preferences from his own, personal ones.

Hence, these three actor figures are positioned in-between the user and the manufacturer as shown in Figure 2.5.

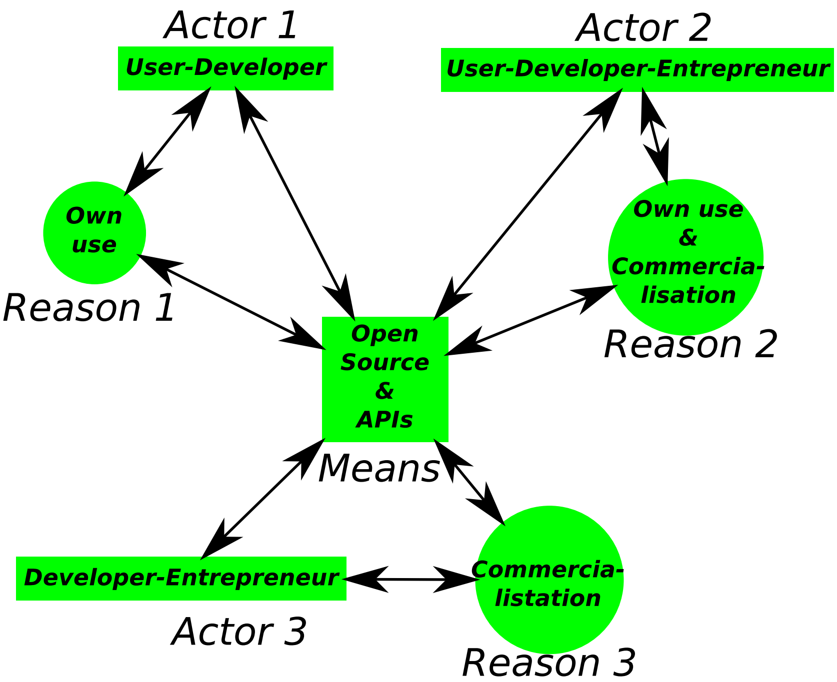


Figure 2.4: Part outcome: Actors, means and reasons configuration.

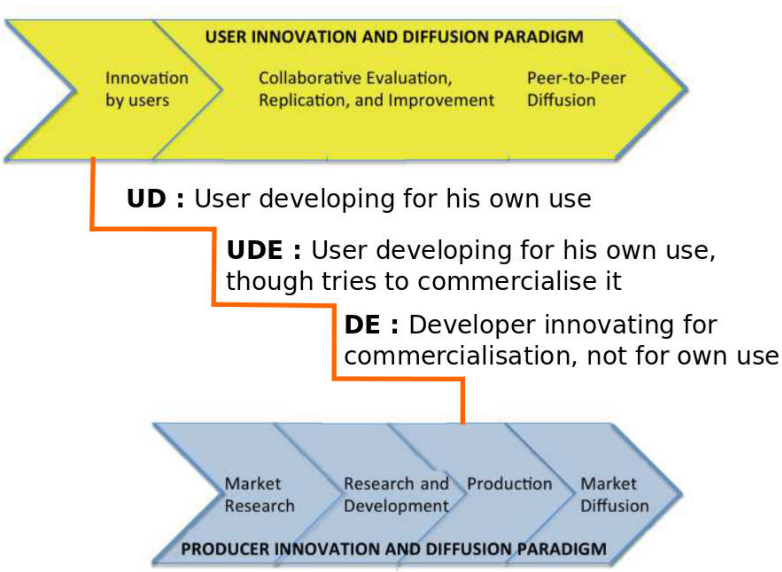


Figure 2.5: Actor figures identified as compared to the model of Raasch and von Hippel (2012).

2.4 Part overview

Table 2.1 presents a synoptic overview of the chapters to follow. Initially, I will construct an analytical framework for the study of Web services innovations drawing on the related literature. Then, I will describe the methodological approach to be used in the different fields studied. Then, the three following chapters will explore the phenomenon using different entries. Firstly, I will be interested in identifying the peculiarities of this field, as expressed in service providers' discourse. Then, I will further study the action norms of a novel figure, the developer, as exercised during the development of a Web service and as understood by those providing the necessary knowledge for this action, namely the authors "Developer Cookbooks". Finally, I will study a specific case of a User-Developer-Entrepreneur (UDE) creation, an *eBay* sellers application.

Theoretical Concepts

Chapter 3 is dedicated to the literature review, where emphasis is given to the concepts that will be used later on. Revisiting the experience of the "Dot.com" phenomenon, and the way it has been addressed by the scholars of that time (Section 3.1) I remark that there is a need for a careful peculiarities investigation (von Krogh et al., 2012) when faced with new phenomena, as a counterbalance to the influence of"management fashion" (Abrahamson and Rosenkopf, 1997).

Section 3.2 reviews the different approaches that have been used by different disciplines to describe the use of the Web services. Economists have proposed three very influential conceptualisations of the value of use for online services, extensively mobilised by the literature on Web business. Hence, the notions of "transaction" and "information costs" reduction, as well as the approximations of the value of a network for its users are widely used both by practitioners and scholars. Still, when it comes to practice, other values come to light. Management scholars have highlighted that, beyond facilitating information exchange, "social software" services, when used as a tool within the enterprise context, lead to "unconventional" relationships that call for new management practices. From a different standpoint, sociologist analysing the use of such services propose that their distinctive value of use is associated with personal identity. Nevertheless, all those approaches investigate the use of Web services and related interaction setting, as well as its implications, without looking into the specific process of their development.

Then, Section 3.3 will review the distinction between the user and producer innovations paradigms (Raasch and von Hippel, 2012), which will serve as a reference throughout this chapter, for the comparative positioning of the activity under investigation. As the object under investigation is the development of Web services, paragraph 3.3.4 will discuss the general theories describing services management. When studying the academic discourse on online services, a "paradox" appears during the passage "from products to services" (Cusumano, 2008) that I am studying: while service delivery highly depends on the "moment of truth" when clients meet the front-office employees, in this particular field there is no employee to meet.

Methodological Approach

Chapter 4 reviews the major methodological approaches used by management scholars and notes the difficulties implied for their use in a research field potentially revealing a new phenomenon. Because of these limitations, I will construct a methodological posture consisting

	Chapter/ Section	Outcome
3	**Theoretical concepts**	
3.1	The trend trap. The case of the Dot.com phenomenon	Need for a careful peculiarities investigation. Proposition: to focus on new technologies and new actors
3.2	Web services literature: focussing on use, missing development	There is a rich exploration of uses and a poor exploration of web services development process
3.3	Manufacturer and user paradigms. Who develops the applications?	Review of the distinction of user and manufacturer paradigms, using the perspective of Web apps development
4	**Methodological approach**	
4.3	An object - discourse investigation approach	A methodological posture for the emerging fields study
5	**Indications on a novel *modus operandi*: the discourse of service providers**	
5.1	Service provider managers interviews	Service providers discourse restitution
5.2	Outcomes	Peculiarities indications. New technologies: interfaces for application development; New actors: "free lance" developers
6	***Modus operandi* identification: in-between user and manufacturer paradigms**	
6.2	Theoretical concepts	A critical review of the "private-collective" model for innovation
6.3	Methodology: distinction and exploration	A two-steps methodology for phenomenon exploration
6.3.2	First step: early norms distinction	Immersion in the developers' action and discourse, through participation in the development of a Web service
6.3.3	Second step: exploration	Analysis of developer figures through the underpinning norms of their action through the study of developers' "Cookbooks"
6.4	Findings (first step): development for profit	Unlike user innovation paradigms, user-developers activity is driven by profit expectation
6.5	Discussion (First step): keeping a foot on both camps	UDEs action comprehends norms from both user and manufacturer paradigms
6.6	Findings (second step): different developers configurations	UD, UDE, DE distinction
6.7	Discussion (second step)	The three configurations of UDE figure act in the interplay between user and manufacturer paradigms
7	***Modus operandi* exploration**	
7.2	Theoretical concepts	A comparative framework setting
7.3	Methodology: narration as a phenomenon illustrator	The use of a story as an argument illustrator
7.4	Outcome: design and diffusion of a "spare-time" product	UDE action in practice
7.5	Discussion: Design and knowledge issues in third-party application development	UDE personal skills influence services competition

Table 2.1: Part 1. An overview: chapters of the current part and their synoptic outcomes.

in the parallel examination the object under development and its developer's discourse, as a privileged way to explore potentially original innovation phenomena, and more specifically the Web services one.

Indications on a novel *modus operandi*: the discourse of service providers

Chapter 5 will effect a restitution of the service providers discourse on the originalities of contemporary Web service innovation, aiming at an early identification of originalities that could suggest there is a novel *modus operandi*, beyond what service providers usually do.

Some early indications will thus be identified on a peculiar actor, the developer, who appears to explore the potential of a given service for its own profit, potentially creating a start-up, in the case of an exploration leading to a concrete concept. New technologies supplied by services, and in particular interfaces conceived to enable third party application development and, thus, the exploration of the potential of a given service by third party developers, aim to be adopted by a dispersed public of developers to create new services.

Modus operandi identification: in-between user and manufacturer paradigms

Chapter 6 further explores what is suggested to be a novel *modus operandi*, examining the actors, reasons and means that can describe it. This configuration will be undertaken using a phenomenon based research strategy (von Krogh et al., 2012), which will deploy in two steps: the first one consists in my observant participation in an ephemeral developers' team building a Web site, the second consists in the systematic study of the "Cookbooks" providing the knowledge required for this activity.

Both steps aim at identifying the action norms (Argyris and Schon, 1978) of this figure and comparing it with the "private investment" and the "private-collective" innovation models (von Hippel and von Krogh, 2003, 2006). The resulting configurations (UD, UDE, DE) emerge as an hybrid of action norms met in the two models, though the "personal investment" (necessary skills and knowledge acquisition) of the developers appears to have a structuring role for this activity.

The emerging actor figures can be described through the reasons behind their action as summarised by three actor figures: User - Developers (UD), using their skills to create something that can be useful to them, much as in the user innovation paradigm (von Hippel, 1975), User - Developer - Entrepreneurs (UDE), commercialising this creation and Developers - Entrepreneurs, creating something that does not correspond to their own needs but instead corresponds to their projections of what a potential clientele would buy.

Modus operandi exploration

Finally, Chapter 7 will further explore how different elements of this mode of action are mobilized in practice, through the "telling" of a story of UDE innovation.

Using a narrative approach, this chapter explores the way in which an application developed for personal use becomes a commercial good. Given the particular case is not a "success story", it illustrates the dilemmas a user-developer faces when trying to commercialise his creation.

Chapter 3

Theoretical concepts for an industrial development analysis of third party innovation in Web services

Contents

Introduction. The problem of an encompassing literature in Web services.

In management science there are very few approaches proposing generic models or theories to describe the management of Web-based application development, as most studies focus on specific aspects of features or cases encountered on-line, such as user forums or Wikipedia. In my view, this fact is due to the difficulty of distinguishing between "inherited" business practices and methods and "original" ones, either based on an ontological description of these services, on a genealogical study or on general theories. This kind of analysis becomes even more complicated as the field of online business is characterized by a significant fluidity.

Concerning business in online services, we can distinguish two currents of literature. In the beginning, before 2001, there were attempts to propose some general models for online services. However, after the dot-com bubble, management research encompassing approaches were less frequent. Rather than researching for general analytical frameworks, more recent management studies have focussed on particular cases and features.

The only analytical framework for on-line business is perhaps the one based on the notion of multi-sided markets, which will be explicitly discussed.

In this dissertation rather than seeking the "keys of success" in the online business ecosystems, or highlighting specificities of particular Web platforms, I will attempt to identify the shared and specific business attributes in contemporary Web services, their *modus operandi*, and propose a model for their management.

3.1 The trend trap. The case of the Dot.com phenomenon.

Abrahamson and Rosenkopf (1997) studied the fashions in management research and practice, proposing that they evolve according to life-cycles. At first, "emotionally charged, enthusiastic, and unreasoned discourse characterizes the upswings of management fashion waves", while during the downswings management discourse is characterized as "more reasoned, unemotional, and qualified". According to Abrahamson (2009), fashion is a general phenomenon observed in different scientific disciplines.

Paradoxically, in the case of the *Web 2.0* wave upswing, management scholars not only lacked enthusiasm, but they were rather reserved[1]. These reservations can be explained by the fact that this upswing appeared only a few years after the *dot.com* bubble burst, the latter characterized by a recent *OECD* study as an event that "helped lay the ground to for the [actual] financial crisis" (Keelee and Love, 2010).

However, as the Figure 3.1 suggests, there has been a considerable interest on the part of the general public in *Web 2.0*. This figure shows the evolution of the relative popularity of the term *"Web 2.0"* in *Google Search* requests since autumn 2005 when the term first appeared in public. We see that this public interest follows an upswing until the end of 2007 and then a progressive decline until the time of writing. This popularity curve raises the question of the term content, that is whether it describes a bubble, not having any original content after all, or, on the contrary, whether its content has progressively become a commonplace within the specific community concerned. In advance, since the form of the curve showing interest diminution is gradual and not sudden, one could suggest in anticipation that we may be in the second case. Unfortunately, similar data do not exist for the case of *dot.com* (as the Web - and *Google* in particular - was far less developed at the time), in order for us to be able to make a comparison.

Hence, a review of the management literature during this early wave of Web business is suggested as the best way to commence with the literature review. During the late 1990's, the *"dot.com"* wave had shown signs that it could lead to growth in the global economy. What

[1]While the documentation of the absence in generally, and the absence of enthusiasm in particularly, is a difficult task for researchers, the following title of a seminar organized in *École de Paris de Management* in March 2008 is illustrative of the ambivalence in the academic world:"Should we take the Web 2.0 seriously?"

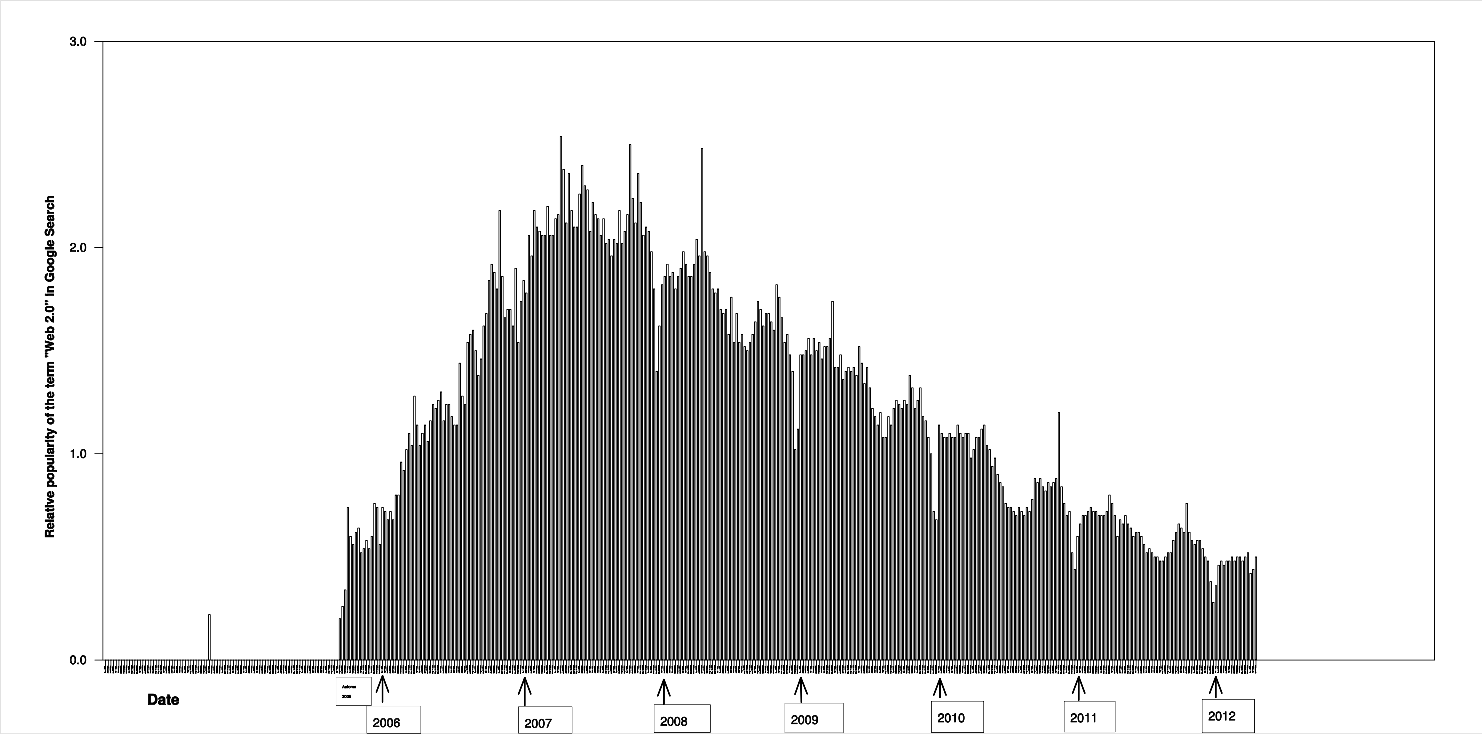


Figure 3.1: Popularity of the term *Web 2.0* in *Google Search* requests. We observe a downsing of the term popularity during the years. Source: *Google Trends*.

was named the "Dot.com bubble", in 2001, illustrated in a painful way that these economical predictions were too optimistic. Christmas 2000 had no Santa Claus for online business, as the projections of a dramatical rise in online sales were proven wrong, disappointing stock market investors (Aspray and Ceruzzi, 2008) and leading to the closing of a great number of business. Far from seeking out those responsible, a look into the literature of the time on how management research community addressed the issue can provide insights for the current study, regarding the theoretical and methodological posture to develop.

Using a term while not defining its referent.

The first paper of the pre-dot-com period that I will review is entitled *"Five steps to a dot-com strategy: How to find your footing on the Web"* (Venkatraman, 2000) and was published in the *MIT Sloan Management Review* just before the burst of what has remained in the collective memory as the "dot-com bubble".

The article calls managers to answer to the following questions, in order to develop their "dot-com strategy":

1. *What's your strategic vision for dot-com operations?*
2. *How do you govern dot-com operations?*
3. *What's your operating infrastructure for dot-com operations?*
4. *Is your management aligned for the dot-com agenda?*

The author concludes by stating the importance of having a *"dot-com vision"*:

> *They [established companies] need to blend their traditional and dot-com operations while confronting the challenge of brain drain as their top talent jumps ship for other dot-com operations. The game is far from over, and we will see powerful transformations as companies embrace the Net and craft innovative strategies that successfully blend physical and digital infrastructures. It's up to managers to take the necessary actions to align their visions to the dot-com world.*

This paper illustrates the tone of discussion beyond even beyond the academic community of management during that era. No definition or description is presented as an answer to the question "what is a dot-com?", since this question is not posed. Yet, while a "dot-com" is considered as an evidence of visionaries, the bubble of dot-coms, which was expressed with a dramatic fall of the stocks of the "high-tech" enterprises in 2001, illustrated in a painful way that there was no evidence on the nature of a "dot-com" for business, beyond the simple fact that they all had a Web portal of which the *URL* ended in ".com".

A similar effect was reproduced during the first period of the "Web 2.0" era (one could find the term "Web 2.0" as a generic adjective for business), evoking thus the suspicion on the part of the academic community, I've already mentioned. Yet, this time more explicit descriptions were initiated, often focusing on the new technical methods and features used in Web development. In this dissertation I will enter those methods of development in order to identify the novel ways of business that they suppose or impose.

Success stories without the keys to success: The missing link of knowledge sharing

A second example of the literature refers to the studies based on exemplar cases, in which "success stories" are studied. The researcher's attempt in this case is to induce the "keys of success" of projects usually undertaken by large enterprises. Yet, in the absence of a reference to a set of solid theoretical framework(s) or a positioning of the issue on a historical basis, beyond the temporal euphoria, new business phenomena could only be interpreted as "better applications" of already known approaches, when studied. In such circumstances, the potential that is hidden behind the success cannot be revealed by the researcher.

The article "*Pathways to E-Business Leadership: Getting from Bricks to Clicks*", by Willcocks and Plant (2001), also published in *MIT Sloan Management Review*, was developed on the bases of the following research question:

> *How do leading business-to-consumer corporations harness the Internet to acquire new customers and increase their market share?*

One of the case-studies used, was a grocery retailer, *Tesco*, expanding its activity in the on-line environment. The case is interpreted as a typical example of *"brand as a strategy"*, which passed from *"brand reinforcement to brand repositioning"*. The case was presented as follows:

> *In the United Kingdom, supermarket chain Tesco moved from brand reinforcement to brand repositioning over two years. First, in 1998 it reinforced its brand by creating Tesco.com, a wholly owned Internet subsidiary that allows customers to order groceries online for delivery and uses existing retail outlets for supply. In 1999, although the online business had lost £11.2 million on £125 million in sales, it also had attracted 300,000 users and was anticipating a profit in two years. By the end of 2000, Tesco had invested £56 million in its online retail business, dedicated 7,000 staff members to it, and had almost all 600 local stores online. At the same time, Tesco used the power of its existing brand and relationships with shoppers to reposition Tesco.com as a seller of services and goods other than food and to launch Tesco Personal Finance, an online joint banking venture with the Royal Bank of Scotland.*
>
> *Senior executives said they expected non-food goods ultimately to comprise half of e-sales and both Internet businesses to move into the market quadrant and reach profitability in 2001.*

In the presentation of the case by the authors we observe a typical myth of the *dot-com* era, that is the idea that creating a site is a sufficient action to enforce a brand. Another hypothesis, considered as evidence during that time, indicated in the paper by the metric of users, was the affirmation that a fast rate of user base growth guaranteed a future enterprise prosperity. In a book edited by the historian Paul Ceruzzi, this approach was named "Get Big Fast" (Kirsch and Goldfarb, 2008), and its aim was to exploit the "first mover advantage". To this purpose, many enterprises used to spend important resources on advertisement. Yet, many of those enterprises saw their stock options dramatically losing value or shutting down their business, despite their large user base. In other words, the dot-com world, at least as we can judge *ex post*, was not a "canonical" world, in which managers could align their visions.

As the discussion on the installed user base is still one of the main arguments concerning Web services managements, I will refer to it more explicitly in the next section, concerning the "network externalities" approach.

Nevertheless, *tesco.com* was one of the sites that indeed managed to survive during the bubble. Yet, a study realized in 2002[2], comparing *tesco.com* with other on-line grocery stores, such as *Webvan*, which reached a stock market value of $7.9 billion at the end of its IPO, had a different interpretation of *Tesco*'s success. As presented in an article in the *MIT Sloan Management Review* (Ellis, 2003):

> *The failure of Webvan and the success of Tesco, among other online grocers in Britain, illustrate the need for sellers to carefully educate customers about new services and coach them in how to get the most benefit from the service. (. . .)*
>
> *In addition, many failed online retailers concentrated too much of their marketing efforts on attracting a broad range of customers and too little on retaining target customers.*

The article of Willcocks and Plant (2001) uses a structure that is common to all papers of that time: based on some example cases, they propose a list of advice for enterprises entering the on-line business. However, in the absence of reference to a specific theoretical framework, a genealogical analysis or a description of specific methods, techniques or tools used by the enterprises in these cases, the reader has the difficulty to judge whether these advices are "general truths" - such as the different modes of branding for companies addressing large public - or whether they correspond to the specificities of on-line business. In the case of *Tesco* for instance, one could not state that "brand repositioning" would be a false strategy. Nevertheless, "brand repositioning strategy" wasn't a pertinent description of the *originality* of *Tesco*'s practice, which consisted in *accommodating and educating users in the use of its service*.

3.2 Different disciplinary approaches on Web services use

This section reviews the major conceptualisations of Web services in three disciplines: economics, management and sociology. Table 3.1 outlines the level of analysis and the use descriptions explored and adopted by different studies. Economics analyses the phenomenon at an abstract level, interested in value measuring. To this end, their notions are mobilized: a) information cost reduction, b) transaction cost reduction and c) network externalities.

In management, contemporary Web services (described as *Web 2.0* or *social software*) are studied through three perspectives, corresponding to different levels of use: a) within the enterprise context, b) between an enterprise and its clients and c) beyond the enterprise, at the level of autonomous communities.

Finally, sociologists enact a more detailed study of user practice, proposing that the peculiarity of those services resides in their use for personal identity affirmation and construction.

However, all the above approaches adopt an approach of methodological separation between the developer or designer and the user actors (Callon, 1992), and pay attention to the user side, this is generally the case aside from most studies in the current of STS

[2] Internet Disintermediation of Food Delivery: Spanning the Last Mile, 2002.

(Science, Technology, Society) (Oudshoorn and Pinch, 2008). In the current study, I will not "choose" andy of these use fields to further explore, since I am interested in the development rather than the use of such settings. However, the literature review operated in the following paragraphs will be useful, as an illustration of the variety of use contexts and fields to which Web services and applications are addressed, thus indicating a great potential for innovation from a development perspective.

3.2.1 Economics use conceptualisation: information & transaction costs, network value

There are three major theoretical approaches in economics that have been influential in online business management studies. Two categories can be distinguished in these approaches regarding the perception of users' benefit: on the one hand, *information and transaction costs* theories, developed in different periods though sharing some common elements, which focus on costs and on the other hand *network externalities* theory, which focus on added value. The level of analysis of these studies has initially been at that of the market or the consumers. Later, the same framework has been used to analyse communication costs within a value chain.

In the field of online services, use is addressed either indirectly, diminishing costs for consumers through the facilitating future transactions, or directly, adding value to a given service through rendering it more useful for its clients.

This section briefly reviews these concepts and their use in the study of online services, noting the limited potential of theoretical concepts use for the design and development process of a new Web service or application.

Information and transaction cost theories: application in online services and limits.

Most management scholars investigate the case of online services using the framework of information economy. A very influential study on the 'information' or 'knowledge economy' was the doctoral work of Porat (1977) at Stanford University, *The Information Economy: Definition and Measurement*. Porat identified as a major source of value the 'information cost' preceding a transaction, and proposed a way to measure it[3]. For him, this cost represented a great part of the overall economy and information technologies that had emerged provided the chance for business to profit from its reduction. The foundations of the approach of the information economy are to be found in the analytical methodology of 'Input - Output Economics' (Neisser, 1941; Leontief, 1941; Walras, 1896; Quesnay, 1759), a quantitative economic technique that represents the flows of value within different elements of an economy.

A similar approach is found in *transaction costs reduction* (Wallis and North, 1986). Both Wallis and North on the one hand, and Porat on the other, share a common ground in

[3]A different approach was developed in France, during the same period. A report of the *Inspection Générale des Finances*, that became very known among public at large as the Nora-Minc report, concerned the "Informatisation of the society" (Nora and Minc, 1978). Adopting a wider view, beyond that of the market level, it underlined the need for a national plan for the "révolution informatique". It introduced the term "télématique", binding together the notions of telecommunications and "informatique". For Nora and Minc new technologies were to transform society in an horizontal way, penetrating all domains of activity, within a context of radical social changes, regarding existing institution, as well as consumer preferences that were urging for innovation.

Discipline	Level of Analysis	Use as. . .	Characterisation	Description	Limits	Indicative literature
Economics	Consumers, Market	value				
			Information/ Transaction costs reduction	Transactions are cheaper to undertake, due to low search cost.	Non-informational values (e.g. sentimental expression) are not taken into consideration.	Bourreau and Gensollen (2004); Eisenmann et al. (2006); Zott and Amit (2010)
			Network effects	The greater the number of users, the greater the value of use.	Network design and user interactions not examined.	Caillaud and Jullien (2003); Rochet and Tirole (2004); Brousseau and Penard (2007)
Management	Group	action norm				
			Autonomous communities	Communities are self-governed on the basis of common interests or know-how.	Regards already formulated and well defined communities.	Giuri et al. (2008); Haefliger et al. (2009)
			Harness external communities	Web platforms provide enterprises the opportunity to address external actors, clients or "contributors".	Relevance of contributions, firm-community interaction issues.	Whitla (2009); Ebner et al. (2009); Wirtz et al. (2010)
			"Enterprise 2.0"	A potential identification for *HR*.	Employees self-expression issues.	Mcafee (2006); Denyer et al. (2011)
Sociology, STS	Users	practice				
			Identity affirmation and construction	Users construct their identity through self-expression.	Service design process is not explored.	(Georges, 2009; Cardon and Delaunay-Téterel, 2006a; Cardon, 2008; Beuscart et al., 2009)

Table 3.1: The use of Web services through the perspective of different disciplines. The current study will focus instead on the development of these services.

an understanding of markets as places where future clients search for a good to purchase, those clients reasoning according to a bounded rationality decision-making model (March, 1978). The costs encountered in order for this search process to reach its conclusion are considered very significant for both clients and vendors, thus a source of value is proposed to provide ways to limit them. For instance, a real-estate agency putting a tenant in contact with a house owner is a typical example of the added value described by both approaches[4]. Transaction cost economics focus exclusively on market transactions facilitation. As such, they do not analyse production, design or organizational issues.

These concepts have been used for the study of innovation phenomena in the case of user innovation as well as in the field of platform management. The notion of *"sticky information"* (von Hippel, 1994) has been based on this theory, suggesting that use-related information is costly to transfer from the user to the manufacturer side. From a different perspective, modularity is proposed to be an efficient way to reduce transaction and information costs both within an organisation and a value chain, as group autonomy enabled by common design rules reduces the need for communication (Baldwin and Clark, 2000; Baldwin, 2008). Hence, the concepts of sticky information and modularity have acted as interdisciplinary boarders between economics and management, enabling the exploration of their further implications in collective action fields.

Most management studies on Web services adopt the approach of the transaction costs reduction as a means to analyse the value created (Bourreau and Gensollen, 2004; Caillaud and Jullien, 2003; Eisenmann et al., 2006; Baldwin and Woodard, 2010, and others).

Following this way of thinking, one can interpret the success of Search Engines on the Web, such as *Altavista* and, later on, *Yahoo* and *Google*, as 'facilitators' of the decision making process for future buyers or sellers using the Web to reduce the 'information' or 'transaction' costs preceding a commercial operation.

For instance, Baldwin and Woodard (2010), consider buyers and sellers considered (platform) *complementors* and *eBay* illustrates the importance of "bringing them inside the walls", reducing their *transaction costs*:

> *The buyers and sellers on eBay want to transact with one another, and are willing to pay a fee to the platform if it reduces their transaction costs. The same holds for the merchant and customer in a credit card transaction, and the searchers, searchees and advertisers on Google (Baldwin and Woodard, 2010, p. 39).*

The same value had become evident even before the Web, during the broad expansion of the *Minitel* online service in France, and specifically the immediate success of the *Annuaire Électronique*, an online user directory, and, later on, the *Kiosque*, an online directory for services available within the *Télétel* network.

Particularly, Bourreau and Gensollen (2004), studying forums of cultural goods retailers observe that user communities exchange knowledge that is necessary to buy a cultural or "experience" good. Of course, consumer forums are a fraction of the total amount of online forums, though more interesting for market studies.

Limits of the information economy concept: semantic Web and social networking platforms

By continuing this reasoning, scholars have predicted that "Web 3.0" - as opposed to the "Web 2.0"- would be deployed on the basis of the semantic Web concept (Lassila and

[4] For an in-depth comparison of the two approaches, see Engelbrecht (1997).

Hendler, 2007; Hendler and Golbeck, 2008; Hendler, 2008). The idea of the semantic Web is based on the classification of information according to hierarchical information categories. A very simplified version of this view, concerning business innovation, is the following: since information search has been a principal innovation trajectory for the online business, the semantic Web, where information could be more structured and, thus, more easy to access, would be the next great innovation trajectory for the domain.

Although semantic information categories were popular to the scientific community, and most frequently appear in disciplines such as data base management, early indications have suggested that they don't constitute the only way for a public of users to search and access information. As engineers had noted back in 2005, 4% of Web searches concerned queries on individual's names (Guha and Garg, 2005), while the equivalent part was estimated at a rate of 5-10% in 2007 (Kalashnikov et al., 2008). While *search engines* display information on the basis of user requests (typically by the use of keywords), *social media* 'push' information from user to user, mixing it with expressions of personal feelings, viewpoints or experiences.

While there have been evolutions on this approach in Web services infrastructures, more significantly in the organisation of large databases, end-user services being exclusively based on the semantic categorisation of information by the users, also called *"folksonomy"* (Auray, 2007), such as *Delicious*[5], had a limited success compared to social networking platforms, though they emerged during the same period.

Hence, the emergence of social networking services indicates a change in the value of information technologies, related to the information or transaction cost reduction, described by Porat and Wallis and North. Within the discipline of economics, the network approach better describes the social aspects, though always on an abstract level.

In parallel, transaction cost approaches do not enter the development and design process challenges. An exception is the work of Baldwin (2008), where she argues that modular design can reduce transaction costs[6]. Baldwin refers to a transaction network, such as a

---

[5]Delicious is one of the services that signified the *Web 2.0* era, being one of the first Web-based cloud computing services. The service consists in saving one's bookmarks or favourite Web pages online, in ones *Delicious* account. Moreover, users categorise these links by attributing one or more key words (tags). It was developed as a spare time project by Joshua Schachter, working at the time in Wall Street, and presented in one of the early conferences of the milieu of Silicon Valley developers/entrepreneurs, the *Foo Camp*, in 2003.

Following a course of growth, the service was acquired from Yahoo! in 2005 for about $ 30 millions. It was the period wherein most of the features of the Web 2.0 (such as blogs, forums, tagging systems, photo-sharing sites) have been developed, often by developers/entrepreneurs that later sold afterwards their platforms to actors such as *Google*, *Amazon* or *Yahoo!*.

However, the service did not manage to grow as quickly as the ones of its generation. In 2008, it had nearly 6 millions users, while *Friendster*, one of the early social networking sites founded a year earlier than *Delicious*, measured 70 million users in 2008. Eventually, Yahoo! sold Delicious in 2011.

See Lacy (2009); *YouTube Founders Acquire Delicious*, Slashdot, April 27, 2011. URL: http://news.slashdot.org/story/11/04/27/2112239/YouTube-Founders-Acquire-Delicious?utm_source=headlines&utm_medium=email . Retrieved on August 20, 2012;Charles Arthur, *Yahoo to sell Delicious for $1m*, The Gardian.18 March 2011. URL: http://www.guardian.co.uk/technology/2011/mar/18/yahoo-sell-delicious-stumbleupon . Retrieved on August 20, 2012.; *Friendster dying? More like growing*, Bitbot, June 29, 2008. URL: http://bitbot.wordpress.com/2008/06/29/friendster/ . Retrieved on August 20, 2012.

[6]Baldwin (2008) mentions:

> *Modularizations, whatever their stated purpose, create new module boundaries with (relatively) low transaction costs. Modularizations thus make transactions feasible where they were previously impossible or very costly (Baldwin, 2008, p. 42).*

market, of which the architecture may imply reduced transaction costs. Still, this study does not refer to the *development process per se*, but to its modular outcome.

The value of the network

Another popular approach for the interpretation of the "Web 2.0 phenomenon" is the one of network externalities (Shuen, 2008; Hendler, 2008; Deshayes and Bourguinat, 2008; Lee et al., 2010). According to this approach, the value of a good (e.g. a Web service) increases the more people use it. A common example in this literature is the case of the FAX, which becomes more valuable the more people use it. This approach was funded by Katz and Shapiro (1986) in order to propose *"a formal model on network competition"*, founded on the argument that *"consumers will base their purchase decisions on expected network sizes"* (p. 426). Since then, various "laws" have been proposed to capture the exact value of a network as calculated by its externalities[7].

While there is no consensus in the academic community on the right equation, there is however a certain reality of calculating Web services value by the number of end users. Web services are most frequently evaluated by financial circles on the basis of the number of users, as well as the estimated value-per-user.

Limits of the network externalities approach

Beyond stressing the importance of an end user base, the attempt to provide a precise numerical value of a service induced exclusively by the number N of network externalities, underestimates a number of variables, such as the quality of relations between end users, the number of third party applications available in the service, the number of developers making those applications, or the extension potential of the platform (as expressed for instance by the variety of available or future applications) and so on.

Moreover, we should note that this approach already existed before the dot-com bubble and influenced the entrepreneurs of that time. Kirsch and Goldfarb (2008) report the fact that during that time, enterprises used to make use of extensive advertisement, in order to *"get big fast"*. Nevertheless, the number of end users did not help many of them even pay back the cost of advertisement, leading to bankruptcy.

Furthermore, the FAX communication networks have a relatively fixed identity, concerning the terminals, the uses as well as the networks themselves. Nevertheless, as has been remarked (Le Masson et al., 2006) computers have a relatively weak identity. A consequence

[7] Metcalfe's Law (Metcalfe, 1995), proposed by the homonym inventor of the *Ethernet* network and further utilised by Shapiro and Varian (1999b), proposes that the value of a network is proportional to the following number:

$$P = n^2 - n, \tag{3.1}$$

where n is the number of end users, or network externalities. In fact, the value of a network is considered analogous to the number of possible pair connections among end users ($\frac{n^2-n}{2}$).

Reed (1999) proposed his own law, where the value of a network should be analogous to the following number:

$$2^n - n - 1 \tag{3.2}$$

or, as he put it in his paper in the *Harvard Business Review*, the value of a "group-forming network" increases exponentially, in proportion to 2^n (Reed, 2001). The reasoning here is that one has to take into account not only the pair connections between the end - users (one-to-one), but also the group connections (many-to-many). One can find different variations of similar laws (eg. the *KK-Law* by Kilkki and Kalervo (2004), or *Zipf's Law* used by Briscoe et al. (2006)).

of that is that networks, terminals and exchange modes can be a field of innovation in the case of the Web.

The utility of a numerical value to be attributed to a service or a network refers to the need for a market value of the service as a whole. Moreover, this market value is useful when an enterprise enters the stock market. In fact, entering the stock market (IPOs[8]) was the basic economic model for Web start-ups before 2001. Since the bubble though, and in the framework of the "Web 2.0" entrepreneurial wave, IPOs were replaced by acquisitions from large enterprises. However, when facing the issue of acquisitions, the monetary value of a service is to be taken into account partially, as more criteria are to be considered, such as technological and cultural integration issues, not included in the laws described above.

The multi-sided market approach

Further observations on networks, using in parallel the logic of transaction economics, led to the expression of the "chicken and egg" problem (Caillaud and Jullien, 2003). In networks that act as markets, such as *eBay* for instance, the value depends on the number of sellers and buyers. However, for a network to emerge a "critical mass" of either category is necessary to attract the other. Price moderation from the service provider has been proposed as a regulating strategy within this framework. Hence, Rochet and Tirole (2003) proposed that free access to "Internet portals" resolves this issue, by attracting potential buyer in the first place.

However, on the developer level, beyond the requirement that a service has to have as many users as possible, the network externalities approach does not provide further insights on how to conceive and develop an online service.

Another slightly different approach used by the literature is the one of multi-sided markets (also referred to as double-sided markets or networks). The review of this literature can be facilitated by the case of *eBay*. It is perhaps the most studied or referred to case in the domain of Web business, which makes it a horizontal case across different management approaches, thus consisting of a meeting ground for management scholars.

eBay has been one of the few companies founded before the dot-com bubble that has managed to become a leading enterprise in the Web services sector. Consequently, it has been mentioned as an exemplary case of Web service in a number of studies in Management (Lai and Turban, 2008; Karakas, 2009; Levy, 2009; Baldwin and Woodard, 2010; Suarez and Cusumano, 2010; Chantepie, 2010, and others). All these studies refer to the initial business model of *eBay*, described by the following sequence of processes:

1. An end user auctions a good on the Web site of the service,
2. he or she defines a starting price and an auction period,
3. then an auction within the community of end users takes place,
4. the good is sold to the bidder once the pre-defined period of auction ends,
5. *eBay* gets reimbursed for the transaction,
6. the seller sends the good to the buyer,

[8]IPO: Initial Public Offering.

7. the buyer may evaluate the seller on the latter's *eBay* profile, providing to the service the knowledge for the liability of the seller, to be used from other end users in future transactions.

In Caillaud and Jullien (2003), one of the basic articles explaining the approach of "two sided markets", used *eBay* as an example of a two-sided market with the following description:

> *Auction websites charge fees that are proportional to the transaction price or even piecewise linear, but sellers also have to pay registration fees that depend on their reserve prices.*

The authors propose a model on the "equilibrium market structures" that emerge as well as different pricing strategies. The intervention assets that double sided markets have in their disposal are the informational intermediation as well as the price discrimination (namely the transaction fees). Using a similar framework, Bourreau and Gensollen (2004) highlight the economic importance of the search tools in Web platforms, regulating the access to new products and thus, competition within this market.

Like most approaches by economists, Caillaud and Jullien (2003) and Bourreau and Gensollen (2004) use a "matchmaking" reasoning, that is an hypothesis that demand and supply pre-exist the market, and that the market just puts them together. The possibility that the Web service imposes or enables design rules on the development of the offer is not addressed.

As expressed in their more conceptual article, Rochet and Tirole (2004) describe a multi-sided market as platform with the following characteristics:

> *A platform enables or facilitates the interaction between the two sides provided that they indeed want to interact. The interaction can be pretty much anything, but must be identified clearly. In the case of video games, an interaction occurs when a buyer (gamer) buys a game developed by a seller, and plays it using the console built by the platform (p.5).*

In addition, the authors state that they share the view of the network externalities approach in the hypothesis that "there are non-internalized externalities among end-users", as "an end-user does not internalize the welfare impact of his use of the platform on other end-users", or in other words end-users act as individuals and not as groups (a proposition that has been criticized by *Reed's Law*, as we've seen in the Section 3.2.1).

The same analytical framework is utilized by Suarez and Cusumano (2010), (referred to as *two-sided networks*), also using the case of *eBay*:

> *Each side of the network represents a different type of user, such as bidders and sellers in the eBay system, and platform companies incur costs in serving each group but can potentially collect revenues from each group as well (p. 83).*

Here, the authors refer to the theory of Eisenmann on multi-sided networks. Eisenmann et al. (2006) also describe a similar function of a platform as a multi-sided market, as do Caillaud and Jullien (2003). Moreover, they refer to the potential of a platform to "embody an architecture" to facilitate user interactions as well as a set of rules "that govern transactions".

In fact, the transaction cost theory is one of the fundamental analytical tools used in the case of online services and for that reason it will be separately reviewed in the next section.

The limits of the market approaches

Market approaches described in this section have a limited view on network externalities. End users, either sellers or buyers, are taken into account only as unrelated individuals acting according to a logic of bounded rationality and having *cost* as their sole criterion of action. Active contribution to the service by or its content by end users or - even more - their innovation is not taken into account. Similarly, the specific architecture of the network (its topology) or the activity of third party developers/entrepreneurs on the extension of the overall platform are overlooked.

Chapter 7 will study the case of an *eBay* application development, where it will become clear that Web services become starting points for development, beyond their nature as networks of product use and diffusion.

3.2.2 The management use approach: unconventional action norms

The concept of the use of Web services platforms as an infrastructure for internal enterprise organisation came about as a "colonisation" of enterprise action norms by those met outside, rather than an extension of the community of practice logic.

Particularly, Mcafee (2006) argued that *"wikis, blogs, group-messaging software and the like can make a corporate intranet into a constantly changing structure built by distributed, autonomous peers - a collaborative platform that reflects the way work really gets done"*. While CoP approach emphasises the long term relationships built through the work tasks as a major factor for internal communities deployment (Wenger, 1998), the *"Enterprise 2.0"* approach focuses on the potential of online practices diffusion within the enterprise.

Consequently, the issue has been tackled as a matter of "technologies adoption" and not of technology design and development. Denyer et al. (2011) argued that adoption of such technologies has major implications within the enterprise environment, calling for a change in *"organisational culture"* and *"leadership style"*. More specifically, they summarised the challenge as the requirement that *"employees need to believe that it is safe to speak up"* (Denyer et al., 2011, p. 392). What is thus observed is how these technologies turn the question of "technological adoption" to one of "personal expression", even when they are introduced to environments characterised by a standard division of roles and labour, such as the enterprise.

Nevertheless, the concept of *personal expression* is different to the one of *labour*, and by extension to the one of collaboration, at least as it has been defined by classic scholars like Adam Smith or Frederick W. Taylor. Moreover, it is also different to the concept of CoP, as what constitutes an identity in this case is the common (practice, knowledge, interest) and not the personal. Besides, the results of Denyer et al. (2011), far from being local observations, match research results on general tendencies of the labour transformation, as expressed by the work contract. Lefebvre (2009), conducting a genealogy of the work contract concept from the late 18^{th} century to our days, found that there it undergoes a slow but stable transformation, suggesting that we have crossed a phase of transition leading to the establishment of a *"personal professional"* contract type.

Autonomous communities: "mechanism" users, not designers

Management research studying online communities has historically been interested in the open source phenomenon, where a distributed coordination of software development processes occurs (Bergquist and Ljungberg, 2001; Shah, 2003; von Krogh et al., 2003a; Bagozzi and Dholakia, 2006, and others), where users learn and advance their skills through source code sharing.

Still, the advancement of online interaction technologies lead to the use of such "mechanisms" of interaction by a broader public, beyond the developers. An exemplar case is *Wikipedia*, where the close dependence of articles and editors relationships has been highlighted (Kane, 2009), as well as the very fact that users construct and share a common information corpus (Gensollen, 2003; Benkler, 2006). Another case studied, also distant from the technical communities, has been the one of video gamers. Haefliger et al. (2009) observed how user-evaluation mechanisms, such as marking a comment or a user with one or more "stars", enable a self regulation of the community, through resulting, "bottom-up" user and content evaluation. Further research in such communities focussed on the different roles between the users, principally leaders and followers (Giuri et al., 2008; Ho and Huang, 2009; Sutanto et al., 2011), or brokers and spanners (Fleming and Waguespack, 2007). Overall, these studies explore norms of action that are rather novel, as they are unfamiliar to the formal organisation norms used within the enterprise context.

Such examples lead to the declaration of the *"Contribution Revolution"*, (Cook, 2008), according to which companies can harness the products of communities. The relationship between such communities and enterprises is often problematic and occurs in an indirect way, through the individual engagement of employees (Lakhani and Wolf, 2003; Dahlander and Magnusson, 2005; Jarvenpaa and Lang, 2011).

However, the design process of these interaction mechanisms is almost always neglected, while those studies who mention it clearly imply that they play a major role in structuring the interactions in question. More explicit description though would have enabled an exploration of the development conditions of such technologies or services from a user-developer-entrepreneur perspective.

Knowledge and information sharing among consumers.

Bourreau and Gensollen (2004) have studied user communities of online retailer services, such as *Amazon* or *Fnac*. In their paper they undertake a comparison between two theoretical cases, on a basis of a mathematical model of market simulation: a retailer who sells only very well-known books, and another one who sells many unknown titles. The originality of the study is that it takes into account the tools offered by the service in searching or evaluating a good, as this parameter is typically ignored in studies that consider Web services as "n-sided networks". They conclude that retailers have strong incentives to differentiate their catalogues of cultural goods to reduce the intensity of competition.

Gensollen (2003), referring to similar cases of online communities, characterizes in a more conceptual article the relations among consumers, as well as between consumers and enterprises hosting their communities. He proposes that the specificity of these communities is the absence of links among the consumers, as these communities are completely mediated by the platform. In addition, he stresses the fact that enterprises can have a direct relation with their clients, exploiting the use-related feedback as an input to their internal processes of innovation. He also suggest the need for a "limited intimacy" for participants, in order for their contributions in the online discussion to be more efficient for the whole community.

Gensollen (2007) stresses the role of an informational corpus jointly build by the community of customers (such as in the case of clients forums), important for the purchase and consumption of 'experience goods', such as books, films or music.

From a different perspective, though in the same context, Richard (2010) highlights the challenge for the business of customer relations, as it has to engage in the public discourse on its products and to implement "Internet sites" to "adjust end user information" on a unique knowledge base.

Given the expansion of online communities, the question of harnessing the benefits of their activity was soon posed by enterprises. Wenger's approach on communities of practice (CoP)(Wenger, 1998; Wenger et al., 2002) had been initially used as an interpretive framework (Hara, 2009). However, the three dimensions of CoP, community, practice and domain, defining the shared identity, knowledge and practices, cannot be "cultivated" by the enterprise in cases where the communities act beyond its boundaries: studies have suggested that there are different incentives between user communities and producer firms.

Hence, Burger-Helmchen and Cohendet (2011) called for further research attention to the enterprise-community relationship, as related to different types of users, since end-user communities refer to a different knowledge domain (Wenger, 1998) than user-developer communities.

A different approach has been within the framework of open innovation, where external communities are structured around idea competitions (Piller and Walcher, 2006; Ebner et al., 2009; Huber et al., 2009). trajectories of study emerged in the research community: on the one hand, Piller and Walcher (2006); Huber et al. (2009) emphasise the importance of platform design to harness this type of communities. On the other hand, Blohm I et al. (2011) explore the potential of collaboration between the participant teams as an alternative setting, as well as combinatory approaches (Ebner et al., 2009).

However, more recent research has proposed that "intrinsic" motivations, such as contributors' enjoyment, tend to lead to more substantial contributions than "extrinsic" ones, as monetary rewards usually assigned to contests winners (Frey et al., 2011). In the same line, Hienerth et al. (2011), studying user community animation cases from well-known enterprises, observed that none of the companies use monetary rewards. Instead, they rely on *"user's willingness to co-create in return for a) being valued as an equal partner, b) having the opportunity to work on new new products and services or on the improvement of existing ones that better fit their needs, c) being recognized by peers, and d) being allowed to take up ideas generated during the ideation process"* (Hienerth et al., 2011, p. 356).

Overall, while "crowdsourcing" logic more or less implicitly suggests that harnessing community value begins with addressing a task-division to a dispersed "crowd" (Howe, 2006b,a), what is observed in practice is a rather inverse process, where users participation is more a personal engagement, triggered by their interests, their desire to learn, to be recognised and to create.

Hence, what research has observed is that external contributions come about as a result of a personal identity process expression and construction, though this process does not fit with typical enterprise culture or professional identity action norms. Still, how can these incentives become design requirements for developers desiring to create Web-based applications? In order to identify some concepts that allow us to explore these questions, we need to further tap into these dimensions of personal activity - a work that will be undertaken in the second part of the current chapter.

Harnessing external communities

Crowdsourcing: a Web platform as a communication medium for innovation.

The literature on *crowdsourcing* sees the Web as an opportunity for enterprises to exploit the potential of end user communities. The term was coined by Jeff Howe, editor in the *Wired Magazine*. In his article entitled "5 rules of the new labour pool" Howe (2006a) describes the following principles of crowdsourcing:

1. *The crowd is dispersed.*
2. *The crowd has a short attention span.*
3. *The crowd is full of specialists.*
4. *The crowd produces mostly crap.*
5. *The crowd finds the best stuff.*

According to Howe's reasoning, the "crowd" can operate tasks, which can be of important specialization, as long as they are divided in elementary parts necessitating "less than 30 minutes to complete". Open calls for submission of ideas or solutions produce by default low quality contributions. Nevertheless, a filtering of these contributions can be operated by the user community itself.

Studies on crowdsourcing, mainly examine user contests, where a problem is posed by an enterprise and a good solution is sought. These studies examine the behaviour of the participants in crowdsourcing initiatives (Haythornthwaite, 2009) and their relation with expertise (Roman, 2009; Poetz and Schreier, 2010), the designing of new products, such as t-shirts by the crowd (Brabham, 2008; Piller, 2010), the design attributes of competition platforms (Huber et al., 2009) as well as task formulation issues (Kittur et al., 2008a).

Thus, literature on crowdsourcing faces Web platforms as a communication medium for innovation, either studied from the perspective of knowledge or ideas sharing, or from the perspective of competitive problem - solving. We should note though, that crowdsourcing does not influence the platform used. While a "good platform for crowdsourcing" can be a question of research in this literature, the object of the contest is usually beyond the device of the platform (e.g. a t-shirt). Participants do not modify the design parameters or functionalities of the crowdsourcing platform.

The consumer - producer

The capability of users to publish information on the Web has lead to a different model, very popular in the "Web 2.0" era: the model where users consume the information produced by themselves. We propose the distinction between two different categories of cases: the cases referring to a common corpus of information and the ones referring to a private corpus. Usually, the first category of information is shared in public, while the second one in an intimate circle of "friends".

The most common case of a common corpus of information is Wikipedia, which constitutes one of the study fields of interdisciplinary discussion among management science, sociology and computer science, though public enterprise forums are also studied in the same context. Research questions include contribution modalities (Levrel, 2006; Cook, 2008), including "best ways" of participation coordination (Kittur et al., 2008b), motivations of the "Wikipedians" (Nov, 2007), issues related to on-line discussion (Kittur et al., 2007; Garfinkel, 2008; Auray et al., 2009; Hansen et al., 2009), consumer communities governance issues (Forte et al., 2009; Fredberg, 2009; Haefliger et al., 2009; Nambisan and Watt, 2010), contributors relations (Kane, 2009).

In the second category, recent research examines the field of online 'social networks', such as *Facebook* or *Twitter*. Management studies focus on the use of these services from a marketing perspective, rather than their design and development. Research includes questions on the use of these services by large enterprises (Rybalko and Seltzer, 2010), on the relation between online and local networks (Felzensztein et al., 2010), participation of consumers in company marketing (Muniz Jr and Schau, 2011). A basic concept in these lineage of studies is the concept of the viral marketing, reviewed in the section 3.2.2.

Viral Marketing

Miller et al. (2009) summarize well another influential approach of management studies regarding on-line communities:

> *Consumers' preferences form within communities as individuals exchange opinions about products and services and observe one another's purchases.*

This aspect is studied by the field of online services and is termed viral marketing. This approach highlights the value of the *word-of-mouth promotion*, or *buzz* (Dye, 2000). Viral marketing uses electronic communications to trigger branded electronic messages throughout a widespread network of buyers (Dobele et al., 2005). The virtue of the Internet in this case is described as follows:

> *It makes talking easier for customers, and its low-cost, minimal response time, and potential market impact make it attractive for businesses willing to put in the effort to create and implement thoughtful viral marketing designs and campaigns.*

Viral marketing research questions include the content of the main messages of marketing campaigns (Dobele et al., 2007; Kaplan and Haenlein, 2011; Swanepoel et al., 2009), the importance of pre-existing customers relational networks in the spread of the buzz (De Bruyn and Lilien, 2008), the importance of capturing the customer feedback of consumers' behaviour (Phelps et al., 2004; Kalyanam et al., 2007).

What is important is the circulation of the information containing an enterprise's message. This approach is putting the enterprise into the position of a Web service end user, trying to infiltrate user networks. Yet, in our study we will be interested in the enterprises and entrepreneurs that design and develop Web services, rather than their end users. Nevertheless, to the extent that Web 2.0 economical models largely depend on advertisement, the question of how to construct services in which end users will "perpetually exist" and receive the advertisement remains important.

3.2.3 Sociological and STS approaches: the distinction between user and developer

Sociology as well as STS (Science, Technology and Society) studies are largely based on the separation of use and development processes, in an analogous way to which innovation models are distinguished by Raasch and von Hippel (2012) in the producer and the user paradigms. Mallard (2007) reviews the principal studies in sociology noting a distance between use and innovation perspectives, an *"impossible integration of sociology of uses and sociology of*

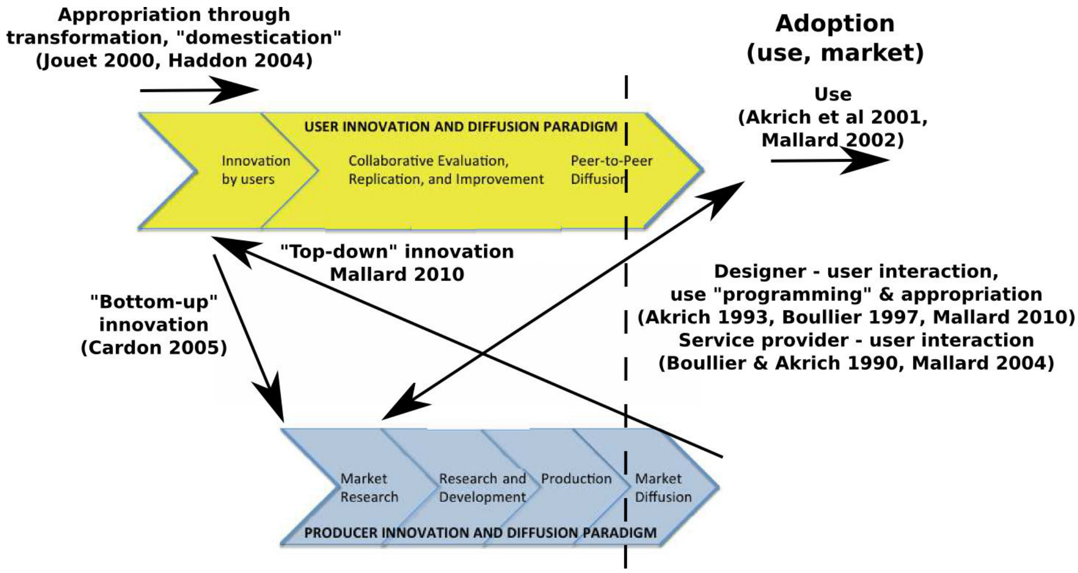


Figure 3.2: Research perspective: the review of studies on Web services use aims in the identification of Web service requirements, considered common both for UDEs and service providers.

innovation" (Mallard, 2007, p. 2), following the general rules of distinction between those developing and those using a specific innovation (Callon, 1992). Figure 3.2 graphically synthesises the different approaches, as mapped in comparison to the categories proposed by Raasch and von Hippel.

Hence, as Mallard synthesizes, innovation is mainly studied by a use and user perspective, highlighting different phases of product life. There are studies focussing on the appropriation through transformation or "domestication" of technology (Jouët, 2000; Haddon, 2004), on the user-designer interaction through "use programming" and adoption or through service provider and client interaction (feedback) (Akrich, 1993; Boullier, 1990) and through the integration of user innovations by manufacturers, or "bottom-up" innovation (Cardon, 2005). To that, Mallard (2007) adds the "top-down" innovation perspective, where provider innovations are transformed by users during adoption.

Particularly in what regards the field of contemporary Web services, *Réseaux* (Networks) review constitutes a corpus of systematic research on uses (Cardon and Delaunay-Téterel, 2006a; Cardon, 2008; Beuscart et al., 2009). Synthesising the research on the field, Cardon (2008) argues that "Web 2.0" services can be categorised through the "identity formats" adopted by users in relationship to the "visibility strategies" proposed by each service. Hence, Cardon proposes a cartography of Web 2.0 services, dividing them into *"civil identity"*, *"active identity"*, *"virtual identity"* and *"narrative identity"* ones. Hence, Beuscart et al. (2009) find that in *Flickr*, a popular photo-sharing platform, users share their photos with a controlled audience, where conversation on the photos marks their quality. A fact that service providers respect, as the forms of "editorialisation", that is the proposition of content to other users, respect this distinction of audiences.

As Georges (2009) observes, the peculiarity of online identity building lies in the fact that it is a process operated through the "traces" users leave, what commonly is referred to as content. Cardon and Delaunay-Téterel (2006a) in one of the early studies of this "school", explored the *"production of the self"* through a user interaction within the *blogosphere*. They analyse how, through using the features of a *blogging* platform (such as *posting, linking,* *commenting* etc.), users build their own identity while interacting with "their public".

The authors identified four modes of enunciation in blogs: intimacy sharing among anonymous users; familiar conversation among relatives; community-based coordination; and public opinion exchanges. Cardon and Delaunay-Téterel remark that, although these modes of blogging practice are characterised by different attributes, they all take place by using the same device, the blogging platform.

While the discussion of the relationship of this mode of use with the business world often leads to a discussion on the use of private information by enterprises in activities such as marketing or recruitment (Rallet and Rochelandet, 2011; Miltgen, 2011; Benraïss-Noailles and Viot, 2012), the studies in the use of "Web 2.0" services reveal an interesting fact for the development perspective: while these studies initially adopt a user-to-user level of analysis, they highlight the importance of the service "object" and its design to the definition, or the "programming" (Akrich, 1993), of user activity potential.

Hence, the conclusions of these studies could be used from a developer's perspective as a set of "general requirements" to respect when designing a new service, consisting in the conception of "places" where individual identity can be reaffirmed and deployed. Thus, concepts and knowledge used in the development process should include personal development. As we are going to discuss in the next chapters, this remark can be consistent with user innovation literature, to the extent to which the innovation in question embodies personal knowledge and ideas - as opposed to innovation embodying knowledge and ideas reflecting personal identities different to the developer's.

3.2.4 The use of the Web for communication amongst developers

A different case of online communities, is the case of developers communities. The issue of developer communities has been extensively studied on the field of open source software development.

As Benkeltoum (2008) noted, these are communities of user-developers. An influential book for management has been *The cathedral and the bazaar* by Raymond (1999), popular in the open source developers community public. Raymond described in his book how he built an e-mail client, *Fetchmail* (a program downloading one's e-mail from the server to his computer) with the help of a community. To the model of centralized software development process (the "cathedral") of an enterprise or an organisation, Raymond opposes the model of distributed development (the "bazaar") of a networked community of developers. As O'Mahony (2003) remarks, open source and free software projects are initiated and managed by a distributed group of people who do not share the same employer.

Raymond describes some good practices and propositions for community based development, later further explored by management scholars.

A first issue addressed was the one of problem solving:

Raymond's Proposition. *Every good work of software starts by scratching a developer's personal itch.*

The same proposition was also formulated as follows:

Raymond's Proposition. *To solve an interesting problem, start by finding a problem that is interesting to you.*

This proposition would be later formulated as the *private-collective model* (von Hippel and von Krogh, 2003, 2006). According to it, innovations comes from a personal problem solving process, to be later shared with the community of users. In addition, innovators benefit from "positive network effects" while diffusing their innovations. An aspect of this is that most innovations address problems faced by a user community (Benkeltoum, 2008).

A second field of propositions concerns community management and participation, where the question of enterprise participation is more explicitly studied.

Raymond's Proposition. *Treating your users as co-developers is your least-hassle route to rapid code improvement and effective debugging.*

Raymond's Proposition. *Given a large enough beta-tester and co-developer base, almost every problem will be characterized quickly and the fix obvious to someone. Or, less formally, "Given enough eyeballs, all bugs are shallow."*

Studies on participation noted that more experienced developers participate more in these communities (Bagozzi and Dholakia, 2006). A number of studies (Bergquist and Ljungberg, 2001; Lakhani et al., 2002; Lakhani and Wolf, 2003; Zeitlyn, 2003) explore the individual motivations for participation in opensource community, such as reputation, knowledge acquisition or personal utility. O'Mahony (2007) examines the governance of such communities, proposing a model taking into account the parameters of pluralism, independence, decentralized decision making, autonomy in participation and representation of the community in central decision making.

The role of enterprises in this framework can either be by contributing code to the community (Lerner and Tirole, 2004) or by communities of enterprises (Henkel, 2003). At the same time, Benkeltoum (2008), noting the elements of solidarity in these communities (beyond individual motivations), also notes that open source software projects are often developed in shifting organisational environments.

Compared to the case of Web services development, there are some important differences. While in the case of open source software the source code of the programs is accessible to all with the liberty (in the case of free software) to copy, modify, distribute and use the program at one's will, in the case of Web services the source code is not available to either end users or to developers. Use (consumption, production or development) of the platforms takes place on the basis of the interfaces, that is in relation with specific (and specified) inputs and outputs. Moreover, the availability of the service is under the control of one organisation, the service provider enterprise, which can control use and access issues, either by the terms of use, or by the devices themselves.

Chapter 17 will explore the use of the Web by developers for problem-solving processes, once the modus operandi of Web-based application development is determined.

3.3 A framework for an industrial analysis. The distinction between manufacturer and user paradigms

In management literature, as well as in practice, two visions of innovation process organisation are often opposed. On the one hand, there is the traditional enterprise-based innovation paradigm, which follows new product or service management methods (NPD) to provide a new good to the market and has been extensively studied in industrial contexts. On the other hand, user innovation often takes place beyond the enterprise context and within user communities, thus raising the question of the relation between this type of innovation, enterprises and markets. The case of Web-based platforms demands a critical review of both currents, as innovation phenomena in this context are located in the intersection between the two approaches.

Late research in user innovation focuses on interaction between user innovators and enterprises. In their recent article Raasch and von Hippel (2012) summarize previous work in user innovation and focus on the rivalry between two innovation paradigms: the "free", peer diffusion one, as studied by the literature on free-revealing of knowledge and innovations (von Hippel and von Krogh, 2003, 2006), and the producer-based 'paradigm'. This opposition has also been underlined by the complementary work of Baldwin and von Hippel (2011), according to whom the criterion for the distinction between the two is whether or not user innovators (either enterprises or individuals) freely reveal their innovations with a community of peers or, in the contrary, follow a closed, proprietary model for their exploitation. Thus, the economical motivations in the two cases are not the same: *"users expect to benefit from using a design, a product or a service"*, while *''producers expect to benefit from selling"* it (Baldwin and von Hippel, 2011, p. 3).

The general model of Raasch and von Hippel (2012), summarized in the Figure 3.3, can be used as a starting point for our literature review. There, the authors juxtapose two different paradigms, the user innovation paradigm (von Hippel, 2005, and others) on the top and the "linear" producer innovation one as described by Godin (2006), the latter induced by an historical study on managerial doctrines.

The arrow connecting the two paradigms represents for the authors the interactions between the paradigms. In their article they explore this interaction in terms of competition and complementarity between these two modes.

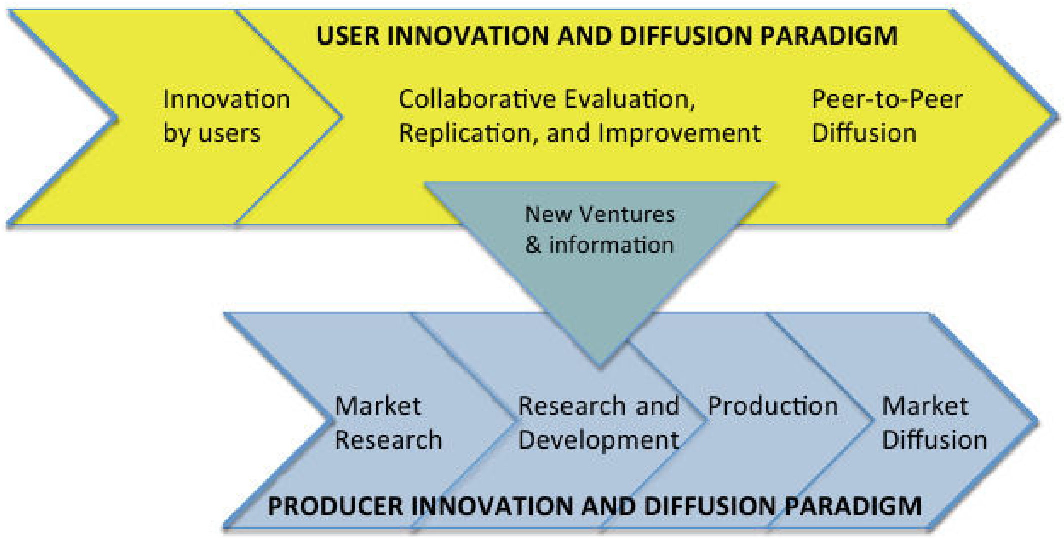


Figure 3.3: The user and producer innovation and diffusion paradigms (Raasch and von Hippel, 2012).

However, what happens in the case where a single developer creates and exploits an application on top of an existing platform? Is such a case valuable for platform providers? We will proceed to an analytical review of the two paradigms mentioned by Raasch and von Hippel taking into account the advances in Management literature studying the different phases outlined in Figure 3.3.

In the Figure 3.4 we indicate the positioning of our research field and problem within the literature on innovation management.

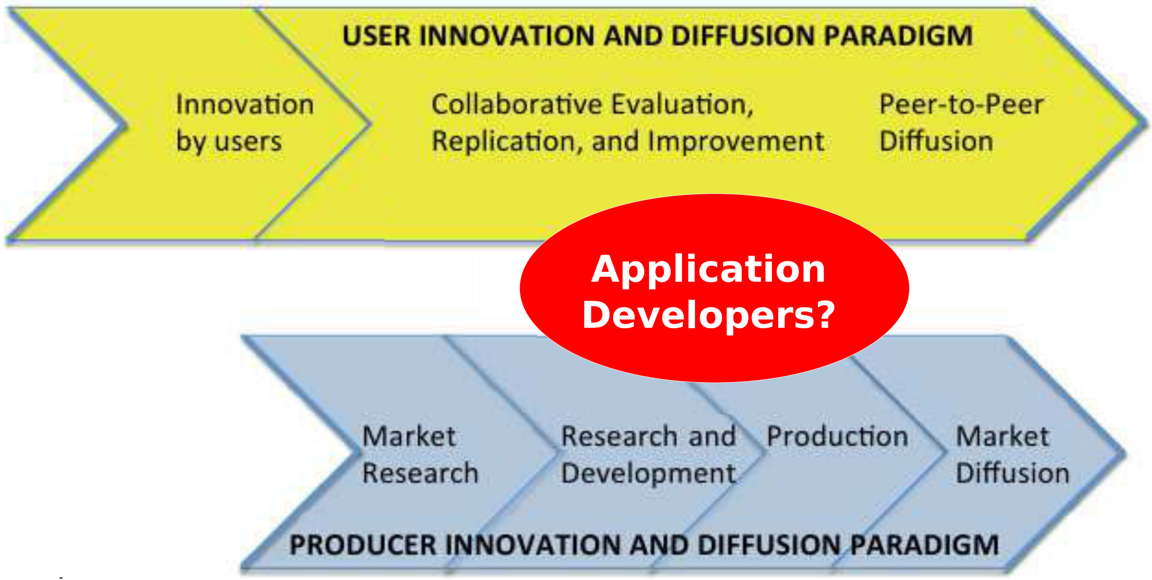


Figure 3.4: The position of our research field and problem within innovation management literature.

3.3.1 The user innovation paradigm

Raasch and von Hippel begin their reasoning by the review of studies on user innovation, which they see in three phases, according to the process observed in the field. Firstly, users innovate on the basis of their own needs (von Hippel, 1986). User innovation is understood as a problem-solving process, where "sticky information" (von Hippel, 1994), possessed by "lead users", yet ignored by the manufacturers (von Hippel, 1986), is mobilized for the design of new objects.

Afterwards, user innovators usually "freely reveal" (von Hippel and von Krogh, 2003, 2006) their knowledge and concepts to a community of fellow users, the latter being able to modify the initial innovations, as in the case of the free/open source software (Lakhani and von Hippel, 2003; Benkeltoum, 2011, and others). That is the phase of *"collaborative evaluation, replication and improvement"* indicated in the Figure 3.3.

Finally, innovations are diffused through *"peer-to-peer diffusion"* channels, such as horizontal user networks (von Hippel, 2007) or communities (Franke and Shah, 2003), where, according to Raasch and von Hippel, *"no producers need to be involved"* (Raasch and von Hippel, 2012, p. 2).

The experience of open source software development has been of an important influence on user innovation studies. Such projects act more as a common infrastructure rather than products. Through distributed collaboration (Lee and Cole, 2003) developers, participating either as motivated volunteers or as a part of their official occupation tasks (Bergquist and Ljungberg, 2001; Hars and Ou, 2001; Ghosh et al., 2002; Lakhani et al., 2002; von Krogh et al., 2003a; Zeitlyn, 2003), develop programs that can be commonly accessed and modified. An open question in this literature is the one of the relationship between enterprises and communities as, in order to lead an open source project, a constant change of the framing organisation is needed (Benkeltoum, 2008, 2011).

Two innovation trajectories for users.

The recent discourse on the interactions between the two paradigms does not take into account the different trajectories that user innovation may follow. The notion of "sticky" knowledge (von Hippel, 1994), while alerting enterprises on their ignorance of some valuable knowledge related to the goods they produce, it has implicitly discouraged the effort to further categorize user knowledge and concepts.

However, looking into the case studies of some of the most important studies on user innovation, we can retrace two different methods the users may follow: the one of problem-solving and that of use-diversion.

The first method, the one of problem-solving, results from the activity of users leading to innovation through the problem solving method. Here, innovators using "sticky knowledge" (von Hippel, 1994) regarding the product in use, identify problems that others cannot, and invent new solutions. This is the case for instance for "knee-activated brake levers", providing greater braking power and helping avoiding fatigue in a very mountainous terrain, in the case of mountain bike innovation studied by Lüthje et al. (2005). This type of innovation modifies the object's design parameters (the brake), though not the identity of the object or its use (in this case, it remains a mountain bike used for mountainous terrain).

The second method, the one of use diversion, results from the activity of users leading to innovation by highjacking the design rules of the object in use. Here, innovators have the knowledge of the use, though utilize knowledge of a distant field to divert the "kind of use", thus the identity of the object. A very enlightening study on this method is the case examined by Haefliger et al. (2010) where a gaming platform became a film producing platform, through user innovation.

Haefliger et al. (2010) study a case where user entrepreneurship takes place in a different market. In their case study, video game users utilize the virtual context of a video game platform (graphical environment, characters etc.) to create and commercialize films, entering into the motion picture market. The resulting filming "school", using video game platforms to shoot films is called *"Machinima"*. The conclusions regard user's creativity when in an entrepreneurial venture:

> *The point here is not that user entrepreneurs' creativity is unlimited but that their accumulated experience in the production of Machinima, and the gaming culture that the team shares with its audience, are sources of new ideas that can give rise to new opportunities and ultimately to commercialization (Haefliger et al., 2010, p. 1210) .*

In other words, the authors observe a double phenomenon: on the one hand, the configuration of the gaming platform (constituted by its features as well as its public) *conditioned* by entrepreneur's creativity, in a way similar to the one we described in Part I: it constituted an initial knowledge base and conceptual architecture, though expansive partitions (Hatchuel and Weil, 2009), in particularly by partitioning the design space by the use of a new knowledge base - the one of film making. It is important to note here that innovation did not entail the modification of platform's design parameters (DPs), but an introduction of new functional requirements (FRs). This *diversion* of the use opened up a radically new design space. On the level of knowledge, entrepreneurs needed to master the platform's design parameters. Moreover, they needed to have some knowledge, not trivial in the community of gamers, regarding film making.

As Haefliger et al. note, the origins of this transformation of use are to be found in gamers' innovation during the 1990s, according to which users often recorded their adventures and then published them. Moreover, they note that the difference between user innovators and user entrepreneurs lies in the latter's skills in applying complementary assets. They note that a "non-feature", a property not having a central role in the gaming experience (in particular the fact that avatars could point down their gun and look straight ahead) could be exploited in artistically (to create the illusion of two avatars engaging in a dialogue). Hence, features of a platform may obtain a higher value than the one predicted by a diversion of the use trajectory.

However, the design parameters were critical to enabling such an operation: this platform made available a feature which users could use to record their experience. The innovators studied used this feature beyond their own gaming experience, to record the avatars in the virtual environment as if they were to record actors, far beyond the game's objectives.

In Figure 3.5 I schematically represent the two innovation trajectories we encountered in the literature regarding user innovations. As in the C-K theory (Hatchuel and Weil, 2009), the schema is divided in two spaces: the concept space and the knowledge one. However, the concept space begins with a known artefact, that is the object of use.

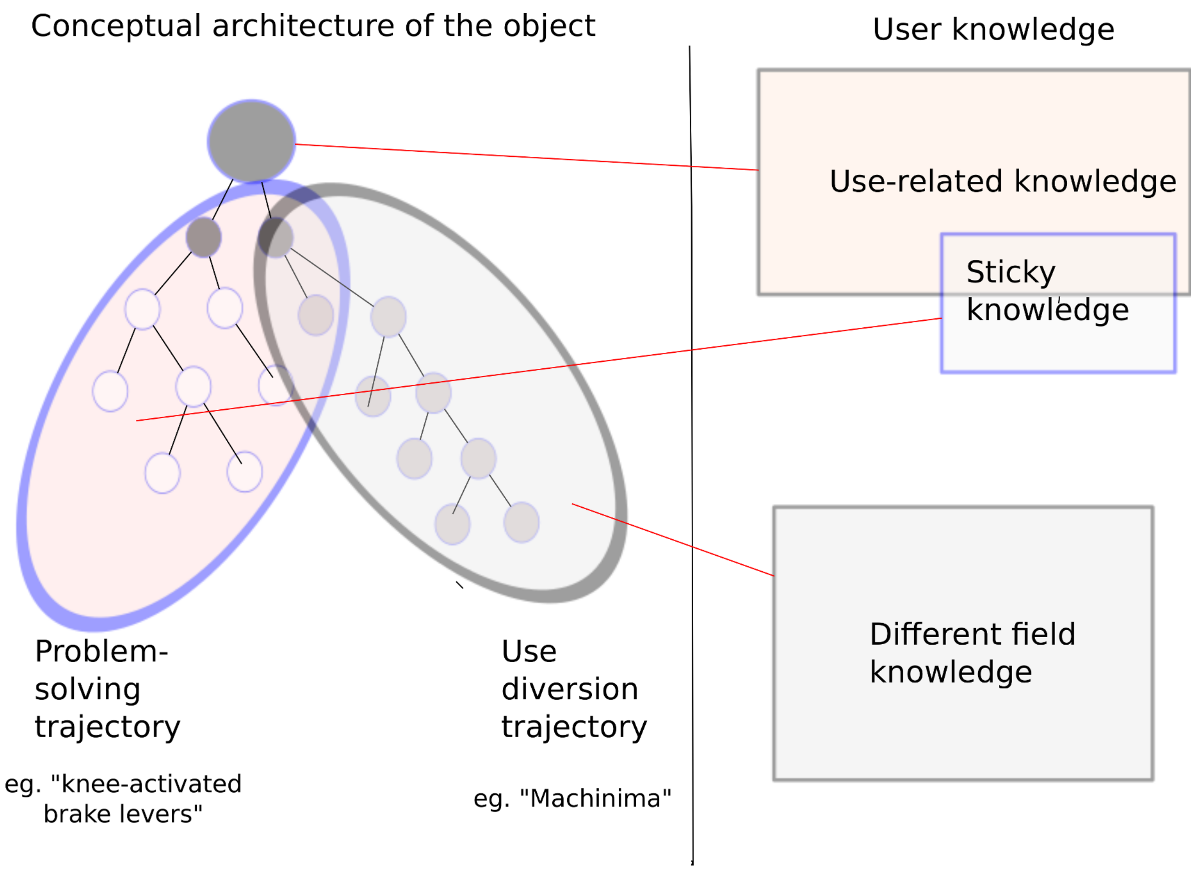


Figure 3.5: Two trajectories of user innovation: problem-solving and use diversion.

This simple categorisation is however important, as it implies different modes of enterprise-UDE interaction. When users use the problem-solving way, enterprises are expected to support the user activity. A specific task entering this regime (Hatchuel, 1999; Segrestin et al., 2002; Garel and Rosier, 2008; Elmquist and Segrestin, 2009; Benkeltoum, 2011) is contributing to the resolution of bugs reported by users, something that in the context of open source software development is undertaken by the community itself (Auray, 2004).

On the other hand, regarding the use-reversion trajectory, things may become more complicated for enterprises. Unexpected trajectories should be identified and platform potential exploration should allow the evaluation of the trajectory before engaging resources to support a new direction.

Of course, enterprises should be able to evaluate user innovation as a whole and be in position to autonomously propose new trajectories, aligning the community at the same time.

3.3.2 The manufacturer paradigm

While user innovation is often triggered by user needs, the methods followed in the business world are a lot more structured. Raasch and von Hippel describe the "linear paradigm" as follows:

> *Producers start by studying user needs, and then perform R&D as needed to develop and produce novel products and services. Next they diffuse what they have created via sales in the marketplace. As producers would lose profits and sales if other producers adopt their innovations without payment, innovating producers generally try to prevent this via such means as secrecy and intellectual property rights (Raasch and von Hippel, 2012, p. 3).*

The four phases presented by the authors (Figure 3.3) represent the main categories of the linear paradigm (Godin, 2006). However, these phases undergo important transformations as well: market research is no longer limited to studying user needs, R&D is called in to integrate the function of innovation, product development management is urged to take into account design as well as market considerations.

Typically, during the *"Market Research"* phase, where *"producers start by studying user needs"* (Raasch and von Hippel, 2012), marketing studies, following qualitative or quantitative methodologies, are undertaken by enterprises and institutions, aiming at an estimation of the demand, on the basis of consumer behaviour and consumption reasoning (Langeard and Meyer, 1975). Still, Drucker (1998) calls for attention on "unexpected occurrences", "demographic changes" and "changes in attitudes" regarding consumers: such transformations *create new market spaces* and trigger innovation processes beyond the already established metrics. The case of *Ford Mustang* is cited by Drucker as a result of the realisation that *lifestyle* was at the time the new segmentation of the car industry market, replacing the one of *income*. Moreover, marketing can introduce new values that did not exist before, what is also referred to as *value innovation* Kim and Mauborgne (1997). More generally, it has been remarked since the 70's that there is a fast rate of consumer behaviour transformations (Carof, 1973; Nora and Minc, 1978), with enterprises responding to those changes with short product life cycles, as the revenue depending on three year old products can represent 80% of the total revenue (Le Masson et al., 2006, p. 68). Thus, I can summarize the first phase as the one of the initial concept/idea elaboration, either being the response to a specified user need/desire, or being the fruit of a new value introduction, which in any case should correspond to a competitive environment of intensive innovation (Le Masson et al., 2006).

In parallel, R&D is placed by Raasch and von Hippel in the second phase of the linear model. However, R& D has also been a study field for research questioning the linearity of innovation process. The discussion in R&D management has taken important dimensions, especially since the 1980s, opening up two challenges (Rosenbloom and Spencer, 1996):

- R&D financing and articulation between science and technology on the one hand and
- capturing the 'fruits of research' through innovation on the other .

As we enter into the realm of competition through innovation, a broader view of innovation becomes necessary beyond the field of specific products, confronting it as *"a process having its own specificities, resources, targets and management"*, guiding and articulating Research and Development (Hatchuel et al., 2001). Hence the imperative of innovation in New Product Development (NPD) questions in practice the linearity of the process: management of concepts and knowledge, time, processes as well as the value included in the realm of innovation the acceptance of the unknown.

On the level of financing, different modes have been applied. While public funds can cover parts of costs, especially for large industrial groups (Gandon and Jacquin, 2001), industrial partnerships are also frequent, resulting in complex "webs" (Rosenbloom and Spencer, 1996) of alliances or joint ventures (Baptista et al., 1991; Inkpen and Beamish, 1997; Peng and Shenkar, 2002). Hence, co-evolution (Eisenhardt and Galunic, 2000), co-development (Chesbrough and Schwartz, 2007) and co-innovation (Maniak and Midler, 2008) incentives emerge to cover the gap in the financial but also the cognitive level within partnerships.

However, even when the phase of development (indicated in the bottom arrow in Figure 3.3) is reached, research and design is often not over, as the notion of "closure" (Bijker et al., 1987) of the design at some point in the process before launching a product to a market is challenged in practice. While useful in process modelling, linear models frequently fail when faced with the element of the unknown implied in innovation. Thus, a certain level of openness in the process has been proposed by different management scholars.

A very influential representation of this process is the funnel metaphore (Cooper, 1987), shown in Figure 3.6, according to which the initial project ideas are progressively filtered through a go/kill decision process:

> *A "new product funnel" builds in tough go/kill decision points throughout the process; the poor projects are weeded out; scarce resources are redirected toward the truly deserving projects - the high value ones; and more focus is the result (Cooper, 2001, p. 116).*

However, linearity in this process is questioned as well. During NPD processes, firms are called upon to acquire the virtues of speed and anticipation (Brown and Eisenhardt, 1995; Eisenhardt and Tabrizi, 1995; Jouini et al., 2004), to "freeze" the design concept in the middle - and not the beginning - of the NPD process (Iansiti, 1993, 1998) and develop "anticipation", "reaction" and "flexibility" capabilities (Verganti, 1999; MacCormack et al., 2001; Buganza and Verganti, 2006). Moreover, it has been proven (Hooge, 2010) that during NPD processes, value is not universally defined among stake holders, as it presents important variations through the course of the project. In the particular case of NPD collaboration, this variation makes mutual engagement difficult, as a contract *"may hide more than it reveals"* (Osborn and Hagedoorn, 1997), hence joint ventures may end in *"divorce"* (Peng and Shenkar, 2002). Thus a different kind of engagement, a *"contract for exploration"* (Segrestin, 2006) remains to be invented.

Besides, as enterprises act and react within a business environment implying among different actors, knowledge and ideas are exchanged along different organisations. Bayart et al. note:

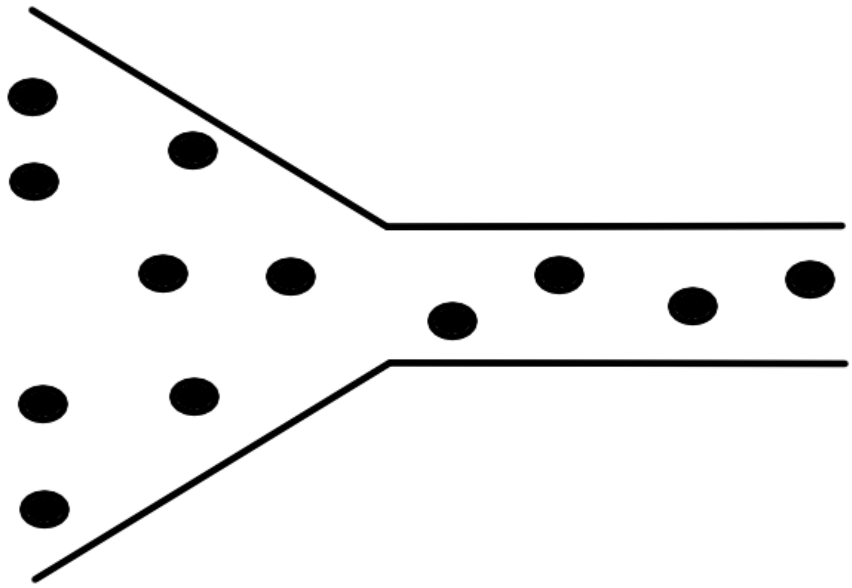


Figure 3.6: New product development *funnel* (Cooper, 2001). Poor projects are "killed", deserving projects "go".

> *By introducing the possibility of "trading", buying or selling projects (wholly or partly) at different stages in their development, the relevant business model is modified to take into account the management of the R&D portfolio. This takes us from the conventional R&D "funnel" to one which we might describe as a "porous funnel", leaving room for trading with other firms as shown in the diagram below (Bayart et al., 2000, p. 10) [Diagram reproduced in Figure 3.7].*

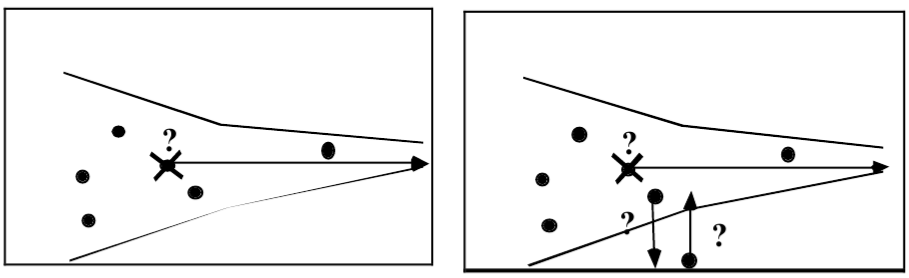


Figure 3.7: New product development *"porous" funnel* (Bayart et al., 2000). New projects can be bought during the NPD process.

For Open Innovation literature (Chesbrough, 2003a,b; Chesbrough et al., 2006, and others), these purposive *"inflows and outflows of knowledge and ideas"* can be managed through extensive licensing of inventions that are *"sitting on the shelf"* (Chesbrough, 2003a, p. 38).

3.3.3 Enterprise - users innovation interaction: linking the two paradigms.

Returning to the work of Raasch and von Hippel summarized by the Figure 3.3 (page 56)and the interaction between the two models reviewed above, the authors identify two different conditions, two different states of this interaction, *'user-contested market'* and 'user complemented market'. The first condition is when commercial products face competition from self-supplied users. This "contestation" excerts "price discipline" onto producers as, in some conditions, user innovations can exert greater competitive pressure than rivals. The second condition is when products are complemented by user innovation. This is the case when a product may be sold on the market, while techniques for operating that product may be diffused peer-to-peer.

The scholars also provide examples for both conditions. A typical illustration of contestation is the competition between "open source" and "closed source" software suppliers (Benkeltoum, 2008, 2011; Sen et al., 2008), while typical illustrations of complementary relationships are described by the literature on customization and user innovation toolkits (von Hippel, 1994; von Hippel and Katz, 2002; Piller, 2010; Oliveira and von Hippel, 2011). The relation between these user communities and enterprises, however is a research question for different studies, most of them based on the notion of "lead users" (von Hippel, 1986; Leimeister et al., 2009; Mahr and Lievens, 2012), where enterprises are called upon to reach the lead users and their innovations (Herstatt and von Hippel, 1992). Yet, this process can fail in the case of complex products (Olson and Bakke, 2001) as firms are often not in position to integrate innovations realized by communities (Dahlander and Magnusson, 2005; Le Masson et al., 2010).

Still, the specific condition where users have advanced development skills and use them to innovate on a given service through the creation of third party applications is not addressed by the two paradigms reviewed above. As regards my research, this is the very condition that I explore.

The specific condition of user-entrepreneurs.

A specific condition cited by Raasch and von Hippel, though not particularly explored, is when the *roles change*, and *users* become *producers*, the case of "user-entrepreneurship". The authors refer to other studies exploring this particular issue. Yet, since this regime (Hatchuel, 1999; Segrestin et al., 2002; Garel and Rosier, 2008; Elmquist and Segrestin, 2009; Benkeltoum, 2011) is closer to our research topic, regarding the way innovation is deployed on the basis of *Web 2.0* platforms, we will analytically review literature exploring this field.

Baldwin et al. (2006) study cases where users become entrepreneurs. Their research is based on a design reasoning, according to which *"user innovation begins when one or more users of some good recognize a new set of design possibilities - a so-called "design space" - and begin to explore it"* (Baldwin et al., 2006). Thus, user innovations cover this particular design space. Then, once an innovation starts to be diffused in a community of users, the authors observe that first *"user-purchasers"* appear, preferring to *buy* rather to *create* innovations themselves. Hence, the first manufacturers to enter the market are *"likely to be user-innovators"* using the same technologies they used to build their own prototypes (Baldwin et al., 2006). In their paper they develop an economical modelling of this process based on the work of Baldwin and Clark (2000) on design costs. Moreover, unlike the current of entrepreneurship research highlighting the importance of psychological and behavioural characteristics of individual entrepreneurs (Gollwitzer and Brandstätter, 1997; Zhao et al., 2005), Baldwin et al. highlight the collective aspects of entrepreneurship, proposing that user community encompassed entrepreneurship is more efficient in terms of cost and design exploration than solitary entrepreneurship.

Shah (2003) studied the case of innovation in sports equipment, specifically in windsurfing, questioning market actors about their early entrepreneurial steps. She found that *"of all expert practitioners who innovated, 71% sought to profit from their innovations by forming small, lifestyle firms that would produce their innovations for sale to others."* (Shah, 2003, p. 48). Moreover, she observed that existing sporting firms were not present in innovations during the emergence of the windsurf market.

Lüthje et al. (2005) also studying innovation in sports, though this time in the mountain bikes market, propose that expert users are more likely to innovate as they have already been involved with use and thus possess a high level of relative knowledge and experience. This way, they escape initial investment costs that outsiders would have to spend, in order to reach a level of formulating new problems and thus resolving them.

Shah and Tripsas (2007), in an effort to propose a theoretical framework for user-entrepreneurship, agreeing with Lüthje et al., propose the following formulation on their specificity:

> *User entrepreneurs are distinct from other types of entrepreneurs in that they have personal experience with a product or service and derive benefit through use in addition to financial benefit from commercialization.*

They also distinguish two categories, the end-users (that use a good in their day-to-day life), and professional-users (that use a good as a part of their professional duties). Their investigation concerned juvenile products manufacturers, where they found that many of the firms in the market came from the parents of the users. They argue that users are *"accidental" entrepreneurs*, as the development of the idea, experimentation, adaptation and preliminary adoption often occur before the formal evaluation of the idea as the basis of commercial venture. Their model consists in the following steps of user-entrepreneurship (Shah and Tripsas, 2007, p. 129):

1. The beginning comes from the existence of user's unmet needs.
2. User creates a novel solution to satisfy their own needs.
3. The innovation is illustrated within a community or in public, attracting though the interest of others and obtaining a first feedback.
4. Innovators identify the business potential and
5. forms a firm.
6. The firm enters the market and uses market feedback to improve the product.

Using the literature - phenomena mapping of Raasch and von Hippel (2012) (Figure 3.3 on page 56), I locate the steps proposed by Shah and Tripsas (2007) in the two paradigms, as shown in the Figure 3.8. The user-entrepreneur is considered by Shah and Tripsas as a transition from the user paradigm to the producer one, "skipping" the two early producer phases (*"Market Research"* and *"R&D"*). In the end, what differentiates user-entrepreneurs from typical producers is not their intermediary position, but their former user experience which is exploited to manage feedback, once their product is on the market. In parallel, the market replaces the *"peer-to-peer diffusion"* phase that would occur if they had remained in the user community.

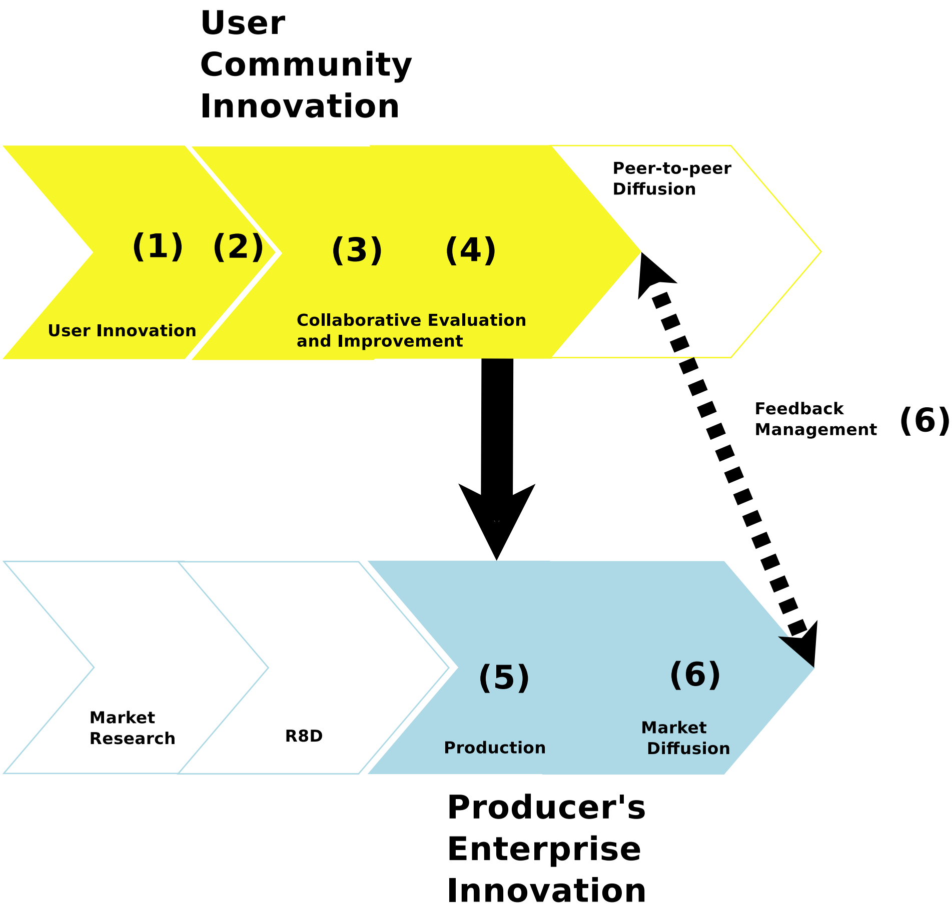


Figure 3.8: User-entrepreneurship steps according to Shah and Tripsas (2007) localised in the mapping proposed by Raasch and von Hippel (2012).

In all of the above case studies, user-entrepreneurs utilize their innovation to enter the initial market (in most cases considered, the sporting equipment one). Moreover, in all the above-mentioned studies on user-entrepreneurship, once users become manufacturers they are considered independent from the initial product manufacturer, who is rarely mentioned.

As I am going to show in my study, in the case of Web services platforms, that third-party innovations remain functionally attached to the initial platform, thus rendering the user innovators' autonomy impossible.

This impossibility of autonomy has to do with the nature of services in themselves. In services, as outlined by many scholars (Fixari et al., 1997; Bancel-Charensol and Jougleux, 1997; Jougleux, 2006), information plays a central role in the everyday delivery of the good. In addition, information changes over time, making it crucial for providers to have a permanent relation with their sources. Hence, the platform remains dominant over third parties, as long as it controls the flow of information.

The question of Web-based application development

In comparison to the above mentioned studies, Web based application development presents an important original phenomena as they aggregate in their platform third party innovations while the framing organisation (the enterprise) remains the same. That is possible because those service platforms are characterized by an important design originality: while keeping the code of their platform closed, firms (such as *Facebook* or *Google*) provide to exterior developers the means (called *APIs - Applications Programming Interfaces*) to expand their product. At the same time, the exterior developers benefit from the resources provided by the platform.

3.3.4 Service management: the automated services "enigma"

Unlike products, in services there is a direct interaction between user and enterprise during its delivery. This process, can be described as a potentially innovative design process, since it aims at making *"something that is partly unknown and partly specified with things that already known and/or discovered during the process"* (Hatchuel and Le Masson, 2007).

Moreover, there appears to be a paradox when faced with the case of Web services. On the one hand, management scholars highlight a strong tendency "from products to services", specifically in the field of information technologies (Cusumano, 2008; Gawer, 2010a; Suarez and Cusumano, 2010). On the other hand, as Cusumano notes, enterprises "productize" their service, so they can be delivered more efficiently. *Google, eBay* and *Amazon* are some of the most frequent examples of this tendency. In those services, there is no employee to help clients find information or buy something. Instead, actions such as suggesting similar information or items are integrated in the technology (the user interface) as "features". Gallouj and Weinstein (1997) noted that in the case of automated service, *"interfaces or front office technologies"* supply *"certain [service] characteristics directly to the customer"*, instead of them being delivered by an employee.

However, while the benefit in terms of efficiency is clear for enterprises, the issue of service effectiveness remains open. Of course, one could state that this issue will always be open, since the criterion of effectiveness is the rather difficult to define one of, *client satisfaction*. What appears as a paradox is the expression of the fact that, as Chase (1978) has observed early on about employees delivering services, *"interaction with the customer makes the direct worker in fact part of the product and therefore his attitude can affect the customer's view of the service provided"*. Jougleux (2006) argued that, even if marketing methods can provide a clear image of clients' expectation, not much is known about the means to effectively provide this service. Codello-Guijarro et al. (2011) found that employees in the front office often deliver more than a single service, while a collaboration both at the front and the back office are necessary for improve service effectiveness. citetNormann1991 used the term *"the moment of truth"* to describe the situation where an agent and client are face to face and service is about to be delivered. Gallouj and Weinstein (1997), based on the work of Lancaster (1966), remarked that service delivery is also dependent on a client's competencies, while, particularly in what regards information services, Delaunay and Gadrey (1987) (p. 185-189) also highlighted the importance of *"interactivity"* between service consumer and provider *during the delivery* of a service, the rules that characterize this interaction and the consideration of the social situation of the two.

Drawing from the case of banks' databases, Delaunay and Gadrey indicate a series of problems in the process of service delivery:

- joint definition with users of data input procedures and need and practice relevant references,
- qualification of the individuals operating this, often not codified, procedure for the users,
- consultation modes that take into account adaptation services;
- global knowledge of the procedures and of the model of function,
- system flexibility to deal with the arrival of unexpected data or new requirements.

For the authors, service delivery implies a relation between humans (as opposed to relations mediated by objects). They also add that service relation cannot be flattened in a relation of technical information management. Nevertheless, Gallouj and Weinstein (1997) highlighted some cases demonstrating the feasibility of user interaction automation, the example of the *ATM* has already being one of them. :

> *Interfaces or front-office technologies, mobilised by the service provider, by the client or, more generally, by both at the same time, supply certain service characteristics directly to the customer, and in that respect have something in common with the internal technical specifications of goods. Home banking is undoubtedly the archetypal example of this scenario, in which all the customer has to do is 'press a few buttons' to obtain the service he or she requires (Gallouj and Weinstein, 1997, p. 543).*

Yet, in their study they consider it an exception and do not advance their analysis further. Moreover, while 'home banking' is presented by the authors as an exemplary case of automation, in the terms of Simon (1965), where a client can obtain the service required in a few clicks, home banking is found to be home of innovations, as well. Von Hippel (1998) examined the case of a specific type of interface, *Application Programming Interfaces (APIs)* of computer-telephony integration, which *"refers to a field of specialized computing applications that draw upon both computing and telephony functions to accomplish a task"*. These APIs permit one:

> *to incorporate basic telephony functions such as "answer phone" or "transfer call" in their programs in the same way that they incorporate traditional computing functions such as "add" or "create a file".*

Unlike ATMs, these interfaces were addressed to developers, for them to create their own banking applications. Oliveira and von Hippel (2011) returned to the field of e-banking, studying applications created with Web-based APIs, this time, finding that developers had invented many new services, in a similar manner.

In the field of the Web, similar interfaces have been identified. Returning to the example of *eBay*, Suarez and Cusumano (2010) refer to its need for complementary services ("delivery services and secure payment methods"), in order for the platform "to function properly". Iansiti and Levien (2004) mention *eBay* as "a good example of a keystone company that effectively creates and shares value with its *ecosystem*". They also note the existence of additional tools addressing to the end-users:

> *[eBay] has developed state-of-the-art tools that increase the productivity of network members and encourage potential members to join the ecosystem.*

Among these tools, exists one which “helps new sellers prepare professional-looking online listings” as well as another which “tracks and manages thousands of bulk listings on home computers”.However, the role of the external *eBay* developers and their relation with the service provider is not discussed in the above mentioned studies, as their analytical framework focuses on two actors: *eBay* and its end users (the latter group split in two: buyers and sellers). Still, 25% to 30%[9] of items listed in the service do so by using third-party tools and applications[10]. *eBay*'s “community of developers” counts more than 100.000 members who have created over 13,000 active applications, according to the enterprise[11].

Overall, the service literature has shown that there is an inherent “element of the unknown” in service delivery, usually covered by the employees skills during his/her interaction with the client. Automated services - such as e-banking or eBay - have implemented user interfaces rendering service delivery an issue of bounded rationality (March, 1978), and enabling a multiple choice user interaction (“clicks”). Still, in many services - such as e-banking or eBay - there appear other types of interfaces, for other types of actors, the developers, who are not about choosing (“clicking”), but about creating additional automatic services, or applications. This early sign in the literature, may imply that in such services the place of the employee has been taken by a developer, who addresses the potential cases of service delivery which are not included in the initial options. This possibility will be explored in the current part of the thesis, aiming at the identification of the Web-based application development's modus operandi.

[9] According to statistics published to the official Ebay Blog, addressed to the eBay developers community, the percentage of the listings coming through third party applications were 25% in 2007 and 30% in 2009. Unfortunately, such statistics are quite rare for Web platforms.
Source 1: Ebay Blog, Blog: Certified Provider, “Last week's CP Fair in Salt Lake City”, Mar.21.2007. URL retrieved on the 11th of August 2011.
http://developer.ebay.com/community/blog/?category=Certified+Provider
Source 2: Ebay Blog, News Blog: Product News, “We are 9! The 9th Anniversary of the eBay Developers Program”, Dec.02.2009. URL retrieved on the on the 11th of August 2011.
http://developer.ebay.com/community/blog/default.aspx?category=Business+News See also

[10] We should note that data on the traffic or the use of a service due to third party applications are rarely communicated by service providers. Most commonly, such statistics are announced either during conferences or on blogs and forums addressing to the developers' community.

[11] Ibid.

Chapter 4

Methodological approach

Contents

In my work, I will use an exploratory, qualitative approach, studying multiple cases and utilizing multiple sources to triangulate my contributions.

4.1 The scope of case studies methodologies

Many case studies in Management Science have been based on the contribution of Eisenhardt (1989), who in turn built on Mintzberg:

> *No matter how small our sample or what our interest, we have always tried to go into organisations with a well-defined focus - to collect specific kinds of data systematically (Mintzberg (1979, p. 585) cited in Eisenhardt (1989, p. 536)).*

Another approach widely used by management scholars[1] is the one put forward by Yin (2003), defending a richer proposition for case studies research. As Yin put it, case studies are the preferred strategy when "how" or "why" questions are being posed, when the investigator has little control of the events, and when the focus is on a contemporary phenomenon with some real-life context (Yin, 2003, p. 1). For him, case studies constitute a comprehensive research strategy, by relying on multiple sources of evidence, with data needing to converge in a triangulating fashion, while they benefit from prior development of theoretical propositions to guide data collection and analysis (Yin, 2003, p. 14).

In the upper side of Table 4.1, we see how Yin positions case studies within the broad spectrum of research methodologies. For him, the questions starting with "who, what where,

[1] During the discussion of my work with members of the academic community (conferences, journals), reviewers would often suggest the use of the methodologies proposed by Yin (2003).

how many, how much" are better addressed by other methodologies, namely surveys or archival analysis. Case studies are most appropriate for answering questions about "how" or "why", like history studies. Still, the latter focuses on past events, while case studies focus on contemporary ones.

	Strategy	**Form of Research Question**	**Requires Control of Behavioural Events?**	**Focuses on Contemporary Events?**
Yin (2003, p. 5): Relevant Situations for Different Research Strategies (quoting COSMOS Corp.)	**Experiment**	how, why?	Yes	Yes
	Survey	who, what, where, how many, how much?	No	Yes
	Archival Analysis	who, what, where, how many, how much?	No	Yes/No
	History	how, why?	No	No
	Case Study	how, why?	No	Yes
	Genealogy	how, what, where, why?	No	Yes

Table 4.1: The genealogical situated in comparison to the typology of Research Strategies as outlined by Yin (2003).

Yin (2003) summarized his proposal on the relation between research questions and methodologies as follows:

> *In general, "what" questions might either be exploratory (in which case any of the strategies could be used) or about prevalence (in which surveys or the analysis of archival records could be favoured). "How" and "why" questions are likely to favor the use of case studies, experiments or histories (Yin, 2003, p. 7).*

At this point, let us return to my own research questions:

Q. 1) What are the specificities of "Web 2.0"? This question addresses my research goal to model its distinctive characteristics.

Q. 2) What is the genealogy for the Web 2.0? This question addresses my research goal to identify its dynamics in time.

Q. 3) How do we manage those platforms? This question addresses my research goal to identify management practice peculiarities.

From those questions, only the last one may be explored by using a case study methodology, as described by Yin (2003). According to his typology, the first question is to be explored using any methodology (as is an exploratory question), while the second one does not enter in the categories proposed: we propose the study of the history while focussing on contemporary events, a *genealogy*. In fact, *Q. 2)* is a necessary "bridge" between *Q. 1)* and *Q. 3)*.

As my research questions complicate the review of methodologies, I should note that researchers are often troubled when facing new phenomena in what regards the methodological issue. In fact, Mintzberg himself questioned the utility of research questions at all when studying strategy, practically receding from his 1979 claims, as used by Eisenhardt (1989):

> *Asking a right question in strategy is analogous to an explorer's finding his or her bearing before starting the journey. There is no standard methodology for coming up with questions: intuition and experience play far too important a role in the process (Mintzberg et al., 2003, p. xii-xiii).*

However, giving up the research questions is not considered in the current thesis. Intuition has played indeed an important role, especially in its early phase, a fact that is difficult to formalize. On the other hand, experience, while also difficult to model, was programmed and organised.

To summarize, I am going to use a case study approach in the third part of my document, exploring the *Q.3)*. There we will further analyse our methodological approach utilising recent academic propositions on phenomenon-based research(von Krogh et al., 2012) to study three "tools" for User-Developer-Entrepreneur collective action management.

In order to design a research methodology for *Q. 1)* and *Q. 2)* I aim to review the literature on research methodologies on surveys and archival analysis, as proposed by Yin (2003). After identifying their limits, I will review less popular methodological approaches, taking into account researcher's experience and the genealogy of the problem.

4.2 The scope of surveys and archival analysis

Surveys are often used in Social Studies for further understanding a question. In the particular case where categories are to be the output and not the input, questionnaires or semi-directed surveys and interviews are proposed by the literature. As Spector (2001) summarizes, such studies aim at investigating the causes of or the responses to an effect (Spector, 2001, p. 15). Interviews are preferred over surveys in cases when in-depth study is required, where questions may be more open, though demanding more time.

Archival analysis refers to the study of archives, of an institution or an organisation. Archives are typically filed according to some existing categories (administrative or other), which can be useful to the researcher applying quantitative methodologies.

However, information technologies have brought about another type of archives, the *traces of online communication*. Thus, Benkeltoum (2008) has used online surveys and chat interview methods, to enable the evaluation of open source software projects by a dispersed developer community, while Chanal (2004) has used emails and texts to study organisational innovation through employees communication. Mailing lists and online forums have been used as research material by a number of researchers studying online communities (von Krogh et al., 2003b; Bourreau and Gensollen, 2004; Cardon and Delaunay-Téterel, 2006b; Haefliger et al., 2009; Fayard and DeSanctis, 2010, and others). In this latter methodology, the researcher has the advantage of studing this field without influencing his/her sample, though *seeing without being seen*.

From surveys to semi-directed interviews and from archival studies to *traces of communication* researchers gain in scope, though a greater effort of *a posteriori* formalisation is required.

Still, in the current study this appears to be a challenge: while previously mentioned studies were able to limit the scope of their sample through organisational boundaries (be it a specific community or an enterprise), how might we determine the sample of our study in an emerging business environment?

To answer this question, we need an additional methodological levier, **the study of the field through an *object-discourse* approach.**

4.3 An *object-discourse* approach: studying an emergent field

I describe the methodology I will apply in the current research as an *object-discourse* methodology. The focus of my research will be on the novel technical substrates used by the developers of third party applications and the relative discourse of the actors on these technologies, as well as the resulting applications.

My approach resulted from the fact that, during semi-directed interviews, interviewees 'clearly' identified the peculiarities of the field, although they used their own, technical language to express it.

For instance, the centrality of interfaces for the development of third party applications (*Application Programming Interfaces, APIs*) has been a recurrent issue in our interviews both with UDEs and platform providers. An exemplar case of an enterprise extensively using *APIs* for innovation has been *Twitter*. Here is how a *Twitter* engineer described the emergence of their platform[2]:

> *Once the API was out there, for several months it was really just a Web site added feature – API added feature, Web site added feature – API added feature. (...) But I think the nice thing that Twitter has helped out with is that almost every site now has an API. And Twitter is a good example in this, it enables innovation around the platform, people adding value that you just don't have time to add if you're a five person start-up*[3].

As suggested by the last lines of the above quotation, the use of *APIs* seems to play an important role in innovation, value adding and growth.

Nevertheless, the recursive method of *"Web site added feature - API added feature"*, also expressed by *Google Developer Relations Engineers* as the fact that in *Google*'s case *"there is always going to be a steady stream of APIs, just as you have a steady stream of products"*[4], outlines a trend which is not easily described by Management terminology. While a "stream of products" is easily understood by the New Product Development literature, the meaning of a "new stream of APIs" needs to be further explored.

In parallel, the actors of the 'milieu' highlight the importance of specific technologies, often developed or advanced by others, though commonly used by developers to innovate through the use of such "APIs".

Here is how a *Mozilla* executive described the increased ability of developers innovate today[5]:

[2]Interview taken on August 13, 2010. San Francisco, California, USA.

[3]Interview taken on 10/08/2010.

[4]Interview taken on August 10, 2010. Mountain View, California, USA.

[5]Interview taken on August 12, 2010. Mountain View, California, USA.

For me the most important bit is that modern APIs are really easy to consume. I think the more exciting thing that happens today with the APIs is that they are either JSON or RESTful. And both ways you can really easily manipulate. So, you don't need to be a C++ developer, as you need to be for a platform API like Linux. You can write your stuff in HTML and Javascript and you can write mashups in Javascript and you don't need to learn new techniques and system-level knowledge. Which I think is the most interesting think about it. This is super-powerful because it enables more and more people to build [their applications].

From a management research standpoint, one can understand from the above quote that there is something that excites the actors of the field, regarding the capacity of developers to innovate, and that is related to the use of some programming languages instead of others.

Still, what does this change in technical terminology mean for business? Can it be related to the tendency to move from products to services (Cusumano, 2008)? Do these technical terms "hide" some kind of answer to the current debate on services platforms (Gawer, 2010a)?

The study of conversations has already been a common practice for Management scholars within an organisational context, enforced by the extensive use of information technologies in organisations, rendering conversations readable by researchers *a posteriori* (Krogh et al., 1994; Adam and Murphy, 1995; Vaast, 2003; Chanal, 2004, and others).

However, while these studies have focussed on conversations within an organisational discourse - the latter based on older or emerging task divisions - in my study the discourses concern the definition of a new, complex object by those who use it to innovate. These actors do not belong to the same organisation, though they share a common discourse of the milieu.

Hence, in order to understand the meaning of the discourse, I will have to study the object itself: **as the discourse is not limited in organisational boundaries, I propose that the best way to delimit it is through the *object* of discussion.** From this perspective, the technical language would normally be considered by management scholars as a *problem*, in my study I consider it as the *solution* to a *challenge*.

The challenge is to identify and characterise the field problem (Hatchuel and Molet, 1986), common to different actors engaged in a great variety of institutional settings within the same business environment. To this aim, I will use as an entry point the new technical language used by the developers to describe their activity.

Hatchuel and Weil (1992), studying the implementation efforts of *expert systems* in different industrial settings, conclude to a prediction about the emergence of the technical language within the enterprise, as a result of further rationalisation of information technologies:

Given the complexity and heterogeneity of knowledge [amongst autonomous divisions], communication is difficult without a base of shared knowledge and without specific efforts to maintain this common base. (...) [In the future] We will probably talk less in terms of expertise, knowledge or imitation as long as more abstract and more technical concepts appear that will simply say that we know how to treat some specific information structures (Hatchuel and Weil, 1992, p. 150).

What I observe in my study is a common technical language, shared among developers in different countries and enterprises (mainly in France and in the United States and in start-ups as well as large Web services providers). By reversing the argument of Hatchuel and Weil (1992), I use this technical language as the expression of a common knowledge among the experts of the 'milieu'. Then, I perform an "interpretation" of this knowledge in business terms, based on an extensive review of on New Product and Services Management.

4.3.1 The use of the object-discourse methodology to reply to the hypothesis of "Web services platforms"

The contemporary discourse of User-Developer-Entrepreneurs is analysed on the basis of a concrete case study. Consequently, the open question of the relation between today's online services platforms and the traditional ones, identified by the literature, will be explored on the basis of a case of a collective design of an online market **??**. The discourse on Web services architecture will then be analysed by the constitution of a "dialogue" between the literature and the actors of the field on the issue of Web platform architecture.

Similarly, the question of the transition from products to services, raised through a market study by Cusumano (2008), will be in-depth studied by a concrete case, in Section **??**. The discourse in this case will regard the specific technologies that enable the Web to be a platform for services, corresponding to the need for a constant information flow exchange (and not a static Web, as it had been the case before *Web 2.0*). Hence, we find that the "new words" for developers correspond to a new environment for online business, opening the way for the expansion of Web-based services.

4.3.2 An historical perspective: the role of UDEs in industrial development

In Part II of my study, I will use a genealogical approach, on the basis of the problematization regarding the specific role of UDEs in industrial development.

To this regard, I borrow Foucault's notion of genealogy to trace the history of the thought leading to the knowledge and the concepts appearing today as Web 2.0 peculiarities, responding to the question "how can knowledge be constituted" (Foucault, 1984) [6].

In this task, I will analyse at the same time the objects and the discourses of those developing or using them while they appear in history, concentrating in describing their differences, transformations and mutations (Kendall and Wickham, 1999; Foucault, 1969). Practically, I will be studying the discourse of the two notions, collective intimacy and computers, as opposed by previous disciplines and as connected with emerging ones.

On the one hand, I will study the emergence of "collective intimacy settings", focussing on the coffee-shops institutions, identified by a series of scholars as the foundations of what I

[6] Foucault typically mentions:

> *Histoire de la pensée, ça veut dire non pas simplement histoire des idées ou des représentations, mais aussi la tentative de répondre à cette question : comment est-ce qu'un savoir peut se constituer ? Comment est-ce que la pensée, en tant qu'elle a rapport avec la vérité, peut avoir aussi une histoire ? Voilà la question qui est posée (. . .) [Problématisation] est l'ensemble des pratiques discursives ou non discursives qui fait entrer quelque chose dans le jeu du vrai et du faux et le constitue comme objet pour la pensée (que ce soit sous la forme de la réflexion morale, de la connaissance scientifique, de l'analyse politique, etc.) (Foucault, 1984).*

call collective intimacy. On the other hand, we are going to study the emergence of computer industry, to which Web 2.0 platforms services and their developers are genealogically bound.

Reviewing the collective work "Foucault, Management and Organisation Theory" (McKinlay and Starkey, 1998), Hatchuel (1999) and discussing organisational ideals, such as "autonomous teams" or "democratic dialogue", proposes the following teachings of the works on Foucault's investigation method:

(a) understand the genealogy of the rationalization forms which produced the disciplinary regimes of corporate history;

(b) think of the forms of rationalization which correspond to values that are now important to us, while remaining attentive to the disciplinary forms they prepare - in other words, by trying to ensure that the invitation transmitted to Dr. Jekyll does not result in the arrival of Mr. Hyde —;

(c) avoid building up relations and knowledge which leave others with the figure of Mr. Hyde as the only feasible way of constructing themselves as subjects (Hatchuel, 1999, p. 518).

Hence, in our particular case, I will investigate the genealogy of rationalization forms that produced the various disciplines in the regimes of computer industry, as well as the forms of rationalizations corresponding to the value of collective intimacy.

Additionally, when tracing the genealogy of Web 2.0, we will try to respond to the question of the role of User-Developer-Entrepreneurs. Hatchuel (1999) designs two conditions for the study of the "forms of rationalization":

> *First, the consolidation of a set of "actor figures" (Hatchuel and Weil, 1995): making places, relations and identities visible through discourse and practices; secondly, says Foucault, these "figures" must act as forms of "subjectification", they must become the truth through which actors perceive themselves as "subjects".*

In this study, responding to the first condition, we will be investigating the consolidation of the set of the UDE actor figures, their making places (such as clubs or associations), the relations and identities that become visible through their discourse and practices (scientists, entrepreneurs, hackers, users). As I have defined the collective intimacy as the place where the "true self", the truth about subjectivity is expressed, tested and experienced, the study level of these actors will be the historical configurations of their collective intimacy. Thus, we will less focus on their family life or their enterprise tasks, unless expressed within the collectivity in question.

Chapter 5

Indications on a novel modus operandi : the discourse of service providers

Contents

In this chapter, using semi-directed expert interviews I trace three indications of possible contemporary Web services peculiarities, regarding the means for value exploration and the actors engaged in the process of service creation, through the reconstitution of the platform providers discourse. I conclude with the indication that the originality of Web 2.0 services lie in the parallel exploration and exploitation of the service while in the market. This original development mode seems possible only through a synergy between platform providers and user-developer-entrepreneurs, enabled by the provision of specific interfaces for application development.

5.1 Methodology: managers interviews

I trace the first intuitions (Mintzberg et al., 2003) on the peculiarities of a possible phenomenon to be distinguished (von Krogh et al., 2012), the *Web 2.0*, using a set of semi-directed interviews of experts, product developers or community managers for large Web platform providers. The "open-ended" question (Eisenhardt, 1989) I am exploring is **"what are the specificities of "Web 2.0" platforms ?"**. The question being broad, interviews are semi-directed, leaving space for interviewees to themselves indicate the field of possible answers. Borrowing from the article of Eisenhardt (1989), *"Building Theory from a Case Study"*, I do not assume any hypothesis or theories at this level. I enter the field by *"overlapping data and analysis"*, being limited here to a single *"data source"*, experts interviews, and I identify the common views of the experts on the specificities of the phenomenon.

Position	Enterprise	Date	Place	Duration
Technology Advisor	*SUN Microsystems*	19/07/2007	Paris, France	46 min.
Product Manager	*Yahoo!*	25/07/2007	Paris, France	44 min.
Product Manager	*Netvibes*	17/08/2007	Paris, France	57 min.
Product Manager	*Google*	02/05/2010	Paris, France	27 min.
Developer Relations Manager	*Google*	18/05/2010 - 19/07/2010	Online	2 months
Labs Director	*Orange*	10/08/2010	San Francisco, USA	1h 44min.
API Developer	*Twitter*	10/08/2010	San Francisco, USA	40 min.
Labs Director	*Mozilla*	12/08/2010	Mountain View, USA	55 min.
Platform Architect	*eBay*	13/08/2010	Mountain View, USA	32 min.
Developer Relations	*Google*	14/08/2010	Mountain View, USA	55 min.
Ubuntu Product Manager	*Canonical*	3/10/2010	Paris, France	56 min.

Table 5.1: The discourse of platform providers. Interviews taken.

Starting from a pragmatic, evidence-based entry, I compare the outcomes to a *"broad range of literature"* (Eisenhardt, 1989, p. 544). I then formulate three indications, using the evidence provided from the field, to be further explored in the following sections.

In the Table 5.1, I list the interviews undertaken and analysed. My interlocutors are managers engaged in product management processes. The sample of our interlocutors is taken from the three models: the open source model (*Mozilla, Canonical*), enterprise paradigm for companies that existed before the Web (*Orange* and *SUN Microsystems*) and the Web 2.0 case (*Yahoo!, Google, eBay, Twitter, Netvibes*). To enable a peculiarity distinction (von Krogh et al., 2012) for contemporary Web services to emerge out of this material, my analysis is undertaken three to six years after the interviews, after having being exposed as a researcher, in the meanwhile, to additional sources of information.

In the next Section I present the outcomes of those interviews, identifying some early intuitions of the field actors regarding what can be a business phenomenon to distinguish. At this stage of my research, I adopt an exploratory posture, since the very existence of a particular phenomenon contained in what was discussed under the term *Web 2.0* is not yet proven. The field intuitions found in the interview material are recognised *a posteriori* and used as early elements for the construction of my thesis' argument, to be further explored in the following chapters. Then, by discussing those intuitions, I identify three indications of the peculiarities of a possible phenomenon regarding business, as compared to other community or enterprise-based innovation models (Raasch and von Hippel, 2012; von Hippel and von Krogh, 2006, 2003) the latter two models having already been reviewed in Section 3.3.

5.2 Outcomes: "open products" and "developers-entrepreneurs"

The *"Web 2.0 wave"* was seen by platform providers themselves as an encompassing phenomenon, including three dimensions: the value eventually served to end-users, the technologies enabling third parties to innovate with this service and the figure of the third parties themselves. Typically enough, a *SUN Microsystem* expert summarized the phenomenon as follows:

> *For us, we reckon that the Web 2.0 is at the same time a set of services, a set of communities, and the overall set of tools enabling the interaction of these communities, either at an internal level or between different communities.*[1]

The above quotation is typical of the discourse I met, the latter characterized by the omnipresence of the term "community" to describe both the services, the related technologies and the interactions among the stakeholders. For most of my interlocutors, while elements of both the technology and the social interaction modes had already been there, the rapid expansion of these services to a wide public of users rendering it "easy to share on-line", has modified the quality of both the technical substrates and the relationships of end-users, ultimately diffusing a "communitarian" relational value. This approach joins the academic

[1] Translated from French. Original quote:

> *Pour notre part on estime que le Web 2.0 c'est à la fois un ensemble de services, un ensemble de communautés, et l'ensemble des outils permettant à ces communautés d'interagir, soit en interne au sein de ces mêmes communautés, soit d'une communauté à l'autre.*

Interview taken on July 19, 2007.

discourse on Web services platforms, which investigates the use and the different community and personal expression settings explored (Haefliger et al., 2009; Ebner et al., 2009; Gensollen, 2003; Cardon, 2008; Cardon and Delaunay-Téterel, 2006a).

Moreover, as I adopt a perspective of industrial development (outlined in Section **??**), my study will focus here on the level of "those who build the service", rather than the one of the end-users. Hence, I consider the end-user value from the perspective of those designing the service - as a design requirement (Suh, 1990), not from the ones who consume it.

Explicitly, despite the activities divergence between the different actors interviewed, I identify two peculiarities of contemporary Web service platforms to which experts' views emerged:

1. The *central place* of *Application Programming Interfaces (APIs)* for the creation of new services.

2. The role of User-Developer-Entrepreneurs as *important actors* in the creation of new services by extending greater ones.

These dimensions will be further explicated in the following paragraphs.

5.2.1 Technology: enabling third party application development

A second set of intuitions on possible peculiarities of Web services is to be drawn from the part of the providers' discourse regarding the technologies enabling third party innovation. Research in management has shown that there is a correlation between experts' common knowledge and technical language. In fact, Hatchuel and Weil, studying information systems conception within different enterprise contexts, presumed that the emergence of an abstract, technical language would substitute the debate on experts knowledge (Hatchuel and Weil, 1992, p. 150). In my research, I followed a reverse reasoning: I used the technical language utilised by the providers to enter their debate on third party innovation enabling.

Studying the first "boom" of Internet business at the end of 1990s (which eventually lead to the so called "dot.com bubble"), Cusumano and Yoffie, in their widely respected study, highlighted the importance that the "battle of browsers" (between *Microsoft* and *Netscape*) had not only for those business actively engaged in online services at the time, but for *"Competing in Internet Time"* in general:

> *We are not exaggerating when we say that the Internet and the World Wide Web, with the browser as its user interface are revolutionizing mass communications, as well as mass networking technology (Cusumano and Yoffie, 1998, p. 2).*
>
> *For companies competing in the new information economy, the Internet is forcing managers and employees to experiment, invent, plan and change their ideas constantly while they are trying to build complex new products and technologies (Cusumano and Yoffie, 1998, p. 5).*

If the latter argument of Cusumano and Yoffie about the Internet *"forcing managers and employees"* to adapt to a faster rate of innovation is generally accepted today, the former one, regarding the infrastructure should be given the same attention.

The "revolution in mass communication" was based, according to the authors, on the potential which Web browsers opened up: through those devices and by the use of their interfaces, a great public was enabled to access information, hosted at an enormous number of Web sites. The early era of Web business was then deployed through the development of commercial Web sites (recognised by their ".com" suffix).

Regarding this very potential, the interview I had with with a *Mozilla Labs* director, questioned the above mentioned perception on the value of Web browsers:

> *What came out the Web 2.0 for me is the nearly omnipresence of APIs. You've got APIs everywhere. And that's really interesting because it allows you to do things very very differently. It's interesting because, I think, one of the manifestations of APIs are now apps. Because an App consumes an API and presents the data in an non-web format. It's a native App. But it consumes Web data. So, APIs is a really interesting topic. And I see companies today, before they build the service, they build the API.*

This statement from a *Mozilla* executive suggests a radical change. To further grasp the content of this statement, we have to enter the underpinning technical knowledge. While in the period studied by Cusumano and Yoffie browsers were the unique means to access Web-based information for the broad public, our interlocutor talked about applications ("Apps") as an alternative for end users. Consequently, what is indicated is that *Application Programming Interfaces (APIs)* are presented, thus, as a way for the service provider to enable the creation of these Apps by third parties. While in the early period, that studied by Cusumano and Yoffie, the crucial part of competition was between the major browsers (*Microsoft Explorer* and *Netscape*, the latter becoming *Mozilla Firefox* afterwards), today the corresponding part is alluded to the competition of a Apps multitude galaxy. To the extent that platform providers use *APIs* to distribute information, end users access it ("consume it") not only through the browser, but by a multitude of "Apps", applications that can be "native", that is proper to specific devices connected to the Internet beyond the PC (such as *smartphones*, *tablets* or other devices). Hence, while the early Web was characterized by the "omni-presence" of browsers, today's "omni-presence" of APIs is said to change the business environment.

Along these lines, while in the first phase (the dot-com era), of which the conditions were described by the study of Cusumano and Yoffie (1998), innovation trajectories regarded "Web pages" (web sites with the suffix *.com*), the era of the *Web 2.0* signified the opening of an innovation trajectory through the "Apps".

Regarding this suggested shift towards Apps, during the interviews I found signs that they constitute both a continuity and a rupture with the computer industry history, eventually shifting innovation outside the boundaries of the enterprise. Characteristically, a *Canonical* product manager commented:

> *Typically, we have been developing APIs that were accessible only on the level of a single computer. Then, we were enabled to render those APIs usable on a computer network, though a local one (...) Afterwards, there was a guy that came and said "that is stupid, we already have a communication language between different elements, it's named HTTP, which is already in use by security interfaces, why not simply re-use the same way of interaction as HTTP, that is by passing the orders in the request and simplify the most possible".*
>
> *And with that, we arrived at an API that really exited the walls of the enterprise and passed to APIs usable on the Internet[2].*

[2]Translated from French. Original quote:

What I keep from this quote is the insight that the APIs, and consequently Web based Apps, were inherited from the history of computers by successive changes in both the technologies and the practices used and that this process resulted in the ability to "exit" the boundaries of an IT enterprise in respect to the innovation process in the Internet economy.

In other words, what is suggested above is a coupling between the historical evolution of Web substrates with the historical evolution of the enterprise leading to what was characterized as an Open Innovation era (Chesbrough et al., 2006), in the particular domain of computer industry.

Hence, I can formulate the following insight, to be developed, concerning the importance of the evoked shift towards "Apps" and the technologies that enable it (the APIs):

> **Intuition 1.** *The shift from the Web browser to Apps as user interface for accessing Web-based information, enabled by the provision of APIs, is a shift of a historical value and has major implications for innovation.*

Advancing my exploration of the providers' discourse insights, and trying to understand the nature of these "APIs", I identified a case frequently cited in the "milieu" of developer entrepreneurs, which highlights the value of this "substrate": the *Twitter API* and the way it helped the company grow. Here is how an *eBay* platform architect commented on the case of *Twitter API*:

> *They certainly invested into APIs because they seem to, I don't know . . . but that's what it seems they do. They have got a lot of great APIs that they come out with and as a result you saw a lot of third party applications and, you know, I used to read this, that 2/3 of all access at Twitter is done through APIs.*

The above estimation indicates that *Twitter API* worked as an exemplary case for the developers of the field, causing a discussion on its features and its success. Like in open source communities (Lakhani et al., 2002; Zeitlyn, 2003; Osterloh and Rota, 2007, and others), reputation on technical achievements circulates within the milieu. However, unlike open source, one cannot *know* the details of the technology and its use (e.g. the platform source code). Hence, reputation seems to result out of "rumour spread than by exact knowledge of the inner nature of a platform.

In *Twitter* itself, the invention of the first *API* has taken the form of a *"rational myth"*, an *"action model allowing the mobilisation of the organisation on the basis of an objective in which the actors believe (myth), though of which the formulation and objectives are realistic and adaptable (rational)"* (Hatchuel and Molet, 1986). Such a "myth" was built on the experience of the first API, expressing the memory of the rationalisation of a new mode of

---

Typiquement c'est que pendant longtemps on a développé des APIs qui étaient accessibles uniquement au sein d'un ordinateur. Et puis en suite, on s'est permit de rentre ses APIs utilisables à un réseau d'ordinateurs mais en local (. . .) Puis un gars qui est arrivé en disant « mais non, ça c'est quand même stupide, on a déjà un langage de communication entre différents éléments, qui s'appelle HTTP, qui a déjà des interfaces de sécurisation, pourquoi ne pas tout bêtement réutiliser la même façon d'interagir que HTTP, c'est-à-dire en passant les ordres dans la requête et on rendre la chose la plus simple possible.

Et là on est arrivé réellement à une API qui est sortie des murs de l'entreprise, et passée à des APIs utilisables sur Internet.

innovation management not only for *Twitter*, but the entire emerging sector. Here is how a *Twitter API* developer, hired when *Twitter* was beginning to grow, described this experience:

> *You know, in the beginning we put an API out. It was basically what the Web site had. So we had a Web site, by virtue of choosing the Ruby Rails technology we kind of got an API more or less for free,* [3] *and it was sort of a side project first as I understand it, I wasn't here, and took off, got attraction, people started building alternate desktop clients*[4] *and that kind of spread the API as a thing (...) So, once it was out there, for several months it was really just "Web site added feature - API added feature, Web site added feature - API added feature".*

Twitter API is thus discussed as a platform feature allowing others to build upon their service. The attraction of individuals capable to do that is referred to as another factor of success.

In the same spirit, a *Yahoo* product developer summarized the value of *APIs* for the enterprise as follows:

> *To deploy, popularise, extend the functionalities of Yahoo to third party services*[5].

For the *Yahoo* product manager, as well as the others interviewed, the API mode is a way to render the service more popular, but also to enrich its functionalities, while the service in the market. From the above, we have some further insights on the characteristics, to be further explored:

- There seems to be a "mirror-relationship" between information provided by the Web site and those provided by the API, as their relation is commented as a reflective one (*"Web site added feature - API added feature, Web site added feature - API added feature"*);
- APIs are about *"attraction"* of *"people starting building"* upon a specific platform;
- An important challenge for an API is *"to spread"*, to be adopted by others;

I am going to further explore those insights in the discourse of platform providers in the following paragraphs, starting from the first point.

A "mirror-relationship"

Beginning from the first point, an *overlapping field between the different information interfaces* is suggested, also supported by other platform managers interviews. Matching features of Web sites to APIs seemed to be a common concern of all platforms. Our *eBay* platform architect has been very enlightening on this regard:

[3] One of the first engineers employed by *Twitter*.

[4] A desktop client is an application for a desktop computer that receives and displays information from the Internet. Browsers or the "Fetchmail" application (Raymond, 1999) can be considered desktop clients, as well.

[5] Translated from French. Original quote:

> *Déployer, populariser, étendre les fonctionnalités Yahoo sur d'autres services tiers.*

> *If there's a Web page on eBay, we will probably create an API that does something similar. So, you have My-eBay where you can see things you are bidding on, things that you're watching, things you're selling. So, we'll create an API that gives you all the same data.*

For *eBay*, where users bid for items in an auction, information provided by the APIs is on this particular activity: online selling and buying goods by auctions. Hence, generally, what is evoked above, is that the same information provided on the Web site (on *"things you are bidding on, things that you're watching, things you're selling"*, is to be provided by the API as well. This insight can be expressed as a "mirror-relationship" between the Web site and the API: both provide information on the activities of the same service.

Still, what is the use of an *API* if it provides the same kind of data? The answer is already suggested by the statement of the *Mozilla* executive: these substrates enable, in some way, the creation of applications - they are not to be directly used by end-users. So, there is something in the nature of these interfaces, as compared to the browser or the Web page, that makes third party innovation possible.

Hence, what is evoked is that APIs provide streams of information, already in the Web site, using the same categories of information as the site, for third parties to further develop the functionalities of the Web site through their own applications.

Another parameter of the assumed "mirror-relationship" between a Web site and an API is its reflective relationships: provider actions on the first one induce similar actions on the second one. Here is how a *Google* manager put it:

> *There is always gonna be steady stream of APIs, just as you have a steady stream of products (...) So for the past five years it looks like we've been releasing API at fairly steady base. So I would guess it will continue that way.*

Consequently, there emerges the suggestion that the *API* is a parallel mechanism to product development, at least as the latter has been studied by the new product management literature (Clark and Fujimoto, 1991; Midler, 1993; Brown and Eisenhardt, 1995), or by the two models proposed by Raasch and von Hippel (2012): this mechanism is attached to the product, shares common attributes with the latter, while it enables a synergy between the provider and third party developers on the further development of the product.

This close relationship between the final service, as perceived by the end users, and the application development interfaces, as used by entrepreneurs for the invention of applications, will be further explored in Chapter 7. A guide - to be tested - for this exploration will be the following intuition:

Intuition 2. *In contemporary Web services, there is a "mirror-relationship" between the service provided to the end users and the means provided to developers for the development of applications.*

Next, I will further look into on third party application building aspect of APIs.

"People starting building"

Judging from a more traditional enterprise perspective, the director of *Orange Labs* in San Francisco characterized the APIs as quite revolutionary in the new paradigm that emerged through the Web 2.0:

Offering APIs for others to come and use your data to manufacture other products (for example, look at Google Maps, it has been one of the examples) is a very original innovation development strategy, because you have people who work with your data, whom you don't pay, perhaps it's them who pay you a bit. And if you like the product, either you buy it, either you buy the firm - it is what often happens - or these people will create more value on your product with their developments. APIs have been quite revolutionary in this model[6].

Furthermore, a metaphor by the Canonical product manager for those devices helps further understand their particular nature of these devices:

The API is something like a plug. Not only the plug, but also the power, etc, but it corresponds to the way that it will interact with the one who provides us the power[7].

The metaphor of a "plug" has been used by the literature (Gawer and Cusumano, 2002) to describe cases such as the *USB key*, enabling the development of a "complementary" industry of computer peripherals, based on *Intel*'s microprocessors. This way, I trace once again a parallelism between the typical computer industry and Web services. However, in the former case, the common design rules (Baldwin and Clark, 2000) were configured on the basis of commonly recognised and standardized instrumental values (such as the calculating power of a microprocessor). Still, can one argue that the values related to personal identity development can meet the same "universal" recognition by the different actors of the business environment as do computer interfaces?

The comparison between the different use of APIs from *eBay* and *Twitter* enlightens this dimension. For *eBay*, *APIs* were used to enable innovation on a very specific field, already identified by the company's business model. The latter being user transactions (with *eBay* revenue coming from its mediation), APIs were focussed on the creation and development of tools for expert sellers, that is sellers beyond the average user, operating a higher volume of transactions. On this, the *eBay* platform architect was very clear:

We really focused most on APIs around sellers. There were things that we knew we weren't going to go after but we knew that there was a market there that could enable other folks. Maybe if we wanted to go after this market, maybe we wouldn't release the API so often, I don't know. It was a long time ago, hard to say.

[6]Translated from French. Original quote:

Offrir des APIs pour que d'autres viennent utiliser vos donner pour fabriquer d'autres produits (par exemple, regardez les Google Maps, ça a été un des exemples) c'est une stratégie de développement d'innovations très originale, parce que vous avez des gens qui travaillent avec vos données, que vous ne payez pas, peut-être que eux vous paient un peu. Et si le produit vous plait, soit vous l'achetez, soit vous achetez la boite - c'est ce qui est arrivé souvent - soit les gens vont valoriser vos produit encore mieux avec leurs développements. Les APIs ont été assez révolutionnaires dans le modèle.

[7]Translated from French. Original quote:

L'API c'est en quelque sorte la prise. Pas seulement la prise, mais c'est aussi la puissance, etc, mais ça correspond à comment est-ce que va interagir avec ce qui nous fournit de la puissance.

While the market for *eBay* seller tools was already identified and *APIs* came about as a result of a particular enterprise scope (Gawer and Cusumano, 2002) understanding, this was not exactly the case for *Twitter*.The development of the *Twitter API* was described as a more adventurous process, being transformed through the interaction of the enterprise with user-developers. *"Firehose"* was been the *API* that provided user-developers with a stream of "tweets", user posts in the service. This stream was the basic "material" for user-developers to innovate, by inventing new services. Here is how the *API* developer described the process:

> *So the first notion was, well we'll take the Firehose and we'll offer sampled feeds, so you'll get 1% of the tweets. So if I'm a developer at home - I think at that time the rate was maybe a hundred tweets a second - I get a couple of tweets a second. And that's pretty easy for me to handle. And then there are other ways to start creating more interesting slices of the full Firehose. "I want tweets that contain some key-words", "I want tweets that happen in a certain location or have certain hashtags", or from a specific list of people. So there we've really grown around what users are asking for.*

What is described above is a synergy between *Twitter* and user-developers on the exploration of the value of a "stream of tweets". The design of the *Firehose* had taken into consideration the capacity of a single developer working on his computer to process information (therefore they provided the 1% of the tweets). Still, this quantitative segmentation was not enough. Further, qualitative restrictions of the stream based on key-words regarding brand names, locations, topics of discussion (*"hashtags"*) and others were introduced for the "material" to be more relevant to emerging segments of the "tweets" market.

Enterprises like *Google* have also embraced the experimental, exploratory potential of this operation mode. Concretely, *Google* used the *Labs* concept to indicate the unstable nature of an *API*. Here is how a *Google* developer relations manager described this practice:

> *So by having that system [Labs] we can put out our API without having to worry about the three years compatibility. So that makes it easier for putting out APIs. If every time we have put out an API we had to, like, swear to having the same interface for three years and supporting it we probably wouldn't have put out as many, 'cause that's rather difficult.*

The more experimental the nature of the interface and the resources it provides for developers to innovate, the closer the interaction has to be between platform and developers to acquire a feedback and identify the specific use segments. At the same time, the less engaging this process can be for the platform provider, such as *Google*, as the value of a feature (and thus the necessarily resources invested in it) remains to be explored.

Building on the Intuition 2, I describe the peculiarities of the services under study by their double identity: being used by end user *as they are*, while being enriched by third parties *on the basis of what they are.*

> **Intuition 3.** *We can describe contemporary Web services as "Open Products" (Chrysos et al., 2010), embodying two characteristics:*
>
> 1. *They provide a standard set of uses for end-users.*
> 2. *They provide the means to further explore and develop the product to developers, based on the attributes of the initial set of uses.*

"Spreading" the API

Moreover, the management of the relationship with user-developers is another condition for this model to be possible. This dimension was expressed by the Canonical product manager as follows:

> *If I launch an API that is only usable for myself, there is no interest. After [the launch] there is the marketing problem for this API, hoping that there will be users that will begin to use it.*[8].

In the next section I am thus going to outline the findings regarding the management of the relationship with user-developers-entrepreneurs. For the moment, I can draw some additional working hypothesis, summarizing the part of the provider discourse on the importance of *APIs*.

> **Intuition 4.** *For an API to be successful, it has to be actively used by UDEs. This requirement implies the need for management of the particular relationship between platform provider and UDEs.*

5.2.2 Enterprise - UDE relationship

Beyond the technical conditions for a service to be extended by third parties, there is another condition highlighted above, the *intimacy* between developers and platform providers, something that is present in the Web environment though not similarly in the close, telecommunication sector. In fact, this model is different from the case of enterprise partnership, studied in different business contexts (Midler et al., 1997; Hagedoorn et al., 2000; Bozeman and Dietz, 2001; Segrestin, 2005, and others). The novelty comes from the capability provided to nearly everyone to use those substrates to innovate. Here is how a *Netvibes* product engineer outlined this dimension:

> *Two companies that work together to propose a product, that's not new. It always existed. What is a lot more novel, is these open APIs, which are given to everyone, which enable everyone to make mashups*[9]. *And the popularity of a service is also measured by its APIs*[10].

[8]Translated from French. Original quote:

> *Si je lance une API qui n'est utilisable que par moi il n'y a aucun intérêt, ensuite c'est le problème de marketing pour cette API, en espérant qui il ait d'utilisateurs qui se mettent à l'utiliser.*

[9]A *mashup* is an application using more than one sources of Web data (APIs).
[10]Translated from French. Original quote:

However, launching an API does not lead to an open product strategy at once. An *Orange Labs* executive commented on the effort made on behalf of telecommunication operators to use the same model:

> *Telecom operators all try to provide APIs on the networks, for people to be able to develop on the network. It works a little bit, not as we could have imagined, because actually there is no intimacy between the world of telecommunications and the developers, as there is an intimacy between the world of Web and the developers*[11].

As we have already shown in the previous paragraph, APIs are conceived taking into account the requirement of enabling a single developer to experiment (as in the case of the *Twitter Firehose API*). This requirement is the rule in *Yahoo* as well, user-developers can experiment without even asking the permission of the provider. This development process precedes the creation of the start-ups by the developers. As also described by the *Orange Labs* executive, the successful innovations have the chance to be acquired by the platform provider (and thus are not freely revealed, as in the case of the *private-collective model* in open source (von Hippel and von Krogh, 2003)). Here is how the *Yahoo* product manager described the purpose and the process of API innovation:

> *They [user-developers] create their mashup and use it in their corner . . . Generally, they don't ask our opinion. That's the reason why APIs were invented and distributed, for people to appropriate them and create their own services beginning from those APIs. So, there is not necessarily the need to come back to Yahoo and ask for a specific validation to advance. It is automatic, which allows it to go a lot faster rather than burdening the process, which would take longer.*
>
> *On the contrary, it's true that today we trust start-ups, providing them with APIs so that they can start up and [then we] begin making strategic partnerships with those start-ups that we estimate that will develop themselves very fast. There is a real policy today that goes in this direction. Specially in France, but also in general for Yahoo.*[12].

> *Deux sociétés qui travaillent ensemble pour proposer un produit, c'est pas nouveau. Ça a toujours existé. Ce qui est beaucoup plus nouveau, c'est ces APIs ouvertes, qui sont données à tout le monde, qui permettent à tout le monde de faire des mashups. Et la popularité d'un service se mesure également à ces APIs.*

[11] Translated from French. Original quote:

> *Les opérateurs de télécom essayent tous de fournir des APIs sur les réseaux, pour que des gens puissent développer sur le réseau. Ça marche un peu, pas comme on pourrait imaginer, car en fait il n'y a pas d'intimité entre le monde des télécom et les développeurs, comme il y a une intimité entre le monde du Web et les développeurs.*

[12] Translated from French. Original quote:

> *Ils créent leur mush-up et l'utilisent dans leur coin. . . En général, on ne nous demande pas notre avis. C'est la raison pour laquelle les APIs on été inventées et distribuées, c'est pour que les gens se l'approprient et créent leurs propres services à partir de ces APIs. Il n'y a donc pas forcément un réel besoin de revenir sur Yahoo et de demander une certaine validation pour avancer là-dessus. C'est automatique, ce qui permet d'aller beaucoup plus vite plutôt que d'alourdir les process, ce qui est forcément très long.*
>
> *En revanche, c'est vrai qu'on fait d'avantage confiance, aujourd'hui, aux start-ups en leur four-*

Hence, platform providers "trust" developers helping them start up. At the same time, what the "intimacy" between the two suggests in addition, that developers should also "trust" as well the provider. For a Google Developer Relations engineer, the biggest risk is losing trust from the user-developer community:

> *The big risk with APIs is probably losing trust, because when you're using an API you're trusting. It's kind of having like a reasonable uptime, cause you're taking part of your site and you're making it depend on an API so if that API starts breaking a lot, or it's flaky or gets slow you're gonna lose trust in there, right? Or if you're trying to use it you can't get any responds you file a bug and you never hear back you're gonna lose some faith.*
>
> *I think that's the big risk, right? 'Cause if you have a bad experience with a Google API you're probably not gonna use another Google API, right? And then you might tell other people, you might do a blog post that says Google App Engine sucks. There are certainly posts like that. And then other people have that attitude as well. It could actually be that it sucks and it's something we obviously need to fix, right?*

Hence, what is described above is a trust through the liability of the substrate, on top of which developers build their own services. A problem on the interface will have effects on the entire set of services that are "plugged" into it instantly, as what is in use is a constant flow of information. Moreover, user-developers are not isolated. The capacity of user-developers to blog or complain to others may diffuse the information, engendering negative network effects for the provider's reputation.

Still, when I asked a *Google* product manager how they manage the relationship with those developers, he answered:

> *It's our strongest point. Without developers, you don't see the word Google. A developer is really a person that's saying "what business can I do? Can I develop this? You know what? I will not start from scratch. This is the platform that I need for a base. And I build my stuff on top of it." And that's where the API is coming in.*
>
> *So, you could say developers are actually an extension of Google. Because they take our technology, they add their technology and they build something in the middle.*

We find thus a particular actor, the user-developer-entrepreneur (UDE) who is given enough value from the platform providers in what regards the expansion of their platform, in terms of popularity, functionalities and deployment. Still, the platforms do not engage in a typical partnership relationship with them. At the same time, UDEs do not reveal freely their innovations: they either create their own services or get acquired by the providers if they present a fast rate of growth.

A key element in this process is the establishment of intimacy between providers and UDEs: providers have to trust UDEs, while UDEs have to trust, on their terms the providers.

---

nissant des APIs pour qu'ils puissent démarrer et commencer à faire des partenariats stratégiques avec des start-ups dont on estime qu'elles vont se développer très rapidement. Il y a une vraie politique aujourd'hui qui va dans ce sens. Spécialement en France, mais aussi en général pour Yahoo.

What is assumed from the discourse represented above, this "chicken and egg problem" has two dimensions: on the one hand, the Web business environment seems to have begun with an *a priori* trust among the two, something which has not been the case for instance for telecoms. On the other hand, this relationship is maintained and managed by the platform provider throughout the life of the platform.

Hence, we can conclude this paragraph with the formulation of the following working hypotheses:

Intuition 5. *User-Developer-Entrepreneurs constitute a particular actor for innovation in Web platforms. The synergy between UDEs and platform providers is a critical requirement for the expansion of the platforms*

Intuition 6. *While UDEs are not organised in innovation sharing communities, such as in the User Innovation paradigm (von Hippel and von Krogh, 2003), they are linked with each other, as well as with platform providers through "intimate" ties.*

Intuition 7. *Web services open product strategy, beyond the provision of specific substrates for UDE innovation, is due to two reasons:*

1. *The historical competitive advantage of this business domain in what regards an intimate relation with developers.*
2. *The active management of the communities of UDEs.*

5.3 Discussion

In the Table 5.2 I synthesize the Intuitions as resulting from the identification of the business domain peculiarities through the platform providers' discourse.

During the interviews, service provider executives stressed the importance of three major peculiarities, as well as the prerequisites rendering them possible and illustrating the leading direction for enterprise strategy.

The peculiarities regarded two elements of online business: the technical substrate and the actors of the sector.

Hence, Web services act as Open Products (Chrysos et al., 2010), providing a specific value of use to end users, while giving UDEs the means to extend the platform, while the latter is in the market (Int. 3). Between the attributes of the means provided to developers (of the APIs) and the attributes of the final service, as perceived by the end-user, there appears to be a "mirror relationship" (Int. 2), as UDEs will develop their services using the categories of information already there. For instance, in *eBay* they will use information about auctions, while in *Google Maps* they will use information about mapping. This mirror relationship comes as a result of an historical evolution consisting in the shift from Web sites to APIs (Int. 1) enabling the expansion of Web services over different devices and the displacement of innovation trajectories form the Web sites to the Apps.

The figure of the UDE is central in this process: his activity goes where partnerships cannot go. UDEs also appear as a condition for platforms to quickly be explored and extended, enriching the final offer to end-users and rapidly tracing new tendencies in their

	Technical Substrate	Int.	Actors	Int.
Peculiarities	Mirror relationship	2	UDEs : conditioned innovation condition for innovation	4
	Open Products	3	Synergy with providers, intimate ties	5& 6
			Collectivity Management	7,2.
Prerequisites	Shift from Web sites to APIs	1	Industry - Developers ties	7,1.

Table 5.2: Synthesis of the peculiarities identified.

public through the development of start-ups (Int. 4). Hence, a synergy between platform providers and UDEs is suggested by their common interests (Int. 5), though this synergy can only be based on an intimacy between the two actors, as partnerships cannot be deployed at this level (Int. 6).

The fast development of Web services, with services reaching hundreds of millions of users on a global level within few years, is among others, based on this synergy. The Web services sector has had this competitive advantage because of its historical formation (Int. 7, 1). Still, the further deployment and popularisation of these services demands a management method for this specific kind of relationship (Int. 7, 2.).

Overall, the innovative activity of UDEs enters in-between the two known paradigms for innovation, the producer and the user innovation models.

In Figure 5.1 I indicate the major findings as compared to the two innovation paradigms as outlined by (Raasch and von Hippel, 2012) (reviewed on page 56). Firstly, I identified the specific value of use for Web services, the value of collective intimacy, which is an entry point for innovation processes in both cases. Then, the particular actor identified in this chapter constitutes a bridge between the two models: he comes from the user public while he moves towards the enterprise. This specific move is what interested my interlocutors and the reason for the provision of APIs is to facilitate it. Of course, the public of UDEs is smaller than the one of end-users, still considerably larger than an enterprise's engineering department, when talking about Web services such as *Google*, *Yahoo* or *Twitter*. At the same time, UDEs have an expertise which is higher than the average user, though still not as advanced as enterprise R&D engineers.

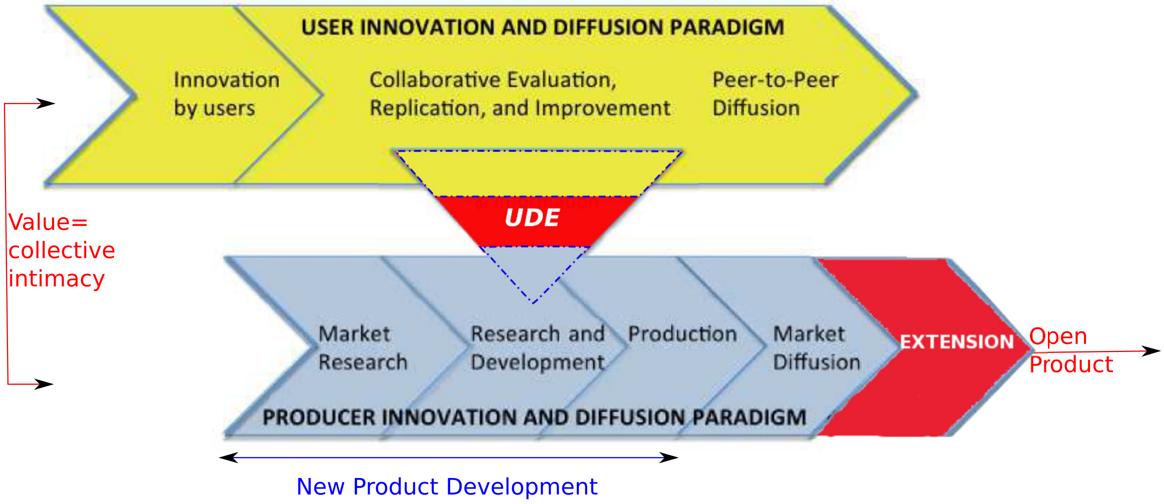


Figure 5.1: Peculiarities of the Web services innovation environment.

5.4 Conclusion

In this section I identified two indications on the questions of who develops applications based on contemporary Web services and how this is done, as compared to existing studies on online business, new product development and user/producer enterprise models. Enterprises appear to share this challenge although they have an original model of innovation management, based on the sharing of exploitation and exploration with a large public of individual user-developer-entrepreneurs (UDEs), emphasizing the phase prior to the start-up formation. UDEs innovation is placed in-between the user and the producer innovation model, calling for novel management methods of peculiar innovation collectivities.

In the rest of the current Part of my study, I will further explore the actual peculiarities identified in this chapter. Then, in the Part II, I will study the formation of the prerequisites that allowed to the Web sector to produce such a particular innovation model.

In the last part of my study, I will distinguish, explore and propose three methods for the management of such collectivities, taking into account all the particular characteristics of this business environment, as well as their dynamics, as identified by the study of the prerequisites formation.

Chapter 6

UDEs: in-between the manufacturer and the user innovation paradigms

Contents

Chapter 5 traced some early indications of the possible peculiarities of contemporary Web business, regarding a) the value of "online identity" provided by Web services, b) the specific figure of User-Developer-Entrepreneurs (UDEs) and c) the particular technologies supplied to the latter by service providers, enabling the extension of online services through UDE innovation.

The current chapter explores the possibility of UDE activity constituting a third innovation model, placed in-between user and manufacturer paradigms. Using a phenomenon-based research approach (von Krogh et al., 2012), I distinguish some elementary characteristics of such a model, by revisiting the well-known "private-collective" model for innovation (von Hippel and von Krogh, 2003, 2006).

Examining such actors in action and triangulating my observations with the study of specialised books for online service developers, I find that typical norms of the existing innovation models are not followed. Hence, I further explore this way of action and suggest that UDEs oscillate in-between user and manufacturer roles through an entrepreneurial activity, which may have an important impact on innovation strategies for the Web services sector.

6.1 Introduction

This study explores the ways in which third party innovation on Web services can be studied, as compared to the general paradigms of user and manufacturer innovation. The distinction between user and manufacturer innovation von Hippel (1975), as described by the "private collective" model (von Hippel and von Krogh, 2003, 2006), has provided a new perspective on B-C (Business to Consumer) relationship understanding, enriched or event contested by the C-C relationship. This chapter explores yet another figure that should be taken into account, beyond B and C, the *developer*, particularly active in the Web services domain.

The current chapter will suggest that the role of developers can be analysed by taking into account his three different configurations: user-developer (UD), creating tools or applications for his own use, user - developer - entrepreneur (UDE), attempting to commercialise such creations, and developer - entrepreneur (DE), using similar methods for direct commercialization of creation but not for personal use. Two different methods will be used to distinguish the action norms of this figure, both being undertaken at the level of an individual developer (and not at the one of an enterprise or a user community).

This exploration will be deployed in two steps, as outlined by Figure 6.1. The first step explores in a qualitative mode by means of an "observant participation" the way individuals active in the domain operate. The outcome is that their activity seems not to fit in either of the paradigms, as individuals don't reveal their creations, although aren't manufacturers, in the business sense, either. Moreover, they use a "palette" of tool-kits, coming from different origins, both free software tools, as in the open source paradigm and Web service based ones, the *Application Programming Interfaces (APIs)*.

The second step proceeds in a more systematic exploration of the action norms of these individuals, as described by the technical "Cookbooks" they read, to be able to develop their own applications. Verifying the propositions of the first step through a more detailed analysis, it suggests that their action mode can best be described as a "personal investment" one, situated in-between the "private investment" and the "private-collective" models (von Hippel and von Krogh, 2003, 2006).

The chapter concludes with the proposition of three alternative concepts for research, the User-Developer (UD), the User - Developer - Entrepreneur (UDE) and the Developer-

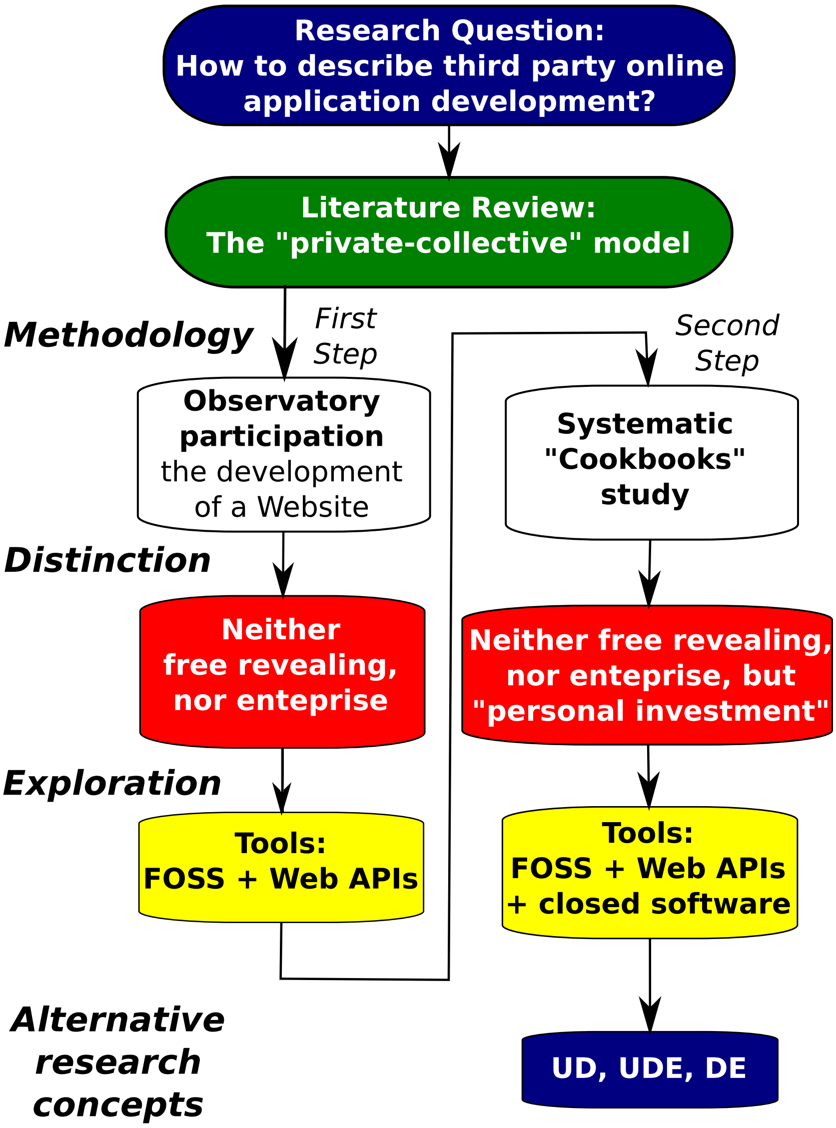


Figure 6.1: Chapter overview.

Entrepreneur (DE), each having a different attitude towards the innovation process.

6.2 Theoretical concepts

The question I am studying in this chapter is the following:

Question. *Should the activity of people using existing Web services to develop their own applications be described according to the user innovation or the manufacturer innovation paradigm?*

My objective will thus be to identify and explore (von Krogh et al., 2012) the peculiarities, if any, of their particular actions and allow for their eventual classification. Hence, the principal concepts of the "private-collective" model will be reviewed in order to construct a comparative exploration and analysis framework.

6.2.1 The *"private-collective"* model

In their influential work, von Hippel and von Krogh (2003, 2006), using the open source development as an example case, introduced a novel model for innovation. The *"private-collective"* model has been a theoretical framework for the interpretation of the articulation between individual and collective incentives, for what had already been identified as *"user innovation"* (von Hippel, 1990; Herstatt and von Hippel, 1992; Olson and Bakke, 2001; von Hippel and Katz, 2002, and others). In brief, there are two fundamental questions according to which von Hippel and von Krogh (2003, 2006) classify the different innovation paradigms:

- How is knowledge produced and shared?
- How is value appropriated?

More specifically, the *"private-collective"* model was introduced as a junction of two other models, the *"private investment"* and the *"collective action"* ones, *offering the "best of both worlds" under many conditions* (von Hippel and von Krogh, 2003, p. 209). Table 6.1 summarizes the arguments proposed by the two authors, regarding the dimensions of knowledge and added value production, which are reviewed in the following paragraphs.

	Knowledge		**Value Appropriation**	
Model	**Production**	**Sharing**	**Private**	**Public**
Private investment	R&D	Patent	Monopoly	Loss
Collective (public) action	R&D, public research	Knowledge Control Reliquishing	Reputation, monetary subsidies	"Common pools" resources
Private-collective	"Sticky" knowledge	"Free revealing"	Reputation, Learning, problem resolution	"common pools", common solutions

Table 6.1: Private-collective model as opposed to the Private Investment and the Collective Model. Compiled from von Hippel and von Krogh (2003, 2006).

The "private investment" model

On the one hand, as a private investment model, von Hippel and von Krogh refer to what elsewhere is termed the manufacturer paradigm (Raasch and von Hippel, 2012, and others). The authors adopt the definition of Demsetz (1967), according to which this model is based on the following fundamental assumption:

> *that innovation will be supported by private investment and that private returns can be appropriated from such investments (von Hippel and von Krogh, 2003, p. 212).*

Under this reasoning, *"to encourage private investment in innovation, society grants innovators some limited rights to the innovations they generate via intellectual property law mechanisms"* (von Hippel and von Krogh, 2003, p. 212). Hence, the *monopoly* granted to innovators marks a *"loss"* for the society, as the knowledge created within these norms, can be exclusively used by the innovators only (von Hippel and von Krogh, 2003, p. 213). In parallel, mechanisms such as *patents* come to counterbalance this public loss, since, by definition, they commute monopoly granted to public knowledge disclosure for the period of the exclusive rights granting, after which this knowledge becomes public. Typically, knowledge created and then exploited through innovation, is explored within an enterprise *R&D* department, though it has been additionally proposed that innovation should be treated as a distinguished enterprise process (Hatchuel et al., 2001), organising a dialectic relationship between research and development.

The "collective (public) action" model

On the other hand, by the term *"collective action model"* von Hippel and von Krogh refer to a model which *"applies to the provision of public goods, where a public good is defined by its nonexcludability and nonrivalry"* (von Hippel and von Krogh, 2003, p. 213). For this model to be actionable, collective action contributors are required to *"relinquish control of knowledge they have developed for a project and make it a public good by unconditionally supplying it to a 'common pool' "* (von Hippel and von Krogh, 2003, p. 213). *Monetary* or *reputation* subsidies are usually attributed to contributors by the public to foster this type of innovation and ensure participation.

The "private-collective" model

Subsequently, the "private-collective" model comes about as a junction of the two other models mentioned above. User innovators *"freely reveal"* their creations (von Hippel and von Krogh, 2003, 2006) and benefit in a non-profit way: the efforts for the resolution of a problem they face, which is considered as the starting point of innovation, are shared within a community of users through the public disclosure of the innovation itself.

One of the fundamental propositions of the user innovation is that the "locus" of innovation, that is *where innovation actually occurs*, is often in the user, not on the manufacturer side. So, in his seminal work on innovation in scientific instruments, von Hippel highlighted that *"it is almost always the user, not the instrument manufacturer, who recognizes the need, solves the problem via an invention, builds a prototype and proves the prototype's value in use"* (von Hippel, 1975, p. 20). Moreover, the user innovator does not keeps innovations for himself, nor does he commercialize it, but *"encourages and enables the diffusion of his* *invention by publishing information on its utility and instructions sufficient for its replication by other users - and by instrument manufacturers"* (von Hippel, 1975, p. 20).

According to von Hippel and other researchers of the user innovation "School", innovation is localized on the user side because use-related *"sticky information"* also lies on the user side. By that term, von Hippel refers to information or knowledge that is costly to transfer (von Hippel, 2005, pp. 67-68). In general, such knowledge may be obtained either by the manufacturer or by the user. More specifically, as outlined in Table 6.2, users usually obtain information about their *needs*, to which user innovation comes as a response, since it is cheaper and more beneficial for these actors. The "buy or innovate dilemma" expression has been used to illustrate the motives of user innovation. On the contrary, manufacturers are more likely to innovate during new product development when this information is about *technology*, where their expertise is useful (Ogawa, 1998; von Hippel, 2005).

Innovation Actor	**Sticky information**	**Innovation locus**
Manufacturer	Technological	New product development
User	User need - related	Use
Developer	*Under exploration*	

Table 6.2: Likeness for Users and Manufacturers to innovate using "sticky" information. Compiled from (von Hippel, 1994; Ogawa, 1998; Lüthje et al., 2005; von Hippel, 2005). This chapter explores the values for the corresponding fields for the developer case.

As von Hippel (2005) puts it, one of the main values of "sticky" information for innovation, resides in the modelling of users own needs (von Hippel, 2005, p. 8). This "sticky" information is then the resource on which "each innovator will draw on" (von Hippel, 2005, p. 71). As a result, user innovations tend to be closer to the specific needs and context of use, than the ones of manufacturers:

> *In the specific case of product development, this means that users as a class will tend to develop innovations that draw heavily on their own information about need and context of use. Similarly, manufacturers as a class will tend to develop innovations that draw heavily on the types of solution information in which they specialize (von Hippel, 2005, p. 70).*

In general, user innovation implies a series of advantages for innovators, such as the low cost for innovation processes, development and maintenance through the activity of user communities (von Hippel and von Krogh, 2006; von Krogh et al., 2003a), the wide, community-based diffusion, as well as the learning process itself (von Hippel and von Krogh, 2003, 2006).

The use of these categories to describe the development of online applications is not evident: while technology is an easy to guess requirement for the development of an application, the indications of chapter **??** suggest that one does not have to be an enterprise to possess this information. At the same time, one can safely assume that users are not in a position to develop an application, say for a service like *Facebook*, unless they are able to master some of its technology.

The analytical exploration of the different types of "sticky information" and "innovation loci" used by the people developing online applications is thus required to characterise these actors. Before that, a closer look at the literature regarding the open source paradigm as well as the figure of entrepreneur in the innovation process is put forward for consideration.

The exemplary case of open source

The open source phenomenon has been an exemplary case for the "free revealing and private-collective model" of development, allowing to explore additional dimensions of user innovations, such as collective learning processes (von Krogh et al., 2003a,b; O'Mahony, 2003, and others). The activity of user-developers deployed thanks to a 'pool' of open source software projects enables users to build upon existing resources (von Krogh et al., 2003a). In parallel, the public earns from the resolution of the problem itself, as the community freely reveals its innovations. Such a process is structured around the challenge of a common problem resolution (von Hippel, 2007; Benkeltoum, 2008).

More specifically regarding the open source software paradigm, von Hippel and von Krogh remark on the specificities rendering this case unique:

> *What may be unique to knowledge and information products is that in these fields we see users carrying out the entire innovation process for themselves - no manufacturer required. Thus, open source software projects encompass the entire innovation process, from design to distribution to field support and product improvement. Such "full-function" user innovation and production communities are possible only when self-manufacture and/or distribution of innovative products directly by users can compete with commercial production and distribution. In the case of open source software this is possible because innovations can be "produced" and distributed essentially for free on the Web, software being information rather than a physical product (von Hippel and von Krogh, 2003, p. 219).*

Hence, in the specific case of software, the "private-collective" model takes a more global dimension, through the practice of *open source software*. This aggregation of user innovation characteristic of the open source software phenomenon is attributed to the specificity of *"knowledge and information products"*, which gives users the possibility to carry out *"the entire innovation process for themselves"*, from design to distribution and maintenance, without the help of manufacturers. *"Essentially free"* distribution is then enabled by the Web, as software is *"information rather than a physical product".*

Yet, in the field of Web 2.0 application development, one can observe the same characteristics: similar to the open source software case, Web 2.0 applications can generally be characterised as knowledge and information products, while the Web is also used for distribution. Nevertheless, as we traced in chapter 6, an "intimate" synergy between enterprises providing services and UDEs is suggested. Given this identity of attributes between Web 2.0 applications and open source software, and judging from the open source specific characteristics as outlined by von Hippel and von Krogh, the synergy indicated in my field seems paradoxical: why would UDEs and providers come together if each actor could act individually?

In other words, which model is the appropriate one to describe the activity of people using Web services, such as *Facebook* or *Google Maps* to conceive and develop their own applications?

At this level, we should distinguish another form of individuality beyond the user, also discussed in innovation studies, the figure of the entrepreneur. The following paragraphs begin by reviewing the literature connecting the two figures.

6.2.2 The user - entrepreneur

Studies on the open source phenomenon contributed to the further orientation of literature in user innovation towards the innovation capacities (Nassimbeni, 2001) of individual users. While the initial conceptualisation of user innovation includes both individuals and firms who innovate to serve their own needs (as opposed to marketing those innovations) (von Hippel, 1976, 1977a, 2011), later research explicitly focuses on individual users (consumers) as innovator actors. Their particularity consists in going beyond the provision of concepts for innovation to the undertaking of the entire process themselves (von Hippel, 1978a,c; von Hippel and Katz, 2002; Olson and Bakke, 2001), allowing for the projection of the current era as *"the age of consumer innovation"* (Fa et al., 2011), where end-user innovation would obtain autonomy from the manufacturer paradigm.

Hence, while the experienced, individual user takes a more or less central role regarding innovation, his relationship to another figure, traditionally highlighted by innovation studies, the entrepreneur, is less explored.

The few studies exploring the specific condition of user-entrepreneurs have already been reviewed in Section 3.3.3 (page 63). Baldwin et al. (2006) study the case where users become entrepreneurs. Their research is based on a design reasoning, according to which *"user innovation begins when one or more users of some good recognize a new set of design possibilities - a so-called "design space" - and begin to explore it"* (Baldwin et al., 2006). Thus, user innovations cover this particular design space.

However, research interested in the phenomenon of user-entrepreneurship (Shah, 2003; Lüthje et al., 2005; Shah and Tripsas, 2007; Haefliger et al., 2010) concludes with a rather linear model of transition from the user to the manufacturer paradigm, as modelled in Figure 3.8 (on page 65) following a literature review, which is doubled in Figure 6.2 for ease. These studies are based on cases where innovations are successfully commercialised. However, success is mostly an exception in what regards entrepreneurship as a whole.

The Schumpeterian analysis of innovation is largely based on the figure of the entrepreneur. Schumpeter recognises that the vast majority of "would-be" entrepreneurs fail (Schumpeter, 1939, pp. 116-117). Moreover, he underlines the absence of common social characteristics among entrepreneurs, who may originate from a variety of social groups (Schumpeter, 1939, pp. 101-103). Hence, unlike the user, who can be clearly defined by his relationship with a given good, the entrepreneur can be defined merely by a mind-set. Therefore, management scholars felt the need to distinguish *entrepreneurship* from *entrepreneurial orientation* (Lumpkin and Dess, 1996, 2001), the former being the result (e.g. a new market entry), the latter being the way to it. Consequently, it is legitimate to argue that research on user-entrepreneurs describing a rather linear passing from the user to the manufacturer role, touches only on a part of the phenomenon, the tip of the iceberg, to the extent that this research only studies successful cases.

There are two major currents in the literature on entrepreneurs. On the one hand, there are studies highlighting the subjective characteristics, such as personality, favouring entrepreneurship (Lüthje and Franke, 2003; Fauchart and Gruber, 2011, and others). On the other hand, other studies highlight the objective characteristics, such as entry cost, that constrict entrepreneurship (Willcocks and Plant, 2001; Sawhney, 1998; Iansiti and Levien, 2004; Eisenmann et al., 2006; Osterloh and Rota, 2007). In the current chapter, I explore entrepreneurship as an aspect of the "Web 2.0" phenomenon, that enables people to develop applications. Thence, I will focus on the objective characteristics favouring entrepreneurship, that is the conditions rendering it possible for (more or less) charismatic people to innovate

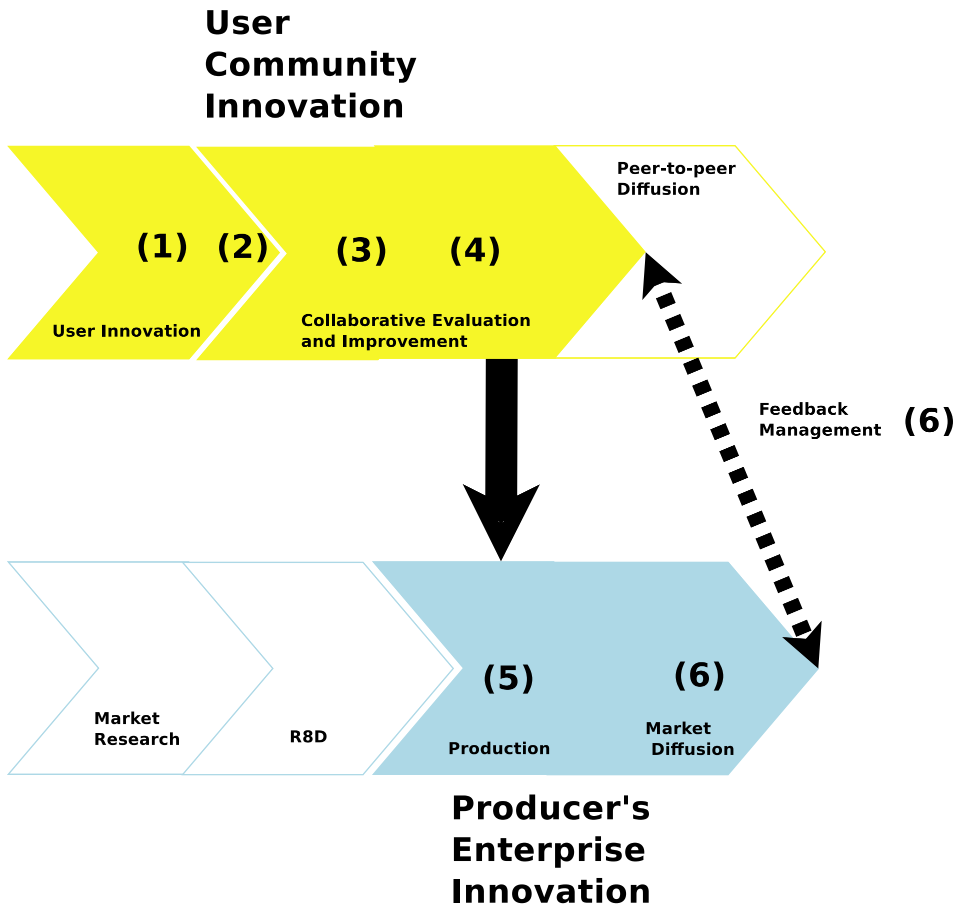


Figure 6.2: User-entrepreneurship steps according to Shah and Tripsas (2007) localised in the mapping proposed by Raasch and von Hippel (2012).

using Web services. Design space opportunities (Baldwin et al., 2006) identification will be a guide for this exploration.

What I met in the field and will develop in the following paragraphs, is that entrepreneurship appears as a "continuous oscillation" between the service user and the service provider roles, enabled by a common practice of the service providers, to allow innovation on their platforms. By exploring the eventual action norm peculiarities of people developing Web applications, I expect thus to determine in a finer way the transition of the user to the manufacturer condition.

6.2.3 An exploration framework

As already mentioned, the aim of the current chapter is to distinguish and to explore (von Krogh et al., 2012) the action norms of people using Web services for their own applications development, in relationship to the two figures mentioned above, the user and the entrepreneur. For this, two innovation paradigms, the user and the manufacturer ones (Baldwin and von Hippel, 2011; Raasch and von Hippel, 2012) will be used as a reference. Still, it should be noted that this opposition between the two paradigms is not absolute: by definition "lead user" innovation comes before its manufacturer industrialisation and, thus, the distinction is rather a matter of different innovation phases. Nonetheless, this dichotomy will serve more as an emphatic reference for an early form of modelling, than as a deterministic classification criterion.

Using similar distinction criteria as von Hippel and von Krogh (2003, 2006), I will explore the questions of knowledge production and value appropriation qualitatively, in respect to application development on top of existing Web services. As already mentioned, the level of my analysis will be at that of people developing such applications, which I will describe in comparison with their models.

Table 6.3 outlines the main characteristics of the two paradigms. Design and development as well as diffusion innovation phases are typically structured in different ways. I review these differences by separating action *goals*, judged by the actors in terms of effectiveness, and the action *norms*, describing the rules, the assumptions and the conditions under which the action takes place (Argyris and Schon, 1978).

More specifically, according to the user paradigm, design and development emerges from the incentives to satisfy personal needs, not fully addressed by an available product, and are characterised by the use of "sticky" or "local" knowledge, lead users have from their own experience (von Hippel, 1986, 1994, 1998; Lüthje et al., 2005, and others). In addition, there is a variety of individual incentives for participation in the collective process, globally characterised by the sharing of development efforts and the corresponding reputation effects within the community of users (Hars and Ou, 2001; Lakhani and von Hippel, 2003; Shah, 2003; Lattemann and Stieglitz, 2005; Shah, 2006; Arena and Conein, 2008, and others).

On the contrary, profit expectations rule in a high degree design and development process in the manufacturer paradigm. Enterprises are based on Market Research to identify and configure potential client needs, using its results as requirements for further advancements (Mowery and Rosenberg, 1979; Sahal, 1981; Clark, 1985; Krishnan and Ulrich, 2001, and others). R&D is the typical enterprise norm for innovation organisation, while new product development is managed according to the criteria of speed, quality and cost play a crucial role in decision-making, as well as in formulating the agenda of future issues to explore (Clark and Fujimoto, 1991; Iansiti, 1993; Verganti, 1999; Hatchuel et al., 2001, 2002; Midler and Navarre, 2007; Midler and Beaume, 2010).

	User Paradigm		Manufacturer Paradigm	
	Goals	**Norms**	**Goals**	**Norms**
Design & Development	Satisfy own needs; Share efforts	"Sticky" knowledge; User community development	Satisfy (potential) client needs; Speed, cost, quality	Market Research; R&D
Diffusion	Reputation; Maintenance	"free revealing"; User networks diffusion	Merchandise; Client support	Marketing; After sales service
Indicative Literature	Herstatt and von Hippel (1992); Shah (2003); Lakhani and von Hippel (2003); Lüthje et al. (2005); Osterloh and Rota (2007); Benkeltoum (2008)		Clark and Fujimoto (1991); Iansiti (1993); Verganti (1999); Hatchuel et al. (2001, 2002); Midler and Navarre (2007); Midler and Beaume (2010); Andreani and Conchon (2001); Fombelle et al. (2012)	
	von Hippel and von Krogh (2003, 2006); Baldwin and von Hippel (2011); Raasch and von Hippel (2012)			

Table 6.3: Exploration framework: norms of the two innovation paradigms.

Finally, innovation diffusion takes place in the first case through user networks, where innovators freely reveal their creations, benefiting from reputation effects, while in the particular case of open source software they also benefit from community-based maintenance (von Hippel and von Krogh, 2006; von Hippel, 2007). The discipline of Marketing, on the other side, has dedicated its greatest interest in determining and accessing a relevant public to merchandise the good produced (Denner, 1971; Eiglier and Langeard, 1975; Millier, 1989; Andreani and Conchon, 2001; Benavent and de la Villarmois, 2006; Brown et al., 2006; Fombelle et al., 2012, and others).

From the above mentioned norms, I will use as a reference the most distinctive ones for each paradigm. While innovating *for use* or *for profit* constitutes the major distinction criterion between the two paradigms (Baldwin and von Hippel, 2011; von Hippel, 2011; Raasch and von Hippel, 2012), there are overlapping characteristics, while when it comes to development R&D practices, historically bounded to the enterprise context, can also be used for user innovations, as early studies on the field of scientific instruments innovation have shown (von Hippel, 1976, 1977b; Riggs and von Hippel, 1994; Shaw, 1985; Lettl et al., 2006). Hence, I will not take in account the R&D dimension in my exploration, being limited to less ambiguous action and more singular norms.

Overall, if there are to be peculiarities in the way people developing applications act in comparison with the above mentioned paradigms, they should be expressed on the level of the elementary norms of action mentioned. The goal of the current study will thus be to qualify whether or not innovation activity of these people enters one of the two paradigms mentioned. For this, I will examine their elementary "norms of action" in relationship to end-users and providers, following a two-steps investigation method, as developed in Section 6.3.

	Distinction Step	Exploration Step
Question	*Can the activity of people using Web services for their own application development be described by the already known innovation paradigms?*	
Level of analysis	People developing their own goods, on the basis of existing online services“	
General Research Goals (von Krogh et al., 2012)	*"Encounter bracket peculiarities"; "describe context in broad cultural terms"; "identify inadequacy of given body of theory and knowledge in the field"; "identify relative concepts for study"*	*"Intensify data gathering"; "Generate more solid concepts that can serve as filter for further data gathering";*
Method	Observant participation in a Web site development process	Systematic analysis of“Cookbooks by and for developers"
Outcome	User-Developer-Entrepreneur figure (early distinction); Use of both Web services and open source modules	From user to entrepreneur: transformations of interest

Table 6.4: Distinction and exploration steps my methodology, based on (von Krogh et al., 2012).

6.3 Methodology: distinction and exploration

As already mentioned in the introduction, a two step methodology will be used to distinguish and explore Web-based application development phenomenon. This section elaborates the research goals and the methods used in each step. The findings will be discussed in Sections 6.4 and 6.6.

6.3.1 Research Question and Goals

To distinguish and to explore the specific characteristics of the action of people using existing services to develop their own ones, I will use a phenomenon-based research strategy (von Krogh et al., 2012). Table 6.4 summarises the steps of the methodology used and will be further analysed in the following paragraphs.

The research question is posed as follows:

Question. *Can the activity of people using Web services for their own application development be described by the already known innovation paradigms?*

For this question to be answered, the activity of these people should be explored in a way that enables the identification of possible peculiarities (von Krogh et al., 2012), as expressed through their action norms (Argyris and Schon, 1978). Hence, once the action norms are identified, a comparison to the user and manufacturer paradigms (von Hippel and von Krogh, 2003, 2006; Baldwin and von Hippel, 2011; Raasch and von Hippel, 2012) can be operated, as discussed in the exploratory framework, paragraph 6.2.3 and outlined in the Table 6.3 (page 104).

According to von Krogh et al., studies distinguishing a new phenomenon often do it by using narratives. Early research into the open source phenomenon research research used the works of Raymond (1999) and Stallman (2002), identifying its distinctive characteristics. Both authors had been invested in the activity of open source communities and their writings had the intention to share this experience with the other members of the community. For Raymond this experience concerned the benefits of a distributed development process, while for Stallman those of free software and the related copyright license. Von Krogh et al. synthesizes the following research goals for studies aiming at a new phenomenon distinction: *"encounter bracket peculiarities against existing body of knowledge"; "describe context in broad cultural terms"; "identify inadequacy of given body of theory and knowledge in the field";* and *"identify relevant concepts for study"* (von Krogh et al., 2012, p. 290). This approach will be used during the first step of my exploration, where I will use an "observant participation" method by joining the process of a Web site development. I will analytically describe this step in paragraph 6.3.2.

Then, always according to von Krogh et al., during the exploration phase, researchers should *"intensify data gathering inside and outside the focal concepts"* and *"generate more solid concepts that can serve as filters for further data gathering"* (von Krogh et al., 2012, p. 290). Citing the work of Lewis (2000b) on exploration research, the authors agree that *"at the outset, a phenomenon is typically defined in terms of what it is not"* (von Krogh et al., 2012, p. 286). This approach will be used during the second step of my exploration, to which the results of the first step will be an entry. This step consists in the systematic study of "Cookbooks" edited and read by developers, further described in paragraph 6.3.3.

6.3.2 First step: early norms distinction

The first step of my research methodology is deployed through the "observant participation" in the collective development of a Web site and aims at distinguishing (von Krogh et al., 2012) the action norms of these people, as compared to existing innovation modes.

Exploration method: *observant participation*

The method used in the first exploration step is situated on the frontiers of action-research and participant observation. The method consists in using a field challenge in which I participated to identify the action norms of the field actors, the people I collaborated with. The challenge was the development of a Web site, where the team actors belonged to an exemplar milieu. Sharing elements with both, it is distinguished by the exploitation of an ephemeral group action, a fact which positions the researcher in a privileged place.

Observant participation compared with research intervention

There is a similarity of the method used to the research intervention approach, which lies in the fact that I actively took part in the tasks of the team, and more specifically in *"objectives formulation"* and *"collectivity mobilisation"*, while, as a researcher, I had the *"opportunity to interact with the actors on these challenges and acquire an in-depth knowledge"* of the issue (Hatchuel and Molet, 1986). Still, the collectivity in question was not an enterprise, as in the case of most research intervention studies, while the results of my research were not to be *"taken into consideration"* by the collectivity, as the theory suggests, since the group was to be dissolved after the three days term.

Generally, within the framework of action research, *"low level implication"* in the field has several advantages and inconveniences (Mitchell, 1993; Baumard, 1999). The researcher is in this research mode *"not considered a threat"*, as long as he can *"understand the specific language of the actors"*, while this *"independence provides the possibility to dissociate his/her research from the field challenges"* (Baumard, 1999). Moreover, in this specific case, the collectivity being ephemeral, everyone had a minimal engagement (of three days), making my dissociation not specific to myself as a researcher but in fact a common ground.

Observant participation compared with participant observation

Unlike research intervention, participant observation approach privileges an external researcher position. It is used as a means to *"access events or groups that are otherwise inaccessible to scientific investigation"*'(Yin, 2003, p. 94). Yin highlights the importance of the researcher being external, to avoid the danger of him becoming a group supporter. In addition, as participant observation often goes on over a long period of time, Yin notes the difficulties implied by factors such as good *"timing and attention for the right observations to occur"* (Yin, 2003, pp. 95-96). However, the ephemeral character of the action I joined, rendered all group members "external" (as the participants had not met before). The group's limited life period also rendered observation easier, compared to long term activities observation. Hence, my position as a researcher, provided that this investigation was parallel to a relative familiarity with the specific terminology and technologies involved through the corresponding *"Cookbooks"* (discussed in the next paragraph), was more of an "observant participation", since I observed the group's action while taking part in it.

Field entry: joining the development of a Web site

The specific group I joined was built around an ephemeral project designed and developed in three days, during a *"hackathon"* event for *Silicon Valley Google Technologies Users Group* effected in *Google*'s headquarters in Silicon Valley[1]. The team was met and formed in place to develop a Web site, an "online application store".

A *hackathon* is a three days competition, where developers and designers are called to use a provider's technologies to innovate. Developing teams are formed on the first day, after the keynote speeches, through the self-selection of members on the basis of a concept proposal (a *"pitch"*). Group leaders candidates present their ideas to the participants and form a group that will work three days (and three nights) to develop the concept.

As Silicon Valley is home to many important Web service providers - such as *Google*, *Yahoo*, *eBay*, *Facebook* and *Amazon* - as well as to an important number of dynamic startups (Saxenian, 1994, 2000; Sturgeon, 2000; Weil, 2010; Lecuyer, 2006; Lécuyer, 2006), I judged that this event could provide access to an exemplary sample of Web developers to observe.

I chose to participate in a group that was to build an online store for Web-based applications, as the concept of online markets is already familiar to management literature

[1]The event itself as a means to manage UDE communities will be explicitly studied in Chapter 16. While participation on the event was open to everyone, I personally came to attend it after the proposition by a *Google* Developer Community Manager, during an online interview regarding the Developer Support process.

(Brousseau and Chaves, 2004; Iansiti and Levien, 2004; Eisenmann et al., 2006). The group had six members, myself included.

The results of this exploration will be discussed in Section 6.4 and used as input for the second step of my methodology. Table 6.3 (on page 104) will be used as a reference for the discussion of the findings as related to the paradigms already reviewed.

6.3.3 Second step: exploration

To further explore the peculiarities (von Krogh et al., 2012) of the activity under investigation, the second step of this study will be based on material where a more systematic reflection on their practices is undertaken, beyond isolated cases. The entry chosen for this is a set of books providing advice to people who want to use Web services for their own development processes, and is inspired by historical research on technological actors' process rationalisation.

Exploration method: the study of editions "by and for developers"

The research method used in this second step is inspired by the historical research on the rationalisation processes leading to the emergence of both new disciplines and new actors in different industrial contexts.

The historians Chatzis and Ribeill (2008) outline a *"panorama"* of editions, written *"by and for engineers"* in France from 1750 to 1950. In their original study, they prove the contribution of the technical editions and early engineering communities in the capitalisation, communication and sharing of knowledge among the early engineer communities of the 19^{th} century, during what is known as the second industrial revolution (railways, chemistry). Those editions, generic or specialized, responded to a diversity of challenges of the industrial revolution, framing the formation of the "identity" of engineers around a discourse on innovation.Significantly, the constitution and the legitimacy of the "corps" of state engineers was constructed in the middle of the 19^{th} century around such reviews, in competition with autonomous ones (Chatzis and Ribeill, 2008, p. 123).

The diversity of historical and technological contexts where rationalisation processes, regarding both technological knowledge and technical actors, emerge through the coupling of new technical communities and related editions, is encouraging for research into Web contexts that are based on similar manuscripts.

Studying a different period, the decades 1945-75 (the *Trente Glorieuses)*, Chatzis (2008) shows how new actors in French enterprises, the maintainers, managed to illustrate the value of their work while rationalising it, through the creation of the Maintenance Department. There, he underlines the importance of *"specialized professional journals"* providing *"both the infrastructure required to develop rationalization techniques and a collective self-image for the actors involved in rationalization"* (Chatzis, 2008, pp. 81-82).

Unlike archival analysis, having as its goal to answer "who" and "what" questions, in a quantitative and non-explanatory mode (Yin, 2003, p. 6), the study of technical editions providing advices to "peers" will be used to understand the underpinning norms, governing the "best practices" described in those books.

Entry: Systematic analysis of "Cookbooks" for Web services-based development

Retaining this teaching on the importance of the institutional or autonomous editions "by and for engineers" in the framing of a discourse of an innovation, regarding both the content and the strategic importance for future evolutions in the milieu, the second investigative step of the current study will be based on the study of analogous editions, addressed today to people developing applications using Web services. Still, this study doesn't aim at presenting a comprehensive panorama of those editions, but at a revelation of the business stakes "hidden" in the technical discourse. More specifically, exploration will be based on the study of *"Cookbooks"*, as they are often called, written "by and for developers" and providing advice on how the development ought to be. Hence, this material will be examined as a second-order reflection on developers action, not constrained by the objectives put forward by a single, local developer group. Then, the norms identified will be compared with those of the user and the manufacturer paradigms..

Author (Year)	**Title**
Calishain and Dornfest (2003)	*Google hacks: [100 industrial-strength tips & tools]*
Karp (2003)	*eBay hacks : [100 industrial-strength tips & tools]*
Bausch (2003)	*Amazon hacks*
Erle and Gibson (2005)	*Google Maps Hacks*
Bausch (2006)	*Yahoo! hacks: Tips & Tools for Living on the Web Frontier*
Stay (2008)	*FBML essentials : Facebook Markup Language fundamentals*
Makice (2009)	*Twitter API: Up and Running*
Goldman (2009)	*Facebook Cookbook*
Balderas (2011)	*Paypal APIs: up and running*
Hudson (2012)	*eBay commerce cookbook: using eBay APIs*

Table 6.5: List of Developers' "Cookbooks" studied. (The books studied were edited by O'Reilly Media Inc).

Table 6.5 provides the list of the manuals studied. Those manuals are edited by O'Reilly Media[2]. Their study will be based on the categories already summarised in Table 6.3 (on page 104), similar to the first step. Moreover, during the second step, more attention will be given to *roles* and *means* that people developing services are assumed or advised to adopt by the books authors.

Data collection: "Sticky information" and innovation localisation

The second exploratory step benefits from the results of the first one: to intensify data gathering and generate more solid concepts (von Krogh et al., 2012). As discussed in Section 6.4, one of the results of the first step was the indication that a "sticky information locus" (von Hippel, 1994, 2005) of the Web site development lay in entrepreneurial knowledge

[2]O'Reilly Media is a reference in the developers milieu, known for editing comprehensive and action-based manuals for new programming languages. Besides, the editor, Tim O'Reilly, is the one that has popularized the term "Web 2.0".

(specifically regarding the business models of potentially competitive Web sites). Thus, in the second step, the loci of such knowledge will be studied in a more systematic way.

Hence, regarding the loci of service users sticky knowledge, my exploration will be based on the working proposition of distinction between the following types of users:

- *End - user, consumer*. This is the type of user, typically a "visitor" of the Web site or application buyer, who is consuming whatever the service provides (information or products).
- *End - user, seller*. This is the type of user corresponding to those who sell items on *eBay* or those who sell books in *Amazon*, using the service as a two-sided market Rochet and Tirole (2004); Eisenmann et al. (2006). Their profit comes from the transactions of the items that could be sold to other markets as well (e.g. second hand library stock).
- *Developer*. This is the type of user that has the skills to develop a software program. This skill gives him the potential to act upon the functionalities of a given platform and construct his own extensions.
- *User-Developer-Entrepreneur*. This is the type of user-developer who attempts to merchandise an application or a service, based on the features of a given service, such as for instance *Facebook* games, which end-users can play with their friends or localisation services, using *Google Maps* to display information to their clients.

Those types are schematically represented in Figure 6.3, along with the symbols which will be later used for the results analysis (on page 117). The different types are structured in reference to the two innovation models, the user and the manufacturer one: the upper side of the reversed pyramid includes the two types of end-users, the *consumer* and the *seller*. The bottom of the schema is dedicated to the *entrepreneur*, who is closer to a manufacturer logic. The *developer* role, linking the two, is put in the middle.

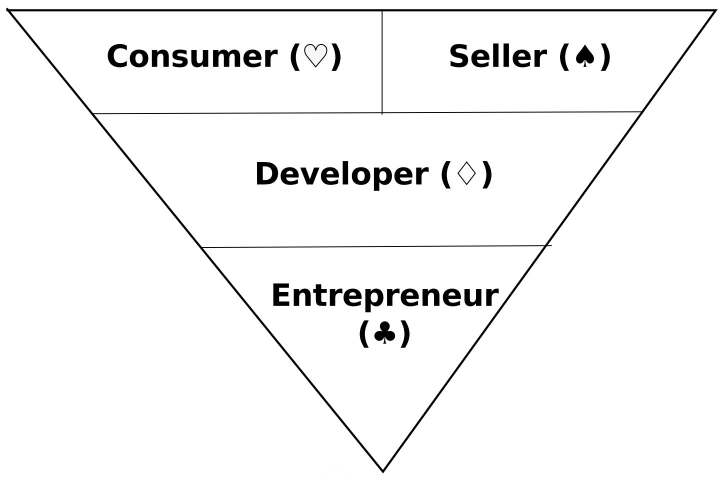


Figure 6.3: "Loci" of user "sticky" information.

Overall, there is a major qualitative difference between end-user knowledge and developer knowledge, even for what the authors of those books consider as simple tasks. For instance,

Figure 6.4: A simple Google Search form, resulting from the code provided by Calishain and Dornfest (2003).

an elementary example of a use of the popular *Google* Search service is the following *"very simple search form"* provided by Calishain and Dornfest (2003) in the beginning of their book:

```
  <!-- Search Google -->
<form method="get" action="http://www.google.com/search">
<input type="text" name="q" size=31 maxlength=255 value="">
<input type="submit" name="sa" value="Search Google">
</form>
<!-- Search Google -->
```

This form produces a box, like the one shown in Figure 6.4 in any given Web site, if added in its code. Visitors can then use the resulting form to use *Google Search*. While this is a "simple" example for the readers of the books under study, it is not as "simple" for the average *Google Search* user. It is thus evident that, even the elementary use case described in these books is different from the use cases one most end-users are familiar with.

Hence, the results presented here regard a second-level understanding of users' "sticky" information. Since on a first level of analysis the technicality of the knowledge provided by the books studied is *de facto* putting a line between the majority of end-users and the developers, my analysis will tend go beyond this cognitive distance to identify a possible conceptual distance: do developers share the same needs and context of use with the rest of end-users, or is there a "Great Wall" separating the two categories, in terms of desired innovations? In other words, do the concepts come from the developers incentives to satisfy their own needs as users or do they come from an entrepreneurial logic, to satisfy other users' needs, different from their own ones? This will be the secondary question enabling me to judge on the *"locus"* (von Hippel, 1994) of sticky information.

For instance, the example mentioned above is part of advice on how to integrate *Google Search* in a Web site. Hence, it is about advanced use of the service by user - developer - entrepreneurs (UDEs), to the extent that it is a commercial site: Users, because they use the service, developers, because they have to program to use it, and entrepreneurs because of their commercial activity, for which the site is designed.

Regarding the means used by the developers, the three following types will be taken into consideration:

- Free and open source software (FOSS), which is used as *"communal resource"* (von Krogh et al., 2003a) by open source software developer communities.
- Proprietary platforms, which are used as a resource by firms being *"complementors"* (Gawer and Cusumano, 2002) of platform leaders to develop their own peripherals.
- Web service interfaces (*Application Programming Interfaces - APIs*), which provide information flows to third party services.

The findings of this step will be discussed in Section 6.6.

6.4 Findings (First step): development for profit

The design and development process of the online application market was characterised by attributes of both the user and the manufacturer innovation models, without it fully "fitting" into any of the two models. Group members used "sticky" knowledge on entrepreneurship, while they were motivated by a potential commercial success of the Web site. The following paragraphs present the findings regarding the two different phases: the design & development phase and diffusion phase.

6.4.1 Design & Development Phase: group development for profit

Table 6.6 summarises the findings regarding the Design & Development phase of the online service. Unlike typical user innovation, as it became clear early on in the process, potential profit had an important role in the reasoning behind the design. The group was interested in developing an online application "mart", where *other developers were targeted as potential clients*. Hence, during the first hour of group discussion about the nature of the service to be developed, many of the concepts proposed regarded business-related requirements: the way developer clients would reimburse the service for hosting their application, as well as the means by which end-user clients would purchase available applications.

Design & Development Phase

User Paradigm		Observation
Goals		
	Satisfy own needs	✖
	Share efforts	✔
Norms		
	"Sticky" knowledge	✔
	Community-based development	✖
Manufacturer Paradigm		
Goals		
	Satisfy (potential) client needs	✔
	Speed, cost, quality	✔
Norms		
	Market research	✔

Table 6.6: Design and Development phase: online service development peculiarities observation.
Symbols:
✔ $\rightarrow$ *Observed;* ✖ $\rightarrow$ *The opposite was observed.*

During the same discussion, a "draft market research" was performed, with participants sharing information on existing application stores, such as the *Apple App Store* or the *Android Market Place*. Then, they shared their knowledge on those services from an entrepreneurial perspective: those who had relative informations to share, either had already sold applications through those services, or had friends who did. For instance, the project leader told us that when one sells applications through the *Apple App Store*, he has to earn an income over $200 per currency to get reimbursed: hence, one selling applications in the Canadian or the UK market cannot get reimbursed for these sales unless they reach the $200 threshold. In addition, he reported that the provider retained 30% commissions for each transaction. Therefore, one competitive advantage of the service would be the immediate reimbursement of application sellers, charging only 10% for our service.

Consequently, much of the group's "local" information (von Hippel, 1994, 1998; Lüthje et al., 2005), regarded *entrepreneurship* as a common experience. Yet, no systematic market research was made at any level of development, while group members often visited web sites to confirm information mentioned in the discussion.

Efforts were shared among the team on the basis of a task division. Nevertheless, those efforts were not shared with other users, neither online communities nor other participants in the event.

In parallel, the development speed was imposed by the event itself: the Web site was to be presented on the final day of the event. Eventually, what was ready was a prototype of the service, which was presented to the other participants.

The cost of the site development was very low. No member had to pay anything, while the leader of the group offered himself to "host" the project on a Web address (www.h5mart.com).

6.4.2 Diffusion Phase: no free revealing

In the Table 6.7, I summarise the observations as compared to the two paradigms, the user and the manufacturer ones.

Provided that the final product was only a prototype to be presented in the competition, there was no in-depth consideration on the diffusion of the service. Still, it is important to remark that, despite the fact that the project was left incomplete, the group members did not suggest that the source code should be revealed to other users, in order for them to further develop it. The source code having been shared among the members during the development, we left each other with the oral, friendly agreement that "everyone could do whatever he or she wants with it", with no further specifications regarding the terms of disclosure.

At the same time, the Web site remained available for online visitors to see. There, visitors could also read a short presentation by each member with a photo of him/her. We also exchanged contact information after the event. This "self-promotion" and "networking" of members reveals a desire for reputation, something however that could not be guaranteed on a large scale given the modesty of our final result. Other considerations, such as service maintenance, client support, service marketing or after sales services were not discussed.

6.4.3 Other observations: the use of FOSS and Web services

While a comparison of the group action with the user and the manufacturer innovation paradigms presents the picture of an autonomous, isolated group having no relationship with free and open source software (FOSS) communities or service providers, a more careful look into the source code of the project revealed a slightly different image.

Diffusion Phase		
User Paradigm		**Observation**
Goals		
	Reputation	✔
	Maintenance	☐
Norms		
	"Free revealing"	✖
	User networks diffusion	☐
Manufacturer Paradigm		
Goals		
	Merchandise	✔
	Client support	☐
Norms		
	Marketing	☐
	After sales services	☐

Table 6.7: Diffusion phase: online service development peculiarities observation.
Symbols:
✔ $\rightarrow$ *Observed;* ☐ $\rightarrow$ *Not observed;* ✖ $\rightarrow$ *The opposite was observed.*

On the one hand, for the development of the project, group members used both FOSS and Web services as components. Much of the instrumental part of the Web site was based on FOSS (particularly, *Python* and *jQuery* were used in the back-end and the user interface correspondingly).

On the other hand, different Web services were used to integrate features to our service. In particular, *eBay API* was used to enable financial transaction, *Gmail API* to enable client identification and *Facebook API* for allowing visitors to comment on the applications hosted on the Web site.

Hence, the low development cost for the group, came out of exploitation of existing "modules", taken either from FOSS or existing Web services. At the same time, our project consisted in an extension of each of those platforms separately and all together, that would have contributed (if achieved) to the extension of their user base or the enrichment of functionalities for the existing ones.

6.5 Discussion (First step): keeping a foot in both camps

The group observed used methods of both the user and the developer paradigms (Baldwin and von Hippel, 2011; Raasch and von Hippel, 2012), in a rather opportunistic mode. "Local" information (von Hippel, 1998, 1994; Lüthje et al., 2005) coming from the members' experience of application development was used though not to develop something for their own use (e.g. a community-based "pool" of applications, such as the open source software repositories, studied by von Krogh et al. (2006)), but to develop a commercial application for the market, thus seeing their peers as potential clients.

Moreover, despite the fact that, within the time-frame of three days, the Web site wasn't completed, group members didn't choose to open-source it, to "freely reveal" their creation, to share the efforts with a wider user community, as the private-collective model (von Hippel and von Krogh, 2003, 2006) would suggest. Instead, they shared both efforts and source code within the closed, intimate group, with no further specification on its potential disclosure terms.

Moreover, there were variations on what was observed in the two main paradigms. On the one hand, instead of community reputation effects observed within open source communities, according to the implication of each individual (Lakhani et al., 2002; von Krogh et al., 2003a; Auray, 2004, and others), what was observed was rather a simple "socialising", a will to meet (and be met by) new people active in the field.

On the other hand, instead of the market research used by enterprises, the group produced and used a "draft benchmarking" of other similar services that existed, based on personal experience or even rumours, to be confirmed in place, using the Web as a resource.

Group members, during the design and development process,"kept a foot in both camps", not only in regards to their collaboration. They also made extensive use of products coming from both paradigms. In fact, tools and components of the both models were used as resources for the service development: the developers used open source software as well as Web services APIs to structure their prototype.

The above suggests that there may be two significant implications for business practice and theory. Firstly, a modification of the actual narrative of user-entrepreneur innovation is evoked. Assuming that group members were experienced users (von Hippel, 1983) of open source and Web platforms all together, we should still recognise that this experience in question is of a radically different nature, when compared to the end-users one: they have developed an expertise on services, such as *Facebook*, not through intensive use, but rather through purposive design space exploration for the development of applications. Consequently, they were not "accidental" entrepreneurs, as identified by Shah and Tripsas (2007) regarding user-entrepreneurs, but they were in a more typical entrepreneurship configuration, in a conscious quest to make a business.

Secondly, ease of service prototyping in the particular field is rather surprising. While costs (entry costs, prototype costs, development costs) are often cited as barriers to entrepreneurship (Teece, 1986; Willcocks and Plant, 2001; Sawhney, 1998; Iansiti and Levien, 2004; Eisenmann et al., 2006; Osterloh and Rota, 2007), such costs were observed to be insignificant and largely limited to personal time investment. In turn, this fact has implications to the overall competitive environment for providers.

Research in platforms has focussed on competition between different providers. The use of modules coming from different providers has also been studied from a platform competition perspective. Gawer and Cusumano (2002) has proposed that "complementors" should try to anticipate platform providers' moves to try to *"assess who will win the war for platform leadership"* (Gawer and Cusumano, 2002, p. 54), as complementors commit resources to their own innovations. More recently, Eisenmann et al. (2011) proposed more complex strategies, where a provider of a given platform can be at the same time a complementor of an other one. Using features of one platform (say *eBay*) on another (say *Facebook*)[3] is one possible strategy to "tap in" to competitors user bases. Platform competition has been the case since enterprises started using platforms. Still, the fact that those using existing platforms

[3]Such a possibility is enabled through the APIs.

to innovate needn't be enterprises, but can be people prototyping a new application at a very low cost, may indicate a change in the overall "rules of the game".

6.6 Findings (Second step): different developers configurations

Tables 6.8 (on page 117) and 6.9 (on page 119) show the outcomes of the developer "Cookbooks" study, regarding the design and development phase and the diffusion phase correspondingly, as compared with the user and the manufacturer innovation paradigms action goals and norms (already outlined in Table 6.3, page 104).

The results are categorised by book, as well as by role assumed by authors when proposing an advice. Hence, advice assuming the role of *end-users consumers* are represented in the tables by the rows beginning with the mark "♡", advice for *end-user sellers* are represented by the rows beginning with the mark "♠", advice for developer with the mark "♢" and, finally, advice for user-developer-entrepreneurs with the mark "♣". The entrepreneur role is assumed here through the commercialisation of an application developed by the actor, not the selling of an object manufactured elsewhere (as in the case of the *seller*).

When one of the goals or norms is explicitly advised by a book author, I noted it with the "✔" mark, in the corresponding row. When such advice is not mentioned at all, I used the mark "☐". Finally, when authors explicitly disapproved a specific goal or norm, I noted it that using the mark "✖". The last column summarises the findings of action norms for each role, marking when they suggest an end-user role (U), a developer role (D) or an entrepreneur one (E).

In the following paragraphs I present and analyse the findings of this exploration. As *"a [new] phenomenon is typically defined by what it is not"* (Lewis, 2000b; von Krogh et al., 2012), I will start with the elements of the user innovation model that are absent in the specific innovation field, and then I will proceed to the description of those elements of both user and manufacturer innovation paradigms that are present, albeit taking a particular form.

Table 6.8: Design & Development Phase: "Cookbooks" analysis.
Symbols:
End-User (consumer): ♡ ; *Seller (of items)*:♠ ; *Developer*: ♢ ; *UDE*:♣ ;
Explicitly advised: ✔ ; *Not mentioned*:☐ ; *Explicitly disapproved*:✖.

Book: **Author (Year),** **Service**	Role	**User Paradigm**	**Goals**	Satisfy own needs	Share efforts	**Norms**	"Local" knowledge	Community-based development	**Means**	FOSS	**Manufacturer Paradigm**	**Goals**	Satisfy client needs	Speed, cost, quality	**Norms**	Market Research	**Means**	Proprietary software	Web APIs	**Roles (summary)**
	Criterion No:																			
	1			*2*	*3*		*4*	*5*		*6*			*7*	*8*		*9*		*10*	*11*	
Calishain and Dornfest (2003),	♡			✔	☐		✔	☐		☐			☐	☐		☐		☐	☐	**U**
	♠			☐	☐		☐	☐		☐			☐	☐		☐		☐	☐	
	♢			✔	☐		✔	☐		✔			✔	✔		☐		✔	✔	**D**
Google Search	♣			☐	☐		✔	☐		☐			✔	☐		☐		☐	✔	**E**
Karp (2003),	♡			✔	☐		✔	☐		☐			☐	☐		☐		☐	☐	**U**
	♠			✔	☐		✔	☐		☐			✔	☐		☐		☐	☐	**U**
	♢			✔	☐		✔	☐		✔			✔	✔		☐		✔	✔	**D**
eBay	♣			☐	☐		☐	☐		☐			☐	☐		☐		☐	☐	
Bausch (2003),	♡			✔	✔		✔	☐		☐			☐	☐		☐		☐	☐	**U**
	♠			✔	☐		✔	☐		☐			✔	☐		✔		☐	☐	**U**
	♢			✔	☐		✔	☐		✔			✔	✔		✔		✔	✔	**D**
Amazon	♣			☐	☐		☐	☐		☐			☐	☐		☐		☐	☐	
Erle and Gibson (2005),	♡			✔	☐		✔	☐		☐			☐	☐		☐		☐	☐	**U**
	♠			☐	☐		☐	☐		☐			☐	☐		☐		☐	☐	
	♢			✔	☐		✔	☐		✔			☐	✔		✔		☐	✔	**D**
Google Maps	♣			☐	☐		☐	☐		☐			☐	☐		☐		☐	☐	
Bausch (2006),	♡			✔	☐		✔	☐		☐			☐	☐		☐		☐	☐	**U**
	♠			✔	☐		✔	☐		☐			✔	☐		☐		☐	☐	**U**
	♢			☐	☐		✔	☐		✔			✔	✔		✔		✔	✔	**D**
Yahoo	♣			✔	☐		✔	☐		☐			☐	☐		✔		☐	✔	**E**
Stay (2008),	♡			✖	☐		✖	☐		☐			☐	☐		☐		☐	☐	
	♠			✖	☐		☐	☐		☐			☐	☐		☐		☐	☐	
	♢			✖	✔		✔	☐		✔			✔	✔		☐		✔	✔	**D**
Facebook	♣			✖	☐		☐	☐		☐			✔	☐		✔		☐	✔	**E**
Makice (2009),	♡			✔	☐		☐	☐		☐			☐	☐		☐		☐	☐	
	♠			☐	☐		☐	☐		☐			☐	☐		☐		☐	☐	

. . . continued in the next page . . .

. . . continues from previous page . . .

Book:	Role	**User Paradigm**	**Goals**	Satisfy own needs	Share efforts	**Norms**	"Local" knowledge	Community-based development	**Means**	FOSS	**Manufacturer Paradigm**	**Goals**	Satisfy client needs	Speed, cost, quality	**Norms**	Market Research	**Means**	Proprietary Software	Web APIs	**Roles (summary)**
Author (Year),	*Criterion No:*																			
Service	*1*			*2*	*3*		*4*	*5*		*6*			*7*	*8*		*9*		*10*	*11*	
	♢			✖	✔		✔	☐		✔			✔	✔		☐		✔	✔	**D**
Twitter	♣			☐	☐		☐	☐		☐			✔	☐		✔		☐	✔	**E**
Goldman	♡			✖	☐		✖	☐		☐			☐	☐		☐		☐	☐	
(2009),	♠			✖	☐		☐	☐		☐			☐	☐		☐		☐	☐	
	♢			✖	✔		✔	☐		✔			✔	✔		☐		✔	✔	**D**
Facebook	♣			✖	☐		✔	☐		☐			✔	☐		✔		☐	✔	**E**
Balderas	♡			☐	☐		☐	☐		☐			☐	☐		☐		☐	☐	
(2011),	♠			✔	☐		✔	☐		☐			✔	☐		☐		☐	✔	**U**
	♢			✔	☐		✔	☐		✔			✔	✔		☐		✔	✔	**D**
PayPal	♣			☐	☐		☐	☐		☐			✔	☐		☐		☐	✔	**E**
Hudson	♡			☐	☐		☐	☐		☐			☐	☐		☐		☐	☐	
(2012),	♠			✔	☐		✔	☐		☐			✔	☐		✔		☐	✔	**U**
	♢			✔	☐		✔	☐		✔			✔	✔		☐		✔	✔	**D**
eBay	♣			☐	☐		☐	☐		☐			☐	☐		☐		☐	✔	**E**

End of table

Table 6.9: DIFFUSION PHASE: "Cookbooks" analysis.
Symbols:
End-User (consumer): ♡ ; *Seller (of items)*:♠ ; *Developer*: ♢ ;
UDE:♣ ;
Explicitly advised: ✔ ; *Not mentioned*:☐ ; *Explicitly disapproved*:✖.

Book:	**Role**	**User Paradigm**	**Goals**	Reputation	Maintenance	**Norms**	"Free revealing"	User networks diff.	**Manufacturer Paradigm**	**Goals**	Merchandise	Client Support	**Norms**	Marketing	After sales services	**Roles (summary)**
Author (Year), Service	*Criterion No:* *1*			*2*	*3*		*4*	*5*			*6*	*7*		*8*	*9*	
Calishain and Dornfest (2003),	♡			☐	☐		☐	☐			☐	☐		☐	☐	
	♠			☐	☐		☐	☐			☐	☐		☐	☐	
	♢			☐	☐		☐	☐			☐	☐		✔	☐	**D**
Google Search	♣			☐	☐		☐	☐			☐	☐		✔	☐	**E**
Karp (2003),	♡			☐	☐		☐	☐			☐	☐		☐	✔	**U**
	♠			✔	☐		☐	☐			✔	☐		✔	✔	**D**
	♢			☐	☐		☐	☐			☐	☐		☐	☐	
eBay	♣			☐	☐		☐	☐			☐	☐		☐	☐	
Bausch (2003),	♡			✔	☐		☐	☐			☐	☐		☐	✔	**U**
	♠			✔	☐		☐	☐			✔	☐		✔	✔	**D**
	♢			☐	☐		☐	☐			☐	☐		☐	☐	
Amazon	♣			☐	☐		☐	☐			☐	☐		☐	☐	
Erle and Gibson (2005),	♡			☐	☐		☐	☐			☐	☐		☐	☐	
	♠			☐	☐		☐	☐			☐	☐		☐	☐	
	♢			☐	☐		☐	☐			☐	☐		☐	☐	
Google Maps	♣			☐	☐		☐	☐			☐	☐		☐	☐	
Bausch (2006),	♡			☐	☐		☐	☐			☐	☐		☐	☐	
	♠			☐	☐		☐	☐			✔	☐		☐	☐	**U**
	♢			☐	☐		☐	☐			☐	☐		✔	☐	**D**
Yahoo	♣			☐	☐		☐	☐			☐	☐		☐	☐	
Stay (2008),	♡			☐	☐		☐	☐			☐	☐		✔	☐	**U**
	♠			☐	☐		☐	☐			☐	☐		☐	☐	
	♢			☐	✔		☐	☐			☐	☐		☐	☐	**D**
Facebook	♣			☐	☐		☐	✔			✔	☐		✔	☐	**E**
Makice (2009),	♡			☐	☐		☐	☐			☐	☐		✔	☐	**U**
	♠			☐	☐		☐	☐			☐	☐		☐	☐	
	♢			✔	✔		☐	☐			☐	☐		☐	☐	**D**
Twitter	♣			✔	☐		☐	✔			✔	✔		✔	✔	**E**

. . . continued in the next page . . .

. . . continues from previous page . . .

Book:	**Role**	**User Paradigm**	**Goals**	Reputation	Maintenance	**Norms**	"Free revealing"	User networks diff.	**Manufacturer Paradigm**	**Goals**	Merchandise	Client Support	**Norms**	Marketing	After sales services	**Roles (summary)**
Author (Year),	*Criterion No:*															
Service	*1*			*2*	*3*		*4*	*5*			*6*	*7*		*8*	*9*	
Goldman (2009),	♡			☐	☐		☐	☐			☐	☐		✔	☐	**U**
	♠			☐	☐		☐	☐			☐	☐		☐	☐	
	♢			☐	✔		☐	☐			☐	✔		☐	✔	**D**
Facebook	♣			☐	✔		☐	✔			✔	✔		✔	✔	**E**
Balderas (2011),	♡			☐	☐		☐	☐			☐	☐		☐	☐	
	♠			☐	☐		☐	☐			✔	☐		☐	☐	**U**
	♢			☐	☐		☐	☐			✔	☐		☐	☐	**D**
PayPal	♣			☐	☐		☐	☐			☐	☐		☐	☐	
Hudson (2012),	♡			☐	☐		☐	☐			☐	☐		☐	☐	
	♠			☐	☐		☐	☐			✔	✔		✔	✔	**U**
	♢			☐	☐		☐	☐			☐	☐		☐	☐	
eBay	♣			☐	☐		☐	☐			☐	☐		☐	☐	

End of table

6.6.1 No innovation revealing

In the books examined, there is a complete absence of two major action norms of the private-collective model as developed and used in numerous fields, including the open source phenomenon (Raymond, 1999; Stallman, 2002; von Krogh et al., 2003b; von Hippel and von Krogh, 2003, and others). On the one hand, there was no advice or indication for readers to *freely reveal* their own creations, as shown in the *Column 4* of the Table 6.9. On the other hand, there was no sign either of suggestions for community-based development, as shown in the *Column 5* of the Table 6.8. The fact that no book refers to such a process reveals a "common ground" among the actors of the field that their activity is *not about freely revealing* their creations to user communities, therefore a particular description - or even mention - of how to do it is not among the requirements of such a book.

> Hence, there is no evidence of collaboration for application development on the basis of existing services: private development is commonly considered as a rule.

A more subtle look into these books, though, suggests a *sense of community*, yet not about innovation sharing itself. What developers have in common is the fact that they share development methods and tools. This sense comes from the common development methods, requiring the use of an expanded set of tools.

6.6.2 Not an enterprise-based development process

While taking a distance from the user paradigm, in the sense that developments are not to be freely shared, the development process described is not one of an enterprise either. Typically enough, the books do not provide explicit advice on development speed, quality and cost performance, criteria which are omni-present in enterprise new product development in business (Clark and Fujimoto, 1991; Iansiti, 1993; Bhattacharya et al., 1998, and others). Instead, authors qualify operations as "easy to do" or "not so easy to do". Overall, the reader, as portrayed by these books, is not a product manager running a team of developers, he is a single developer.

Such qualifications are made along with the discussion of different means in the disposal of developers. These means can be open source software or proprietary platforms, as well as the service API itself.

6.6.3 Application development: individual "innovation" palette, common method

Application development, as presented in the books examined, is a tools - skills individual method, a developer's continuous effort to master new tools and use them for his own good. Goldman 's introduction to his book, is very typical in the way it presents the "new opportunity" opened up for developers by the *Facebook* platform:

> *The barrier to entry [in Facebook application development] is very low and requires only that you retrain some of your existing web development skills (or learn some basic new ones), all of which you can master with this very book (Goldman, 2009, p. 4).*

Overall, far from describing an organised enterprise development process, these books describe a solitary one. Developers undertake - what during the interviews with actors of the field was referred to as a *side project*, different from one's *day job* - a personal project that makes use of both the skills of a specific person and a variety of tools, the latter being provided by services as well as by free and proprietary software.

In this realm, a sense of community does not come about as a result of a communitarian spirit, as literature suggests for the open source community (Benkeltoum, 2008, and others), but in an utilitarian mode, being "remixed" with other tools, Web APIs and proprietary platforms. Hence, one of the very rare references to the participation in a developer community is made by Goldman, in the context of free software use (namely *PHP libraries*) as an auxiliary means for the development of a *Facebook* application. A different sense of community is referred to by Makice (2009), though not in what regards common modules. He refers to the *Twitter* developer community in a broad sense, on the basis of a common concern on "what kind of application" could be created, thus entering more into a perspective of qualitative "benchmarking", using examples and in no way as common development. A third indication of a community sense comes from the references to online forums that the provider has put into the disposal of developers, where they can ask for further information or address questions on particular problems they face.

The developer forum indication will be further explored in the Chapter 17. In the paragraph 6.6.2, I will take a deeper look into the two other indications of a community sense, the sharing of common concerns and the use of *FOSS* modules.

6.6.4 Types of "sticky information" and "innovation loci" identified

Overall, exploration results propose that the development activity of user - developer - entrepreneurs (UDEs) undertakes the costs of innovation on the basis of sticky knowledge located both in the user (user need related) and the manufacturer (technology related) sides. In this paragraph I will further analyse the types of sticky knowledge identified, corresponding to different types of users.

Depending of the type of service, the nature of this information and the technical subtract where innovation is localised differs, as summarized in the Table 6.10 and illustrated in Figure 6.5.

Development actor	Service type	Sticky information	Innovation locus	Books
Consumer-Developer	Instrumental	Functional	Web site development	Calishain and Dornfest (2003); Erle and Gibson (2005); Bausch (2006)
Seller-Developer	Market	Commercial	Seller tools development	Karp (2003); Bausch (2003); Balderas (2011); Hudson (2012)
Developer-Entrepreneur	Online identity	Social	Application development	Stay (2008); Makice (2009); Goldman (2009)

Table 6.10: Sticky information and locus in Web services development.

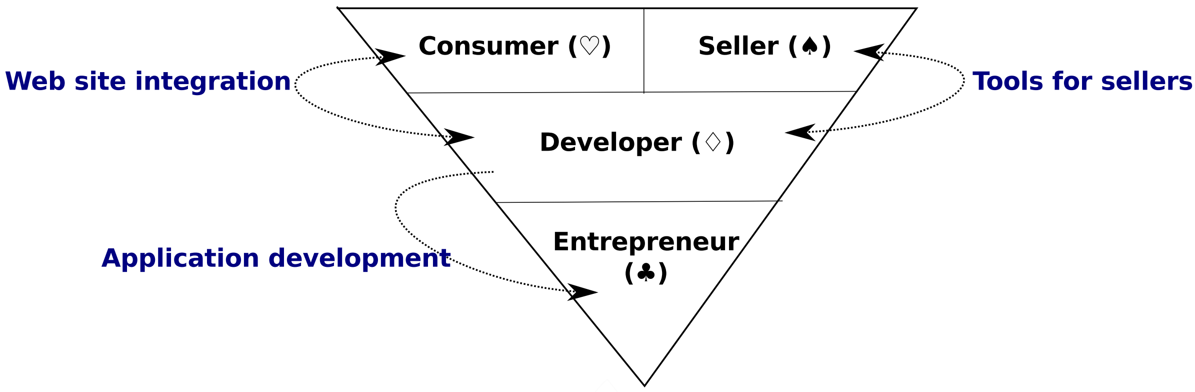


Figure 6.5: Sources of use related sticky information in the field of Web services development.

In most developer books studied (Calishain and Dornfest, 2003; Karp, 2003; Bausch, 2003; Erle and Gibson, 2005; Bausch, 2006; Balderas, 2011; Hudson, 2012), reference is made to lead user knowledge as a requirement for development. In general, these books enrol a development process in the continuity of the advanced personal use, enhanced by *personal development skills*. Advice suggesting the normative exploitation of end-user "sticky" information in development process comes from two different end-user perspectives: the consumer and the seller ones.

Development using instrumental services.

More specifically, there are, on the one hand, books tackling the issue of development on the basis of rather more *instrumental services* (such as search engines, mapping or money transfer services). There, authors explicitly provide "tips and tricks" for information and functionalities for consumers. Sticky knowledge shared by the authors concerns the *functional* aspects of the service. Consumers could benefit from this knowledge as well, as early chapters reveal information that can be exploited without having programming skills.

Still, while there is a recognition of the use as the place where valuable for development, i.e. "sticky" knowledge resides, the authors do not assume that their readers will give away their creations for free. Those books typically consider developers as "Webmasters", already developing a Web site and wanting to integrate functionalities and more competition from other services (e.g. it appears higher in search engines' ranking), hence for their own use: the locus of development is, in this case, their own Web site. As a result, their Web site becomes richer in functionalities, by integrating the service provider's ones,

Development using Market services

On the other hand, cookbooks referring to *two sided market services*, provide in their first chapters "tips and tricks" regarding *commercial* knowledge, such as best online selling practices. Typically enough, the second edition on *eBay* is a lot more focussed on a public of sellers than on a public of buyers. These books mainly consider developers as item sellers, who create tools for themselves. Hence, entrepreneurial activity resides in item selling, thus development comes about as an auxiliary process for entrepreneurship and not as a potentially lucrative activity as such. In this case, the locus of innovation is the seller's instruments, as he uses both information and tools available for his own good, to be more efficient in his selling activity.

For the list cited in the "Seller-Developer" raw of the Table 6.10, only the last chapter of Balderas (2011) addresses development as an entrepreneurial activity as such, when providing information on how to develop smart phone payment applications using *PayPal*. Hence, in the particular field of online market services, the locus of innovations is found in the development of applications for merchandise only relatively recently.

Development using online identity services

However, there is also a third way, in what regards developer - end user knowledge and relationships. In the *Facebook* case, developers are explicitly called upon not to develop an application for their own use. While the first edition (Stay, 2008) is rather subtle in advising this detachment (*"Always look at your application through the eyes of your users"*), the second one (Goldman, 2009) is less so:

> ***Who should I build for?*** *Some of the most successful software products are born out of a need their developers felt wasn't satisfied elsewhere. If that's the case for you, build for yourself and your friends. However, many of you will be reading this book because you hope to make money from your work, in which*

case you need to consider your audience more carefully. Sometimes you and your friends aren't the ones who are going to pay for your villa in Maui, so make sure you spend the time to understand who is (Goldman, 2009, p. 11).

Hence, using the profit motivation, one is advised to take a distance from his own intimate circle and study the public to which his/her application will be addressed, using typical market research methods, such as polls.

In a similar spirit, the book on *Twitter* (Makice, 2009) also assumes that developer's "sticky" knowledge is not sufficient for the creation of *Twitter* applications. On the one hand, it advises developers to *"understand the culture of Twitter users"*, a culture originating from previous online chatting communities. On the other hand, it lists a series of applications that are representative of this culture, to inspire readers to begin development. The question is "what kind of application" could there be to address a need, and the author performs a qualitative "benchmarking", using examples and in no way as common development, to illustrate possible trajectories.

6.7 Discussion (second step)

This study has shown a difference regarding the action norms of developers as compared to the models reviewed and proposed by von Hippel and von Krogh (2003, 2006). These differences are found both in the cognitive conditions and the implicit modes of value appropriation's norms of development. Moreover, this research helped refine the transition modalities from user to entrepreneur roles, in the case of Web-based application development. While the distinction of user and manufacturer paradigms suggests the division of innovations between the class of users and the one of manufacturers, according to the sticky information each class possesses (von Hippel, 2005, p. 70), developers who use online services to innovate exploit knowledge from both classes.

Globally, what is indicated by my investigation is the emergence of an intermediary way of action, placed in-between the user and the manufacturer paradigm, where individuals may exploit available resources coming from a set of service providers for their own good. In the following paragraphs I will review the characteristics of what could be called a *personal investment* model, as compared to the ones of the known paradigms.

Table 6.11 summarizes the arguments of comparison of the field findings to the models of von Hippel and von Krogh, which will be discussed in paragraph 6.7.1. Pagraph 6.7.2 elaborates on the transition modes from user to developer states, describing the three different configurations of developers identified.

6.7.1 Cognitive conditions: neither free revealing, nor R&D, an individual "innovation palette"

In respect to developers of online applications, both as met in the Web site development challenge and as studied through the "cookbooks", there is no structured R&D and new product development process, as in the case of enterprises (Iansiti, 1993, 1998; Verganti, 1999; Bayart et al., 2000, and others). Instead, innovation capacity (Nassimbeni, 2001) of individual developers depends on their ability to use a diverse set of tools. Web APIs provide the chance to innovate on a given service, yet additional open or closed software tools are indispensable for the development process. On these bases, developers use both use-related and technology related "sticky information".

	Knowledge		Value Appropriation	
Model	**Production**	**Sharing**	**Private**	**Public**
Private investment	R&D	Patent	Monopoly	Loss
Collective (public) action	R&D, public research	Knowledge Control Relinquishing	Reputation, monetary subsidies	"Common pools" resources
Private-collective	"Sticky" knowledge	"Free revealing"	Reputation, Learning, problem resolution	"common pools", common solutions
"Personal investment"	User and manufacturer "sticky" knowledge, personal skills	No revealing	Competition	Public-private pools

Table 6.11: A "Personal investment" model: comparison with the models outlined by von Hippel and von Krogh (2003).

As I have already stated, the distinction between user and manufacturer innovation is based on the distinction between the sticky information, the knowledge that is costly to transfer from one actor to another (von Hippel, 1994; Ogawa, 1998; von Hippel, 2005). Still, what I observed in action and is further supported by the study of books addressed to developers is that in the process of developing new services or tools, UDEs normally use a set of diverse knowledge resources, constituting a plural knowledge base.

Globally, what distinguishes developers from common end-users, is their possession of technological knowledge, usually attributed to manufacturers expertise (Ogawa, 1998; von Hippel, 2005). At the same time, they often exploit use context specific knowledge, typically obtained by end-users.

An important part of the sticky information utilized for development comes from the end-users' "best practices". When developers use instrumental services for development (such as search engines, mapping services or payment ones) this goes along with the use of functional end-user sticky information. The use of market services (such as bidding or buying services) is related to the use of commercial end-user sticky information. Finally, the online identity services (enabling the expression of their users) utilize sticky information on how social interaction takes place.

Exploitation of end-user sticky knowledge

Hence, sellers may develop tools for themselves (in services such as *eBay* or *Amazon*), while Webmasters may extend the service they provide to information consumers by integrating features of another service provider (such as *Yahoo Search* or *Google Maps*).

However, when it comes to services were the gap between developers and users is big, such as the ones enabling the construction of an online identity, what is recommended to developer is to take a distance from their own needs and try to seize the needs and desires of potential clients, thus adopting a clearer entrepreneurial posture.

Exploitation of manufacturer sticky knowledge

An omni-presence of service interfaces' use was noted both in the setting of a Web service development and the study of the cookbooks. This information is typically service-specific, as the functionalities proposed by different services are different.

These application programming interfaces (APIs) are publicly available. Everyone can access their use, as long as he/she knows how to use them. That constitutes a particular kind of synergy, as no authorisation is needed. Developers using them do not enter in a typical partnership relationship (Midler et al., 1997; Segrestin, 2006) with the provider, but they agree instead in the terms of service.

In addition, they also use proprietary software as well as free software, in a rather opportunistic mode, in order to complete their own creation.

Similar phenomena have been studied by the platform literature, through the lenses of modularity (Baldwin and Clark, 2000), platform leadership (Gawer and Cusumano, 2002) or platform envelopment (Eisenmann et al., 2011). Still, my exploration reveals an actor who keeps a foot in both manufacturer and user camps, allowing for a different perspective on the phenomenon of Web services innovation to emerge. While design costs have been at the centre of interests of those studies, the low design costs of User-Developer-Entrepreneurs opens up a new field for entrepreneurship.

For this lowering of design costs there are both subjective and objective factors. Subjective factors lie in the utilisation of user sticky knowledge (von Hippel, 1994), while the objective factors lie in the existence of new sets of design possibilities (Baldwin et al., 2006), simultaneously coming from different platforms. Overall, circumstances lead to the ability of developers to construct a *plural knowledge base*, utilising both user and manufacturer knowledge. This dimension will be reviewed in the following paragraphs.

6.7.2 Three configurations of the developer figure

The study of the "Cookbooks" revealed an actor who does not enter into the typical B-B or B-C frameworks. The three different configurations of the developer, as related to the user and the entrepreneur roles, are schematically illustrated in comparison to the user and manufacturer paradigms in Figure 6.6. These configurations are clearer when seen through the design and development criteria set out early on in this chapter, than the diffusion ones. The books being more about how to create value and less about how to exploit it, diffusion tips were rare.

Hence, as shown in the last column of the Table 6.8, there are three configurations of developers. A first configuration is the one of User - Developers (UD). It regards a public that uses the Web service tools (*APIs*) along with other development tools (FOSS or proprietary software) in order to create tools for themselves, to enrich their use experience or to be more efficient in it. This is the case for the books by Karp (2003) on *eBay*, where the developer is viewed as either a seller or a buyer of items on auction, but not a seller of applications for other users. This is also the case for the book by Bausch (2003) on *Amazon*. The book of Erle and Gibson (2005) on *Google Maps* is even more centered on use issues, as it judges it necessary to illustrate the value of use for *Google Maps*, being a rather new service at the time of its publication. This category is thus closer to the typical user innovation model (von Hippel, 1975), even though sharing practices were not explicitly discussed in those books.

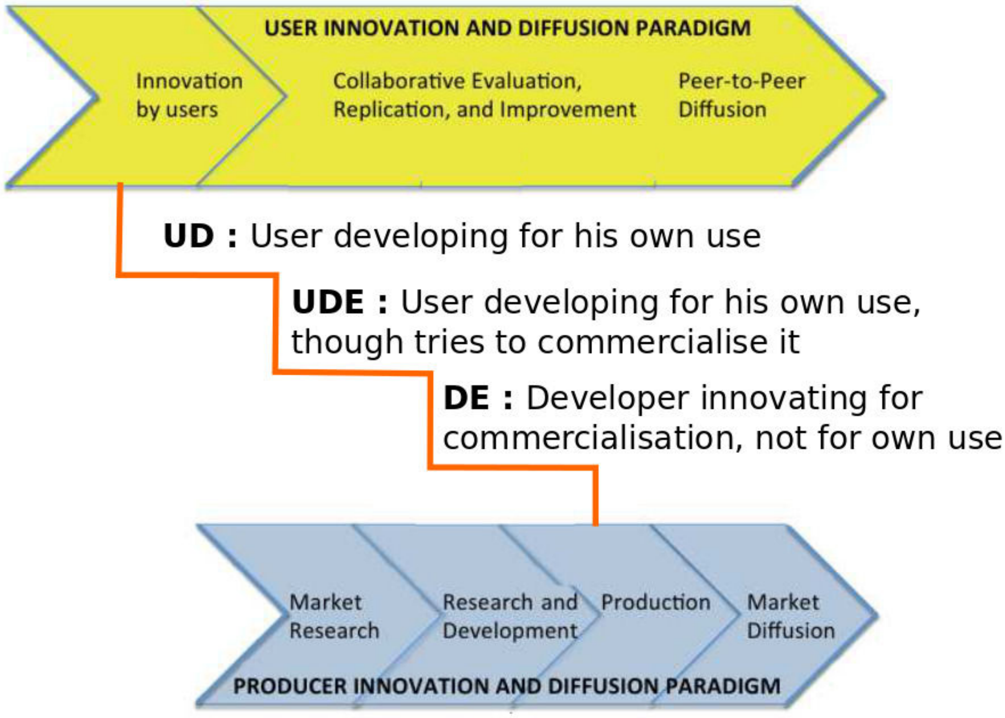


Figure 6.6: Three developer configurations: UD, UDE, DE.

A second configuration is the User - Developer - Entrepreneur. Here, applications are initially built through a similar reasoning to UDs, that is the serving of one's own need, or the "buy or innovate" dilemma, as referred to by the literature (von Hippel, 2005). Still, this innovation supports an entrepreneurial activity. On the one hand, that can be done through its use to enrich an ongoing business. That is generally the spirit of the books about search engines (Calishain and Dornfest, 2003; Bausch, 2006), where functionalities such as searching can be embedded into a commercial site and add value to it. On the other hand, there is the option to merchandise the creation as a separate good. This possibility is more or less explored by the books on *PayPal* (Balderas, 2011) and *eBay* (Hudson, 2012), though the user condition is still largely present.

A third configuration is the Developer - Entrepreneur, where a complete distinction between developers and end-users occurs. The needs or desires to satisfy or evoke are no longer the ones of the developer, they are the ones of his potential clients. Hence, books on *Facebook* and *Twitter* (Stay, 2008; Makice, 2009; Goldman, 2009) take as granted that their readers do not develop for satisfying their own needs, but for money. Hence, a call is made by the authors to "understand" the public to which they address, before they start designing, while Goldman goes further, being rather tough on developers that innovate for themselves. DEs are thus in the frontiers of the two paradigms, the user and the manufacturer one (Raasch and von Hippel, 2012), though clearly not belonging to any of them.

6.8 Conclusion

Literature has focused on the opposition and the complementarity between user and manufacturer innovation (von Hippel and von Krogh, 2003, 2006; Baldwin et al., 2006; Raasch and von Hippel, 2012). While these two modes of innovation are clearly defined by the ways each actor benefits from innovation (for use or for profit), in the specific field of Web services innovation I identified an actor operating in-between the two modes.

The contribution of this study, beyond the identification of this actor, lies in the proposition of three different configurations of developers of online applications: user - developers, developing for their own use, user-developer-entrepreneurs, attempting to commercialize such creations and developer-entrepreneurs, directly creating for commercialization as opposed to innovating for use. As this activity implies the use of a multitude of tools, some provided by enterprises, others by communities, these tools are a meeting point for diverse innovation actors.

Strategies on innovation should thus include the existence of this actor, who may innovate in a low cost manner, as entrepreneurship can catalyse industrial development.

Chapter 7

Modus operandi exploration: the case of an *eBay* seller application

Contents

7.1 Introduction

In order to further report (von Krogh et al., 2012) on the potential distinctive characteristics of user-developer-entrepreneurs, as explored in Chapters 5 and 6, the case of a single application, that was developed and marketed by a single UDE, is studied in the current chapter.

While in the previous chapter I used a rather modular approach, examining the characteristics of UDE action through a norm-by-norm comparison, this case will be explore the coherence of UDE activity during the passing from use-based to market-based innovation, and more particularly the boundaries between UD, UDE and DE roles.

Using the story of a UDE developing and commercialising an *eBay* seller application as a "spare-time" project, I further explore the action of this figure, comparing it to the literature on open source, on the one hand, and user-entrepreneurship, on the other.

I conclude that this activity, placed in-between the user and the producer paradigms in the specific field of online application development, benefits from both.

7.2 Theoretical concepts

As we have already discussed in previous chapters, at the core of the user innovation approach resides the notion of "sticky" information (von Hippel, 1994, 2005). Within this framework, users are more likely to innovate by drawing on use context related information, while enterprises are more likely to innovate by drawing on technological information. The "stickiness" of these two kinds of information, according to von Hippel, regards the high transfer costs, from one actor to another (as already reviewed in paragraph 6.2.1). Thus, it is costly for enterprises to learn about the problems lead users face, while it is also costly for lead users to master the technology of a given product. In addition, users face the "buy or innovate" dilemma, according to which use related innovation is more beneficial than the purchasing of a product that has similar attributes. Moreover, user innovation regards specific attributes of a given product, the ones directly utilised by users (von Hippel, 2005).

In Chapter 6, I suggested that UDEs innovate by using both user and manufacturer "sticky" information. In this endeavour, they have at their disposal tools (APIs) that come from a variety of service providers. I also suggested that the entry costs being relatively low in the sector of Web services, the development of a service or an application can be advanced, at least at a certain level, by a single individual. The current chapter further explores this potential through a single case study.

Moreover, I generated *"concepts that can serve as filter for further data gathering"* (von Krogh et al., 2012), namely the three configurations illustrated in Figure 6.6, quoted in Figure 7.1 for convenience. Comparing the distinction between user and manufacturer innovation models (von Hippel and von Krogh, 2003, 2006; Raasch and von Hippel, 2012), concluding chapter 6 I suggested the distinction of three types of developers: user-developers (UD), creating tools for their own use, user-developer-entrepreneurs (UDE), attempting to commercialise those tools, and developer-entrepreneurs (DE), creating application for commercialisation, as opposed to the development for own use.

The current chapter further consolidates the three configuration concepts (UD, UDE, DE), by examining a case illustrating an attempt to pass from the user to the manufacturer paradigm. Two settings that utilise the user innovation approach and will be used to discuss the experience of UDEs in this chapter are the one on user-entrepreneurs and the open source

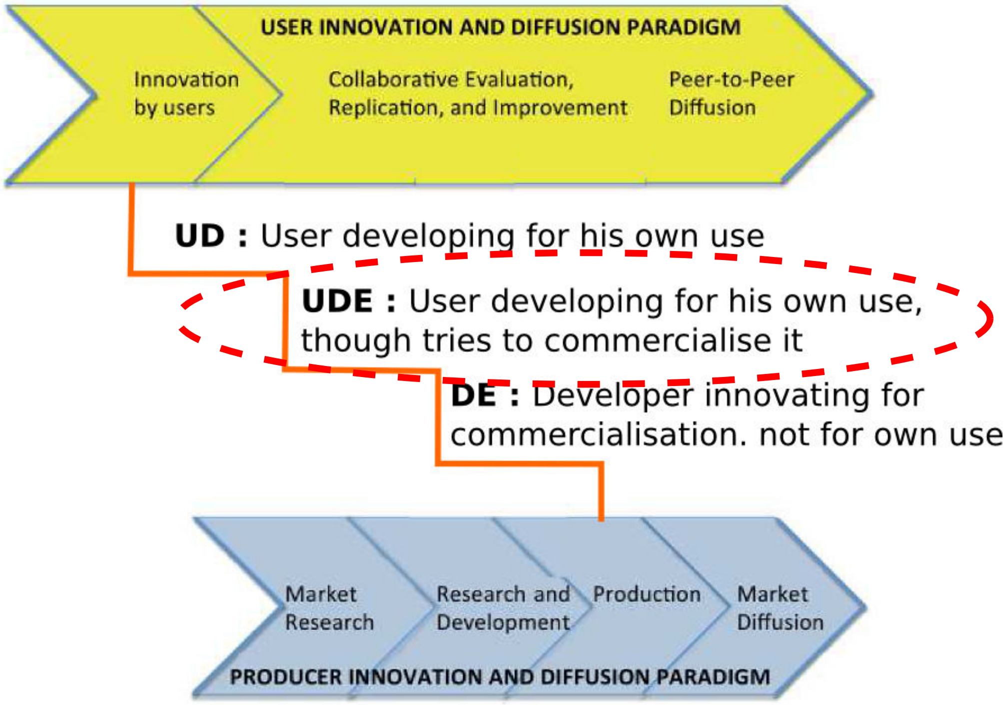


Figure 7.1: Three developer configurations: UD, UDE, DE (copy from Figure 6.6, on page 127.

model. Table 7.1 summarises the use of the theoretical concept by these studies, that will be reviewed in the next paragraphs.

7.2.1 User-Developer-Entrepreneur: comparison to the user-entrepreneur approach

As already reviewed in paragraph 3.3.3, on page 63 and in chapter 6, literature on user-entrepreneurship shares the category of user innovation approach regarding early concept formulation and early diffusion, as user "sticky" information and lead user communities are identified as start up resources for the actors (Shah, 2003; Lüthje et al., 2005; Shah and Tripsas, 2007; Raasch and von Hippel, 2012). Moreover, this literature describes a "transformation" of user to manufacturer during the diffusion phase, which can be summarised in two propositions:

- User-entrepreneurship is an "accidental" phenomenon, whereas idea development, experimentation, adaptation and preliminary adoption occur before the formal evaluation of the idea as the basis of a commercial venture.

- Once the venture succeeds, users become manufacturers, their relationship to the user role is limited to managing the feedback of their own users.

This independence from product manufacturers is also claimed in the case of innovation networks "by and for users" (von Hippel and Katz, 2002), though in the field studied indications suggest the contrary. More specifically, previous chapters advanced two propositions, in the light of Web services development phenomenon:

Setting	**Design Phase**	**Diffusion Phase**	**Indicative Literature**
User - entrepreneurship	"Accidental" entrepreneur; User "sticky" information as a resource	User-manufacturer transformation; Initial peer diffusion, then marketing and feedback management	Shah (2003); Lüthje et al. (2005); Shah and Tripsas (2007)
Open source model	User-developer; User "sticky" information + open source as resources	Peer community; Free, Web-based diffusion, using "repositories"	von Hippel and von Krogh (2003, 2006); von Krogh et al. (2003a,b)

Table 7.1: User innovation approach for entrepreneurship and open source development

- In the case of Web services, a purposive effort and personal investment is observed, of which entrepreneurship is the fruit. Hence, entrepreneurship is often the fruit of this effort, rather than "accident".

- In addition, the specific informational nature of the online services suggests that even when users become manufacturers, they are still bounded to the initial service, as they need provider's information flows for their own service to be operational.

Thus, rather than a complete, accidental transformation of users to manufacturers through entrepreneurship, Chapter 6 suggested a distinction of three configurations linking the user to the manufacturer paradigm, UD, UDE and DE. For that, a case of a user developing a tool and the commercialising of it will be studied.

7.2.2 Comparison to the open source model

Section 6.2 has already reviewed the way open source has become an exemplary case for user innovation studies, as it illustrates a *"full function"* for this approach (von Hippel and von Krogh, 2003, p. 219).

Hence, user-developers utilise their "sticky" knowledge as well as existing open source code as initial assets for innovation (von Krogh et al., 2003a,b). Then, diffusion takes place by the help of online "repositories", around which a peer community is built (von Krogh et al., 2003a,b; Lakhani and Wolf, 2003). An exemplary case of an open source software entirely developed by users, is the *Apache* Web server (Von Hippel, 2001), very competitive in its market. As Bonaccorsi and Rossi (2003) mention, the wide diffusion of open source software owes a lot to its particular licence, the *General Public Licence, GPL*, which obliges users to further diffuse their work freely, to the extent that they have used free software modules for its development.

Still, the previous chapter proposed that similar assets, such as Web-based diffusion and open source software, are jointly used with Web services specific ones (APIs) to market rather than freely reveal developers' creations.

7.2.3 Setting of the study

Referring to the model of users developed on page 110, I will study the case of an application built by a *seller-developer* using the *eBay* service to innovate and then attempting to market his creation. The oscillation between user (seller) and entrepreneur I will explore is illustrated in Figure 7.2.

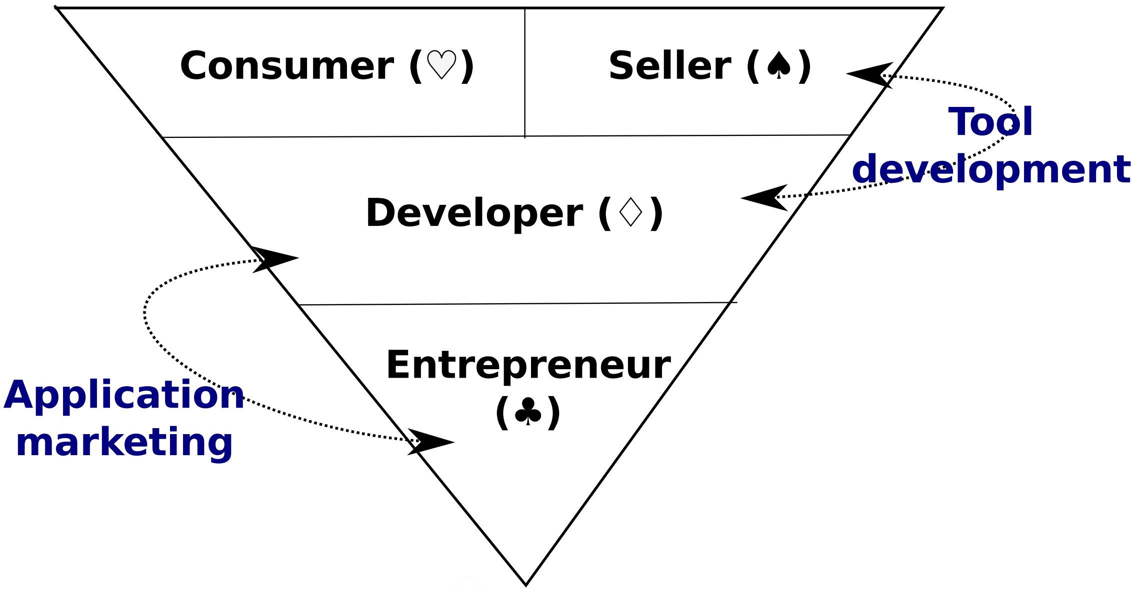


Figure 7.2: Seller - Developer - Entrepreneur oscillation: an exploration hypothesis.

The working proposition in this study will be the conclusions of the previous chapter. Thus , I expect that such a development can take place through a "personal investment" mode, where innovation will draw on a "plural knowledge base", including "sticky information" on both use context and provider technology. Development is expected to be undertaken by the use of *eBay APIs*, aided by additional open and closed source software tools. Commercialisation is then to take place within a competitive environment.

7.3 Methodology: narration as a phenomenon illustrator

While their efficiency as a general method for social science is a topic of debate, narratives have been proposed as a particularly valuable method for early phenomenon distinction (von Krogh et al., 2012). Von Krogh et al., analysing the emergence of the academic community studying the open source phenomenon, highlighted the importance of narratives for the very early steps of phenomenon distinction.

Table 7.2 , synthesises the different uses of narration in management and organisation studies, in relation to the different levels of analysis. The following paragraphs review each approach. Then, the current section concludes with the need to "discover" a narration, by contacting and interviewing a UDE, provided that no book has been identified throughout my research that could serve that research need. Afterwards, design and diffusion process of this peculiar case are analysed and compared to the literature, through the use of multiple sources, including interviews, archival documents and an analysis of the eventual application design.

Type	**Edited by**	**Addressed to**	**Academic value**	**Example or method description**
Autobio-graphical	Leaders	General public, peers	Phenomenon distinction	Raymond (1999); Grove (1997)
"Narration studies"	Researcher, anonymous members (on-going debate)	Academia	Dominant narration critique	Czarniawska (1998)
Story telling	Knowledge managers	Enterprise	K.M. tool	Soulier (2005)
Rational myths	Researcher	Enterprise	Theory testing, rationalisation guiding	Hatchuel and Molet (1986)
Story discovery	Researcher	Academia	Novel action model illustration	*Current chapter*

Table 7.2: Types of narration methods

In the following paragraphs I will briefly review each of the mentioned categories.

7.3.1 Leader stories for phenomenon distinction

The first category regards narratives created by leaders and addressed to the general public or their peers, often being autobiographical books about the deployment of their ventures. Exemplary narratives of this kind are those of Raymond (1999) and Stallman (2002) on their own contribution to the success of the open source phenomenon. As von Krogh et al. (2012) analyse, these particular narratives gained an *a posteriori* appreciation within the academic management community, as they contributed to the distinction of the proper characteristics of the phenomenon, as compared to more traditional ones.

Similarly, the narrative of Andy Grove, the leader of *Intel*, on how his company managed to become a leader in the sector of microprocessor platforms (Grove, 1997, *"Only the paranoid survive : how to exploit the crisis points that challenge every company and career"*), has been recognised by later research on platform management as insightful (Gawer and Cusumano, 2002; Baldwin and Woodard, 2010). Thence, in a way, it contributed to subsequent platform management literature.

Overall, success stories written by their protagonists are privileged narratives, from which management science may draw indications for further exploration of new phenomena. In a way, the "truth" of their arguments is proven by the outlined phenomenon's economic impact, before further academic research is undertaken (as in the case of open source software diffusion or *Intel*'s dominance, correspondingly for the cases mentioned above). Still, for management science, what is more important is not the truth *per se*, as a static picture, but the models of action that generate the status of truth (Hatchuel, 2005c). Thus, editions about success stories may, in some cases, help researchers further identify and explore the peculiarities of new phenomena.

Still, while success stories are easily shared by their actors, as well as being interesting for manager scholars, because of their effects within the business field, failure stories are harder to share. In parallel, entrepreneurship as a general phenomenon is far more characterised by those failure stories than the successful ones, as entrepreneurs most frequently fail (Schumpeter, 1939). As leadership is judged by its efficiency, leaders are not likely to share their failures, unless they already have found a way to overcome them. Entrepreneurship is thus often explored through the success stories, as can be seen in literature on user entrepreneurs (Shah and Tripsas, 2007; Haefliger et al., 2010), though leaving aside failure stories.

However, the proposition of an oscillation between user and manufacturer roles cannot be explored, as successful cases do end up with a complete separation of these two roles. Unfortunately, no book of a failed entrepreneurial story has been identified in the framework of my research, which could summarise the characteristics of the oscillation between user and entrepreneur I proposed in the previous chapter.

7.3.2 "Narration studies" and "minority report"

Narratives have been proposed as a methodological entry for organisation studies, to conceptualise and highlight organisation communication phenomena (Czarniawska, 1998; Boje, 2002; Adorisio, 2009), what can be described as "narration studies". They are addressed to the academia, while their editing opens up a discussion on the relationship between the researcher and the narrators.

These studies in social sciences originate from the post-modern position. As Hatchuel (2005c) comments, this approach, of which Lyotard is one of the principal theoreticians, challenges the possibility of common meaning creation, due to the diversity of approaches in the field. Hence, a model of action that would possibly include diversity management is not included in this literature. Typically, Hatchuel mentions:

> *They emphasize that it is worthwhile to avoid domination and that it is important to protect critical minorities within an academic field (Burrell, 2002). One can also remark that in this perspective a model of collective action, the protection of diversity, is not discussed as such and is taken as a universal solution for the production of knowledge (Hatchuel, 2005c, p. 140).*

In narration studies, the narrative is both the starting point, the question and the answer to a critique: dominant narratives can be questioned or "de-constructed" by the existence and the diffusion of different ones. They fulfil the requirement described by metaphors such as "talking pig", for studies managing *"to provoke thought and new ideas, rather than to poke holes in existing theories"* (Siggelkow, 2007) or *Black Swans* (Taleb, 2007) , attracting scientific interest because they are rare, highly impactful and predictable in retrospect (von Krogh et al., 2012). Of course, the place of the researcher is delicate in this setting, and the distance between the story and its writing is a recurrent question in this literature.

However, what I am interested in developing in the current chapter is not limited to a critique of the user and the manufacturer model (Raasch and von Hippel, 2012), but a further exploration of the figure of User-Entrepreneur-Developer, on the basis of the categories proposed on the previous chapter, exploring thus a potential intermediate action model.

7.3.3 Story telling for Knowledge Management

A third category of narratives are the ones that are created and diffused within the framework of a specific organisation. While methodologically close to the previous one, this approach emphasises the utility that narratives may have in knowledge management. Hence, stories are seen as a Knowledge Management tool, to be used within the framework of a specific organisation.

A typical case is the study of Patriotta (2003) on stories in *Fiat* industry shop floor, while a conceptualisation of stories as Knowledge Management tools is operated by Soulier (2005). There, the challenge of generation, codification and diffusion of stories within the enterprise through an information system is proposed as a way to manage knowledge. To the extent that these narratives take part of the organisation's tacit knowledge (Nonaka et al., 1996; Nonaka and von Krogh, 2009), communities of practice can be a way to cultivate them (Wenger et al., 2002).

7.3.4 "Rational myths"

A different category of stories are the ones that are diffused and created during research intervention. The researcher has an active role in story formulation, while the story is addressed to the enterprise. It has both a theoretical and practical value, as it enables theory testing and field rationalisation.

A "rational myth" is an action model allowing the mobilisation of the organisation on the basis of an objective in which the actors will believe (myth), though of which the formulation and objectives will be realistic and adaptable (rational) (Hatchuel and Molet, 1986). It follows a demand made by of enterprise actors facing a problem, seemingly due to dysfunctions or a need for improvement. In the next step, *intervention and interaction*, the researcher proposes here one of many management tools replying to the problem. The diverse reactions during its implementation will allow an in-depth knowledge acquisition through the interaction of the researcher and the actors on the basis of the myth. Thanks to what is learnt from the previous experimentation phase, researchers, using their specific status, are in position to model the attitudes of the implied actors either accepting or rejecting the rational myth proposed. The previous questioning of the tools proposed and the associated collective learning will induce a cross transformation process of tools and organisation (Hatchuel and Molet, 1986).

A "rational myth" is described, in its complete expression, by a *technical substrate*, knowledge on the field of values and on "how to do better" (*progress knowledge*) and by set *action figures* (Hatchuel, 1998). An action figure can be defined by the specification of his intervention attributes and the relationships he maintains with other actors. Hence, using this terminology, my field could be described by the technical substrate of *APIs*, the knowledge on how to use them to develop new services, what we described in Chapter 6 as norms of action, and the figure of UDE. Accordingly, the difference of the rational myth to previously mentioned narratives resides in the fact that it constitutes a means to explore an action model, from which its rigorousness and relevance depends. In addition, this myth creation and diffusion is a tool for further exploration, not the objective of an intervention.

Thus, in this very particular case, a narration becomes a working hypothesis, enabling further exploration of a specific problem.

7.3.5 Research objective: a story of a UDE

For the needs of further exploration of potential distinctive characteristics of UDEs activity, a new approach will be used, which can generally be described as "story discovery", having its objective both its identification and formulation. Such a story is to be used in the academic debate on user and manufacturers innovation models, while its value resides in its illustrative capacity regarding the distinctive characteristics of Web UDEs, explored in chapter 6. For the design of this method, elements of the previous approaches will be borrowed.

From the literature highlighting the importance of leader narrations in phenomenon distinction (paragraph 7.3.1), the objective of a distinctive narrative's identification will be adopted. The narration sought is one that describes the *process of line crossing* between the user and the manufacturer roles. However, in this particular field, no book describing this process has been identified. Such an absence may be due to the fact that entrepreneurs would be keen on writing a book on their experience only if it had a very successful end. Still, even if such a book was identified, a distinctive line crossing process description could not be guaranteed.

Hence, the possibility of "non-dominant" narrations (as reviewed in paragraph 7.3.2) will be explored. Still, the stories of the milieu, already analysed in Chapter 5 were rather partial, focussing on technical issues and successful projects of manufacturers. In fact, one interviewee did talk about an early phase of development, even before funding a start up, where *APIs* are particularly useful (Section 5.2.2, page 88). Such stories may thus be insightful. A difficulty imposed though, comes from the non-existence of an organisation for these actors, where these stories would circulate for me to identify.

Therefore, the discovery of such a story implies the direct engagement of the researcher in "provoking" the story telling: since no such story "circulates" in the field, it will have to be elicited from an actor who has experienced it. For this, I will exploit my researcher status as a facilitator, similar to research intervention (Hatchuel and Molet, 1986), though not being limited in the boundaries of a specific enterprise. Moreover, I will focus on the description of the "action figure" (Hatchuel, 1998) of the UDE, to further distinguish the specificities of his intervention attributes.

Similarly to the reasoning developed in Chapter 6, in what regards the level of analysis, this study will focus on UDE action during the most distinctive phases, as described in the discussion of the user and the manufacturer model (Raasch and von Hippel, 2012), that is, the design and the diffusion ones.

7.3.6 Use context

The general *eBay* service context of use has already been explored by the literature, as we've seen in Section 3.2 (Caillaud and Jullien, 2003; Rochet and Tirole, 2003; Parker and Alstyne, 2005; Eisenmann et al., 2006; Baldwin and Woodard, 2010). In this service, there are two types of end users: sellers and buyers. Typically, a seller puts an item in auction and then potential buyers compete on buying it by bidding. Bidding lasts a specific period of time, after which the item is sold to the higher bidder, in so far as buyers have indeed expressed their interest, otherwise it remains unsold.

The enterprise has a control over the transaction thanks to a rigid user registration process. To access the service, each *eBay* user has to obtain an account, for which several steps are required, including a telephone call confirmation. Users provide their personal information by completing a form, where they also indicated their preferred payment method (either by credit card, or by *PayPal*[1]).

The provider is remunerated on the basis of transactions operated within its service. The elementary service that *eBay* provides to sellers of goods is called in the service jargon a "listing". A listing is the action of putting an item for sale among the end-users of the platform. Service remuneration is previewed both for the action of "listing" and the transaction itself, if it eventually takes place[2].

While most sellers use the *eBay* Web site to list their goods, advanced sellers use additional services or tools to manage their operations, once their complexity becomes more important. Hence, a market of "tools for sellers" has been created to address those needs.

7.3.7 Case identification and exploration

The application I will study is called *Auction Street*. It was designed and developed to be used by *eBay* sellers and can be used as a tool to handle information on transactions in between the transactions themselves.

The identification of the case to study came through an examination of the seller tools listed in the service directory[3]. An indication that privileged the study of this particular application is that in its Web page there[4] figured a video illustrating its use, where the narrator was the developer who created it, often speaking in the first person. This fact suggested that no organised enterprise was behind the development of this application, thus the case could further enlighten us as to the action of UDEs as suggested in previous chapters.

The application was developed in 2005, by Jay Brown, a software development consultant. The contact with the developer was rather spontaneous. One of the use tutorial videos was not properly displayed. Hence, I visited the developing company, as featured at the bottom of the product's Web page[5]. There, a note described quite clearly what the hypothesis suggested:

> *Jay Brown is the principal technical member of Heartland-IT. He wears many hats ranging from webmaster to architect and project manager to developer. When Jay is not consulting with business customers he is actively involved in product development and day-to-day operations.*

---

[1] *PayPal* was one of the first companies active on the sector of online payments security. When *PayPal* started to operate there was a great insecurity whether or not providing credit cards information was a good practice online. *PayPal* handled this intermediation, between bank accounts and online transaction. Today, despite the fact that credit cards are widely used in online commerce (with banks reimbursing *a posteriori* their users in cases of fraud), an important amount of transactions are paid using *PayPal*. The service was acquired by Ebay in 2002.

[2] Listings can take place either by *auction*, which is the by default method, or by *immediate order* ("buy now" option). In the first case, a starting price and an auction period is defined by the seller. Each insertion of an auction is charged from $0.10 to $2.00, according to the level of the starting price. In the case that the item is sold, ebay gets a 9% fee over the transaction. For the "buy it now" option, ebay charges an extra $0.50 for each *listing*. Yet, each listing can contain more than one item. Ebay also charges for additional features. For instance, each listing can have one picture of the item listed for free, while each additional picture is charged $0.15.

[3] Ebay seller tools applications: http://applications.ebay.com/selling?EAppsByCategory. URL visited on August 18th, 2011.

[4] Web page of the application: http://www.auction-street.com/. URL retrieved on 18 August 2011.

[5] URL of the developing company: www.heartland-it.com/ . Visited on August 20th, 2011.

View Jay's LinkedIn profile.

Email him at jay.brown@heartland-it.com.

Hence, I contacted him reporting the problem on the video and asking for an online interview. Following the open ended interview I conducted, I studied the design of the application itself, as well as the additional archives on the application development he provided to me.

As it turned out, the application was developed within the framework of Brown's course in the University of Whichita, Kansas, USA, where he had studied computer science during the 1980s. The documents included a full archive of courses on an "Introduction to Software Engineering", given in 2005, having as a laboratory project the development of an application. This laboratory course used a project management structure to organise the development of the application. The estimated cost of the project was minimal, as it was to use open source tools, while Brown donated his "time and equipment"[6].

While starting to investigate this case and being conscious of the fact that during exploration it is often difficult for the researcher to identify and record the proper information (Yin, 2003), as well as that Web sites, unlike documents, can be modified by their editor any time, I downloaded the Web sites under study onto my computer's hard disk with the help of an appropriate software, the *WebHTTrack Website Copier*.

7.4 Outcome: design and diffusion of a "spare-time" product

7.4.1 Overview

The overall creation and marketing process of the *Auction Street* application is schematically illustrated in Figure 7.3. Brown used multiple resources for the development of the application as a "spare-time" activity. These resources included user "sticky" information, himself being an *eBay* seller, open source software tools and "technological" or "manufacturer" sticky information, as he used a combination of "closed source" and Web API tools for the design and the development of his application. Moreover, he used Web services to market it, namely the *eBay* Web site, where the application was indexed for customers to buy as well as *YouTube* videos, displayed in the application's Website, illustrating its features.

Following paragraphs further analyse the design and the diffusion phases.

7.4.2 Design phase

Brown was teaching software engineering in the local university of Whichita, Kansas, USA, in 2005, when he developed this tool as a demonstration of software design for the needs of his class[7]. He himself was an "*eBay* enthusiast" and from this perspective this application could be described as a user innovation (von Hippel, 1986, 2005). The idea for the application came about by personal problem-solving reasoning, as Brown recalls:

> *I had trouble keeping track of the inventory that I had up [to eBay service] for sale*[8]*.*

[6]Source: Auction Street Vision, Jay Brown, November 2005, class document. Provided by its author.

[7]Interview taken on the 22nd of August 2011 by telephone at 17:15 CEST. Interview duration: 46 min 35 sec.

[8]Ibid.

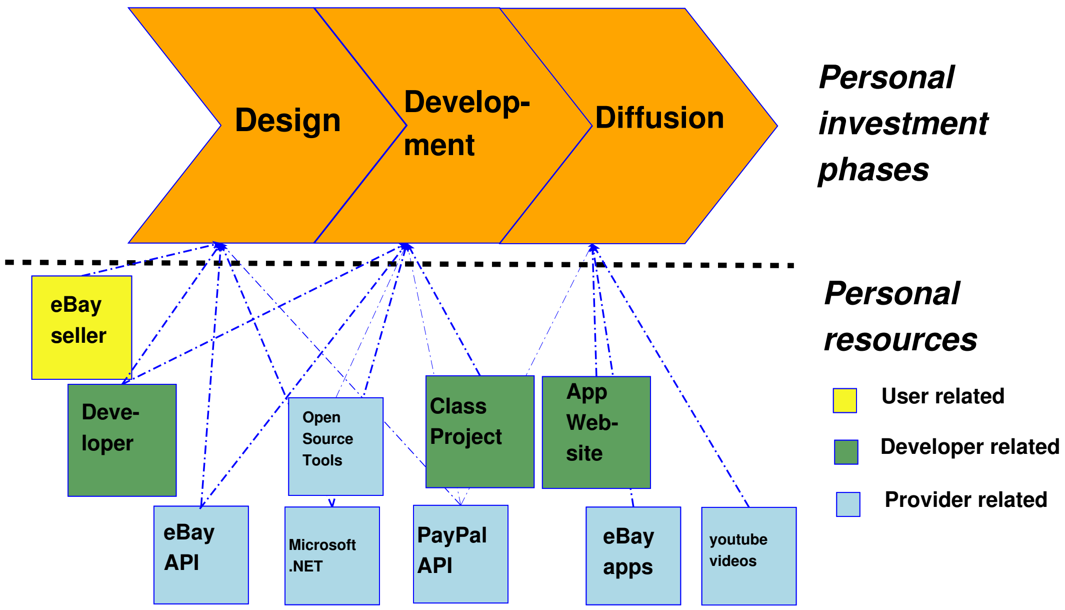


Figure 7.3: Resources for personal investment process in the case of *Auction Street* application.

The value of this tool was determined by use context "sticky" knowledge. Its aim was to *"manage and keep track of the inventory of the things you are going to sell, so you can calculate profit and loss"*[9]. As it happened, *"a lot of people are using software to manage consignment sales"*[10]. "Consignment sales" in the service jargon signify the sales operated by many transactions. Hence, a seller may buy an item at a low price and then resell it on a higher one. Hence, as n item can be bought and sold in *eBay* multiple times by the same seller, the tool was designed to calculate profits and losses between the transactions, including *eBay* and *PayPal* fees for the whole set of transactions.

In Brown's course notes[11], where he described the design and development process, he makes the remarks the following organisational constraints and market assumptions:

1. *Constraints*
   (a) *Spare-time activity - no project team.*
   (b) *Must finish by mid-December*[12]*.*
2. *Assumptions.*
   (a) *Potential Customers will prefer a Microsoft Windows platform.*
   (b) *No automated interface to Ebay will be necessary.*
      i. *Listings will be initiated with Ebay's web-based interface, or TurboLister.*
      ii. *Ebay and PayPal data will be manually exported to csv data files.*

Hence, when the courses started, the concept had already been fixed. The project was meant to be a personal (*"no project team"*), "spare-time" activity, lasting few months.

[9]Ibid.
[10]Ibid.
[11] *"AuctionStreet™- Vision"*, Jay Brown, course notes, Wichita, Kansas, USA, 2005.
[12]The project had started on the 18th of August 2005.

Assumptions about the users prefigured the design space (Baldwin et al., 2006) that would be used. On the one hand, users were assumed to be using Microsoft Windows as well. In addition, users were considered competent in managing different interfaces to manually transfer data from the *eBay* service to the tool (*Web browser* and *comma separated values (csv)* files).

These assumptions were in-line with the developer's resources: his equipment and time, as mentioned earlier. On the one hand, he was actively participating to *".NET community of users meetups"*[13], that is meetings of developers on how to build applications for Desktop computers having *Microsoft Windows*, by the use of the dedicated programming framework, the *Microsoft .NET*. His participation was motivated by the will to *" be informed on changes to the platform and the language", "do professional networking"* and *''keeping his skills"*[14]. On the other hand, further automating the exchange of information between *eBay*, *PayPal* and *Auction Street* may have been a longer process. Hence, a Windows user being able to manually export and import a file from his browser to his application fitted the requirements of a "spare-time" project.

Further product development was undertaken on the basis of a systematic design approach, as illustrated by the project schedule (Figure 7.4). The project uses a schedule clearly influenced by the Systematic Design (Pahl et al., 2007) approach, where design and development is spread in time across different phases. By the time that the semester started, the functional requirements had been set[15].

In the next paragraph, we will look into the final result, as it was commercialised by Brown, and the elements used for its construction.

7.4.3 Eventual application design

The final application design embodied different types of concepts and knowledge. The additional attributes, as compared to the standard *eBay* Website, where end-users access the service, are shown in the Table 7.3 (analytical design analysis of the application is available in Appendix A on page 393).

The application provided the possibility for sellers to add photos of the item to their announcement (a service also available from *eBay* for an additional fee), to manage their contacts, to calculate the profits and the losses between multiple transactions on the same item (*consignments*), print labels for the items to ship and review the history of the transactions.

To achieve these features, a set of different tolls where used, which were embodied as well in the final design. The most important was the *eBay API*. As shown through the design analysis, the new attributes and functionalities depend on the attributes of an *eBay item* - such as the *item ID*, its name (*title*) and description. Those attributes are common in all applications operations, thus impose common *design rules* (Baldwin and Clark, 2000) to all applications.

In parallel, *PayPal API* was also used, to enable calculations on past transactions. Since the application was "native" for Desktops using *Microsoft Windows* - i.e. it had to be installed in the user's computer in order to be used, as opposed to services being accessed through a Web site - the *.NET* platform was also used for this reason, to enable the processing of the service's information locally, on the user's computer.

[13]Interview, op.cit.

[14]Ibid.

[15]A document on functional requirements has been distributed in class in the first sessions.

Attributes of an eBay item	Application use value for sellers: New item attributes	New services
Item ID	Extra Photos	Photos adding
Title	Shipping information	Contact management
Subtitle	Acquisition information	Profit/Loss calculation
Category	Transaction history	Label Printing
Description		Transactions review
Photo		
Quantity		
Price		
Auction duration		
Sate date & price		

Table 7.3: Additional attributes and services provided by the *Auction Street* application to *eBay* sellers.

Finally, open source tools were embodied and used during the development process. As described in a document distributed in the class[16], the tools *NHibernate*, *sharpDevelop IDE* and *NUnit*, were used for the development and the testing of the application.

7.4.4 Diffusion phase

The whole process of design and development can be generally described by the framework of user innovation (von Hippel and von Krogh, 2006; von Hippel, 2007), as both the actual product and the development process had a value of use for the developer (to serve his own needs as a seller and as a teacher, correspondingly). Still, the developer did not freely reveal the application source code. Instead, he preferred to commercialise it, to *"go ahead and market the product"*[17]. It was his first time doing so and the only project Brown "had with *eBay*". In the Web site where one can download the software - for $29.95 - there is a "News" page with announcements on the software updates. There are three of them, dated 2007, 2008 and 2009[18]. Asked for the value of the product and the cost of its maintaining, Brown commented:

> *This was not really a product that I made much money on. And I haven't enhanced it much the last seven years.*

According to him, one of the reasons for its limited business value was due to the existence of a lot of competition from other tools:

[16] *Auction Street Vision*, class document, Jay Brown, 2005.

[17] Interview, op.cit. All quotes in this paragraph refer to the same interview, unless stated otherwise.

[18] Announcements on the Auction Street Product. News page URL: http://www.auction-street.com/Articles.aspx Web page retrieved on 26th August 2011.

> *When this was developed, the eBay tools were not free and they weren't really geared towards inventory management, they were geared towards just sales. They've since offered some free tools and then there is this Auctiva which is a Web-based product which does everything mine does and more. It is out-there, it's rather inexpensive, so I've seen a competition now that we didn't have in 2005.*

While building an application *"that worked"* was a result of the process described above, moving into a larger market presented additional challenges. A principal issue was client support:

> *Releasing software, it's gotta be a little bit higher quality that something you're just writing for yourself. Otherwise, it could be difficult to support in the long term. One of the reasons so many software packages fail is that the users install it, and if something goes wrong or it's not really easy to get started the people don't want to invest much time in making it work, so they'll probably search for something else.*

Hence, the relationship with clients consisting in providing a product that could be usable to customers without them having to *"invest much time to making it work"*, was considered by the developer as a difference between *"writing for yourself"* and *"releasing software"*. Moreover, Brown seemed to have regretted the design choice to develop a Desktop application and not an entirely Web-based service, when talking about commercialisation perspectives:

> *The problem is I do not want to invest too much time in it because I think when these products are more popular, at this point . . . I mean when I designed it, it was modelled after Microsoft Outlook, with pop-up forms and that kind of style . . . It's not Web-based so I don't think there is really much market for [it] right now.*

In fact, when contacted, Brown was focussing on consulting for Web-based software development, experiencing what (Cusumano, 2008) characterised as a move from products to services, in software business. According the developer "*that's definitely what companies want right now".* One of its contracts is *"taking Desktop software and making a Web-site out of it"*. This way his clients could *"get more customers and have more opportunities for sales"*. Yet, that is a *"kind of a challenge"*, as Desktop software *"performs very well, because it is installed in your computer"*. They will have to re-design the software using '*'HTML5 and a lot of Javascript"*. In this they will have to *"send and receive information to the server, without bothering the user".*

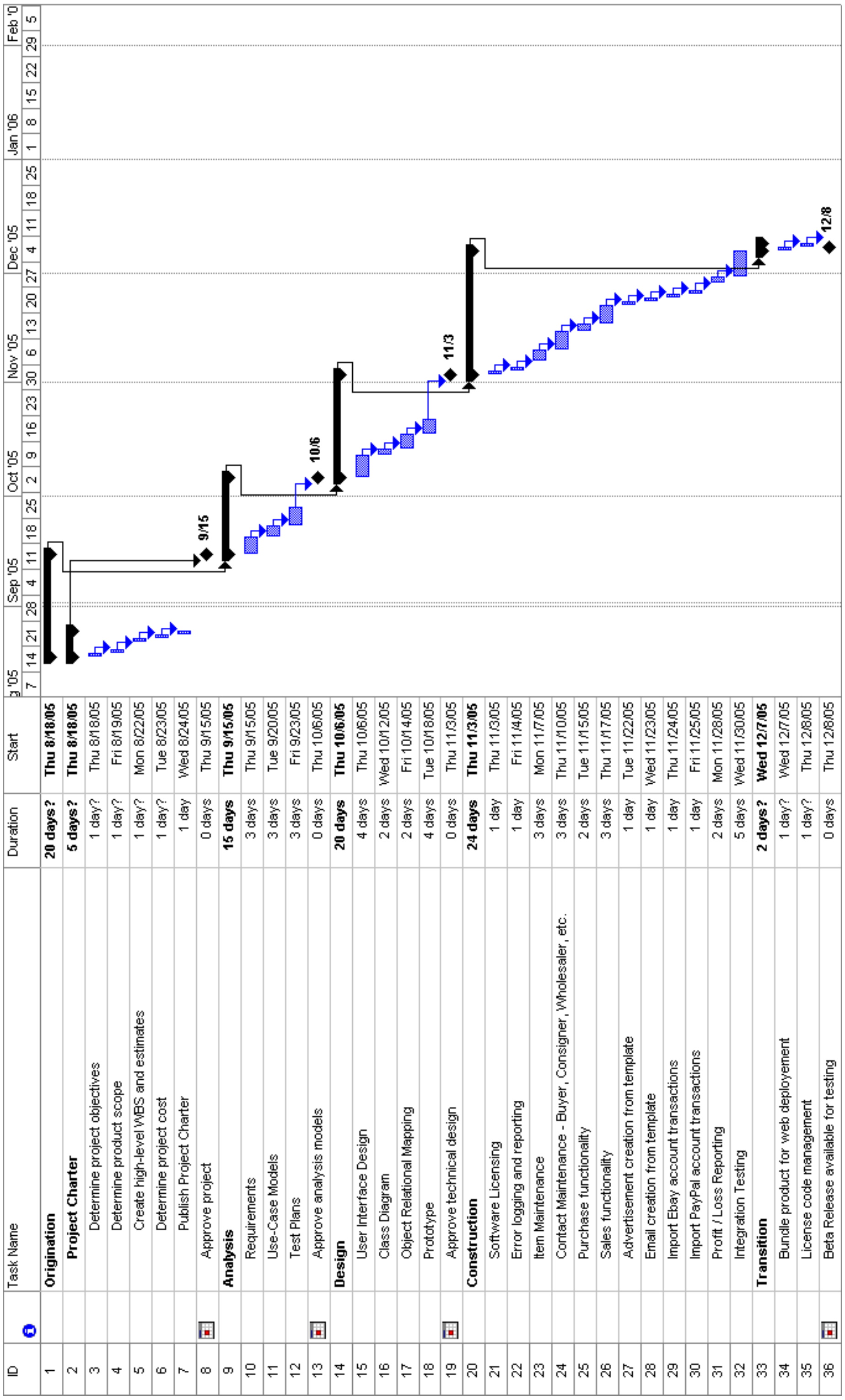


Figure 7.4: Auction Street class project plan. Organisation of the project uses a Systematic Design structure. Source: *Auction Street Vision, Jay Brown, 2005.*

7.5 Discussion: Design and knowledge issues in third-party application development

Table 7.4 summarises the findings of the study and compares them to the attributes of open source and user-entrepreneurship models, as described by the literature. Paragraph 7.5.1 will discuss the case studied in relationship to the open source phenomenon and paragraph 7.5.2 in relationship to the user-entrepreneurship literature.

On the whole, synthesising the findings of the current chapter with those of Chapter 6, I argue that UDE activity benefits from both user and producer paradigms. As I illustrate in Figure 7.5, UDEs benefit from both use-related and technology-related "sticky information" (von Hippel, 2005) during the design process, they utilise both open and closed source software, to be able to use Web APIs for online application development and, finally, they have a relationship with both end-users and service providers once the product is on the market, as in the first case a feedback management is required, while in the second one they can use online services as diffusion channels.

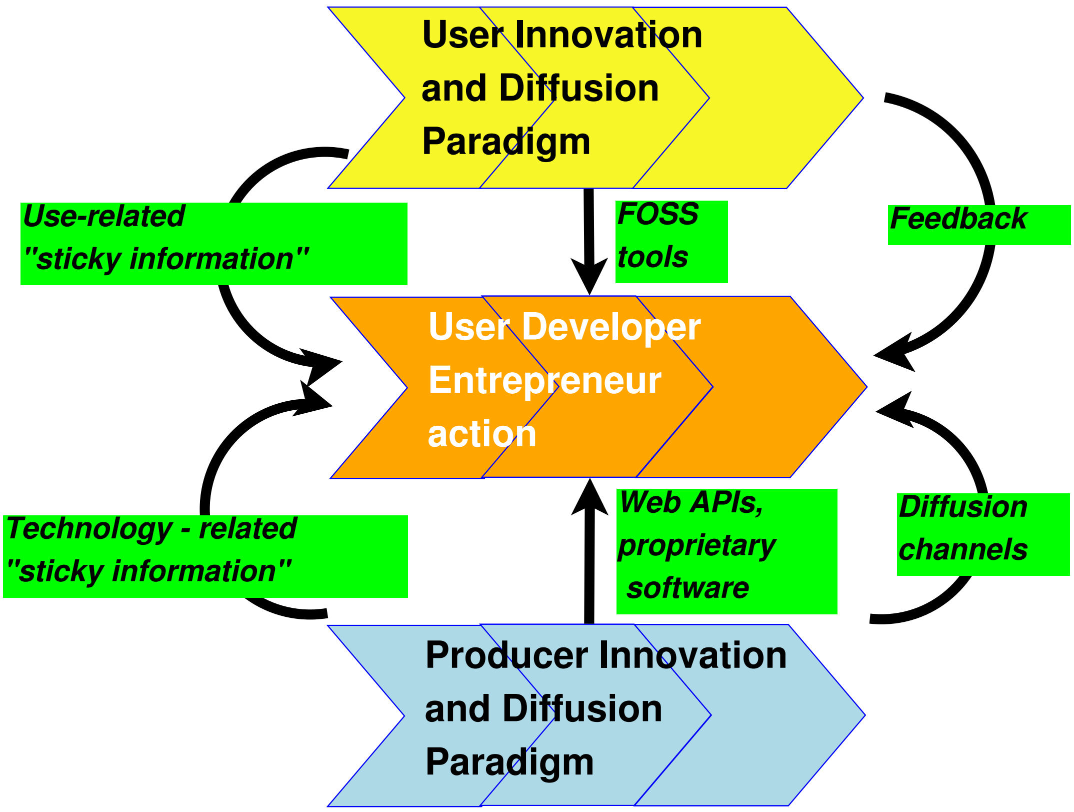


Figure 7.5: UDEs benefiting from both models. A synthesis.

7.5.1 Comparison with the open source software model

As shown in the Table 7.4, there are two major differences between the case studied and the ones described by the literature on open source software. The first regards the developer resources, while the second regards diffusion modes.

Regarding developer resources, open source software development is based on pre-existing modules, which are re-used freely, under the condition that the development result is also

	Design Phase						Diffusion Phase		
	"Sticky" Information		Developer Resources						
Auction Street case	**User**	**Manufac-turer**	**F.O.S.S.**	**Web APIs**	**Closed Software**	**No revealing**	**Compe-tition**	**Marketing**	**Client Support**
Open Source model	✓	☐	✓	✗	✗	✗	✗	✗	✗
User-Entrepreneur model	✓	☐	☐	☐	☐	*	✓	✓	✓

Table 7.4: Open source, user- entrepreneur, and UDE innovation modes
The features of the development mode identified in the current chapter are compared with the characteristics of open source and user entrepreneur innovation, as they have been described by the literature.
✓ : Feature met in the corresponding action mode descriptions.
✗ : Feature not met in the corresponding action mode descriptions.
☐ : Feature not mentioned in the corresponding action mode descriptions.
* : According to the literature, user entrepreneurs discuss their innovation with a lead user community before commercialising it (Shah and Tripsas, 2007).

freely disposed to the public (Stallman, 2002; von Krogh et al., 2003b; von Hippel and von Krogh, 2003, 2006). As Bonaccorsi and Rossi (2003) mention, open source is usually published under the *GPL* licence, which obliges the developer to freely reveal all source code being produced on the basis of open source modules, and thus is "viral".

However, what was found in the case of *Auction Street* was that open source software is *combined* with the use of both proprietary and Web-based tools. Hence, unlike cases like the *"Apache"* web server software, where all knowledge used was user related, since it was developed by users (Von Hippel, 2001), there was a utilisation of both "user" and "manufacturer" sticky knowledge in the development of this application: beyond understanding the needs of an *eBay* seller, the developer needed to understand the technologies of multiple providers, including *Microsoft, eBay* and *PayPal*.

Moreover, the resulting application was not "freely revealed" to the community of *eBay* users, it was *sold* as an application. Thus, it entered *competition* with other services proposing similar features, also facing marketing challenges. On the one hand, the developer initiated some advertisement, through the publishing of *YouTube* videos explaining the application's features to potential clients. On the other hand, he faced the challenge of *customer support* - and not community based maintenance, as the open source model suggests - to further develop the application to be easier to use by a greater clientèle. Typically enough, he cited a competitor (*Auctiva*) who managed to create a service providing similar features, though in a more efficient way.

7.5.2 Comparison with the user entrepreneurship literature

The case studied did not reveal an "accidental entrepreneur" process. Much effort was invested into the development of the application and its commercialisation. That said, the application did have a value of use for the developer himself, as he can be qualified as a lead user.

Still, the attributes of the eventual application reveal an impossibility for the developer to exit the user role, even when passing to the manufacturer one: his application will always need to be updated with information from the *eBay* service to be functional. The design analysis of the application showed that the developer used the design space (Baldwin and Clark, 2006) provided by the Web APIs, to provide an extended service for *eBay* users. This dependence on the initial service platform resembles the particular case of user-entrepreneurship studied by Haefliger et al. (2010), where the attributes of the initial service become starting points for user creative development.

Similar to the literature on user entrepreneurship, *competition, marketing* and *client support* have been challenges faced by the UDE. What is peculiar is the little amount of investment (characterised by a "spare-time activity") that was needed for this venture. Hence, I observed a relatively low "entry cost" (Utterback and Suárez, 1993) for developers, since having already at their disposal a set of tools to start designing and developing their own good, "investment" is largely limited to dedication of personal time.

Moreover, the development was based on a multitude of knowledge bases (use context related, provider related, tool related), which heavily influenced the design itself. Characteristically, the fact that Brown participated in the *.NET* community, is correlated to the fact that he used it as a tool for his project. Hence, the investment of learning was minimal, as he could re-use previous knowledge for this new venture. However, the use of this tool meant that his clients would access it by a specific interface (a Desktop application), thus it would not be an "entirely Web -based" service. When contacted for the interview, Brown seemed to regret this "strategic" choice, as after the development of the application he turned his activities towards the second business model, surmising that "that's what business do" currently.

The most important difference between user entrepreneurship, as studied by the literature, and as faced in the particular field, seems to be the "easiness" of investing and de-investing in the development of applications, without being limited to a sole product, but always being dependent on a given service provider.

7.6 Conclusion

User-Developer-Entrepreneurs benefit from both user and producer paradigms, by developing their own applications. While in the specific field of Web-based application development creating a first version of an application does not require much resources and can, thus, be undertaken as a "spare-time activity", further customer support necessitates more investment from the entrepreneur side. In parallel, when the innovation makes use of Web service provided information in its design, development and use, its use is determined by a continuous information flow from the service provider to the application user, rendering the application dependent on the initial service provider.

Chapter 8

Conclusion of the Part I

Contents

8.1 Problem addressed

The current part addressed the question of "what is Web-based application development", exploring the possibility of it being a novel phenomenon (von Krogh et al., 2012). The possibility of a peculiar *modus operandi* has been investigated in relationship to the synthetic distinction between user and manufacturer innovation paradigms (Raasch and von Hippel, 2012).

This investigation has been based on the breaking down of the question into an actor, means and reasons identification problem. The resolution of such a problem had already been known in management, since enterprises are called to reply to similar challenges when faced by the need for a new rationalisation, as induced from my reading of the work of Hatchuel and Weil (1992).

Hence, to reply to the initial question, I had to explore the questions of "who, how and why" in respect to the development of Web-based applications.

8.2 Methodology used

The methodology used utilised four different angles, enabling me to identify the Web-based application development *modus operandi* by a gradual exploration of its elements.

Initially, in Chapter 5, I explored the potential of new online innovation phenomena by means of some early indications within the discourse of platform providers, composed after a series of semi-directed interviews. The indications I found regarded the means of application development, as well as a first sketch of the figure to identify, the developer.

Then, in Chapter 6 I used a double step methodology to access, distinguish and describe the figure in question, as well as his means and reasons of action. Using the work of von Hippel and von Krogh (2003, 2006) on the distinction between "private investment", "collective action" and "private-collective" models, I constructed an analytical framework for this research step. Both methodological steps used a phenomenon-based research strategy (von Krogh et al., 2012), consisting in distinguishing the phenomenon of this activity and then further exploring it by the proposition of alternative research concepts.

The first step focussed on the figure as a result of "what they do", by joining an ephemeral development team and observing the means they used and the motives for their action ("observatory participation"). In the second step of the same chapter I used a more focussed and systematic approach to further explore this figure through the study of "what he reads", i.e. how he acquires the knowledge required for the development process. There, I investigated the action norms (Argyris and Schon, 1978) assumed by the authors for their readership and compared them to the ones described by the "private investment" and the "private-collective" models (von Hippel and von Krogh, 2003, 2006).

Finally, in Chapter 7 I examined a particular case of such an innovation, which was exemplary not because of its impressive outcomes but because of its compliance to the concepts explored and elaborated in the previous chapters. The reason behind this *Auction Street* application's development was initially self-use, though in the process its developer decided to commercialise it. This story further explored the *modus operandi* in action, investigating the "boundary case" of the passing from the user to the entrepreneur modes.

8.3 Part outcomes

Figure 8.1 synthesises the outcomes of the current part. Three actor figures have been identified, all being placed in-between the user and the manufacturer innovation paradigms (Raasch and von Hippel, 2012):

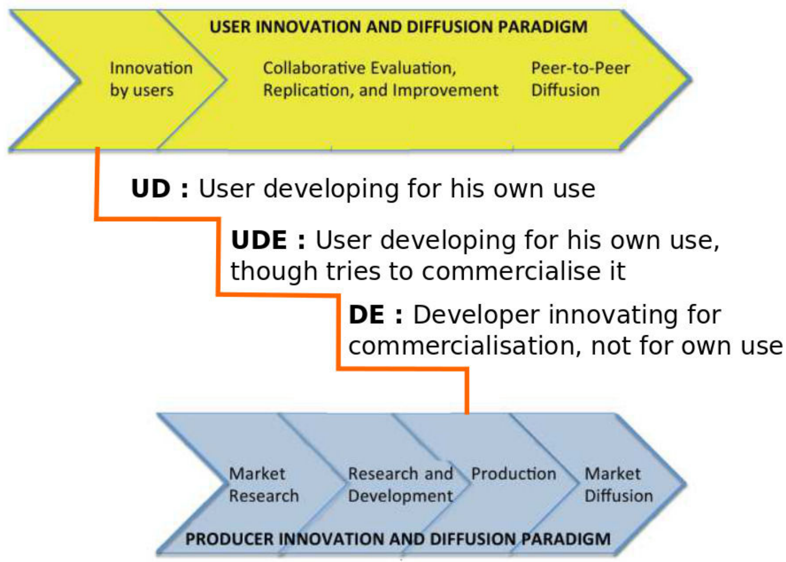


Figure 8.1: Actor figures identified as compared to the model of Raasch and von Hippel (2012).

1. The User - Developer (UD), who uses his development skills to innovate for personal use, much like an "lead user" as described in user innovation literature (von Hippel and

Katz, 2002; von Hippel, 2005; Franke et al., 2006), though possessing both *"use-related sticky information"*, attributed by von Hippel (1990) to users, and *"technological sticky information"*, attributed to manufacturers.

2. The User - Developer - Entrepreneur (UDE), who, while having a similar starting point with the UD, does not "freely reveal" his creation, as the "private-collective" model suggests (von Hippel and von Krogh, 2003, 2006), but he decides to go on, and try to commercialise it.

3. The Developer - Entrepreneur (DE), who, unlike the previous two figures, does not innovate for his own use, but he does so for others, having a commercial goal from the outset of his design process.

At the same time, all three actors use the same means to develop their own applications, combining open source software (which is extensively explored by the literature) with enterprise-provided software, namely *Application Programming Interfaces*. Both kinds of resources constitute for developers an "innovation palette", constituting for them the material conditions to create things they wouldn't have been in position to beforehand.

8.4 Further research

As already discussed, the configuration of the Web-based application development *modus operandi* was undertaken by considering one of its known elements (actor, means or reason), while distinguishing and exploring the others.

As a result, the identification of the three actor figures remains static. Chapter 7 distinguished a specific case where the *modus operandi* identified is put in action as a whole, though there are questions that remain open.

More specifically, the dynamics of the three figures action should be further explored, while the question of the conditions of appearance of this peculiar modus should also be identified. In addition, a return to the starting point should be enacted, that is identifications of some means through which enterprises can manage this phenomenon.

Part II further explores this *modus operandi* as a whole, by examining whether or not it has appeared in different industrial settings. For that, the histories of the enterprise computer, the personal computer and the radio industry will be reviewed, on the basis of my specific research problem.

Subsequently, Part III will return to the field of Web-based application development, studying the particular conditions for the appearance of "UDE settings", exploring some ways in which enterprises can harness the effect of the phenomenon.

Conclusion de la première partie

Problème de départ

Cette partie a posé la question « qu'est-ce le développement des applications Web », en explorant la possibilité d'un nouveau phénomène (von Krogh et al., 2012). Le *modus operandi* particulier a été étudié en rapport avec la distinction synthétique de Raasch et von Hippel (2012) entre les paradigmes d'innovation usager et industriel.

Cette investigation a été basée sur la considération de la question comme un problème d'identification des acteurs, des moyens et des raisons d'action particuliers de ce domaine. La réponse à ce type de problèmes a déjà été connue dans la gestion, vu que les entreprises elles-mêmes sont appelées à résoudre des questions similaires, quand elles se trouvent face à des épreuves de rationalisation (Hatchuel et Weil, 1992; 1995).

Par conséquent, afin de répondre à la question initiale, nous avons été obligés d'explorer les questions de *qui, comment* et *pourquoi* développe ce type d'applications.

Méthodologie utilisée

La méthodologie utilisée a traité le problème de recherche en adoptant quatre angles d'investigation, nous permettant d'identifier le *modus operandi* du développement des applications Web par une exploration progressive de ses éléments constituants.

Tout d'abord, le Chapitre 5 a exploré l'éventuelle existence d'un nouveau phénomène d'innovation en ligne, comme indiqué par le discours des fournisseurs de services Web - un discours qui a été restitué à partir des entretiens d'experts d'entreprises. Ces indications suggéraient l'existence des nouveaux moyens de développement, les *APIs,* fournis par les entreprises à des tiers pour qu'ils créent leurs propres applications. De plus, nos interlocuteurs nous ont indiqué qu'il y avait des individus qui « le font », n'étant « même pas de *start-ups* », puisque leur activité avait un caractère exploratoire, opérée en amont et ouvrant la possibilité d'innovation. Les indications obtenues par ce chapitre on été utilisées par la suite pour une exploration plus approfondie de la manière dont se créent les applications Web en comparaison avec la synthèse de Raasch et von Hippel.

Ensuite, le Chapitre 6 a fait usage d'une méthodologie à deux étapes, afin d'accéder, distinguer et explorer l'action de la figure en question, ses moyens et ses raisons. En utilisant le travail de von Hippel et von Krogh (2003; 2006) sur la distinction entre les modèles différents d'innovation, nous avons construit un cadre d'analyse pour cette exploration. La première étape a étudié la figure d'acteur comme un résultat de « ce que l'on fait », en faisant partie d'une action éphémère de développement d'un site Web (« participation observante »), nous permettant d'observer les moyens et les raisons d'action en question. Lors de la deuxième étape, nous avons utilisé une approche plus systématique, en étudiant la figure d'acteur comme un résultat de « ce que l'on étudie afin d'agir », c'est-à-dire le savoir exigé pour le développement des applications. En explorant les normes d'action (Argyris et Schon, 1978) assumées par les auteurs de ces livres, nous avons été en mesure d'en faire une comparaison avec les modèles décrites par von Hippel et von Krogh.

Enfin, le Chapitre 7 a examiné un cas particulier d'une telle innovation, qui fut exemplaire non pas en raison de ses résultats étonnants, mais en raison de sa conformité aux concepts explorés et élaborés dans les chapitres précédents. La raison du développement de l'application étudiée étant facilement identifiable, ce cas nous a permis d'approfondir l'exploration du *modus operandi* et de mieux saisir le cas où une innovation se conçoit d'abord afin d'être utilisée par son propre développeur, et qui se commercialise par la suite.

Résultats de la partie

La Figure 8.1 (page 150) synthétise les résultats de cette partie. Trois figures d'acteur ont été identifiées, toutes étant positionnées entre les paradigmes d'innovation d'usager et d'industriel, comme décrites par Raasch et von Hippel. Ces acteurs sont les suivants:

1. L'Usager-Développeur (UD), exploitant ses compétences de développement afin d'innover pour son propre usage, semblable au « *lead user* » décrit par la littérature sur l'innovation par l'usager (von Hippel et Katz, 2002; von Hippel, 2005; Franke et al., 2006), sauf qu'il dispose à la fois d'informations liées à l'usage (*«use context sticky information »*), attribuées par von Hippel aux *lead users*, et d'informations liées à la fabrication (*« technological sticky information »*), attribuées par von Hippel aux entreprises.

2. L'Usager-Développeur-Entrepreneur (UDE), ayant une démarche similaire à l'UD, sauf qu'il ne révèle pas librement ses innovations, comme le suggère le modèle « collective privatif » de von Hippel et von Krogh, mais poursuit sa commercialisation.

3. Le Développeur-Entrepreneur (DE), dont la démarche d'innovation est différente de celle de deux figures précédentes, comme la démarche d'une propre usage n'est pas à la base de son activité innovante. En revanche, il innove pour les autres, ayant des préoccupations de profit dès le départ.

En parallèle, tous les trois acteurs utilisent les mêmes moyens pour développer leurs applications, en combinant à la fois le logiciel *open source* (déjà exploré par la littérature en gestion) et des dispositifs fournis par les entreprises, les *APIs.* Ces ressources constituent une « palette d'innovation » pour les développeurs, dessinant les conditions matérielles de leur activité potentiellement innovante.

Perspectives de recherche

Comme il a été déjà discuté, la configuration du *modus operandi* du développement des applications Web a été entreprise en explorant à chaque fois deux de ces éléments constituants (acteurs, moyens ou raisons), tout en considérant l'un des trois connus.

Il en résulte que, l'identification des trois acteurs reste statique. Le Chapitre 7 a étudié un cas spécifique où le mode opératoire identifié est mis en action dans son ensemble, sauf qu'il y reste encore des questions ouvertes.

Plus précisément, la dynamique des trois figures d'acteurs doit être étudiée de manière plus approfondie, au même titre que les conditions d'apparition de ces acteurs devraient être identifiées. Par la suite, ces explorations devraient être utilisées pour la proposition des moyens de gestion de ce genre d'activité par les entreprises.

La Partie II approfondira l'exploration du *modus operandi* identifié dans son ensemble, en répondant à la question de ses conditions d'apparition. À ce propos, l'histoire des trois cadres industriels proches sera relue sous l'angle du problème étudié dans cette partie.

Par la suite, la Partie III reviendra au champ du développement des applications Web pour étudier les conditions spécifiques d'apparition des dispositifs des UDEs, en explorant des méthodes pour que les entreprises soient en mesure d'exploiter les effets de ce modus.

Part II

A historical perspective: the role of UDEs in industrial development

Une perspective historique : le rôle des UDEs dans le développement industriel

La première partie de cette étude a été consacrée à l'exploration et la configuration d'un acteur semblant avoir un rôle important dans l'innovation sur le Web. Trois configurations de cette figure ont été identifiées: l'usager-développeur, exploitant ses compétences pour innover à son propre usage, l'usager-développeur-entrepreneur, poursuivant ses efforts à la commercialisation de ces innovations, et le développeur-entrepreneur, innovant pour les autres, ayant une démarche de profit. Une limite de cette exploration a été sa nature « statique », vu que le souci initial de notre recherche était la caractérisation de cette figure d'acteur, et non pas la description de la dynamique mettant en rapport les configurations différentes de ces acteurs.

La présente partie de cette recherche a émergé comme fruit d'un travail parallèle à l'exploration des phénomènes d'innovation sur le Web. Comme déjà discuté dans la première partie, quant on se trouve face à des phénomènes potentiellement nouveaux, lors de leur propre émergence, il est nécessaire que le chercheur puisse prendre une distance du champ qu'il est en train d'étudier, afin de pouvoir résister à des effets de mode.

Elle est consacrée aux aboutissements d'un exercice de *« problématisation »* du rôle historique qu'a joué la figure identifiée dans la première partie au développement industriel. À ce propos, nous allons proposer une nouvelle lecture de l'évolution historique des cadres industrielles proches, à savoir les industries de l'ordinateur d'entreprise, de l'ordinateur personnel et de la radio, une lecture basée sur le travail des historiens. Curieusement, ce qui ressortira de cette lecture est une apparition assez fréquente de la figure de l'usager-développeur-entrepreneur (UDE), durant des phases de développement industriel très en amont, accompagnant l'apparition d'une nouvelle technologie illustrant un grand potentiel, dont la valeur reste néanmoins à explorer.

À partir de la première expérience étudiée, celle de l'industrie d'ordinateurs d'entreprise, nous avons induit un modèle selon lequel les UDEs semblent jouer un rôle décisif dans le développement industriel, lorsque de nouvelles théories et matérialisations d'objets les incarnant émergent, faits qui conduisent à une émergence de marché précoce. Le rôle de ces acteurs s'étend également dans la phase où ce marché précoce se déploie, tout en explorant le potentiel de la nouvelle technologie. Le rôle dominant des entreprises arrive plus tard, quand une rationalisation de la conception, de la production et de la commercialisation du nouvel objet pourra être proposée. La mise en épreuve de ce modèle dans les histoires des cadres industriels de l'ordinateur personnel et de la radio montre une pertinence inattendue.

Bien que ce travail reste incomplet, il a une contribution double dans la recherche en *business* : d'une part, en ce qui concerne la méthodologie d'exploration de l'originalité d'un phénomène potentiellement nouveau, qui fut par ailleurs la démarche initiale de cette étude historique, cet exercice fournit un premier cadre d'étude comparative. D'autre part, le modèle qui se dégage de cet exercice de manière peu prévisible, contribue à la meilleure compréhension des dynamiques industrielles entre les innovations des usagers et celles des entreprises, qui semblent être catalysées par l'action des UDEs.

La structure de la présente partie suivra le raisonnement initial de notre exercice. Tout d'abord, nous relirons l'histoire du cadre industriel de l'ordinateur d'entreprise, d'où nous induirons un modèle selon l'analyse de ses phases de développement successives. Par la suite, nous utiliserons ce modèle pour analyser l'histoire du cadre industriel de l'ordinateur personnel. Enfin, nous utiliserons la même méthode pour le cas de l'industrie de la radio.

Concernant le *modus operandi* identifié dans la première partie, nous concluons que ses acteurs (UD, UDE et DE) n'ont qu'un rôle temporel, qui sert à explorer un potentiel donné avant qu'une rationalisation industrielle soit conçue et mise en place.

Synthèse synoptique du modèle et de la présente partie

Une synthèse synoptique de la présente partie est illustrée au Tableau 9.1 (page 168) , où les éléments majeurs de chaque étape de recherche sont soulignés en rapport avec modèle aboutissant.

La première colonne comprend les caractéristiques majeures de toute phase de développement industriel identifié (*« Matérialisation précoce », « Émergence de marché », « Compétition dans le brouillard »).* Ensuite, les trois autres colonnes sont consacrées à chaque cadre industriel étudié, celui de l'ordinateur d'entreprise, celui de l'ordinateur personnel et celui de la radio.

Les deux premières phases peuvent être étudiées sur une distinction assez claire du processus de développement à la base soit d'un concept ancien, « mieux incarné » dans la nouvelle technologie, soit d'un concept nouveau, constituant un « nouveau rêve » à atteindre. Cette distinction n'est plus le cas dans la phase de la « Compétition dans le brouillard », puisqu'il y a une fusion des directions d'exploration des anciens et des nouveaux concepts.

Inspiré par Lefebvre (2013), qui propose qu'un cadre industriel peut être étudié en trois axes, la distance conceptuelle, la distance cognitive et les relations d'acteurs sous-jacentes, tous les trois cas seront étudiées à trois niveaux, le conceptuel (marqué dans le Tableau 9.1, page 168, en couleur bleu-vert), le cognitif (marqué en couleur orchidée) et le relationnel (marqué en couleur lavande).

Notre étude s'intéresse davantage à la figure de l'usager-développeur-entrepreneur, dans les configurations identifiées dans la première partie de ce travail, et à sa contribution spécifique au développement industriel. Par conséquent, la figure de l'UDE définira également le niveau de notre analyse.

La limitation majeure de cette étude réside dans l'absence d'une exploration systématique de deux phases, très en amont, identifiées mais pas explorées durant ma recherche. La première, précédant la phase de la *« Matérialisation précoce »,* est la phase de la *« Construction d'une théorie ».* Ainsi, cette partie n'explore pas l'émergence des théories de la Cybernétique où celle de l'Électromagnétisme, même si elle en tient compte dans l'investigation des cadres industriels correspondant. La seconde phase peu explorée est celle de la *« Rationalisation industrielle »,* suivant celle de la *« Compétition dans le brouillard* », et qui a été l'objet d'études renommées, comme celle de Baldwin et Clark (2000) sur la modularisation de l'industrie d'ordinateurs et la théorie de la modularité où celle de Hatchuel et Weil (1992) sur la rationalisation en général.

Les paragraphes suivants exposent les grandes lignes des phases identifiées et explorées.

La phase de « Matérialisation précoce » : des projets uniques, pour des clients uniques

Cette phase est déterminée par l'action des usagers-développeurs (UDs), qui créent des prototypes illustrant un nouveau potentiel. Durant cette phase très en amont, il est exigé que les UDs aient une compréhension profonde des nouvelles théories, sur lesquelles est basé le potentiel en question. Leurs créations correspondent à leurs propres besoins ou leurs propres rêves, et, pourraient donc être décrites comme des innovations d'usagers von Hippel (1976). Plus précisément, on peut distinguer deux types de matérialisations précoces:

- De « meilleures solutions », reposant à un concept connu précédemment et, sont donc comparées à des critères de performance existants, mais dont le développement fait usage du nouveau potentiel.

- Des « nouveaux rêves », incarnant un concept nouveau, souvent reposant sur la projection d'un mode de vie différent, et introduisant de nouveaux critères de performance lors de leur développement.

De plus, bien que le développement d'un prototype soit souvent un projet individuel, les UDs sont liés par des collectivités désinvoltes mais intimes, où les théories sous-jacentes sont discutées ainsi que la possibilité d'exploration de leur potentiel d'application. Ces collectivités sont souvent construites autour des structures académiques ou institutionnelles, sans pour autant être formalisées. Néanmoins, à ce niveau, la valeur des innovations potentielles est loin d'être reconnue par un public pouvant former un marché. Un facteur clef de cette phase est l'existence d'un client singulier, souvent une entreprise ou une institution, qui financera cette matérialisation précoce pour son propre usage.

La Section 10.2 étudiera le déroulement de cette phase dans l'histoire de l'industrie d'ordinateur entreprise, la Section 11.2 l'étudiera dans celle de l'ordinateur personnel, tandis que la Section 12.2 l'étudiera dans le cas de l'industrie radio.

La phase de l' « Émergence du marché » : le tout début de la « production de masse »

Lors de cette phase, qui suit celle de la « Matérialisation précoce », il y a une activité parallèle à la fois sur la base de l'ancien concept et sur la base du nouveau. Comme identifié, c'est d'abord l'ancien concept qui attire l'intérêt des acteurs établis du marché. Par des variations conceptuelles, le nouvel objet est commercialisé dans un cercle plus grand de clients, une mise en marché qui repose sur les savoirs produits dans la phase précédente, à la fois en ce qui regarde son développement et son usage. Les efforts d'innovation se concentrent sur des aspects (modules ou caractéristiques) des matérialisations précoces. Les UDEs développent le produit pour leur propre usage, en même temps qu'ils cherchent des opportunités de commercialisation. Ils peuvent être des individus ou des institutions, mais en tous cas la relation marchande entre fournisseur et client du nouveau produit reste difficile à clarifier.

Par la suite, le développement sur la base d'un nouveau concept prend lieu, utilisant le même savoir, bien qu'il affirme une nouvelle valeur. La conception de son extension par des tiers donne la possibilité d'une exploration conceptuelle plus approfondie, où les UDEs jouent un rôle important.

Dans tous les deux cas, des « cercles intimes » discutant le processus sont de grande importance, car ils permettent un premier partage des avancements conceptuels ou cognitifs majeurs parmi les UDEs et les entreprises qui ont rejoint l'épreuve, à prendre en compte par la suite pendant le processus de développement.

La Section 10.3 étudiera le déroulement de l'émergence du marché dans le cas de l'ordinateur d'entreprise, la Section 11.3 l'étudiera dans celle de l'ordinateur personnel, tandis que la Section 12.3 le fera pour le cas de l'industrie radio.

La phase de la « Compétition dans le brouillard »

Durant cette phase, il y a une expansion du marché précoce, caractérisé par une pléthore de produits utilisant le nouveau potentiel. Un ensemble d'acteurs divers exploitent commercialement ces produits, bien qu'ils explorent encore le potentiel de leurs propres produits. Au long de cette phase, émerge un « écosystème d'affaires », décrit par la coexistence des entreprises, des consommateurs et des développeurs-entrepreneurs (DEs). Un point critique de la compétition est la gestion du rapport entre les entreprises qui sont actives dans ce marché et les DEs, vu que celles qui ont déjà cultivé des rapports intimes avec les UDEs de la phase précédente sont dans une position privilégiée. Le savoir est caractérisé ici par son segmentation, et repose sur des modules, puisqu'une nouvelle synthèse reste à devenir possible.

La Section 10.4 étudiera le déroulement de cette phase dans le cas de l'industrie de l'ordinateur d'entreprise, la Section 11.4 l'étudiera dans celui de l'ordinateur personnel, tandis que la Section 12.4 l'étudiera dans celui de l'industrie radio.

La fracture dans le cycle

Dans un des cas étudiés, bien que la phase de la « Compétition dans le brouillard » suivi d'une phase de « Rationalisation industrielle », des DEs ont réussi de s'en sortir du cycle, en revenant en arrière, c'est-à-dire en s'adressant de nouveau aux UDs. Il s'agit du cas de *DEC*, et du dispositif qui a été appelé plus tard un *mini-computer.* *DEC* s'est adressé à des *hackers* pour qu'ils complètent le développement de leurs produits, des *PDPs.* Ces ordinateurs ont introduit une série d'innovations (notamment en ce qui concerne l'usage des semi-conducteurs dans leur développement ainsi que leur usage en réseau), qui étaient des aspects peu valorisées dans la rationalisation conçue et implémentée à l'époque par IBM. Ce cas sera étudié dans la Section 10.6.

Comparaison avec quelques approches sur le développement industriel par l'innovation

Tandis que la démarche de cette étude historique ne portait pas sur une discussion de la littérature existante sur le développement industriel, mais elle visait à prendre un recul historique de l'objet d'étude (le développement des pplications Web) et mettre à l'épreuve la figure d'acteur configurée, afin de se rendre compte de son originalité ou de sa généricité, les aboutissements de notre étude pourraient avoir un apport dans la grande discussion sur le développement industriel. La Section 9.2 discute le modèle induit en rapport avec les approches conceptuelles les plus proches.

Sur le dilemme de l'innovateur

La paragraphe 9.2.1 discute le célèbre travail de Christensen (1997), *The innovators dilemma.* Dans sa recherche, Christensen met en avant l'importance des *« disruptions »*, des cas où d'innovations technologiques, *a priori* peu performantes et adressées à des marchés non significatifs, arrivent à obtenir une part de marché importante et, au fur et à mesure où la technologie avance, ces nouveaux acteurs arrivent à déplacer des entreprises établies auparavant. Le dilemme de l'innovateur consiste à savoir s'il doit poursuivre la commercialisation d'une nouvelle technologie, tant qu'un marché qui correspondait à son potentiel n'est pas identifié.

À ce propos, la relecture des histoires des trois cadres industriels étudiés confirme, d'une part, le phénomène décrit par Christensen. Par exemple, l'ordinateur a progressivement remplacé d'autres équipements d'entreprises, *IBM* étant le cas exemplaire de la réussite de cette transition. D'autre part, il y a des cas des « sauts » qui ont une nature différente : c'est le cas où ni la technologie, ni le marché ne sont encore là, mais les acteurs impliqués partagent une estimation de possibilité d'aboutissement d'une innovation, sans pour autant être sûrs sur l'ampleur de l'effort à fournir pour atteindre les résultats projetés. Un cas très typique était celui de l'exigence d'une mémoire d'ordinateur temporelle et dynamique, lors des années 1950. La valeur d'une telle mémoire a été pointue très tôt par des entreprises aériennes, projetant la possibilité de réserver des billets « en ligne » , terme qui à l'époque était utilisé pour décrire une opération « lorsqu'un ordinateur est déjà en train de traiter des données ». Plusieurs développeurs-entrepreneurs ont exploré la technologie de mémoire à la base d'un « tambour magnétique », se retrouvant au bord de la faillite. Un peu plus tard, *IBM* inventera la *RAM,* sur la base d'une technologie différente, pouvant satisfaire les spécifications en question.

Sur la diffusion des innovations

Les paragraphes 9.2.2 et 9.2.3 discutent les approches sur la diffusion d'innovation, notamment le modèle de Rogers et le positionnement de la littérature de l'innovation par l'usager à ce modèle.

Selon cette littérature, une innovation s'opère au sein d'une communauté des « *lead users* » avant qu'elle soit reprise par les entreprises, qui la diffuseront aux « *early adopters* », et, plus tard, à une majorité de consommateurs. L'étude actuelle met en lumière la transition dès « *lead users* » aux entreprises : dans tous les trois cadres industriels étudiés, il est observé une transformation progressive des acteurs, des objets, des usages et des marchés avant que le développement du produit se rationalise. Des UDs aux DEs, l'impulsion entrepreneuriale ainsi que les responsabilités qui vont avec vont privilégier des concepts à explorer plus que d'autres. Les « *early adopters* » du départ, ne sont pas de simples consommateurs : ils participent au développement de l'objet, en ayant des liens proches avec les UDEs. Même dans une configuration d'écosystème d'affaires, il n'est pas seulement question d'une seule diffusion, sans innovations supplémentaires, agissant sur la nature de l'objet et explorant des usages et des marchés possibles, avant une rationalisation industrielle. Cette rationalisation, même si elle n'arrête pas l'innovation, impose néanmoins des règles de conception en commun, conduisant à un marché segmenté, où la diffusion passe à une « majorité d'usagers ».

Méthodologie de recherche

Christensen (1997) a étudié l'histoire de l'industrie du disque dur, à la suite d'une suggestion d'un ami. Christensen s'était intéressé à explorer les raisons pour lesquelles des grandes industries échouent, et cette histoire se promettait avantageuse pour la discussion de cette problématique, car l'industrie en question est caractérisée par son déploiement selon des cycles particulièrement courts.

Exigeant une mise en rapport de notre approche méthodologique avec des réflexions scientifiques portant sur l'étude de l'histoire, la méthode que nous utiliserons dans cette partie de notre étude peut être décrite comme une «problématisation limitée». Faisant usage des apports des études en histoire adoptant l'approche de Foucault (Castel, 1994; Lefebvre, 2005; 2009; Kendall et Wickham, 1999; Aggeri et Labatut, 2010; Chatzis, 2008), nous allons explorer les « conditions de possibilité de déploiement » Lefebvre (2005) du *modus operandi* identifié dans la partie précédente en étudiant le champ du développement d'applications Web, dans le développement industriel, comme rencontrées dans les expériences des trois cadres industriels à étudier.

Chapter 9

Introduction: an unexpected model for UDEs and its positioning in the literature on industrial development

Contents

The first part of the current document was dedicated to the exploration and the configuration of a figure appearing to have a particular role in Web services innovation. I identified three configurations of this figure: the user-developer, using his skills to innovate for his own use, the user - developer - entrepreneur, attempting to go on and commercialise this innovation and the developer - entrepreneur, trying to innovate for others, for an audience having different needs and desires to himself. A limit of this exploration lay in its "static" perspective, as my research interest was focused in the characterisation of this figure and not in the description of the dynamics, linking the different actor configurations.

The current part of my research emerged as the fruit of parallel work during the exploration of the Web services innovation phenomena. As already discussed in Part I, when facing potentially novel phenomena while they occur, there is a need to take a distance from the field, in order not to be carried away by the trend. Hence, this part is dedicated to the outcomes of an exercise of a "problematisation" of the historical role of the specific figure, the User - Developer - Entrepreneur (UDE) in industrial development. To do that, I revisited the evolution of close industrial settings, namely the computer industry, as studied by historians of technology and business.

Surprisingly, I found that this figure appears quite often in industry, during phases where new technologies appear, illustrating a great potential, of which the value remains to be explored. I induced a model from the experience of the computer industry, according to which UDEs appear to have a decisive role in industrial development when early materialisations of new theories emerge, leading to the very beginning of a new market, as well as during the phase where the early market is deployed while exploring the potential of a new technology. The dominant role of enterprises comes later, when a rationalisation of the design, production and commercialisation processes can be proposed. The same model has been tested in the history of a different industrial setting, the radio industry, where it also appears to be valid.

While this work remains incomplete, it has a double contribution to make to business research: on the methodological side, which has been the initial motivation for this study, it can provide a comparative framework for the study of new phenomena while they emerge. A second, unexpected contribution, is the emerging model itself. This study opens the field for further research into the interplay between enterprise and user innovation during the passing from one paradigm to another, as the UDE figure appears to have a catalytic role in this transition.

The structure of this document will follow the research reasoning used during my study. Initially, I will review the history of the commercial or mainframe computer industry, and induce a model through an analysis of its successive phases. Then, I use the model to review the history of the personal computer industry. Finally, I will apply the same model to the radio broadcasting industry.

Regarding the specific field of Web services, I will be then in a position to draw conclusion about the limits of UDE figure I identified in the first chapter which has a temporal role, composed of those who explore the potential of Web services, while exploiting the use values discovered by the process, until a new encompassing rationalisation is proposed.

9.1 Model and part overview

An overview of the current part is shown in Table 9.1, where the main points of each research step are highlighted in relationship to the resulting model.

The first column includes the major characteristics for each industrial development phase identified (*Early Materialisation, Market Emergence* and *Foggy Competition*). The three other columns are dedicated to each of the industries studied, the enterprise computer, the personal computer and the radio ones. The two first phases can be studied by the clear distinction of the process on the basis of either an old concept, "better embodied" in the new technology, or a new concept, constituting a "new dream" to explore. On the contrary, this is no more the case for the Foggy Competition phase, as the different exploration directions merge and new ones emerge through the process itself.

Inspired by Lefebvre (2013), who proposes that industrial settings may be studied using three axes, the conceptual distance, the cognitive distance the study and the encompassing actor relationships, in each of the three phases study is undertaken on three levels, the conceptual (marked in the Table 9.1 in blue-green), the cognitive (marked in the table in orchid) and the relational (marked in lavender).

A particular emphasis is given to the figure of user-developer-entrepreneur - as already identified in Part I for the field of Web services - and his particular contribution to industrial development. Thus, the UDE figure will also be a level of analysis.

The major limitation of the current study lies in the absence of investigation into two phases, which were identified during my investigation though not further analysed. The first, preceding the *Concept Materialisation* phase is the one of *Theory Building*. Thus, the current part does not explore the emergence of *Cybernetics* theory or the one of *Electromagnetism*, though it takes them into account in the investigation of the computer and the radio industries respectively. The second phase left unexplored is the one of the Industrial Rationalisation, which follows the *Foggy Competition* phase and has been the object of some renowned studies, such as the one of Baldwin and Clark (2000) on modularisation of the computer industry and modularity theory and Hatchuel and Weil (1992) on the rationalisation process in general.

9.1.1 "Early materialisation" phase: unique projects for unique clients

The "Early materialisation" phase is determined by the action of user - developers (UD), creating prototypes that illustrate a new potential. During this very early phase, it is required UDs have a deep understanding of new theories, on which the new potential is based. Thus, very often, these users are high level researchers. Their creations correspond to their own needs or "dreams", thus are typical user innovations (von Hippel, 1976). More precisely, there are two kinds of early materialisations:

- "Better solutions", that is materialisations based on a previously known concept and its corresponding performance criteria, yet are produced by the use of the new technologies.
- "New dreams", that is original concept materialisations, based on the imagination of a different mode of life and introducing original criteria for their production.

In addition, while the prototype development process is usually an individual undertaking, UDs are tied to loose still "intimate" collectivities, discussing the underpinning theories and exploring the potential of applications. These collectivities are usually based around academic and institutional structures. Still, at this level the value of the innovation is not yet recognised by a public that could form a potential market. Another important factor during this phase is the existence of a singular client, usually an enterprise or an institution, financing this early materialisation development for its own use.

Hence, in the case of mainframe or enterprise computers, the well-known *ENIAC* electronic calculator was developed as a faster fire table calculation machine, using the theory of cybernetics and the knowledge on electronics to illustrate the new potential of what came to be the computer. Its developers, John Eckert and John Mauchly, were high-level researchers, also participating in a series of "intimate circles" discussing the potential of cybernetics and electronics, of which the most important was the *Association of Computer Machinery*, emerging in parallel with the computer industry. While *ENIAC* was a single purpose computer, the

Phase	Industry		
Characteristics	Enterprise Computer	Personal Computer	Radio
Early Materialisation : unique projects for unique users			
→ Old Concept			
"Better embodiment"	Electronic calculator (*ENIAC*, 1945)	Microcomputer (Micral, 1973)	Radio telegraph (Marconi, 1897)
"Use Theory"	"Mechanical Mind", Electronics	Computer Science, semiconductors	Electromagnetism
Comparison to previous criteria	Faster calculation	Cheaper solution	Cheaper telegraph
"Intimate circle"	ACM	[not identified]	Family & Bologna University circles
UD	Researchers	Researchers	Researchers
→ New Concept			
"New dream"	"Universal Computer" (*EDVAC*, 1949)	"Personal Computer" (*Altair*, 1974)	"Spread music in the air" (Audion, De Forest, 1906, FTC)
Proof of concept	"Stored programs"	"Personal use"	Piano concert broadcasting
New criteria	"Computer architecture"	"User Interaction"	"Signal amplification"
"Intimate circle"	ACM	Altair User Group	Institute of Radio Engineers
UD	Researchers	"Hackers"	Researchers, user-developers
Market Emergence: very beginnings of "mass production"			
→ Old Concept			
Concept variations	Defence calculator (*IBM 701*)	Microcomputer (*Micral*)	Radio-telegraph for commercial ships (*Marconi*)
Component innovations	Use of magnetic tape	Prologue Software	Range transmission increase
Previous knowledge exploitation	Computer architecture	Operating System, Programming Language	Use radio-telegraph
"Intimate circle"	*IBM* - Von Neumann relationship	[not identified]	Marconi-Flemming relationship
UDE	"Special project"	Micral users	"Wireless Age" readers
→ New Concept			
Concept value affirmation	"Universal Computer" (*UNIVAC*)	*Macintosh*	Broadcasting, (Westinghouse, 1920)
Extension design	Software tools	Third party applications	Radio programs
Previous knowledge exploitation	Computer architecture	User Interface	Radio receivers
"Intimate circle"	ACM	Apple-Xerox relationship	Conrad's garage - Westinghouse
UDE	Technical Support	Developer Community	"Attic radio manufacturers"
Foggy Competition: parallel exploration & exploitation; enterprises, DE, consumers			
Trajectories fusion, Parallel Exploration	Multiple computers designs	IBM PC 5150, Macintosh, Commodore, Compaq . . .	Broadcasting "chaos"
Component-based knowledge, Synthesis Required	Memory, processor, data processing techniques & Computer Science emergence	USB key & Wintel model emergence	Battery-less, living room box receiver & Emergence of a theory for radio engineers
Product diversity	ex. Multiple IBM product lines	A multitude of PCs	A mulitude of device and manufacturers & Broadcasters
Provider DE Community	ex. Third party IBM computer leasing	ex. *Compaq* Dev. Communities	ex. *FTC* - UDE relationship
"Business Ecosystem"	University Spin Offs & Manufacturers & Enterprise-clients	UDEs & Manufacturers & Consumers	UDEs & Manufacturers & Audiance

Table 9.1: Synthesis: Early Materialisation, Market Emergence and Foggy Competition phases for the three industries studied.

EDVAC one, was developed by the same team having as their ambition to build a "universal computer". *EDVAC* was a first materialisation of many elements still defining a computer, such as the stored program, and proposed a "computer architecture", what became known as the "Von Neumann Architecture". Both *ENIAC* and *EDVAC* had been singular products, addressed to a specific client, the US army.

In the case of the personal computer, an early materialisation of a "better solution" was the one of the "microcomputer", designed by Thi Truong in 1973 as a project for the French institute INRA, in order to be used in the measurement of evaporative transpiration. *Micral*, as it was named, was designed as a smaller and cheaper computer to operate similar tasks with the mainframes, though less demanding in resources. On the contrary, there have been many materialisations of the concept of a "personal computer", initially meaning a computer one develops for oneself. The *Altair* computer was the product that managed to advance the exploration of what a personal computer could be, through the close ties of the manufacturing start-up, *MITS*, with its UD community. Characteristically enough, *Microsoft* was initially founded to develop *Altair*'s programming language. New criteria, such as user interaction, were explored by a large community of "hackers", where researchers and user-developers had come together.

Similarly, the case of radio industry finds its early materialisation phase initially through the use of radio frequencies to provide a "better solution" for the telegraph. Marconi, exploiting family contacts, had the first contract with British Army in 1897 to develop a "radio telegraph" which was experimentally created in house, using knowledge taught in Bologna University. Later on, the contract would be generalised and Marconi would build an empire on "wireless telegraphy". In parallel, Lee De Forest, an American researcher who did his PhD on Hertzian waves would develop and experiment some years later with "radio broadcasting", where "music could be spread". The "audion" or "triode" used to amplify signals was required for such a use and was invented by De Forest in 1906. Later on, De Forest would play an important role in the creation of FTC, a UD-market oriented radio manufacturer.

9.1.2 "Market Emergence" phase: very beginnings of "mass production"

During the *Market Emergence* phase, which follows the *Concept Materialisation* one, there is a parallel activity of both old and new concept based development. Initially, the old concept attracts the interest of established market actors. Through conceptual variations it is marketed in a larger circle of clients, using the knowledge produced by the earlier phase for its production, while innovations concern aspects (components or features) of the early materialisation. User-Developer-Entrepreneurs, developing the product for their own use while exploring the potential of its commercialisation, can be "free riders" or institutional actors, nevertheless they are linked by a client-provider relationship, often difficult to formalise.

Then, new concept based development takes place by using the same knowledge of the early phase, though affirming the new value. The design of its extension provides the opportunity of further conceptual exploration, where UDEs play a crucial role.

In both cases, "intimate circles" are of high importance, as can enable the early sharing of major conceptual or cognitive advancement, between UDEs and enterprises, to be taken into account during the development process.

Hence, in the enterprise computer industry case, old concept based development was deployed by use of the previous phase knowledge, namely the Von Neumann computer architecture, while innovations regarded components - such as the use of a magnetic tape as a memory substrate. The *IBM 701* has, thus, been described as a defence calculator and its production was organised as a special project within the enterprise. The intimate relationships between IBM and Von Neumann were proven valuable to the former, as the *EDVAC*/Von Neumann architecture was the basis of the new product.

In parallel, a new trajectory of the computer was deployed through the *UNIVAC* project by the developers of the *ENIAC*. Always based on the same architecture and discussed in the same intimate circle, the *ACM*, further steps had been started towards the design of its expansion. Software tools were introduced for its users to be able to adapt it to their own particular needs. For that, a close relationship with those users, a technical support structure, was necessary for them to be able to extend the computer.

In the case of the personal computer industry, *Micral* further advanced its first model and managed to create a market for the microcomputer. *R2E*, its manufacturer, used previous knowledge in computer science to enrich the computer with an operating system and a programming language, which could be utilised by its users to adapt the computer to their own, specific tasks.

In parallel, the *Macintosh* computer, created by *Apple* a few years after its foundation, marked the end of the perception of the personal computer as a *Do It Yourself (DIY)* computer and began the era of the personal computer as a commercial product. *Macintosh* had exploited previous knowledge, most importantly conceptual exploration operated within the *Xerox PARC*, *Xerox* being one of the early investors in *Apple*. While *Apple*'s founders had already taken part in the early UD community, Macintosh marked also the "professionalization" of this community, with a developers community building software applications for it.

9.1.3 From Early Materialisation to Market Emergence: design strategies & tactics

Figure 9.1 schematically outlines the design strategies observed in the transition from *Early Materialisation* to *Market Emergence* phases, inspired by the works on CK theory (Hatchuel and Weil, 2009; Gillier et al., 2010; Lefebvre, 2013, and others). The central part of the schema refers to the different trajectories of the new and the old concepts. The left part refers to the intimate relationships developers construct, while the right part refers to the knowledge related to each level. The levels (reading them top-down) are structured from abstract to more specific concepts.

As already established in the previous paragraphs, there are two major design strategies to be used by developers during these two early phases: one which explores an old concept using the potential of a new theory (e.g. *"radio telegraph"*, *"calculator"* or *"minicomputer"*) and another which explores a new concept (e.g. *"radio broadcasting"*, *"computer"* or *"personal computer"*).

Then, there are two complementary tactics, that developers may follow to be able to materialise their conceptual projections. I name the first one "abstraction". It is the tactic according to which developers' early materialisations are task-specific, while the next ones are steps towards the materialisation of a more abstract and, thus, general concept. Typically, Eckert and Mauchly began with a task-specific computer (*ENIAC*) though having in mind the construction of a "universal computer" (materialised with *EDVAC*). The importance of this tactic lies in the fact that task-based products during *Early Materialisation* serve a known

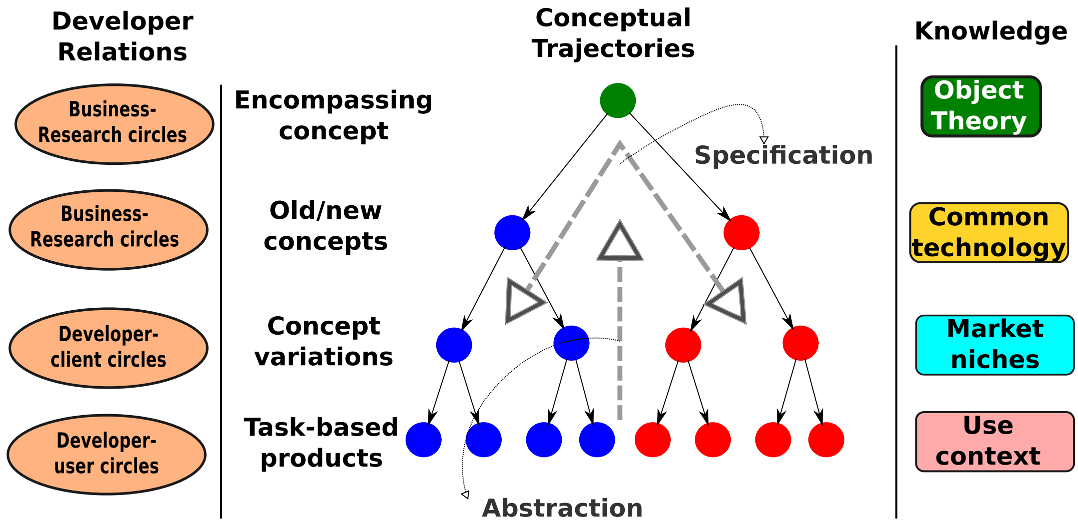


Figure 9.1: From *Early Materialisation* to *Market Emergence*: Design strategies.

and identified value, despite the fact that they use "bizarre technologies". This way an early inventor can have access to resources, in order to continue their exploration.

A second tactic is to try to materialise an abstract, general concept from the outset and then specialise it to specific markets. This is the case for instance of Garry Kildall who developed very early on a generic operating system for microcomputers. Still, during market emergence, Kildall lost in competition with *Microsoft* (as *MS DOS* had greater commercial success than Kildall's *CP/M* during *Market Emergence*). An even more typical case is the well-known example in STS discipline history of the *Diesel* engine (Bryant, 1976), which was not eventually commercialised by the heroic figure of Rudolf Diesel but, instead, by *MAN (Maschinenfabrik Augsburg-Nürnberg)*. Diesel wanted to commercialise a universal machine, but from the beginning, faced multiple commercial and technical disappointments, while MAN was able to tackle the issue with in a more systematic way.

9.1.4 "Foggy competition" phase: parallel exploration & exploitation. Enterprises, DE, consumers

During the *"Foggy competition"* phase, there is an expansion of the market, characterised by considerable product diversity. Actors commercially exploit their goods, while they still explore the potential of their own products. During this phase, there is a "business ecosystem", portrayed by co-existence of enterprises, consumers and developer-entrepreneurs (DEs). A critical point for competition, is the management of the relationship between the enterprises that have emerged during the earlier phases and the DEs, where those having already created intimate relationships with UDEs are in advantageous position. Knowledge is characterised here by segmentation and is based on components, while a new synthesis is required. Such a synthesis may include both the academic level and the enterprise one.

Hence, during the foggy competition phase of the enterprise computer industry, a great diversity of computers was available in the market, with multiple manufacturers (General Electric, Western Electric, IBM, RCA, Burroughs, to cite a few) proposing different materialisations of what a computer could be. Design variations depended on the different market niches addressed and the corresponding modules of the computer architecture added a value to that direction. For instance, an airline computer client valued a computer memory that could be modified "online", that is while the computer was processing, advantaging a direction that would later be materialised by RAM, a temporary computer memory component. This nebulous period ended with the dominance of IBM. IBM managed to dominate the market through the proposition of a rationalisation through modularisation, with the legendary S/360 computer series (Baldwin and Clark, 2000), a little bit after the term "Computer Science" was coined. DEC, an MIT student founded start-up, managed to compete in this environment based on a community of "hackers".

In the case of the personal computer industry, the 1980's were characterised by a great diversity of "personal computers", most of which did not look like the ones the broader public met with after the dominance of the *Wintel* model (Gawer and Cusumano, 2002) in the 1990s. The *Wintel* model along with the USB key managed to rationalise the Personal or Desktop Computer. While the basic elements of this model were included in the "IBM PC", as well as the earlier ones, the role of *Compaq*, based on a DE community, wasa a catalyst in its establishment.

9.1.5 Breaking a cycle, beginning another

In all three industrial settings, a phase sequence was observed: Early Materialisation, where UD played an important role, Market Emergence, where UD started entrepreneurship (became UDE), and Foggy Competition, where the entrepreneur orientation dominated the user one (DE). Then, there followed an Industrial Rationalisation, mainly by a synthesis that was at the same time both scientific and productive.

Still, in one case, this circle was broken. This was the case of DEC, building PDP, described by the enterprise as "not a computer" and addressing it to UD - while in a period of Foggy Competition, just before IBM proposed the rationalisation that helped them dominate the enterprise computer market. DEC's *minicomputer*, as it was later named, gained ground through a close collaboration with the university 'hackers', who used and extended its products. It followed a parallel trajectory to the enterprise and the personal computer, which generally lasted until the rationalisation of the latter one.

9.1.6 From Foggy Competition to Industrial Rationalisation: design and market strategies & tactics

Figure 9.2 illustrated design and market strategies during the transition from the *Foggy Competition* to *Industrial Rationalisation* phases. In parallel to the old/new concept trajectories of the two first phases, other trajectories are explored and exploited in parallel, drawing on newer or older concepts.

Such a case is for instance found in *radio programs*, as opposed to the early, "peer-to-peer" radio communication. Similarly, during the 1950s for the enterprise computer industry or during the 1980s for the personal computer industry, a great variety of "computers" where available in the market, produced either by large companies, such as *IBM*, or by start-ups. At this level, there were no common criteria hence the market looked more like a "bazaar" than a "tidy", market-niche segmented one. Typically, IBM engineers admitted back in 1950's that they knew very little about what computers can do, while the company already produced and marketed different computer products.

Most usually, the shift towards "newer or older" concepts is undertaken by a *"jump"*. This jump can be led by a market opportunity, calling for a technological innovation, or inversely, a new technological potential seeking a market niche. Yet, whether or not this

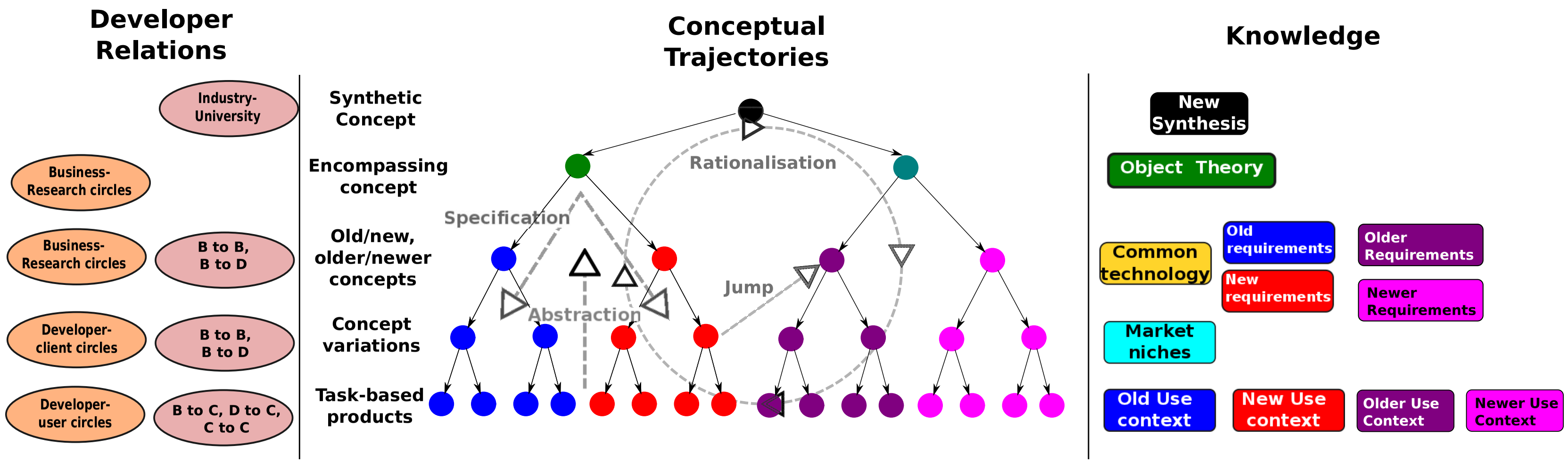


Figure 9.2: From *Early Materialisation* to *Market Emergence*: Design strategies.

"matching" will be realised, is an open question. Hence, during the 1950s, airline companies were much interested in the possibility to book tickets using computers. However, such a task demanded a different architecture than *EDVAC*'s one, as a "dynamic" memory was required. Component innovations (namely "magnetic drum memories", as in the case of *RCA* and *Burroughs Corp.* - section 10.4.1 and 10.4.1) illustrated the potential of fulfilling such a requirement. However, the manufacturers that tried that out failed, as more innovations were required in order for such components to serve those tasks in a professional way. This entrepreneurial mode is not a singular characteristic of DEs, as enterprises also act in the same way.

Foggy competition ends with a new synthesis, which takes into account all exploration experience of this phase and provides an encompassing concept that can introduce common criteria for all known markets and technologies. A famous example is the *IBM System 360* series which allowed the enterprise to reorganise its entire product lines and propose common price/performance criteria. Most competitors had to copy those criteria and follow the same, from then on dominant, design, resulting to a segmented market. Still, as *DEC* has shown, a jump is always possible, opening up a new circle that does not comply with the dominant design.

9.2 A comparison with the innovation-based industrial development approaches

While the starting point of my research was not to discuss the existing literature on industrial development but rather to take a historical distance from the object of Web innovation and the particular figure of the User-Developer-Entrepreneur to "measure" their genericness, the results of my study may contribute to the discussion on industrial development. Hence, in this section I correlate my findings with the major analytical frameworks used by business scholars, participating this way in the large and open-ended discussion on "where does industry come from".

This Section reviews the major conceptual approaches and positions my model has in respect to the academic discussion.

9.2.1 The innovator's dilemma: a new enterprise function?

In his seminal work, *The innovator's dilemma*, Clayton Christensen (1997) addresses the question of *"how can great firms fail?"*. He proposes an interpretation by the distinction between *sustaining* and *disruptive* technologies. Sustaining technologies *"foster improved product performance"* , while the disruptive ones are *"innovations that result in worse product performance, at least in the near-term"* (Christensen, 1997, p. 11), though they have some interesting characteristics:

> *First, disruptive products are simpler and cheaper; they generally promise lower margins, not greater profits. Second, disruptive technologies typically are first commercialized in emerging or insignificant markets. And third, leading firms' most profitable customers generally don't want, and indeed initially can't use, products based on disruptive technologies (Christensen, 1997, p. 12).*

Moreover, disruptive products have another characteristic, which contributes to their diffusion in a very small - insignificant for large enterprises - market: *they propose a new* *value* (Christensen, 1997, p. 175). Christensen also outlines a series of "principles" making large firms vulnerable at disruptive innovations. He observes thus that *"markets that don't exist can't be analysed"* as *"the only thing we may know for sure when we read experts' forecasts about how large emerging markets will become is that they are wrong"* (Christensen, 1997, p. 15). Thus, he formulates the innovator's dilemma as follows:

> *In many instances, leadership in sustaining innovations - about which information is known and for which plans can be made - is not competitively important. In such cases, technology followers do about as well as technology leaders. It is in disruptive innovations, where we know least about the market, that there are such strong first-mover advantages. This is the innovator's dilemma (Christensen, 1997, p. 15).*

Christensen does not provide an answer to the innovator's dilemma, but he indicates some observations, a methodology that can be helpful for managers, a set of principles. Firstly, he proposes that "trajectories maps" may help analyse the conditions and situate the company when faced with a disruption potentially threatening its business. Secondly, he remarks upon the difficulty of allocating resources to disruptive innovation concepts, as the return on investment will probably not be sufficient, until those innovations really come to a mature state. Thirdly, he also remarks upon the difficulty of identifying a market that would appreciate the new value introduced by the disruption. Fourthly, he notes that enterprise capabilities are set within the framework of an expertise and are thus very different from the new ones required for the production and the marketing of a disruptive product. Fifth, he points out that decision making is very difficult, as the risk is high. Sixth, regarding strategy, he underlines that companies don't always have to be leaders in all their activity fields. And seventh, regarding entry barriers, he argues that entry is more difficult than economists claim, as such barriers are related to *"things, such as assets or resources, that are difficult to obtain or replicate"* (Christensen, 1997, p. 172-174).

Hence, as Christensen shows, the adoption of an enterprise level analysis renders the identification of disruptive innovations a very risky and difficult task. The analysis by trajectory mapping he suggests will be used in the current study as well and will be further discussed in the methodology section. Still, while he notes that disruptions introduce new values, he studies their trajectories in comparison to the old performance criteria. Moreover, Christensen illustrates how the typical enterprise structure lacks the tools to envision and address innovations that potentially could *ex post* be proved as disruptive. Therefore, he suggests that *"disruptive technology should be framed as a marketing challenge, not a technological one"* (Christensen, 1997, p. 173), as the typical R&D structure fails to properly situate the issue.

Disruption as a jump during Foggy Competition

The model that resulted from my study may help further describe the phenomenon of disruption. Figure 9.3 shows the model by emphasizing the *jump* strategy. In fact, Christensen (1997) describes the disruption of a specific market by an innovation regarding a new value. This innovation however has to meet a specific market niche to succeed. While this market may be insignificant, further development may threaten pre-existing designs.

Hence, disruption can be described as a jump from a specific trajectory to a new one. Moreover, this jump can not only be triggered by a technological advancement of a specific enterprise seeking a market, as described by Christensen, but by a specific market niche demanding a breakthrough in existing technologies while they are not yet rationalised.

This was the case for instance of computer companies during the 1950's dealing with airline enterprises. The requirements the latter imposed (*to be able to modify data during program execution*) demanded technological advancements (*a dynamic memory*) that those enterprises hadn't been in position to develop - thus leading to severe economic damages.

Beyond modelling disruption, this model also proposes a way in which established enterprises may face it: by providing a new synthesis, by rationalising the design and the production of the different trajectories explored by the whole set of actors, and creating some common *design rules* (Baldwin and Clark, 2000), on the basis of which different product families may be produced, addressing every known market segment. Of course, new jumps may occur, towards concepts that are not included in the synthesis, as they hadn't been explored during the Foggy Competition phase by other actors. This was the case for instance of DEC's PDP (later called the *minicomputer*), which managed to compete with IBM's S 360 computer series in a specific market, the one of University developers (later known as *the first computer hackers*).

To do such a synthesis, as Christensen notes, an enterprise should have the knowledge to do it. Hence, a monitoring of the business environment is required to spot the new trajectories that emerge and be able to anticipate future synthesis necessary. In fact, as we will discuss in Part III, this is what major Web service providers actually do today.

9.2.2 Lead users and UDEs: on the emergence of markets

In line with Christensen's proposition that the emergence of disruptive innovation should actually be an issue for marketing, early studies on lead users highlighted the value user innovation may have for marketing research. Thus, von Hippel (1978b) proposed that, beyond conducting research for the measuring of already identified needs and desires, a challenge for enterprise marketing is *"to search out data on user prototypes, analyse the utility these have displayed in field use, and estimate their potential as commercial products"* (von Hippel, 1978b, p. 19). Later research though, focused on the capabilities of user communities to innovate and diffuse their innovations without the implication of enterprises, leading to an innovation model parallel to the manufacturers' one (von Hippel and Katz, 2002; von Hippel and von Krogh, 2003, 2006; Raasch and von Hippel, 2012), as already discussed in Chapter 6.

The implications of the research on lead users are to be found in a series of fields, varying from market forecasting (Funk, 2005), to further enabling users to innovate via tool-kits (von Hippel and Katz, 2002; Baldwin and Clark, 2006) or policy (von Hippel, 2005; Gault and von Hippel, 2009). Still, as the figure of the lead user is central in this approach, his positioning in industrial development is also crucial for the understanding of industrial dynamics.

Positioning the activity of lead users in comparison to the well-known model of Rogers (1962), Churchill et al. (2009) propose the schema shown in Figure 9.4.

Therefore, lead users are the ones who first innovate for their own use, as opposed to early adopters, who are the first clients of an emerging market. The existence of a market is thus the vertical line separating the two figures. In addition, user innovation literature (von Hippel, 1994; Ogawa, 1998; von Hippel, 1998, 2005; Lüthje et al., 2005) suggests that there is a division of "sticky information" between the technological level, retained by the manufacturers, and the use context one, maintained by lead users. Thus, user innovation is based on the latter information, allowing a "design division" of innovation process between

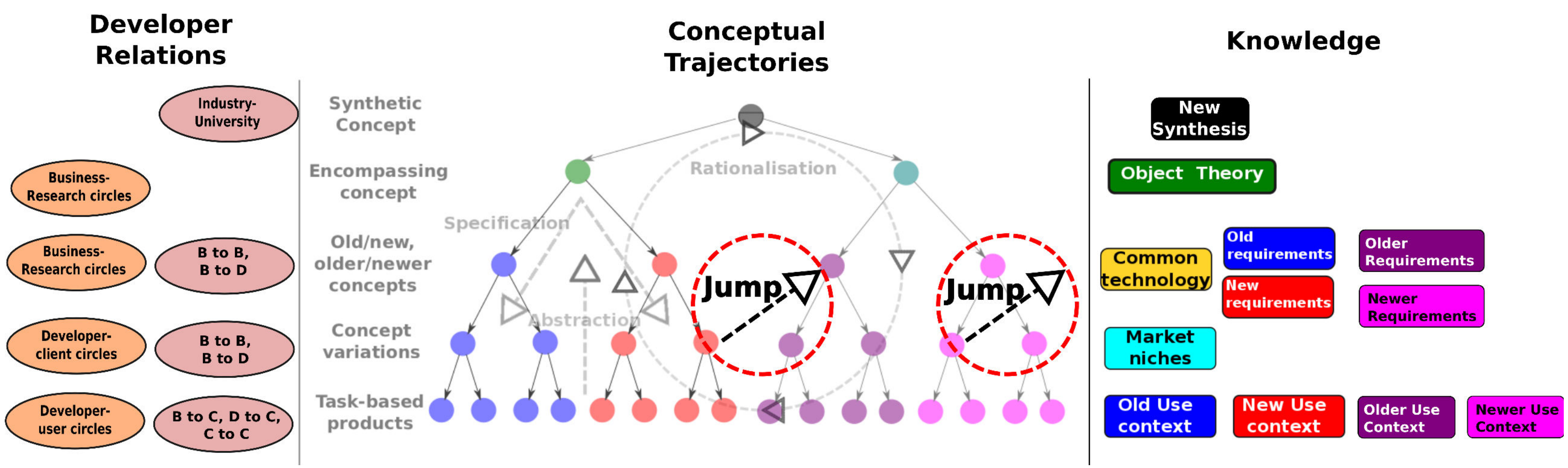


Figure 9.3: Disruption as a jump during Foggy Competition.

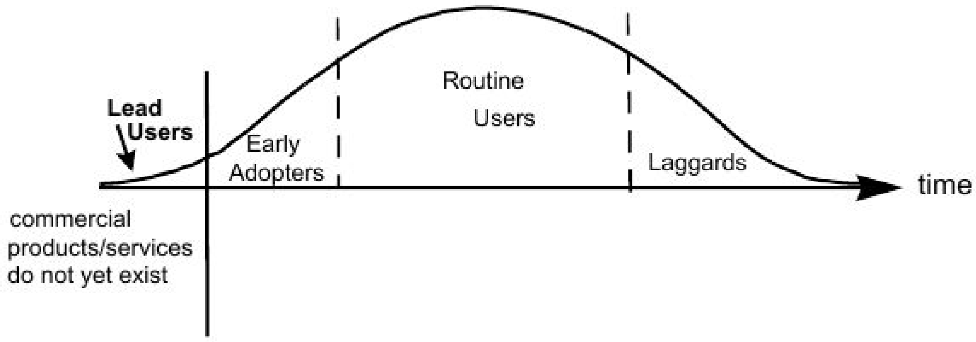


Figure 9.4: Positioning of lead users in Roger's model. Source: Churchill et al. (2009).

manufacturer and user.

However, my findings suggest a different perspective, both in what regards Roger's model and in what regards "sticky information", or knowledge.

On the one hand, they suggest that there is no "sudden" passage from lead users to manufacturers and early adopters. Instead, there is a progressive transformation of both the actors and the objects of innovation, as markets emerge and competition is unleashed. On the other hand, UDEs hold both technological and use related knowledge or "sticky information", while they also create a new knowledge base, related to potential markets.

More specifically, the figures of UD, UDE and DE, as analysed in Part I are active on the frontiers of market emergence. Figure 9.5 illustrates this positioning, as will be further explored in the current part of this thesis. While User-Developers fit the criteria of lead users, in the sense that they innovate for their own use, once entrepreneurial opportunities begin to appear, their action norms begin to shift. Thus User-Developer-Entrepreneurs share characteristics of both lead users and entrepreneurs, but their importance lies in the fact that they are the first to explore the possibility of a new market emergence. Soon after market emergence, Developer-Entrepreneurs do not innovate for themselves any more, but for existing and potential clients, these clients being early adopters.

In parallel, UDEs have to develop both use-context related knowledge and technological knowledge, as noted in the Figure 9.6, as they are the ones to create the early materialisations for their own use before attempting to market them. In their entrepreneurial quest, they also develop the first knowledge about the possibility of markets, at this level insignificant for enterprises, as Christensen notes.

9.2.3 Innovation diffusion and industrial development phases

The different innovation phenomena are very often analysed on the basis of the model of innovation diffusion proposed by Rogers by his seminal work in 1962. In a later version, Rogers (2003, fifth ed.), formulates his model as shown in Figure 9.7. The figures identified by Rogers met during innovation diffusion are the following: (1) innovators, who *"must be able to cope with a high degree of uncertainty about an innovation at the time he or she adopts"*, (2) early adopters, who *"help trigger the critical mass when they adopt an innovation"*(3) early majority, who *"follow with deliberate willingness in adopting innovations but seldom lead"*, (4) later majority, for which *"most of the uncertainty about a new idea must be removed before the late majority feel that it is safe to adopt"* and (5) laggards, who

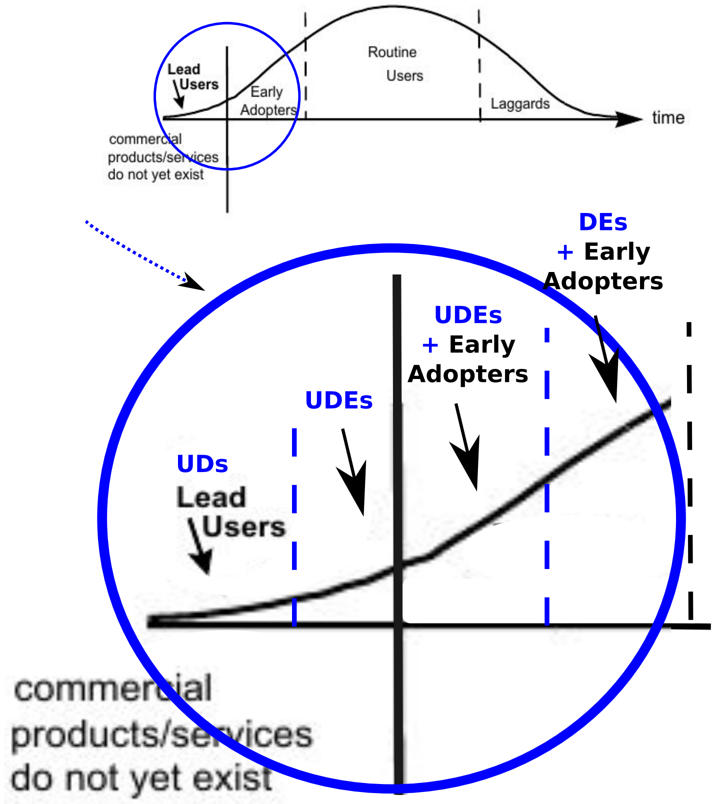


Figure 9.5: The position of UD, UDE and DE in the positioning of Churchill et al. (2009).

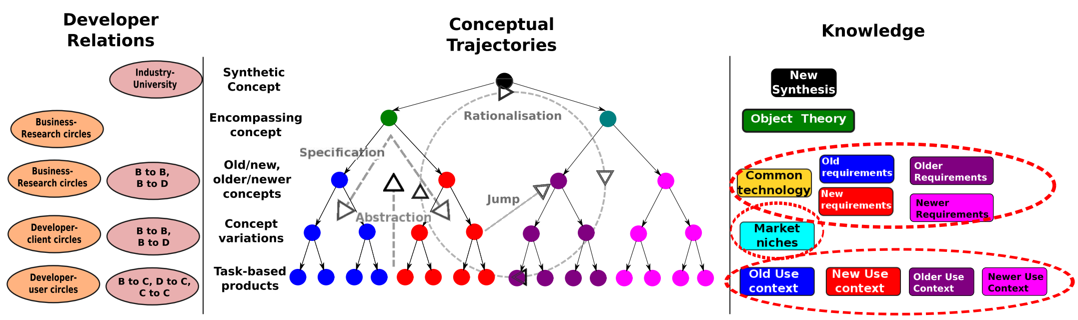


Figure 9.6: UDEs “sticky information”: use-context, technological and market related.

"tend to be suspicious of innovations and of change agents" Rogers (2003, p. 361-366).

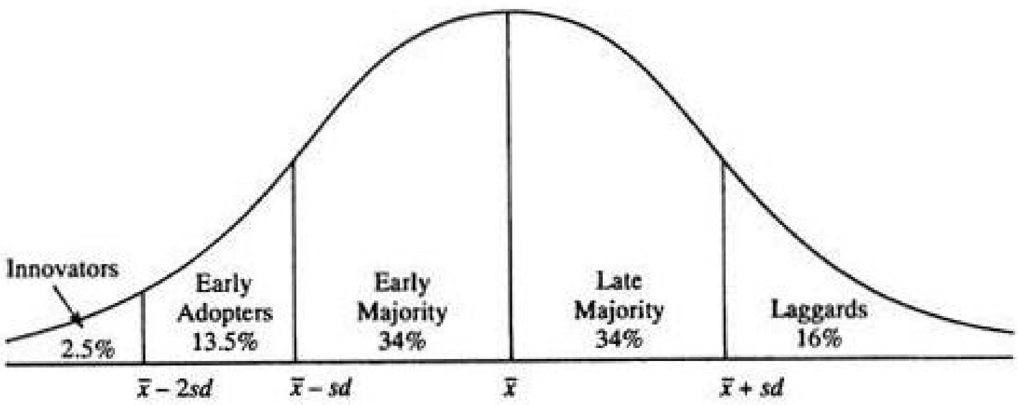


Figure 9.7: Adopter Categorization on the Basis of Innovativeness by Rogers (2003, p. 361).

The model I propose can be situated in relationship to Rogers' one as shown in Figure 9.8. As it is schematically shown in the figure, the overall "S-curb" is the aggregation of many smaller ones. Initially, during *Early Materialisation* two can be identified, as already stated, the old and the new concept trajectories. Then, the *Market Emergence* phase is a lot shorter, and because of this can be characterised as more of an event, when compared to the *Early Materialisation* and the *Foggy Competition* phases. The delivery of the first *UNIVAC* by the creators of *ENIAC* or the period from *Apple II* to the first *Macintosh* are such events, illustrating the market potential and triggering the period of *Foggy Competition*. Then, multiple DEs "enter the game", designing and selling products of similar kinds, though still exploring the potential of both the technologies and the market and, thus, inventing new trajectories. A new "jump" can emerge before *Industrial Rationalisation* (such as DEC's minicomputers), that can disrupt the industry in the future, unless included in this rationalisation. *Industrial Rationalisation*, again, can also be considered an event, as things radically change: the market segments already explored are afterwards coupled with specific product series and product lines and market variety is structured through new, "universal" criteria. For *IBM* it took three years of Research and Design to propose a new set of design rules (Baldwin and Clark, 2000) and more years to fully deploy its System 360 computer series.

Industrial Rationalisation establishes a *"dominant design"* by *"embodying the requirements of many classes of users of a particular product"* and *"enforces standardization so that production economies can be sought"*, eventually leading to a situation where *"effective competition can then take place on the basis of cost as well as product performance"* (Utterback and Abernathy, 1975; Suárez and Utterback, 1995). For this synthesis to be possible, additional innovations are required in order to *"connect the dots"*, as Steve Jobs used to say, and propose a design that takes into account the exploration of the *Foggy Competition* phase.

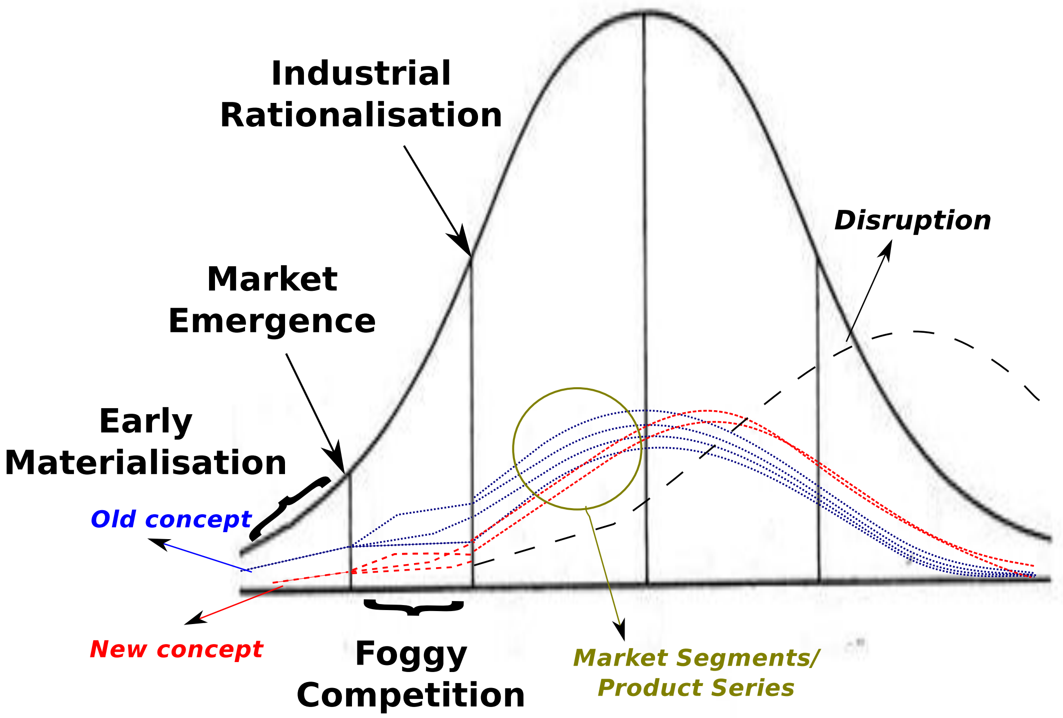


Figure 9.8: Positioning of the model I propose in Rogers' one.

9.3 Research Methodology

Christensen (1997) studied the history of the disk drive industry following a friend's advice, since in this industrial domain *"market structure, global scope, and vertical integration have been so pervasive, rapid, and unrelenting"*, in a manner that could be metaphorically compared to *fruit flies* very short life cycle: they are *"conceived, born, mature, and die all within a single day"* and are studied by *"those who study genetics"* for this very reason (Christensen, 1997, p. 20).

Garel and Mock (2012) studied the story of the famous *Swatch*. This joint work of an academic and an entrepreneur, the latter having taken part in the process studied, used the C-K theory (Hatchuel and Weil, 2009). Unlike the disk drive industry, the watch industry is characterised by very large cycles. In fact, as the authors show, *Swatch* was a revolution, as it has been that an industrial production of watches - in terms of the capacity of manufacturing of millions of similar products - which required a transformation of both the object and the process of its development.

In a similar way, I will study the patterns on three industrial settings, the enterprise computer, the personal computer and the radio. The life cycles of these industries have been greater than the one of disk drives and shorter that the one of watches.

Christensen was interested in answering the question "why do great companies fail?", hence required a history with frequent enterprise failures. Garel and Mock (2012) exposing the history of one of the rare though revolutionary innovations in the watch industry were capable to study the problem of radical innovation *"in the making"*. My research aims to further explore the question "what is the role of UDEs in industrial development". This question emerged from the first part of my study, where I identified a figure acting in-between the user and the manufacturer innovation paradigms (Raasch and von Hippel, 2012), in the particular field of Web services development. The goal of the current part is to identify whether this activity, poorly explored by the literature so far, is a peculiarity (von Krogh et al., 2012) of the Web services phenomenon, or, on the contrary, is an activity recurrently appearing in industrial development in general. For this, I have chosen to study industrial settings that are as vast enough to be able to be compared to the Web services domain. While Christensen's question - enterprise failure - led him to investigate short living industries, mine obliges me to study their "death", and I chose to study longer cycles, for this peculiar actor figure to be identified, and then studied, within the course of industrial development.

My study is inspired by genealogical approaches to the study of history, where the starting point of historical research is a question of the present and the objective is to explore the *"conditions of specificity deployment possibility"* (Lefebvre, 2005). This posture is different to the one of longitudinal studies which are interested in studying the dynamics of a particular system (e.g. an organization), on the basis of "how" questions and in terms of an *input-process-output* analysis (van de Ven and Huber, 1990).

Genealogical approaches are based on the work of Foucault and the notion of *"problematization"*, even though, as Castel notes, *"the work of Michel Foucault does note explicitly accord a central place to the notion of problematization"*. More generally, the impression I had during my own study of Foucault's work, is that, despite the pluralism of terms and methods he introduces, he rarely uses definitions at all, being against the idea that something (anything) can be defined in a singular and universal way, beyond a specific "problematization field". At the same time, he suggested that his work be used as a "tool-kit" for researchers (Kendall and Wickham, 1999). Castel cites Foucault concerning the issue of material treatment:

> *Whoever, on the other hand, wishes to study a* ***problem*** *that has emerged at a given time must follow other rules: the choice of material as a function of the givens of a problem; the focus of the analysis on those elements likely to resolve it; the establishment of relationships that permit this solution. Hence the indifference to the obligation to say everything, even to satisfy the assembled jury of specialists (Castel, 1994, p. 242).*

Hence, many scholars interested in the history of business use this approach to investigate the transformations in time of a present problem. Aggeri and Labatut (2010) use a genealogical analysis to study the theoretical approaches in management that have been based on management instruments, Lefebvre (2009) studies the genealogy of the work contract, , Garel (2003) proposes a history of project management, Chatzis (2008) studies the history of the creation of the maintenance activity in the enterprise context.

Still, while the genealogical approach values the moment of emergence of a specific *"problematization"* and further explores its transformation in a long chronological scale (Castel, 1994) - typically, many of the problematizations studied by Foucault started in ancient Greece - my goal is not to identify the beginnings of the role of User-Developer-Entrepreneurs in industry. My ambition being more modest, I will use a "limited problematization", aiming to explore whether or not UDEs have existed in other industries during the previous century and if they have, whether or not the study of other settings may provide us useful insights on the "conditions of the UDE specificity deployment possibility", to enable us to better understand the kind of stakes and challenges to which today's online services business are called to respond.

Research Material: works on history and user-developer Press

A major reference for my study will be the seminal work of Paul Ceruzzi (2003) *A history of modern computing*. Taking a distance from the "social constructivism" approach, Ceruzzi reviews the history of computing after WW II methodically examining the computer as seen from the *"engineer's workbench"* (Ceruzzi, 2003, p. 5). My reading of Ceruzzi's history will be based on the problematization of UDE's role in industrial development. Thus, placed in between what Ceruzzi calls "social constructivism" and the "engineer's workbench", I will be interested in the workbench, its engineers and the conditions for them to work on the bench.

A second reference work will be the one of Patrice Flichy (1995a), *"Dynamics of modern communication : the shaping and impact of new communication technologies"*, which I will use as an entry for the study of the radio industry. While Flichy studies communications as a social phenomenon and is positioned in the debate between historians which highlights the competition of individual actors, my approach will be based, as previously, in the problematization of UDE's role, while the level of my study will of be themselves as a figure-type and their "bench".

This approach will lead me to look at aspects that the two authors consider as secondary, although they provide precious references in their endnotes to continue the investigation. For instance, Bill Gates is a prominent figure both in the history of computing and today's world business. Still, the fact that he wrote technical articles addressed to the user-developers for the review of a company producing a computer kit called *"Altair"* which had an "insignificant" market, is not something historians will value as a fact of historical importance - incidentally, neither has Computer Science - which is why it is not referred anywhere. Yet, when compared to the givens of my problem, it obtains a different meaning, as it is registered in one of the necessary phases for industrial development, namely the passing from *Early Materialisation* to *Market Emergence*. Such information was possible to trace thanks to the Web, as many Web repositories, like the *DigiBarn Computer Museum* (*www.digibarn.com*), have done precious work in gathering and publishing online historical user-developer addressed Press.

Chapter 10

The invention of the Enterprise Computer

Contents

The historical study begins with the examination of the enterprise computer industry. While popular culture remembers *ENIAC* as "the first computer", the decades that followed were not just a period of "innovation diffusion". Developer-entrepreneurs commercially exploited the "new machine", while exploring its potential and experimenting with different designs. The "foggy competition" phase in the "mainframe" computers ended with IBM's System 360 computer family rationalisation.

10.1 Introduction

The emergence of the computer industry that started after WWII is a complex case, an adventure engaging multiple actors and illustrating the dialectical relationship between innovators, emerging uses, markets and technologies, which frames the course of industrial development.

The current study reveals a richer picture of innovation and its market diffusion than the one provided by the approaches previously reviewed. In fact, there is no clear distinction between innovation and its diffusion, as put forward by Rogers, before an industrial rationalisation: innovations continue to be necessary and critical during market expansion. During the *Foggy Competition* phase, developer-entrepreneurs, often originating from the early circle of user-developers, have a catalytic role both in exploring and exploiting a new technology.

Enterprise computer "early adopters" define new requirements, apt in their own context of activity and the projection of use they make of an innovation. In the case of enterprise computer industry, developers had to further innovate in order to provide computers that could be used in different contexts, while these new machines were being diffused in businesses.

The computer can be generally seen as a disruption (Christensen, 1997) in the enterprise equipment and machinery market. However, innovators' dilemmas were not limited to the identification of an appropriate market: the computer was shaped through the interaction between developers, potential and early adopters, previous equipment manufactures. This interaction occurred on the basis of technologies and theories available or emerging. Early user-developers, such as the developers of the *ENIAC*, opened up a market which would take about two decades to be rationalised and, thus, segmented in clear niches and the corresponding product diversity. Equipment enterprises, such as *IBM* or *Remington Rand*, embraced the new technology, though further development and understanding of it was not a challenge that could be faced with old performance and organisational criteria.

Since the *ENIAC* early materialisation, the question "how to compute" has been the basis of an exploration providing different computer and calculator architectures, as well as software itself. As figure 10.1 suggests, the distinction of early and later phases was not based on a "task division" between lead users, carriers of "use-context sticky information" and manufacturers, carriers of "technology related sticky information", as user innovation literature suggests (von Hippel, 2005). In practice, what can be *ex post* distinguished as a dilemma between "sequential computing" and "parallel computing" has been decisive for innovation both in regard to the use context and the technologies. On the one hand, sequential computing covered much of the requirements of administrative use context, such as the need to calculate and print payrolls every month. The requirements for memory in this case were quite flexible and different technologies (from punch cards to electronic tubes and, much later, semiconductors) helped perform this task efficiently.

On the other hand, services, being based on client interaction, such as airline companies, required much more frequent or "real time" calculations. Hence, while payrolls, for instance are printed once and can be calculated before that, every ticket reservation changes the ticket availability and the next client should be able to take that into account. Here, parallel computing had a specific value for services. Still, a "dynamic" memory was required for such a task, one which could not be embodied within existing technologies. Those developers active in the field innovated with "*magnetic drum* memory", having quite poor results. Until *IBM*'s engineers invent the RAM, passing through the "*magnetic disk* memory", the use of the computer for "real time" programming was a very dangerous "jump" for developers, as compared to other architectures.

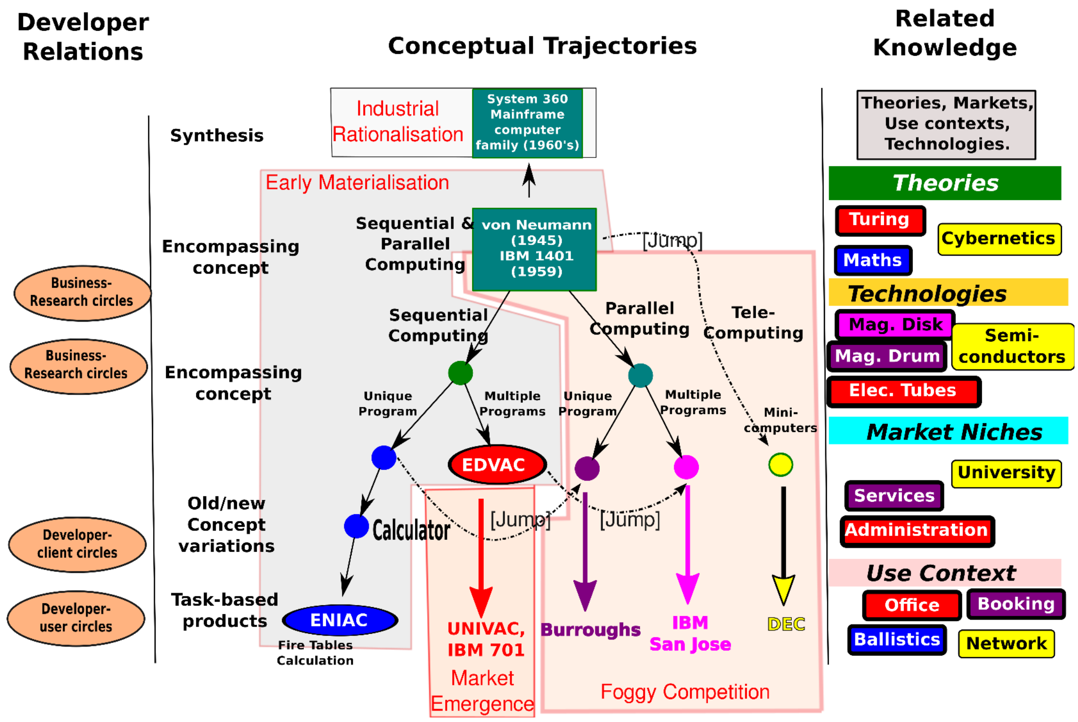


Figure 10.1: The emergence of the enterprise computer industry. Early Materialisation, Market Emergence, Foggy Competition and Industrial Rationalisation.

10.1.1 Chapter overview

Table 10.1 outlines the structure of the current chapter. The chapter begins with the *Early Materialisation* phase, reviewing the well-known case of the *ENIAC* computer as well as a less known but more important computer, the *EDVAC* (Section 10.2). Yet, before that, a review of the intimate circles of user-developers which had emerged some years before this project (paragraph 10.2.1) and their contribution to the *ENIAC* project is judged necessary for the better understanding of their developer's actions. The section continues with the review of *ENIAC* development, which can be analysed as an *Early Materialisation* of a computer: while addressing the functional requirements of its user, the US Army, for being a single

Table 10.1: Chapter overview.

Section	**Phase**	**History**	**Trajectories deployment**	**Previous phase input**			**Phase challenges**			**Figure**				**Role**
				R	**C**	**K**	**R**	**C**	**K**	**UD**	**UDE**	**DE**	**Enterprise**	
10.2	Early Materialisation	From *ENIAC* to *EDVAC* (1943-1951).	Old: ad hoc, task based materialisations; New: a design goal to reach	An exploratory collectivity.	Potential for a "universal computer".	Theory use.	Collectivity stabilisation, client search.	New criteria creation.	"How to"; previous criteria comparison.	✓				Development, discussion, milieu incorporation.
10.3	Market Emergence	From UNIVAC to IBM 701 (1951-1953).	Old (IBM): a "separate project" included in firm's culture; New (Remington): a novel, side product.	From fellows to early clients.	Object architecture.	Object development "how to".	Client support.	Object-task alignment.	From "ad hoc" to multiple products manufacturing.		✓			Further client search & support, production systematisation.
10.4	Foggy Competition	Until *IBM S/360* (1953-1965).	Parallel exploration & exploitation (ex. *Burroughs*, *IBM 1401*).	Market existence.	Some trajectories to further explore.	Some development techniques, theory.	A "business ecosystem".	New trajectories emergence.	Wide, empirical potential exploration.			✓		Conceive new uses, create new markets.
10.5	Industria Rationalisation	IBM S/360 (1963-1967)	Most known trajectories	Market niches	Wide empirical exploration	Dispersed know-how	Market structuring	Additional innovations	Common price/performance criteria proposition				✓	Synthesise knowledge, propose common design rules.
10.6	Phase break	DEC PDP (since 1957)	Trajectory "jump"	>>	>>	>>	UD return	New concept introduction.	Latest knowledge use.	✓				Development, discussion, milieu incorporation.

task machine, for its developers *ENIAC* was a first illustration of the new potential, enrolled in a more ambitious design goal. Using their experience as well as their relations, a second materialisation, *EDVAC*, was the first "universal computer", a future-oriented universalism emphasising computer potential. Its architecture constituted a first knowledge base for the next phase. New criteria, such as the "stored program" were introduced. The exploratory activity of Mauchly and Eckert was discussed and attended by a user-developer "milieu" now more stabilised. This milieu has also been the place where Mauchly first searched for clients to finance the development of the first commercial computer. This early experience created a "how to" that was later proven to be crucial for the industry to emerge.

Section 10.3 pursues the story of Eckert and Mauchly with the development of *UNIVAC*, the first commercial computer. While the two actors and the company they built did not manage to systematise computer production to fit previous criteria (such as manufacturing robustness), their business was acquired by an equipment manufacturer, *Remington Rand*. One year after the first computer delivery, *IBM*, with the help of von Neumann who had studied *EDVAC*'s architecture, created its *IBM 701*, which was internally called a "defence calculator" as a "separate" project. Market Emergence signified two major changes: on the one hand, other actors would soon begin to enter the game. On the other hand, utilising the "universal computer" in different use contexts was proven to be a great challenge, as client support required both the establishment of a training process and the continuation of innovations.

Hence, a period of *foggy competition* started thereafter, where multiple actors entered a market that was still "under construction" (Section 10.4). Computers were generally produced and marketed as "machinery", while exploitation went hand to hand with exploration. A typical case in this phase is the one of *Burroughs* company (paragraph 10.4.1). It was created by the developers of *ENIAC*'s memory and attempted to question the "von Neumann" architecture, trying to address the requirements for tasks such as airline ticket booking. This "jump" was proven to be far too ambitious and *Burroughs* exited the enterprise market, though it continued business through military projects. This problem was solved by the invention of RAM by IBM engineers Hence, while the industry still under emergence had inherited from previous theories an early market, some starting points of exploration and a theory, the business ecosystem had yet new trajectories to explore, new uses to invent and new markets to create, while in competition. A wide, empirical exploration was the result of this phase, which was exploited to propose a synthesis, an industrial rationalisation.

Section 10.5 briefly reviews the rationalisation of the computer development by IBM. *IBM* was based in the previous phase learnings, of which an important event had been the invention of RAM and the IBM 1401 "universal computer" success, developed by RAM inventors. The *IBM* System 360 computer family covered almost every market niche that had been previously explored by synthesising the existing know-how and proposing common performance/price criteria. This was achieved by the institution of common design rules and the modularisation of computer architecture (Baldwin and Clark, 2000).

Still, not all trajectories had been included within the S/360 family. During the late 1950s, DEC drew a different trajectory, that would later be called "minicomputer", using the latest inventions on semi-conductors (Section **??**). DEC was targeted at user-developers in the university, who became known as the early hackers, and collaborated with them in the further exploration and development of its PDP "data processors".

10.2 Early Materialisation: from ENIAC to EDVAC

ENIAC was a fundamental stage in the emergence of the computer industry. Paragraph 10.2.2 will look into the project itself, as well as the role of its two designers in this adventure. While Ceruzzi (2003) begins his history of modern computing from that period on, it is equally important to note the emergence of the very concept of a 'universal computer', that played such a decisive role as we will see. As most historians note, the concept had emerged some years before the actual project of *ENIAC* began.

10.2.1 The "universal machine" notion and its early UD circles

An important contributor to this early phase, as the historian Yates (1997) shows, was Edmund Berkeley. Berkeley was not an engineer, neither did he invent any computer. However, he was deeply involved in the study of life insurance methods, a field also present in the foundations of modern computing methods, such as of Babbage's early work, *A Comparative View of the Various Institutions for the Assurance of Lives* (Babbage, 1826).

After earning a BA in mathematics and logic from Harvard University in 1930, Berkeley joined the British *Prudential Mutual Life Insurance* of New York as an *"actuarial clerk"*. He later moved to the methods department, where as a mathematician he investigated the enterprise's methods and processes, especially symbolic logic types and new types of computing machinery.

Berkeley played an important role in the gathering together of all actors interested in computing methods into an intimate collectivity. One of the early collectivities he animated, was the *New York Symbolic Logic Group*, in 1941. In fact, he kept extensive notes on his activity, on which the research of Yates (1997) is based[1].

Berkeley's contribution in the first phase of the emergence of the computer concept thus included networking, though it went beyond it. He was actually implied in the cognitive part of what came to be the discipline of computing and was also active in creating a place, a *milieu* for researchers, entrepreneurs and scientists working in enterprises and institutions to share knowledge and concepts on computing machinery.

Following the research of that time and Alan Turing's theory on "universal machines"[2], he was the author of a series of papers on the potential applications of "symbolic logic" to insurance company problems during WWII and later of the first popular book for computers, *Giant Brains, or Machines That Think* (Berkeley, 1949).

In parallel, Berkeley visited equipment manufacturers at the time, including *General Electric*, *IBM* and *Bell Labs*, in order to identify whether there could be computing machinery useful for his insurance company. As Yates mentions:

> *While neither of these machines was well-suited to insurance uses, his reports from these contacts indicate that these potential vendors were learning about*

[1]Yates notes about the consistency of this group:

> *The group that met to form this association included several professors from New York universities, as well as several individuals from the "application side of symbolic logic," including Dr. Claude Shannon of Bell Labs (the father of information theory) and three representatives of insurance firms (including Berkeley himself) Yates (1997, p. 63).*

[2]For an extensive review of the emergence of Alan Turing theory see the collective work of Husbands et al. (2008), *The mechanical mind in history*.

market needs from him at the same time that he was learning more about the technology[3].

We could assume that the fact that this intimate collectivity started in New York may be related to the realisation of the World's Fair in 1939 in this city. In fact, Mauchly, one of the designers of *ENIAC*, recalled in an interview in 1973 getting in touch with *Remington Rand* enterprise in this fair, and learning from the company's electrical multiplier, one of the elements he used in the design of *ENIAC*[4]. Mauchly actually had to get in touch with the companies support service and look into the manual to acquire further knowledge. As we will see later, *Remington Rand* was to play an important role in the computer industry during the 1950s.

The legacy of this first intimate circle includes the creation of the *Association for Computing Machinery (ACM)*, which would follow up *ENIAC*'s development long before it became one of the major computer societies, as it is known today. *ACM* did not emerge by a formal decision. In 1947 Berkeley was already connected with every enterprise, researcher or administrative official related to computing. He himself initiated the *ACM*, in an informal way, by gathering the most prominent members of the community that had emerged, John Mauchly and Dick Bloch, the programmer of another computer project in Harvard[5].

In ACM's early conference, there would also be illustrations of prototype computers, while user-developers would exchange information on "how to build a computer"[6].

A few years later, *ACM*, together with the *American Institute of Electrical Engineers*, and the *Institute of Radio Engineers* formed the two computer conferences, the *Western Joint Computer Conference (WJCC)* and the *Eastern* one (*EJCC*), that began in 1951[7]. In 1961, the *American Federation of Information Processing Societies (AFIPS)* was to be formed, and the two conferences were to be renamed *Spring and Fall Joint Computer Conferences*, an Association that would be dissolved in 1990[8], after being the "umbrella trade organisation of computer societies".

10.2.2 The development of ***ENIAC***: an early materialisation based on an old concept

The *ENIAC (Electronic Numerical Integrator and Computer)*, the machine weighing eighty tonnes, has often in the past been praised by the scientific community and the popular press

---

[3]Yates (1997, p. 64).

[4]Tropp (1999, p. 62).

[5]Here is how he recalled the initiative he undertook in 1947, twenty years later, during an anniversary meeting of the ACM:

> *I started putting together a small group who would call for the formation of some kind of an association, a temporary committee for an Association for Computing Machinery, and Dick Bloch was on the committee and John Mauchly was on the committee and I was on the committee and a few more people were on also and we rapidly put something together Tropp (1987).*

[6]Tropp (1987).

[7]Bureau of Ordnance, U. S. N. D. B. of. (1948). *Proceedings of a Symposium on Large-Scale Digital Calculating Machinery*, 7-10 January 1947. In Symposium on Large-Scale Digital Calculating Machinery (p. 370). MIT Press.

[8]Ceruzzi (2003, p. 305).

as the "first computer". However, this "lust" for identifying the "first" of computers[9] over the years has often overshadowed the factor that positioned, in my view, *ENIAC* in a prominent place in the modern history of computer industry: the activity of its own designers, John P. Eckert and John W. Mauchly. Seen from Mauchly's perspective, we could say that *ENIAC* was an intermediary project, something more than a prototype, an early materialisation.

Following their activity, through the work of historians, as well as documents of the time (mainly commercial material or specifications) we will retrace their entrepreneurial trajectory which will provide us with useful insights for today's business.

John Mauchly was interested in computing machines since the 1930s. His extensive use of such machines began after he completed his Ph.D. in Physics at Johns Hopkins University, when, motivated by his passion for weather forecasting, he engaged in research in the area of meteorology[10]. In 1941 he enrolled in an *Engineering - Science - Management War Training Course (ESMWT)* offered by the *Moore School of Electrical Engineering of the University* of Pennsylvania, and designed to teach electrical engineering to scientists to enhance their skills for wartime work[11].

In 1942, Mauchly wrote a memorandum to the School administration, describing the computational advantages of an electronic machine[12], that would exploit the electronic technology already developed in the radar industry. He discussed his proposal with Eckert, a young research assistant in Moore School and a highly skilled technical engineer. Still, the undertaking of such a project became feasible only when the government needed high calculating performance for the needs of the war[13]. Once the contract for *ENIAC* was eventually signed, Mauchly became principal consultant and Eckert chief engineer[14]. The *ENIAC* project, contracted in 1943, was delivered in 1945.

ENIAC was an electronic computer using vacuum tubes, designed to calculate fire tables for the U.S. Army. This task, though very specific, involved the repetitive solution of complex mathematical expressions[15]. In fact, *ENIAC* automated computing tasks[16] by synchronising them with an electronic clock pulse[17]. Many authors agree that the value of *ENIAC* resided

[9]Some, going back the to he *Antikythera mechanism*, dated around 100 BCE and the Hellenistic Period, discovered in the beginnings of the 19th century by sponge drivers in a wreck near Antikythera island. The function of this mechanism, re-established decades later, was to calculate astronomical positions. In modern times, astronomical calculations began to be done in accounting machines during the 1920s: *IBM Type 405* accounting machine could operate an astronomical calculation involving 12 implicit multiplications in 42 hours (Cortada, 2000, p. 134).

[10]Stern (1979).

[11]Stern (1979).

[12]John W. Mauchly, "The Use of High-Speed Vacuum Tube Devices for Calculating," Aug. 1942; reprinted in Brian Randell, The Origins of Digital Computers (New York, 1975), pp. 329-332.

[13]Ibid.

[14]Ibid.

[15]Ceruzzi (2003, p. 15).

[16]"Computing tasks" had already been defined during the 19th century. Charles Babbage, an English mathematician, philosopher, inventor and mechanical engineer, had designed a automated mechanical computer, though he hadn't been able to complete its construction. Its design was the result of his studies 'On the Economy of Machinery and Manufactures'. There, he had proposed that 'the division of labour can be applied with the same success to mental as to mechanical operations'. This proposal was based on his experience of the 'French Tables of Logarithms', which was the response of the 'most distinguished French philosophers' to the desire of French government to 'produce a series of mathematical tables, to facilitate the application of the decimal system which they had so recently adopted'. Babbage has also illustrated that these computing tasks could be mechanically operated by the use of coordinating clocks. Charles Babbage, 'On the Economy of the Machinery and Manufactures, London, 1835, pp 191-202.

[17]ENIAC's arithmetic calculations were performed by 'counting' pulses. As Flichy (1995b) mentions, it

in its capacity to mathematically compute complex mathematical expressions[18], something that Mauchly had early on envisioned. One major step in *ENIAC* design was the utilisation of vacuum tubes (a mature technology at the time, industrially exploited in West Coast radar factories) for a computing machine. This step was to be further explored and expanded with the *EDVAC* computer. Yet, the universality of this value was to be illustrated and further deployed a few years later, through the entrepreneurial adventure of the two designers.

Discussion : ENIAC, an "early materialisation".

Entrepreneurial activity often faces the difficulty of a high "entry cost", that is the initial capital for Research and Development needed (Utterback and Suárez, 1993). Moreover, when the market does not yet exist (in our case a commercial market) the initial cost is even more important: exploration costs can be even higher, as radical innovation consists in operating in the unknown, making it impossible to predict the costs.

Seen from its designers' perspective, *ENIAC* was an *early materialisation*: it allowed them to advance their knowledge on computers as well as the design towards the projected concept of an electronic universal computer (*ENIAC* did use electronic tubes, even though it was not 'fully' electronic). In parallel, this contract ensured an institutional and financial framework for the two entrepreneurs to operate in.

It is important to note here a double property of its design, which might seem contradictory: *ENIAC* was task-specific (fire tables calculation), though its design opened the way for new innovation trajectories. A look into the specifications may enlighten us to this possibility: on the one hand, the requirements were imposed by its sole client (the U.S. Army) to complete a specific operation, thus not a class of users in a market. Nevertheless, the design was completed in a way that the "requirements of many classes of users" (Suárez and Utterback, 1995), *could* be addressed, thus a dominant design *could* emerge. In other words, we propose that *ENIAC* can be characterised as an early materialisation because it *integrated the potential of addressing many classes of users in its design*, beyond the specific client who commanded it, though having satisfied the latter.

In this sense, an *early materialisation* is different from *half-designed solutions* (Hatchuel and Weil, 1999), *not-yet-complete design* (Baldwin and Clark, 2006) or 'incomplete design' (Garud et al., 2008). In those cases, design is not complete, thus there is no evident value of use, as third parties need to intervene in its design and development in order for end users to buy it. In *ENIAC*'s case, the product was complete, fulfilling buyer's specifications, yet its design opened new possibilities for the entrepreneurial activity of its two designers.

10.2.3 The development of *EDVAC*: an early materialisation based on a new concept

An important step forward for the two scientists, as well as for the history of computer business, was the *EDVAC (Electronic Discrete Variable Computer)* project. It was a project signed between the *Army Ordnance Department* and the Moore School in 1944 and was to be a complementary project to the *ENIAC*, though incorporating major design changes. *EDVAC* was the first "all electronic" and "universal computer" and was delivered in 1949.

While the institutional context for the two computers was the same, *EDVAC* was an early materialisation of the new concept of a *computer*, managed to establish the requirements of

had a capacity of 200.000 pulses per second.

[18] For instance Ceruzzi (2003); Flichy (1995b).

a "universal computer", on the basis of which a reference design would emerge.

EDVAC incorporated many innovations that remained a reference for the industry that was to emerge a few years later. Without going into details, we can say that the major requirements (Suárez and Utterback, 1995) that constituted this early materialisation, were the following:

1. *EDVAC* was an "extended electronic computer", as it completely replaced punched cards with electric tubes.

2. It introduced the module of "stored programs" using the same method to store both information (data) and information processing methods (programs)[19], setting out the foundations of the computer programmer profession.

The first characteristic mentioned above revolutionised the machine, as it incorporated in its architecture "state of the art" industry of that time. On the one hand, production of electronic tubes had already been rationalised. On the other hand, their utilisation in *EDVAC* enabled not only faster calculation power but faster programming time. In terms of knowledge, Mauchly and Eckert exploited existing knowledge for the development of a radically new concept.

At the same time, the materialisation of the principle of stored programs (conceptualised on a theoretical level by the English mathematician Alain Turing before the war), opened the way for the creation of a new profession, the one of programmers, that was to be actively present in all the transformations that capitalism met after the war.

Architecture disclosing and conceptualising

EDVAC's design was discussed between the two designers and John von Neumann, a prominent mathematician leading research for the U.S. Army, in March and April 1945, few months after the contract was signed. In the next weeks, *EDVAC*'s architecture was described by von Neumann in his *First Draft of a Report on the EDVAC* and was soon diffused in military and academic circles[20]. This action caused the disappointment of the two entrepreneurs, who envisioned the patenting of their computer[21]. In fact, they filed a patent a year later, to which Moore School administration did not agree. The patent was rejected in the 1970s, as von Neumann's report had disclosed its main principles.

Thus, *EDVAC*'s architecture remained known as the *"Von Neumman architecture"*. This report, while incomplete, clearly described the interfaces of the artefact and remained a reference for Computer Science (except for the human-computer interface). Few years later, von Neumann wrote a three volume book, the *Planning and coding of problems for an electronic computing instrument, report on the mathematical and logical aspects of an electronic computing instrument*, explicitly stating the principles of computing for the Engineering Schools public.

Through this controversial episode, not only was the knowledge diffused, but *EDVAC*-like computers were widely developed in the framework of the "Manhattan project", under the direction of von Neumann. Though not the only computer developed for the Army (another

[19]While *ENIAC* was at the time very fast on calculating, programming the machine necessitated days of work in manipulating its cable connections.

[20]Baldwin and Clark (2000); Norberg (2003); Ceruzzi (2003).

[21]Ibid.

one was the *SAGE* command and control system, developed by *IBM*)[22], *EDVAC* surely installed a considerable knowledge base, both practical and theoretical, into the scientific community of the time.

Entering the path of commercialisation.

At the same period and while computers continued to be seen as scientific and military instruments, Mauchly and Eckert wanted to commercialise their invention. The first business opportunity that opened up for them was their contract to undertake a study for the development of a computer for the *U.S. Census Bureau*[23]. Initially, the two proposed to the Moore School administration *EDVAC*'s commercial exploitation, beginning with this contract. Yet, after the disagreement of Moore School administration the two left to form a partnership in 1946, the *Electronic Control Company* which was incorporated as *Eckert-Mauchly Computer Corporation (EMCC)* two years later[24].

During these two years, John Mauchly undertook a market study identifying twenty-two industries, government agencies or other institutions that could be interested in such a machine and this despite other studies (such as the one by the National Research Council in 1947) stating that there was no place for such an instrument in the market. One crucial reason for Mauchly's market research was the fact that the resources of the contract signed for the first computer study (at a fixed fee of $ 169 000) was quickly proven insufficient for his ambitious project[25]. The list of potential clients included the *Army Map Service*, the *Bureau of Aeronautics*, the *Metropolitan Insurance Company* and various aircraft companies. Mauchly and Eckert were conscious that they were starting a new industry.

However, client demand itself had to be created. In other words, clients had to be seduced by the utility of a new object in their specific enterprise context. While Mauchly undertook this task during his early market search, by contacting organisations that could potentially be convinced to buy a computer, the University undertook a role in knowledge diffusion, necessary for the product to get wanted. Ceruzzi comments:

> *Customers took the initiative and sought out suppliers perhaps after attending the Moore School session in 1946 or visiting a university where a von Neumann type machine was being built. These customers, from a variety of backgrounds, clamoured for computers, in spite of a reluctance among UNIVAC or IBM salesmen to sell them (Ceruzzi, 2003, p. 45).*

[22] *SAGE* computer designed and manufactured by IBM for the U.S. Air Force opened the conceptual space of the User Interface. This computer was a "command and control" device, allowing military officials with low technical skills to control a wide amount of radar information concerning approaching aeroplanes and command actions. The SAGE contract generated half a billion dollars in revenue for IBM in the 1950s. Despite the fact that by the time the system was delivered, missile technology was advanced and, thus, its utility to trace aeroplanes was out of date, this project for IBM and the emerging computer industry was of a great importance. As Ceruzzi (1989, p. 77) mentions, with SAGE the computer industry (especially IBM) learned how to produce high-performance machines on a serial basis in decent production runs.

[23] Stern (1979, p. 10).

[24] Ceruzzi (2003); Stern (1979).

[25] Stern (1979, p. 11).

10.2.4 Discussion. Early materialisation: Conditions for the transition to Market Emergence.

During the phase studied in the previous paragraphs, there has been no innovation diffusion during this early phase, where the model of Rogers (1962) could be discussed. In addition, describing these early user-developers as "lead users" could be problematic, since there was no evident product that they commonly used before the invention of the first computers.Both use-related and technology related knowledge (von Hippel, 2005) were, yet, to be invented - for that to be a competitive advantage later, when computers would disrupt (Christensen, 1997) the business machinery market.

It is important to note however that Mauchly and Eckert entered their adventure with *UNIVAC* having a concept in mind. Hence, while the concept of "universalism" is too abstract to grasp, it led as we've seen to the non-linear design process through different projects (*ENIAC*, *BINAC*, *UNIVAC*) and achieved a design genericness.

In Mauchly and Eckert's activity, the way to address this barrier was by constructing an intermediary product, *ENIAC*. This product had an autonomous identity that didn't depend upon future research. It complied to the buyer's specification and, thus, its development could be financed autonomously. What assured the continuity between the *ENIAC* and the projects to come, was a design reasoning (the one described in Mauchly's memorandum in 1943) in which this product could be attributed an intermediary position. As we are going to see in the next section, the intermediary product method was followed by the two entrepreneurs during the following years as well.

Then, *EDVAC*, exploiting the same institutional and financial configuration, was an advanced materialisation of the concept (in particular in what concerns the full exploitation of electronic technology to enable Turing's concept embodiment regarding stored programs).

However, those elements wouldn't be sufficient to permit the dominance of this design in the market to come. Firstly, the two entrepreneurs had to engage in a journey to the market, which - as we are going to see - posed new challenges for them. Secondly, the fact that von Neumann disclosed and rationalised on the theoretical level *EDVAC*'s architecture was a necessary condition for it to become a 'knowledge platform': Manhattan project computer development as well as academic knowledge on computers were based on these design restrictions. When finally the two entrepreneurs managed to get the first computer to the U.S. market, there had already been an institutional support for the new business profession necessary for the product to be used: the *programmers*. Of course, this entry also implied an openness to competition.

Thus, to summarise, we identify the following steps of a strategy for the emergence of a reference design:

- Developing intermediary products which
  - are positioned within a broader design reasoning of which they constitute necessary steps.
  - They satisfy a particular demand and can, thus, be separately financed, diminishing exploration costs.
- Disclosing architectural knowledge on the interfaces, on a theoretical basis, in a way that *learning* can be possible for third parties. These conditions open up a risk and an opportunity:

- the opportunity for others to invest in the design, diffusing it and supporting its dominance; reinforcing path dependency (Robinson et al., 2007, and others), widening user base (Farrell and Saloner, 1986, and others) and benefiting from future lock-in effects (Farrell and Klemperer, 2007, and others)
- the risk that competition enters into the same market, questioning the enterprise's dominance, though not the design (this potential will be discussed in the next Section).

Challenges	**Business Rationale** Marketing by potential	Single client, targeted projects	Businessmen - University intimacy
Design oriented			
Can a generic design be reached?		Intermediary products	
Can the object be described theoretically?		von Neumann architecture	Knowledge diffusion
Can the generic design be restricted to applications?			Enterprise "modernisation"
Market oriented			
Can there be a client demand?	A network of potential clients		University - business ties
Can the project be financed?		A network of contractors	Demand creation
Can we configure the product for specific tasks?			Extensions demand
Can we scale production for demand?			"Grace" period
		Business environment transformations	

Phase 1 : From a concept to a product

Table 10.2: Challenges during concept materialisation phase (from the concept to a product) and the responses given (business rationale) in the case of UNIVAC. The output of this process is the business environment transformations.

10.3 Market emergence. Entrepreneurship and enterprises: early production systematisation, innovation and marketing.

According to Suárez and Utterback (1995):

> *A dominant design has the effect of enforcing standardization so that production economies can be sought.*

The design of *EDVAC* was not in position to produce effects of an economy of scale. However, it served as a reference for the early user-developer-entrepreneurs to further explore and develop computers.

Still, despite the fact that they shared the same design principles, the path from *EDVAC* (a university project having the army as its client) to *UNIVAC* (a commercial computer of which the success was uncertain at that time) was not linear. Back then, imagining that there could be other classes of clients for computers beyond the Army was rather audacious. Besides, in what concerns the business trajectory having as a sole client the *U.S. Defence Department* - being thus a *monopsony* - it went on until our days[26], proving that it generally was less a risky choice for an entrepreneur to stay in military equipment business. Thus, in *UNIVAC*'s case, Mauchly and Eckert had to invent a new commercial market, and to prove that a universal computer product business was feasible. Moreover, while the *EDVAC* project had already explored the principal attributes of a universal computer, *UNIVAC* (initially called by its designers *EDVAC II*), had to be re-designed in parallel with the emergence of the computer industry to fit its specific needs. The first commercial computer was eventually delivered by *Remington Rand* in 1951, an enterprise that acquired Eckert and Mauchly's company.

However, from 1946 to 1951, various steps were to be followed by the two entrepreneurs to arrive to their objective. A look into this very process can provide us with precious insights for its specific needs, as the model of start-up acquisitions still constitutes a business expansion mode regarding the Web platforms sector.

10.3.1 Other materialisations and the limits of UDEs

In 1947, *EMCC* signed a contract with the *Northrop Aircraft Company of Hawthorne*, California for a small-scale computer, called *BINAC (Binary Automatic Computer)*. A few months earlier, *Northrop* had hired Mauchly as a consultant to explore the adaptability of electronic digital equipment to suit its needs[27]. It was to be a single-purpose computer, for in-flight navigational control of missiles[28]. Yet, for Eckert and Mauchly the project was another case of an intermediary product, a necessary step towards *UNIVAC* - though their client wasn't aware of it. *BINAC* incorporated innovations such as the *stored program* as the use of *magnetic tape* as a data substract (initially similar to the one used in taper recorders, later, in *UNIVAC*, replaced with a more reliable tape treated with an iron oxide coating)[29].

[26] See for instance Ceruzzi (2008).
[27] Stern (1979).
[28] Stern (1979).
[29] Ibid.

Ultimately, the result exceeded the requirements of the contract in terms of design. Nevertheless the artefact presented defections[30]. On the one hand, its manufacturing quality was heavily criticised, on the other hand its complexity and novelty made it difficult for *Northrop* engineers to re-assemble. Concerning their contract, which described the expectations the client had from the provider, it did not clarify whether or not the equipment's construction should also be rigid[31]. The client asked for additional technical support, yet *EMCC* was instead invested in the building of *UNIVAC*[32]. Eventually *BINAC* wasn't ever put into use and Eckert and Mauchly focused all of their technical attention on completion of the six *UNIVACs* which were on order (three for the government, one for the *Prudential Insurance Company*, and two for the *A.C. Nielsen Company*[33]).

While this product met the intermediary design goals set and brought the two entrepreneurs a step closer to their final product design, it exceeded the anticipated exploratory costs: while the computer was contracted for $100 000, its development costed about $278 000[34], putting both *EMCC*'s and *UNIVAC*'s future in question.

10.3.2 Acquisition and further developments

During the late 40's, the start-up, despite a considerable progress both in the development of the technology; market exploration and contracts, was confronted with a debt accumulation. So, the company was acquired by *Remington Rand*[35].

Remington Rand re-negotiated the contracts signed by *EMCC* by augmenting the price. The first *UNIVAC* was delivered to the Census Bureau and many corporation clients followed. *UNIVAC* was rent at a monthly cost of $ 22 410[36].

Large scale production of computers implied an industrial challenge which needed an important production experience to undertake. Actually, completing such a task was an argument for *Remington Rand* to use in its marketing[37]. Early steps of computer modularity

[30]In particular, the computer worked during the demonstration in *EMCC*'s headquarters in Philadelphia, yet it didn't work when moved to California.

[31]Ibid.

[32]Stern (1979).

[33]Ibid.

[34]Ibid.

[35]*Remington Rand* had been active in the typewriting machines since the 19th century and had undergone many organisational and market challenges, arriving to a reinforce its market position after the WW II, also having close ties with military and the *Manhattan Project*. Being one of the major suppliers of office machinery during those years on an international level, *Remington Rand* had both the network and the experience to support a commercial computer, as a new product of enterprise machinery. See for instance Cortada (2000); Stern (1979); Schlombs (2008).

[36]Specifically in Europe, where the computers capacities were way beyond enterprises needs, *Remington Rand* followed a different business model. They created a computing centre in Frankfurt and rented time to European customers (on hourly basis), provided professional personnel for programming jobs, and trained company employees (Schlombs, 2008) .

[37]In a *UNIVAC* commercial flyer, the enterprise presents this challenges as follows:

> *There are 975,000 parts in each Univac, ranging from tiny crystal diodes to the sturdy beams that support the computer frame. This equipment is the most complex ever built, involving intricate scheduling of parts and careful coordination of complicated workloads. These problems and the layout of production and assembly areas have all been carefully worked out to enable Remington Rand to keep pace with the ever-increasing demand for Univac Systems.*

The same flyer mentioned that components were manufactured by "experienced craftsmen". However, the fact that these craftsmen were women and, by that time, women labour was largely un-qualified, illustrates that an advanced rationalisation of this work had been undertaken. For instance, in the case of the semifor the needs of mass production had also been undertaken, with "small, removable service units called 'chassis,' many of which are interchangeable" and can be replaced in case of defection. . However, it was not until the 1960's and *IBM System/360* that modularisation became a complete managerial method in computer industry.

Since the two developer-entrepreneurs set up their company, *EMCC*, and started developing their computer, one of their first preoccupations had been to provide new users with applications as well as tools to build their own. Of course, in both cases, user training has been thought as a requirement for the business to operate. While user training was a business practice already met during the 1930s in the domain of administration machinery, enabling users to build their own extensions was a novel concept.

In the case of *UNIVAC*, user innovation (von Hippel, 1986) was not only something that 'could happen', it was a user potential designed by the company, providing specific means for users to extend it (something that von Hippel and Katz (2002) would call toolkits for user innovation). *UNIVAC* eventually was an open product (Chrysos et al., 2010), prescribing specific uses while enabling user-developers to create their own, secondary products.

This dual relation between (manufacturer) enterprise and (enterprise) user, characterised by product-related knowledge diffusion, on the one hand, and a product extension on the other has since marked the development of this particular industry.

To that aim, *EMCC* acted on two levels. Firstly, Mauchly created a specific department charged with the development of applications for the *UNIVAC*, as well as the development of what we would call today software tools. Applications were focussed to specific, identified market niches (such as payrolls). Software tools were addressed to enterprises that could use the computer as a platform to create their own applications. Many of the people employed for this task had already worked for the *ENIAC* project, and organised their work through design conferences[38].

Provided that Mauchly himself had taken the responsibility of commercial research, the activities of this engineering design group were twofold:

- satisfying the requests of potential customers for specific applications and tools (such as building task-specific applications, like payrolls, procedural flows, accounting and purchasing).

- extending the design of the computer[39].

However, as already mentioned, *EMCC* did not manage to sell any *UNIVAC* computer before being acquired. On the organisational level, one of the first actions of *Remington Rand* was to build a commercial-engineering support department. This department produced a documentation of the computer, trained and assisted the clients. Ultimately, this department was merged with the engineering design department Eckert and Mauchly had already developed, as an in-depth knowledge of UNIVAC's structure was necessary in order for the

conductor industry which emerged about a decade later, the introduction of women labour in the production signified its rationalisation by dividing work in simple tasks that non-qualified workers could perform, after a training conceived by the enterprise, namely the *Fairchild Semiconductors*. See Lecuyer (2006); Weil (2010).

[38]During the "design conferences" of the staff many decisions on the features of the computer and, thus, the research questions to address, were made. Norberg (2003).) mentions the *Conference on the EDVAC II Design* that took place in March 1947, where a series of design issues emerged, building up an agenda for the year to follow.

[39]Norberg (2003).

advices to clients to be efficient. Internal engineers' conferences continued to go on, and the enterprise came up with some important extensions of the initial computer [40].

10.3.3 Discussion. Market emergence: a return to the platform debate: extension design and organisation.

Ceruzzi summarized the period we examined as follows:

> *Commercial computing got off to a shaky start in the early 1950s, Eckert and Mauchly, who had a clear vision of its potential, had to sell their business to Remington Rand to survive. Remington Rand however did not fully understand what it had bought. IBM knew that computers were something to be involved with but it was not sure how these expensive and complex machines might fit into its successful line of tabulating equipment (Ceruzzi, 2003, p. 45).*

To proceed to the commercialisation of their computer, Mauchly and Eckert were based in the initial UD circles in different ways: the used it as a pool for developers recruitment and as a pool for their first clients (computer early adopters). Moreover, keeping up with the concept of a universal computer, they design early on its extension, by specific tools and applications for its user-developers.

However, the transition from user-developers to developer-entrepreneurs required the compliance to those early adopters. In particular, a robustness equivalent to the ones of existing machines was attended. For that, an industrial production of the computers was required, something that *Remington Rand* eventually undertook. Still, the two developers continued their work within this enterprise, organising the interaction with the early adopters.

This way, the computer concept passed from a user to a manufacturer innovation model (Raasch and von Hippel, 2012). Still, this passage did not mean a transition to an era of innovation diffusion (Rogers, 1962; Raasch and von Hippel, 2012): additional innovation were required for this transition to become possible (such as the keyboard input) while yet many more were to follow. The business rationale of this transition is outlined in Table 10.3.

The phase of market emergence marked the evolution of the computer industry at the following levels:

- It provided it with the general principles, the main functions of a computer, as described by von Neumann (processing unit, memory, stored program, input/output), the functional design according to Pahl et al. (2007), where the main 'tasks' of a computer system were defined and which would be the reference of the industry for the years to come.

- It provided an initial model of design extension, providing task-specific application as well as generic tools for the construction of new ones.

- It provided a labour design, resulting from the design of use that would dominate the industry for decades. Valid for the use case of the Census Bureau, it consisted in a sequential use of the elements of the computer (program development, input, processing, output), that was crystallized by a centralised organisation of computer

[40] This team further advanced the the *UNITYPER* (an input keyboard that looked much like a typing machine), developed a card-to-tape converter (enabling the transformation of data that written in punched cards to magnetic tape format) as well as printers (Norberg, 2003).

	Business Rationale				
Challenges	Marketing by potential	Single client, targeted projects	Businessmen - University intimacy	Engineering Design department	Acquisition
Design oriented					
Can a generic design be reached?		Intermediary products			
Can the object be described theoretically?		von Neumann architecture	Knowledge diffusion		
Can the generic design be restricted to applications?			Enterprise "modernisation"	Tools and specific applications	
Market oriented					
Can we inspire client demand?	A network of potential clients		University - business ties		Advertisment
Can the project be financed?		A network of contractors	Demand creation		Enterprise Integration (tension)
Can we configure the product for specific tasks?			Extensions demand	Client support	
Can we scale production for demand?			"Grace" period		Existing methods use (limits)
					Business environment transformations
		Phase 1 : From a concept to a product		***Phase 2 : Market emergence***	

Table 10.3: Phase 2: Market emergence. Same challenges, additional business rationalisation and emerging environment transformations.

operations, having at its centre what were known as the 'clerks', controlling the input and the output of the computer and its peripheral actors, namely the programmers.

- It linked computer production with previous equipment industry. This link provided robust business methods to the industry-to-be (such as marketing, customer support as well as machinery manufacturing), of which the relevance to the new object was however later to be questioned.

One could claim that through the acquisition of *EMCC*, *Remington Rand* managed not to be disrupted (Christensen, 1997) by the computer innovation. However, much remained yet to be invented, in order for the actors of the field to be in position to understand the potential of this new object.

10.4 Foggy competition: business ecosystem creation, design experimentations

In this section we are going to study the first period of computer business growth, when commercial computers started being introduced to the market. We are interested in configuring the "computer value" promotion and exploitation, as well as the conditions for this.

Eckert and Mauchly managed to establish a reference design, whilst pioneering in the emergence of the commercial computer industry[41]. Von Neumann played a crucial role in this process, by producing the theory behind the architecture, as both his *Draft report on the EDVAC* and his book on computer engineering became the standard reference for computer science for years to come. One shouldn't neglect the fact that this architecture was also used for the computers of the Manhattan Project.

This does not mean that, once the possibility of a market for computers was verified with the delivery of the first UNIVAC, other architectures did not emerge. In fact an important number of different computer architectures emerged during the 1950s[42]. Yet, *EDVAC* had already been the reference point for both learning and developing computer systems: the common knowledge base of the early computer history was built according to its architecture, while the production systematisation process followed by *Remington Rand* had diminished the cost by a significant level. By the end of the 1960s, 6000 general-purpose electronic computers were installed in the U.S.A, nearly all of them being descendants of the *EDVAC* computer[43].

From 1951 when the first *UNIVAC* was delivered by *Remington Rand* to the Census Bureau to the mid-50's, the environment had radically changed:

- Von Neumann's theory had contributed in the rationalisation of computer science itself, allowing computer science to be taught in Universities as such.

- *UNIVAC* had illustrated not only that a market for commercial computers existed, but that computers could provide great productivity growth in a series of organisational and industrial activities.

[41]We should note that, at this level, we cannot refer to a dominant design (Abernathy and Townsend, 1975; Abernathy and Clark, 1985), since there has been no market in which to dominate.

[42]For instance the *BIZMAC* computer by *RCA* (Ceruzzi, 2003, p. 56), which included important architectural innovations though didn't meet a commercial success.

[43]Ceruzzi (2003, p. 58)

- Different enterprises had entered the market, exploring different technological trajectories, always having one foot in the military and the other in commercial markets.

UNIVAC had become the synonym fom computer, gaining ground in the public opinion as the technology of the future[44] and had illustrated the potential of this new product. Moreover, the introduction of computers in the industry passed from the large scale use of *UNIVAC* in *General Electric*: payroll, material scheduling and inventory control, order service and billing, as well as general cost accounting were among the first tasks that were 'automated' within the enterprise by the use of computers. Besides, as Ceruzzi notes, the spirit of automation, popularised by John Diebold (1959), was inspiring the industrial transformations of the time[45].

During the early years of the 1950s computers appeared as 'special' enterprise equipment in the pre-existing market of business machinery. In this context, the product systematisation we have seen in the previous section mainly concerned the rules of construction and less the identity, the design of the object. At the end of that decade, common performance criteria were established and society, businesses as well as computer manufacturers themselves all realised the great potential of these devices. Similar to the role of Diesel's engine, which opened up a new industrial era and established petrol as a "universal value" for the economy, computers in the 1960s were transformed from enterprise tools to the engines of the information economy[46].

During the 1950s, with economic growth and the Cold War influencing the general business environment, a wide range of computers and enterprises flourished in the U.S. Computer market which started to blossom, as a niche, high-tech market. Given the fact that many of the features described by von Neumann architecture (including memory and user interface) had yet to become robust enough for them to be suitable for a commercial market, it was generally a decade of costly innovations that didn't provide high revenues. Moreover, as many historians note, the enterprises that, during this period, assumed the role of computer production rationalisation didn't quite well understand the specificities of this new object and practically included it in their strategy as just another type of equipment: the potential of computer was to be gradually explored by its same manufacturers and users. More over, new uses - mainly the use of computers by airline companies - were to question the reference design, on the basis of a different labour design than the one assumed in the beginning. As booking systems necessitate a continuous interaction of the agent with the data, the sequential design of labour in the beginning (with Input/Output being controlled by the clerks) did not fit for this new case. However, a new type of memory, easily accessible and modifiable was to be invented for this purpose. While drum memories gave a first solution, a second one was to be introduced with the invention of the RAM and its embodiment by integrated circuits.

In Figure 10.2 we see the revenues of selected computer and electronics manufacturing firms in 1955. *General Electric* was the largest electronic equipment manufacturer in the U.S. at the time, being at the same time the largest electronic tube supplier of *IBM*, with

[44] Two of the most notable tasks operated by *UNIVAC* computers, the prediction of Eisenhower's victory over Adlai Stevenson in the 1952 presidential election and the "complete Fiscal Budget estimate for airborne equipment spare parts". See Johnson (1952); Ceruzzi (2003).

[45] Ceruzzi (2003), p.32

[46] Following the spirit of Babbage and the 'French Tables', we could say that computers were automating intellectual labour. This industrial trend came to join previous research in the emerging field in computer science and the work of Alan Turing on 'mechanical intelligence' and 'mind machines'. See for instance Turing et al. (2001) and Husbands et al. (2008).

Table 2.1
Revenues of selected computer and electronic companies, 1955

Company	Annual sales	Net profit	Employees
GE	$2.96 billion	$213 M	210,000
Western Electric*	$1.5 billion	$55 M	98,000
RCA	$940 M	$40 M	70,500
IBM	$461 M	$46.5 M	46,500
NCR	$259 M	$12.7 M	37,000
Honeywell	$229 M	$15.3 M	25,000
Remington Rand**	$225 M	$12.2 M	37,000
Raytheon	$177 M	$3.5 M	18,700
Burroughs	$169 M	$7.8 M	20,000

Source: Data from *Fortune* (July 1955).
* Western Electric was the manufacturing arm of AT&T, which owned and controlled it. AT&T's total revenues for 1955 were greater than GE's, RCA's, and IBM's combined.
** In 1955 Remington Rand merged with the Sperry Corporation, a company with $441 million in sales, mostly defense-related.

Figure 10.2: Revenues of selected computer and electronics manufacturing firms in the 1950's. Source: Ceruzzi (2003).

sales of almost $ billion and over 200 000 employees (compared to *IBM*'s sales of $461 million and 46 000 employees or *Remington Rand*'s $225 million and 37 000 employees in the same year[47].

At the same time, electronic manufacturing was not synonymous to computer manufacturing. A fact that is little emphasised by the scholars of that period is that, from 1955 onwards, the major actors of early computer manufacturing were retired from this activity, creating the space for new ones to emerge.

The following paragraphs will briefly review the adventures of the major actors of computer manufacturing during the foggy competition phase. Eventually, the commercial success of the *IBM 1401* Universal computer would trigger a great shift in the enterprise and in the industry, illustrating the commercial potential and triggering the *System 360* rationalisation process.

10.4.1 Computers to the market (of machinery): exploring value and technologies

Remington Rand: constructing the enterprise computer value

A look into the way *Remington Rand* marketed *UNIVAC* and as well as the way *IBM* re-designed the computer will allow us to understand the transformation of the business environment and the breakthrough from a technical substratum to an engine of the new economy.

Although Mauchly undertook market studies at a very early stage of their endeavour, relations with enterprises and marketing were part of *Remington Rand*'s business expertise

[47] Ceruzzi (2003, pp 54-55).

that the two entrepreneurs could not reach.

In *Remington Rand*'s commercial campaign for *UNIVAC* we can identify the core business models that accompanied the first era of the commercial computer.

A film produced by *Remington Rand* in 1955 describes the value added by the installation of the computer within the enterprise, using the enterprise-centric performance criteria, values related to the tasks of the employees and management[48]. In the same film, *Remington Rand* describes how the installation of a computer within an enterprise takes place. Their description on payroll automation illustrates a reflection on the task-related operations in order for them to be automated[49].

UNIVAC was either sold (at the price of one million dollars) or leased. In both cases, *Remington Rand* assured technical support for its clients[50].

A new actor: the early role of IBM

As we already have mentioned, one of the characteristics of this design was the existence of internally stored programs, what gave birth to software. In fact, the hardware standardisation that was imposed by the initial influence of the EDVAC design in the early market, opened the way to the software expansion. Yet, it was not Remington Rand who exploited in the best way this new potential.

One year after the first UNIVAC's delivery, IBM announced the 701, a stored-program computer (called "electronic data processing machine") in the same class. The first shipment

[48]In this film, *UNIVAC* is described as follows:

> *A complete electronic system for sorting, classifying, computing and decision making.*

The same value representation was also marketed in the film by the use of their first contract with the Census Bureau for marketing end:

> *UNIVAC is handling automatically and economically unbelievable volumes of statistical work for the U.S. Bureau of the Census. Work that formely took weeks and months to do, is now being done in a matter of hours by UNIVAC.*

Remington-Rand Presents the Univac, Duration: 17:31, in 2:06 and in 2:20.

[49]The film mentions characteristically:

> *Prior to the actual installation, a team of programmers assists in making a complete analysis of the factory's pay system. Breaking it down into piecework, hourly pay rate, salaries and commissions and other classifications. They provide for overtime rates, tax deductions of all the many variations, social security deductions, insurance deductions, union dues and anything else that affects an individual pay record. Once the necessary tapes have been recorded, and placed on UNISERVOS, the payroll operations become automatic, simply a matter of processing data through UNIVAC. In less than four hours per week and with only a small operating staff, UNIVAC can complete the computation for this payroll for 15 000 employees. A saving in time and money that is tremendous*

Film, op.cit., min. 13:10.

[50]Here is how UNIVAC's renting model was described by an advertising flyer of the 1950s :

> *You and your company can profit from this experience in either of two ways: First, if you want to explore the possibility of purchasing or renting a computer for full-time use, Remington Rand offers a series of training courses in electronic computing equipment. Second, if you have an immediate application for the Univac System which can be handled by occasional use, Univac equipment and personnel are available through our Computer Center services. Operating 24 hours a day on a contract basis, these services are currently being used by business, industry, and government to solve all types of data-processing problems.*

UNIVAC System, Remington Rand Rand, INC, 1955, p. 10.

of the *701* outside *IBM* was to the nuclear weapons laboratory at Los Alamos in 1953. It was based on an advanced design of a computer built by von Neumann - who became its consultant - in the Institute for Advanced Study at Princeton[51].

The *IBM 701* was known as Defence Calculator within the enterprise, illustrating its perceived market. In fact all of the 19 models were installed either for the U.S. Defence Department or military aerospace firms[52]. However, as Ceruzzi reports, *IBM 701* was used for similar tasks as the ones the UNIVAC was performing: logistics for a military agency, financial reports, actuarial reports, payrolls. Moreover, the development of *FORTRAN* programming language by IBM for arithmetical calculations, opened the way towards software [53].

While, as we already mentioned UNIVAC could be bought (at the price of one million dollars), IBM followed a rent-only business model for its 701 (with initial rental fees at $15 000 a month)[54]. However, the enterprise had already been condemned by the U.S. Justice Department, which had alleged that it violated anti-trust laws in conducting its punch card business (before entering the electronic computer market). Thus, a new business model was "forced" to IBM, with companies buying its computers and selling their services[55].

IBM engaged in a process of computer evolution on the basis of its 701. However, as Usselman (2007) comments, even with a rental price of thirty thousand dollars per month, the 705 could never generate substantial revenue, as the enterprises where not ready to implement computer technology.

Overall, this period was not limited in the diffusion of an innovation, the computer, by the enterprises. Many elements were yet to be invented, while both commercial opportunities and technological potential were explored. Either directly, or indirectly, those participating in the early user-developer circles around the developers of the *ENIAC*.

Remington Rand's inability to follow up innovation.

In the middle of the 1950s, *Remington Rand* had serious problems in coordinating its different departments, and specifically research. After IBM's entry to the game, and while having installed about 700 computers, *Remington Rand* merged with *Sperry*, and then was acquired by the latter to form *Sperry Rand*.[56]

Sperry was an enterprise that had been very active in gyroscopes and radar technologies since the 1910s, and thus had an extensive experience with tube-based electronic devices.

[51]Ceruzzi (2003, p. 34)

[52]Ceruzzi (2003, p. 35)

[53]Another computer language was textitCOBOL developed by the American Ministry of Defence as a standard for business computing), providing a design space to customers. Enterprises could thus build their own programs, using the language delivered along with the computer by the provider. There is an interesting parameter on the etymology of the words "software" and "language". Ceruzzi mentions:

> *The word "software" [compared to the word "language"] has less to do with its physical form than with its changeability and intimate connection with the system's purpose, supplied by the computer user (Ceruzzi, 1989, p. 198).*

On the other hand, *COBOL - Common Business Oriented Language* was imposed as a standard by the U.S. government in order to allow managers to program. The debate of "high-level" languages, trying to immitate human language and, thus, reduce the level of expertise needed to access the development process always remains critical.

[54]Ceruzzi (2003, p. 35)

[55]Ceruzzi (2003, pp. 67-68)

[56]Gray and Smith (2004).

Through various re-organisations, *Sperry Rand* didn't manage to follow the way to success, mainly for two major misses:

- *Reference architecture*: The successors of UNIVAC, *Univac File Computer (UFC)* line, broke from the *von Neumann* architecture[57] of a 'universal' computer. At the same time, *IBM* developed the *FORTAN* language for its *701*.

- *Expertise*: given that electronic tubes had been a mature technology, already mastered by *Sperry*, moving away from it and following the semi-conductors revolution was rather difficult.

These two mistakes had been, as we will see, the most common for the industries of the time. *Sperry Rand* eventually managed to survive thanks to a late entry into transistorised computers, that assured the company's transition to the new era, although it had great difficulties keeping up with production, research and support for the previous generation.

General Electric: not entering the market of commercial computers.

General Electric produced a commercial computer in the late 1950's, called *ERMA (Electronic Recording Machine Accounting)*, a system developed along with the Stanford Research Institute and the Bank of America. *ERMA* computer was one of the first to be "transistorised", that is, to use the semi-conductor technology developed in late 1940s by *Bell Labs* and commercialised during the 1950s.

However, GE's management was not interested in entering the commercial computer market, as they decided to concentrate on projects such as jet engines and nuclear power[58]. In fact, at the time the computer market was hardly profitable and existed only as a complementary product to equipment businesses.

AT& T (Western Electric): opening the space for a new market.

Western Electric was the manufacturing arm of *AT&T*. The company was however a regulated monopoly and excluded from the computer market. As Ceruzzi (2003) notes, the company had enough business installing telephones in the booming postwar suburbs. After seven years of litigation, the company settled a lawsuit brought by the U.S. Justice department in 1956 and applied from 1958 and thereafter, according to which *AT&T* was prohibited:

> *from engaging in any business other than provision of common carrier communication services; prohibited Western Electrics from manufacturing types of equipment other than those sold to AT& T for use in furnishing common carrier communication services; and required licensing of certain AT& T patents (Enis and Sullivan, 1985, p. 127).*

From this moment on, there were two parallel processes: on the one hand, *AT&T* was forced to diffuse the knowledge about semi-conductors.For this, they organised a conference between the licensees to share their knowledge. In this context, the research results were also

[57] *UFC* didn't support stored programs, thus had similar problems with the ENIAC in programming Gray and Smith (2004).

[58] Ibid.

diffused. As Gordon Moore, the founder of *Intel* as well as the "Law Moore" on semiconductors recalled, this was the starting point for *Fairchild Semiconductors* manufacturing rationalisation process. Characteristically, in his paper for Bell Labs Journal, Morris Tanenbaum, the inventor of the first silicon transistor, along with his colleague, Thomas,(Tanenbaum and Thomas, 1956) described the transistor fabrication processes, its electrical characteristics, the properties of its structure and the rules of calculation of its design parameters.

On the other hand, *Bell Labs* continued producing computers, mainly for their telecommunication network, beyond some military projects. Through this internal network, some years later, the UNIX system was to emerge, which was to mark the beginning of the history of the internet.

***RCA*: a media actor entering the computer industry.**

RCA (Radio Corporation of America) was an enterprise created in the aftermath of WWI, in the framework of the competition between U.S. and English businesses.It had been a subsidiary enterprise of *AT& T* that during the 1920s was delegated radio stations, while the latter obtained the control of telecommunication networks. GE and Westinghouse were also shareholders of this enterprise, while the American administration was represented in the firm's council (Flichy, 1991). In fact, *RCA* was created after WWI as a patent-pooling enterprise, to oblige the different actors to engage in a cross-licensing strategy, in order for the national telecommunications to evolve, suspending the standards war between the two enterprises (for the latter see Shapiro and Varian (1999a)).

Hence, *RCA* continued as an autonomous enterprise, active in radio and soon, the television network, becoming one of the major electronic tubes manufacturers. That had placed it in an important position concerning the use of vacuum tubes, elementary components of electronic computers, before the dominance of silicon transistors.

RCA entered the computer market in the early 1950s, focussing on core memory. While innovative, with the first commercial computer being *BIZMAC*, the new architectural characteristics didn't manage to compete with others as textitspeed was already a well-established performance criterion, and *BIZMAC* was slower.

The following RCA computers (RCA 301 and RCA 501) also followed the von Neumann architecture.

National Cash Register: computer assembling

National Cash Register (NCR) had been a company specialising in the manufacturing of cash registers since the end of the 19th century, having a dominant place in the market during the 1920s. This case has been studies as an example illustrating the need for enterprise "dynamic capabilities" by Rosenbloom (2000), as a company that managed to "survive the waves of change".

During the 1950s, its activity included three product lines: register, accounting and adding machines[59].

It was in the niche of accounting machines that *NCR* felt the pressure of the computer's introduction. In 1952, it began its entry into this market by acquiring *Computer Research Corporation (CRC)*[60], a company that had been founded by former *Northrop* employees.

[59] Mueller (1953).

[60] Expanding national cash acquires electronic computing machinery plant. (1952). Journal of Accountancy (Pre-1986), 94(000006), p. 739.

This company was integrated into the pre-existing "Product Development Department" of *NCR*, having as a function the definition of questions for the Research and Development department (Rosenbloom, 2000).

A major strategical move was the introduction of the *Class 31* "electromechanical accounting machine", described at that time as:

> *a combination of an electric typewriter and accounting machine (Mueller, 1953).*

This machine, not only would operate as an accounting machine but would be the interface for the different computers to be launched at the suite.

In 1954 it sold its first computer, based on the work of *CRC*, of which the staff had grown to 360 employees (85 in engineering, 186 in manufacturing), a relatively small number, if we consider that the company had 45 000 employees worldwide in manufacturing at that time (Rosenbloom, 2000).

However, NCR was the only company of which the computers were assembled by others Krickx (1995). Having participated in the *ERMA* project along with *GE*, an agreement emerged according to which:

> *GE would build the central processor and the memory for both companies and would buy high-speed card-readers, printers and other peripherals produced by NCR (Rosenbloom, 2000, p. 1090).*

This agreement was based on the previous market context and, thus, division of labour, according to which *GE* was an expert in electronics (tubes) while *NCR* was an expert in printers (ink), an agreement that frustrated *CRC*'s founder, now working for *NCR*, who felt it "taught *GE* everything it knew about computer design" (Rosenbloom, 2000, p. 1090).

While at the time computers represented a small fraction of the revenue of *NCR* (about 2%), the fact that it didn't manufacture computers itself would render its manufacturing capacities, at one time its strongest advantage, obsolete. The firm, after many re-organisations and market pressures, managed to transform its manufacturing capacity for the electronic computer production era in the late 1970s.

Honeywell and Raytheon: the "multi-purpose" enterprise

Raytheon was one of the largest electronic tube manufacturers back in the 1930s. Raytheon was engaged in the design process within the project "Hurricane", in 1950, the object of which was to track and control ballistic missiles[61], though was characterised by an openness on exploring and exploting new concepts[62].

Regarding the computer Raytheon had to deliver for the project "Hurricane" (the *RAYDAC*, for *Raytheon Digital Automatic Computer*), which followed *ENIAC*'s design, had however advanced features in data storage and error checking. Delivered in 1952, it was too slow for real-time missile guidance and tracking, and was thus considered a failure[63].The project however provided the functional requirements for the next one (leading to *SAGE*, discussed

[61] Ceruzzi (1989); Valley Jr (1985).

[62] Typically, during the research in micro-waves, that had already started already in 1945, a *Raytheon* employee, Percy L. Spencer, by stepping close to the micro-wave device (the *magnetron tube*), observed that the chocolate he had in his pocket melted. That was the beginning of the micro-wave oven concept, that *Raytheon* got to the market, supporting Spencer's accidental experience and idea (Singer and Piluso, 2010)

[63] Ceruzzi (1989, p. 58)

in the next section), however *Raytheon*'s computer division was acquired by Honeywell a couple of years later[64]. *Raytheon* itself continued in the missile technology market.

Honeywell was event before WWII an industry that managed to combine different market lineages on the basis of its competencies. While heavily engaged in military projects, it was also active in markets such as home equipment[65].

When, in 1957, it offered a computer to the market (the *Datamatic 1000*), its design was already obsolete. It used vacuum tubes, whereas it had to be clear that transistors could be used for commercial computers[66].

Honeywell would return to the commercial computer market after acquiring *Computer Controls Corporation (3C)*, in 1966. *3C* had be founded by *RAYDAC*'s designers, among whom Louis Fein[67] was the one who coined the term "computer science" (Fein, 1959, 1961). The importance of *3C*'s computers did not lie in their market position, but in the fact that they were later chosen for the *ARPA* project, because of their design.

Burroughs Corporation

Burroughs had started as a calculator company in the beginnings of the 20th century, and diversified its offerings to business equipment (mainly finance) until WWII[68], while it had knowledge on the design of the *ENIAC* memory[69]. The enterprise's activity was mainly axed towards military projects[70].

The machine it offered was the *E-101*, introduced into the market in 1954[71] at a low price, for about $35 000. However, its design did not follow the von Neumann architecture as it lacked a stored-program part[72]. Moreover, like other computers of the time, it applied a specific memory technology (*the magnetic drum*)[73], that hampered its speed. This "jump" towards a different kind of memory was not by chance: *magnetic drum* memory presented the virtue of enabling an "online" computer operation, that is the possibility to manipulate data in "real time", while the program was under execution. Practically, this design parameter could be used to address the requirements of "dynamic" ticket booking. As it was proven by the facts, *Burroughs* did not manage to advance this technology, while the market demand was already there. Hence, it was a typical case of a "jump", as opposed to the "disruption", where technology exists but market demand does not.

However, *Burroughs* remained in the computer industry, managing to develop its expertise through military contracts, mainly the *SAGE* and the *Atlas intercontinental ballistic* programs[74]. *Burroughs* also introduced an innovative design of a transistorised computer

[64] Ceruzzi (2003, p. 54).

[65] For instance, the initial core competencies of the enterprise being .in eating and industrial-process control equipment, the enterprise was active both in military instruments and home thermostats (Kita, 2009, p. 77).

[66] Ceruzzi, op.cit.

[67] National Museum of American History (U.S.). (1969). Computer Oral History Collection.

[68] Gray and Smith (2003).

[69] According to Ceruzzi *Burroughs Corporation* had designed the memory for *ENIAC* Ceruzzi (2003, p. 50). However, Gray and Smith (2003) are more analytical on this aspect: *Burroughs* had hired Moore's School Electrical Engineer Professor Irven Travis, who was implied in the *ENIAC* project, and with whom Eckert and Mauchly had argued on the patent issue.

[70] Gray and Smith (2003).

[71] Koss (2003).

[72] Caillaud and Jullien (2003, p. 67)

[73] Koss (2003, p. 43).

[74] Ceruzzi (2003, p. 67).

quite late, in 1963[75].

IBM: the beginnings of the dominance.

One of the major computer projects during the 1950s was the *SAGE (Semi-Automatic Ground Environment)* project. The *SAGE* computer was the "nerve centre for a complex system of radar installations, aircraft, ships and command centers"[76]. The first computer for the system was delivered in 1956[77], while the system became fully operational in 1963[78]. It remained functional until 1983, while its overall cost was $8 billion, of which half a billion revenue was generated for *IBM* in the 1950s[79].

Its design followed the one of *Whirlwind*, a descendant of *ENIAC* developed by MIT *Lincoln Lab*, which, given the experience of *RAYDAC* had a very fast memory. Without going into details, we should note that this computer was the only one to achieve a fast magnetic drum memory.

The property of this technology was to enable an interactive, command and control mechanism, as users (usually non-computer specialists of the military) could interact with it by a "touch screen" (input-output screen)[80]. Eventually, IBM's engineers replaced *drum memory* with a *disk drive*, which has continued until today as *RAM (Random Access Memory)* - the name signifying the direct ability to access any piece of data.

The breakthrough of *IBM* in the commercial marketwas with the *IBM 1401*, following the "universal computer" architecture, in 1959. At that time, semi-conductors technology was mature enough, and Fairchild Semi-conductors in the Silicon Valley had already been proven to be a reliable partner[81].

While the number of previous models installations (*UNIVAC*s and *701*s) was thirty to forty, eventually over ten thousand 1401s were installed[82].

At the time, IBM made available various computer models (IBM 650, IBM 1401, IBM 7070-7074, IBM 702-705-7080, IBM 701-704-709-7090, IBM 7030), incompatible with each other. By the initiatives of Donald Spaulding (corporate technical *staffer*) and T.V. Learson (vice president), a corporate - wide strategy committee was convened, under the leadership of John Haanstra, responsible for the 1401 product line[83]. Their report, in 1961, called for a New Product Line that would be a success for all existing product lines[84]. A corporate-wide effort then began, described by Baldwin and Clark (2000) as modularisation, that eventually marked the company's as well as the industry's future.

10.4.2 Discussion. Foggy competition: Producing and marketing while exploring its identity.

He now came to a road branching in four directions, and immediately he was reminded of those cross-roads where knights-errant used to stop

[75]Gray and Smith (2003).
[76]Ceruzzi (1989, p. 70).
[77]Ceruzzi (2003).
[78]Ceruzzi (1989, p. 71).
[79]Ceruzzi (1989, 2003).
[80]Ibid.
[81]Ceruzzi (2003); Lecuyer (2006).
[82]Ceruzzi (2003, p. 75).
[83]Brooks (2010, p. 316).
[84]Ibid.

to consider which road they should take. In imitation of them he halted for a while, and after having deeply considered it, he gave Rocinante his head, submitting his own will to that of his hack, who followed out his first intention, which was to make straight for his own stable.

Miguel de Cervantes, Don Quixote.

Like Don Quixote, enterprises are often found in crossroads leading to unknown directions, once finding themselves in an environment of foggy competition. Like Rocinante, Don Quixote's horse, they may choose to return to their stable, their field of expertise and specific market. However, in Cervantes' book, the choice to return home did not prove to be the safest, as Don Quixote met one of his most painful experiences on the way home.

Louis Fein, the man who coined the term "Computer Science", a consultant of *Stanford University* during the late 1950s and one of the first to found a Computer Science School, reported in his article on "The role of the University in Computers, Data Processing, and Related Fields" on the situation during that period:

> *Today, one can sometimes scarcely identify teachers, researchers, designers, and development people as members of a university, industry, or government, for each is involved to a considerable extent with all three. (Fein, 1959, p. 120).*

He summarized the period we studied in this chapter as follows:

> *...the government has set up RAND Corporation(s) for research and project work; it has set up and supported institutes at universities for research and project work; it has indirectly supported development, if not exploratory work, it these fields as by-products of government contracts to industry.*
>
> *Industry and business have set up separate departments charged with overall responsibility of applying these techniques to company operation, management, design, manufacturing, sales, etc. Graduate-level schools for instruction of professionals in these fields are being run by industry itself.*
>
> *The scholars and practitioners in these new fields are uncertain both as to the nature and structure of the fields and their relation to each other. As would be expected, new societies and magazines devoted to these fields have sprung up.*

He then stressed in his article the importance of IBM education programs for Universities, as well as the need to form autonomous computer science departments.

We could describe the period of Foggy Competition as a dialogue between enterprises and society. Enterprises propose different versions of the new, while society declares its preferences. The case of what ended up as being RAM is very characteristic: a series of enterprises, such as airlines, desired a feature that would provide "interactive data-bases". That is, they wanted to be able to modify information "online" (while the computer operated). This case was not contemplated by von Neumann, as none of the computer uses until then had such a requirement. For instance, missile technologies, that constituted the major innovation trajectory for the American DoD during the 1950s were content with traditional algorithmic operations: first the trajectory was programmed and then it was executed. After missile launch, no modification of the program was either necessary or desired.

	Business Rationale			
Challenges	"Misunderstood" Rationalisation	Enterprise expertise evaluation	Research, Experimentation	Alliances
Design oriented				
Can an alternative architecture be followed?	"problem-facing" instead of innovation	Acquisitions of pioneers, design potential exploration.	Appearance of new uses and designs.	Division of exploration.
Can expansion be rationalised?	Client support tensions	Toolkits rationalisation.	Research collaborations.	Interface agreements.
What is the state of the art?	Segmented expertise	Remain in the core of one's expertise or change.	New theoretical formulations.	Common knowledge base establishment.
Market oriented				
What to offer?	New product, old uses.	Strategic dilemmas.	Product variety.	Integration or delegation.
How to scale production?	Pre-existing methods application	Resources attribution issues.	Unknown value of the potential.	
How big is the market?	Volume uncertainty	Tensions on priorities.	Market monitoring.	
				Business environment transformations

Phase 3: Foggy competition

Table 10.4: Phase 3: Foggy competition and business environment transformations.

However, this way of processing did not suit the requirements of ticket booking, as one has to consult available places (retrieve data) and reserve places (modify the data base) in a "random" way.

For this reason, as we've seen, many enterprises tried to rationalise 'drum memory' production and liability, as it corresponded to such a requirement. Finally, IBM - having experienced one of the rare military computers that also had a similar requirement, the *SAGE* - arrived at the RAM invention that satisfied this requirement. RAM became thus an integral part of the design that would dominate the market.

Similarly, while IBM had a great variety of products, after the surprising success of its 1401 model, it took the risk of re-organising its entire production facility on the basis of a plan, of which the head was responsible for the 1401 computer.

10.5 Industrial rationalisation: IBM System 360 and "the power of modularity":.

The next phase in the history of computer business was introduced by *IBM* and its *System 360* modular family of computers. By doing so, *IBM* not only replaced all previous product lines with a 'universal' one, but it also set the standards for the entire industry. This case is very well known in contemporary management studies, as it constituted the major case for Baldwin and Clark's theory of modularity. Therefore, our study on this case will be based on their work, 'Design Rules, The Power of Modularity' (Baldwin and Clark, 2000).

Baldwin and Clark also place *System 360* rationalisation in an historical context where computer knowledge had already been very advanced:

> *By 1962, knowledge about computers had grown to the point where not only could a list of design rules be constructed but designers could argue for and against different choices based on experience with real machines (Baldwin and Clark, 2000, p. 183).*

Consequently, we can say that innovation on computers was not the primarily aim of the project. The objectives of the new design addressed requirements of production rationalisation, under the imperative of product line unification.

10.5.1 The effects of modularisation.

This product line signified a major transformation of *IBM* itself, which managed at the same time to set the standards of price/performance for the entire computer industry[85].

Baldwin and Clark describe the elements coming from the *System 360* that assured *IBM*'s dominance in the market for the years to come:

- *A unified corporation and strategy, providing credibility and continuity at an internal and external level.*
- *A compatible family of computers, rationalising production of different models and giving customers the opportunity to "customise" their choice, excluding or adding components from IBM's list.*

[85] Fisher et al. (1983, chap. 5).

- *A strong technical-commercial department, enduring customer relationships and the possession of a deep knowledge of customer's systems, assuring a "technological follow up" by frequent system upgrades.*
- *A strict property model: computers could be rented, not bought. Interfaces were proprietary (competitors could not build complements unless licensed by IBM).*

Baldwin and Clark (2000).

As we've already mentioned, the last point, leasing, was the "by default" business model for *IBM*[86] even before the computer and despite anti-trust regulations in 1954 that forced the enterprise to sell its machines as well. In addition, the strong technical-commercial department was also one of the historical advantages of the enterprise, too.

The novelty that *System 360* achieved however was in the rationalisation of production (according to a price/performance matrix) as well as the compatibility between different models. On the one hand, compatibility allowed customers to enter computing and then upgrade their system, keeping the same software. On the other hand, it gave the possibility for more advanced computers to be used for programming software for less advanced ones (*"downwards" compatibility*)[87].

Designing different components that were reusable in other systems (*compatibility*) and interchangeable in case of defectiveness, gave *IBM* the possibility to make important marginal benefits, during what Utterback and Abernathy (1975) would describe as a cost minimising phase. Moreover, on the organisational level, modularisation provided *IBM* with a higher capacity to organise intellectual and manual division of labour along the lines of the modularised object, augmenting the expertise on each specific field along the lines of the overall computer architecture (Baldwin and Clark, 2000), something that has been observed in other industries as well (Anderson and Tushman, 1990).

Baldwin and Clark compare business in computing before and after modularisation, where "design options" appeared, causing "the market value of design options to go up by a lot" (Baldwin and Clark, 2000, p. 237).

However, in the framework of our research, we are interested in answering the following question: How did this transition occurr? The answer is significant in two ways. Firstly, regarding our genealogy, this case can help induce the general modes of transition between a period of competition through innovation with no clear market characteristics and the period of a dominant design establishment. Secondly, the "design of the modularisation process", or "the design of the design", as one of *IBM*'s chief designers of the time put it in his recent book (Brooks, 2010), can help us understand the modularity design of contemporary Web platforms.

Hence, we are going to revisit the case of the SPREAD (Systems Programming, Research, Engineering and Development) Committee, which was attributed to design the new design rules (Baldwin and Clark, 2000, pp. 173-194).

[86] Fisher et al. (1983, p. 105).

[87] This ability, also called *emulation* was to become a standard method in computing having various implications, one of which has been the use of mainframe computers to develop software for the PC, by emerging companies like *Microsoft*, in the early era of the PC (see Chapter 11). In addition, many concepts, such as the "control program", the heart of the operating system (Baldwin and Clark, 2000, p. 191), were to be transposed in the early PC platform (see Section 11.2.1), playing a decisive role in its emergence.

10.5.2 Designing the design: the team to rationalise the industry

During the 1960s, the computer industry was to shift to another dominant design. On the one hand, *IBM* advanced its production line, by modularising computer architecture and generating the *System/360* family of computers Baldwin and Clark (2000). On the other hand, the advances in the semi-conductors led to the emergence of Silicon Valley and the beginning of the information revolution.

IBM's vice president, Vincent Learson, established a group of thirteen people drawn from the company's research, development and marketing units in 1961, forming the *SPREAD* committee (Baldwin and Clark, 2000, p. 173)[88]. This committee was conducted informally, as keeping projects secret within the enterprise was one of the organisational methods followed at the time by *IBM*[89]. However, what is surprising is the fact that we have the establishment of an intimate environment among the designers and developers of the project. One of the means of this, were design competitions[90]. This type of "rule breaking process" was oriented towards performance optimisation, a criterion to which both the group's composition and schedule was subject[91].

Regarding the design method itself, the group used the systematic design approach. While the intermediary product method that emerged from the practice of Mauchly and Eckert helped them organise the design process in their long entrepreneurial journey, a systematic design method (Le Masson et al., 2006) provided *IBM* with its development processes framework since its early involvement in computer manufacturing. Moreover, even in the case of pre-war tabulating machines, production was organised by *IBM* in a modular fashion (Baldwin and Clark, 2000, p. 162).

Once *IBM* entered the computer market along the lines of von Neumann computer architecture, it started standardising its different components. Yet, after the semi-conductors

---

[88]The decision to establish this committee came about after a relative failure in the production of a "supercomputer" or a "state-of-the-art" computer, the *STRECH*. It was the most powerful computer commercially available at the time and helped *IBM* further explore the limits of computing. However, despite major innovations that were developed and embodied in the computer, Tom Watson, Jr., *IBM* president, announced that the computer would be sold at half the price projected as it didn't meet performance criteria. The announcement was made in the Western Joint Computer Reference, Fisher et al. (1983, p. 49).This conference, was jointly organised by the Association for Computing Machinery, the American Institute of Electrical Engineers, and the Institute of Radio Engineers since 1951.

[89]Wise (1966), Pugh (1995, p. 47).

[90]Brooks recalls in his book a moment where a specific design, on which the group worked for months, didn't meet the cost requirement. The design responsible, Amdahl, set up a competition:

> *I reckon the design competition, originally suggested by Gene Amdahl, to have been immensely invigorating and fruitful. It put everyone hard to work again after a demoralizing cost estimate. It got each person deeply involved in all aspects of the design, which greatly helped morale and proved valuable in the later design development. It produced a consensus on many design decisions. And it produced a good design (Brooks, 2010, p. 76). (...)*
>
> *Amdahl's proposal for a design competition when our first design ran aground proved very fruitful. It produced great concurrence on many issues, and it quickly spotlighted the crucial differences (Brooks, 2010, p. 328).*

[91]In fact, *IBM* management closely watched the project and did not hesitate to change the SPREAD responsible, Haanstra and send the entire team to a hotel, under the new direction of Evans, to boost the process Wise (1966). The major reason for that is reported to be the fact that Haanstra continued supporting research on the 1401 computer, while the System 360 was to replace every previous computer line (Pugh, 1995). However, as Brooks notes, the two years' time provided was a comfortable time frame for the group to experiment and prepare its report (Brooks, 2010).

industrial revolution, the field was set for mass computer production[92]. Beginning from a 'side project', *IBM* redesigned the entire production line, transforming at the same time the computer's architecture, its own organisation and the market for enterprise computers.

The operation of industrial re-organisation for IBM to correspond to the demands of productivity is described as a process of *modularisation*. The aim of this operation is to *split* the design of reference to small tasks by establishing some common *design rules* (Baldwin and Clark, 2000)[93].

10.6 Cycle break: DEC's network computer and the birth of hackers.

DEC was created in 1957, a few years before the rationalisation of computers by IBM[94]. It was founded by two MIT students, having worked on a computer named *TX-0*[95]. This computer was one of the first to use transistors and was used as an auxiliary instrument to test a bigger one. Because of its auxiliary role, its experimental nature, as well as the mentality of *MIT Lincoln Lab* researchers, *TX-O* operation did not follow the rules already established in the commercial computer market: students could interact with the computer with no mediation of what became known as the 'clerks' (technicians charged with the responsibility of the control of the input and output of computers)[96]: students could simply sit in front of a computer and program it (instead of giving the clerks the program on a punch card and waiting for them to return the results). Hence, they had the possibility to 'play' with the computer during the nights, when it was available and be creative with it. Programs such as games or songs were created in this context, away from the typical trajectories of the time[97].

DEC integrated this culture in its operational mode and built an original business model out of it. As Fisher et al. (1983) note. *DEC* managed to be one of the principal competitors

[92]The emergence of the semi-conductor industry, after which Santa Clara valley was named Silicon Valley, was a critical step in the genealogy of the computer industry. The enterprise that evaluated from a start-up to the major actor of the industry, *Fairchild Semiconductors*, managed not only to take advantage of the economical, cultural, technological and scientific traditions of the region, but also to conceive and project the rules of the industry ever since. Gordon Moore, one of the founders of *Fairchild Semiconductors* and, later, *Intel*, invented - amongst others - the *Moore's Law*, during the early production rationalisation process. The entire semiconductor business and the corresponding ecosystems was developed since according to this *Law*, promising that the performance of a chip, measured by the number of its transistors and its speed, doubles approximately every two years. See Freiberger and Swaine (1999); Ceruzzi (2003); Lecuyer (2006); Weil (2010).

[93]We are not going to enter here the details of the IBM S/360 family development. For a detailed study, see Baldwin and Clark (2000, pp. 169-217).

[94]For the rationalisation of computer manufacturing see section 10.5.

[95]Ceruzzi (2003, p. 243).

[96]During the 1960s a second kind of "computer as a service" emerged, this time in the interior of large organisations (universities or enterprises). This was the case of the "priesthood of technicians" Ceruzzi (2003) or "clerks" Levy (2010). Here is how Ceruzzi describes the operation of this internal service:

> *A typical transaction began by submitting a deck of cards to an operator through a window (to preserve the climate control of the computer room). Sometime later, the user went to a place where the printer output was delivered and retrieved the chunk of fan-fold paper that contained the results of his or her job.*

[97]Levy (1984, pp. 26-32).

of *IBM* in a very specific market:

> *DEC marketed most of its machines announced in the 1960's to "experienced" and "moderately experienced" users.*[98]

Since it did not use an "attendant staff of operators", the *PDP* (*DEC*'s *minicomputer*) was about 20 percent cheaper than the equivalent IBM's *360/50*. Moreover, DEC introduced a decentralized way of using computers: instead of using a mainframe, one could use many different computers to perform the same task, at a much lower cost[99].

Regarding the value exploitation method, *DEC* used to sell rather than lease its products. Moreover, not only did it allowe, but it encouraged its customers to modify and extend their machines[100]. To this end, it published detailed descriptions of the inner workings of its products and distributed them widely[101], unlike all other business practices of the time. Founded in 1957 with an initial capital of $70 000 (provided by a hedge-fund firm), it had accumulated over $3 million in retained earnings by the end of its 1964 fiscal year[102]. While *IBM* imposed quasi-universal price/performance criteria in the computer industry through its *System 360* computer family, *DEC* inaugurated a radically different *modus operandi*. Ceruzzi notes about *DEC*'s first product, the *PDP-1*:

> *Clearly the economics of mainframe computer usage, as practised not only at commercial installations but also at MIT's own mainframe facility, did not apply to the PDP-1*[103].

DEC's products were described by the enterprise's management as not being computers. Fisher et al. (1983) mention characteristically that people involved in the electronic data processing "*could not believe that in 1960 computers that could do the jobs could be built for less than $ million*" (Fisher et al., 1983, p. 273). Thus, *DEC*'s products were marketed as *"Programmed Data Processors (PDP)"*, later called *minicomputers*.

While *IBM*'s rationalisation largely consisted in establishing standard interfaces among well-known components (even though that task necessitated the invention of new ones), in *DEC*'s case, the components to be added by the users were unknown by definition, they were yet-to-be-conceived. Hence, inducing interfaces from the components was impossible. Therefore, *DEC* introduced a 'skeleton', a device were future components were to be 'plugged-in'. What was standardized in this case was the output of this device, that could be used as a design parameter (DP) for the conception of the new components. This device, called the bus, incorporated all the necessary knowledge for experienced users to further expand the computer.

The first client of *DEC* was a consulting company that was to play a crucial role later, regarding the Internet, *Bolt Beranek and Newman (BBN)*. *DEC* products embodied and fostered the hacker culture. Those working on a PDP would have to discover themselves how the minicomputer worked, write their own programs and invent their own uses. And as they were frequently researchers, like in the *MIT Lincoln Lab*, they would share their knowledge on the computers as well as their programs.

[98] Fisher et al. (1983, p. 278).

[99] Ibid.

[100] Ceruzzi (2003, p. 142).

[101] Ibid, p. 143.

[102] Fisher et al. (1983, p. 271-272).

[103] Ceruzzi (2003, p. 143).

10.7 Conclusion

The computer has been a disruption (Christensen, 1997) for the business machinery industry. Still, identifying a market was not the only dilemma for innovators. Until a rationalisation was proposed by IBM, two decades after *ENIAC*, computers transformed business as they themselves were being transformed.

For its developers, Mauchly and Eckert, *ENIAC* was proven to be an early materialisation, allowing them to continue the effort of constructing a "universal computer". Still, once UNIVAC got in the market, innovation diffusion was not just a matter of adoption, as Rogers proposes, nor a question of user-manufacturer sticky information division.

What followed the market emergence was neither a simple adoption (Rogers, 2003) nor a clear division between user and developer "sticky information" (von Hippel, 2005). Technological innovations went hand in hand with computer penetration in different use settings. During the *Foggy Competition* stage, when no-one fully understood what computers could do or what they could be, exploitation and exploration were two parallel processes, nourishing while threatening each other.

IBM's System 360 synthesised a long period of exploration by proposing a rationalisation for computer production and marketing that set the standards for the industry and established a dominant design (Utterback and Abernathy, 1975). Still, *DEC*, still a young enterprise, managed to survive through "turning the clock back" and opening a new cycle, by creating a "mini-computer" for user-developers, exploiting the new technologies, theories and relations (semiconductor, cybernetics, ARPAnet community).

As for the enterprise computer industry itself, as Baldwin and Woodard notes, *"the computer industry became more vertically disintegrated"* (Baldwin and Woodard, 2010, p. 34), with enterprises specialising in different components of the dominant architecture.

In the following chapters the model induced for the role of user-developer-entrepreneurs will be tested in other industrial settings, namely the personal computer and the radio industries, attributing more emphasis to the early phases, where UDEs are found to play a more important role.

Chapter 11

The invention of the Personal Computer

Contents

11.1 Introduction

The Personal Computer revolution, as it is often characterised, was based on a wide exploration period during the *Early Materialisation* phase of the 1970s. Industrial rationalisation came about during the early 1990s with the *Wintel* model, when *Microsoft* and *Intel* combined allowing the sharing of their different enterprises' scope(Gawer and Cusumano, 2002). Le Masson et al. (2010) note that computers - unlike watches - have undergone significant identity changes over time. As this chapter will show, the *Wintel* rationalisation managed to propose a way to integrate not only the exploration experience of the Foggy Competition phase, but a method for permanent identity exploration, as well. This rationalisation can be described as an "open product" (Chrysos et al., 2010) strategy: on the one hand, task specificity is addressed by applications. On the other hand, user and developer interfaces assure that a wide variety of applications can be developed, exploiting the resources of the entire computer, as produced by the coordination of industry-wide forces. The PC cycle was disrupted (Christensen, 1997) by the Web browser application, questioning the boundaries of the PC, as earlier defined by the Wintel model. Though *Microsoft* eventually won the "browser war" (Cusumano and Yoffie, 1998), the new cycle of Web-based innovation continues.

The current chapter studies the *"conditions of specificity deployment possibility"* (Lefebvre, 2005) of the personal computer industry as we eventually have come to know it. The separation between "technology" and "use context sticky information" (von Hippel, 1990, 1994, 2005) came about as solution to the wide diversity of use contexts explored in early phases, that are in general included in the "personal computer" concept. Figure 11.1 provides a synoptic schema of those trajectories.

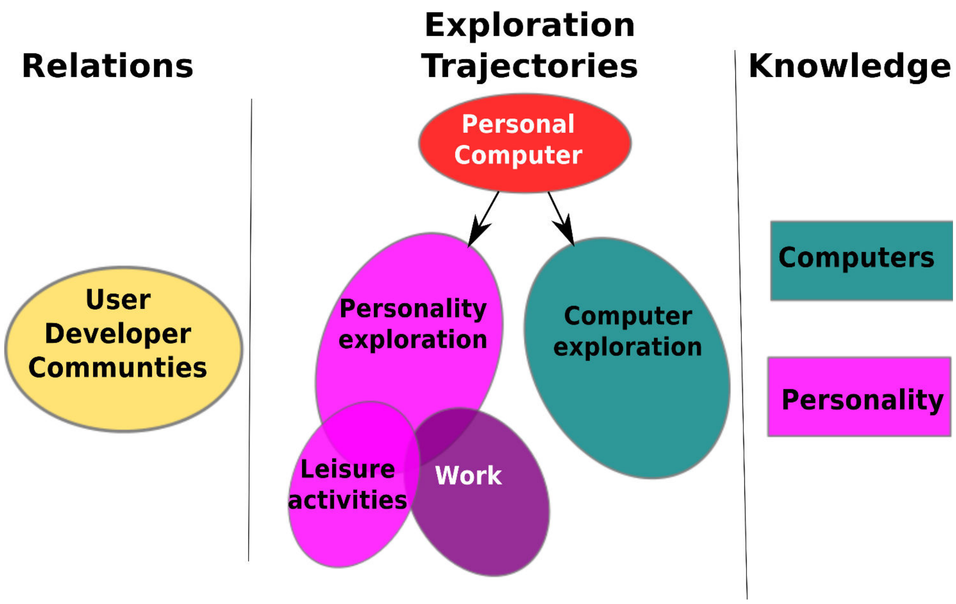


Figure 11.1: A synoptic schema of the Personal Computer exploration trajectories.

The study of personal computer Early Materialisation phase has attracted the interest of researchers from many different fields. The most well-known narration of this phase is edited by Freiberger and Swaine (2000) in their book *Fire in the Valley*, which includes details on the pioneers' private life experimentations. The specific life mode of early user-developer communities is also highlighted by social science scholars, arguing that the role of the cultural atmosphere of the West Coast during the late 1960s and early 1970s was a critical source of inspiration for those pioneers (Turner, 2006; Cardon, 2010, 2012). In fact, the exploration of the personal computer concept required the exploration of personality, as well. The fact that no communities of similar importance were deployed in Texas, home for a flourishing semiconductor industry and of "rocket scientists", where the *Altair* computer was "manufactured" (see paragraph 11.2.5), supports this argument.

However, what ended up being the dominant design of the PC had little to do with the identities projected by those early communities, as Ceruzzi (2003) notes. In fact, the work trajectory, as opposed to the home computer trajectory, was proven to be the one that could economically support the emergence of a world industry. Still, PCs for enterprises were not identical to enterprise computers, as the "microcomputer" concept suggested. Instead of beginning from administration or service processes, the work trajectory remained personal, having as its point of reference the metaphor of the employee's "desktop". Apple also showed that a specific market satisfying the requirements of both leisure and work, for what are also called "creative workers", such as graphic designers, existed. Hence, to the extent that computers became products, those products primarily served office related tasks.

At the same time, "desktop" computers could also be personal. That not only owed to the cost reduction which followed the Moore's Law for semiconductors. It was also based on the possibility of the same computers to be extended in a variety of ways, constituting an innovation starting point for developer-entrepreneurs and subsequently for enterprises. Hence, computer technology itself was transformed by exploration of the personal computer and many of these technological "disruptions" or "jumps" are to be found in the early materialisation and the foggy competitions phases.

This chapter will thus retrace the conditions of possibility (Lefebvre, 2005) of the personal computer industry specificities and the specific role of UDEs, focussing on the concepts that were proven valuable for its rationalisation and using the model developed in the previous chapter. Figure 11.2 shows a schematic outline of the PC industry's development, in terms of trajectories explored and the related knowledge and developer relations for the different phases.

Unlike the enterprise computer rationalisation, where the *IBM System 360* computer series covered the greatest part of the known market, the *Wintel* model rationalisation synthesised industry-wide design rules (Baldwin and Clark, 2000), yet the different market segments were to be covered by multiple enterprises, based on this model. Those enterprises had emerged during the earlier phases of the computer industry. This model of enterprise coordination originates from the early materialisation phase, when *Do It Yourself (D.I.Y)* computers, and most importantly *Altair* computer developed by the *MITS* start-up, had previewed hardware interfaces enabling user-developers to create their own, personal computer. *MITS* controlled this interface, although they taught users how to use it as a basis for their own computer creation. *Microsoft* emerged as a start-up to provide those users with a software development language, opening the era of software commercialisation.

While *MITS* failed to scale production as the demand augmented, the *IMSAI* computer, created by former *IBM* engineers, added a second interface, the *CP/M* operating system, enabling user-developers to create and use in parallel their own software. *IBM PC* would later be based on these *D.I.Y.* systems to define the architectural principles of its computer, providing the possibility for *Microsoft* to develop and diffuse its own operating system, *MS-DOS*, while keeping control of the hardware interface.

Foggy competition was marked by two "waves". Initially, *"IBM clones"* appeared by

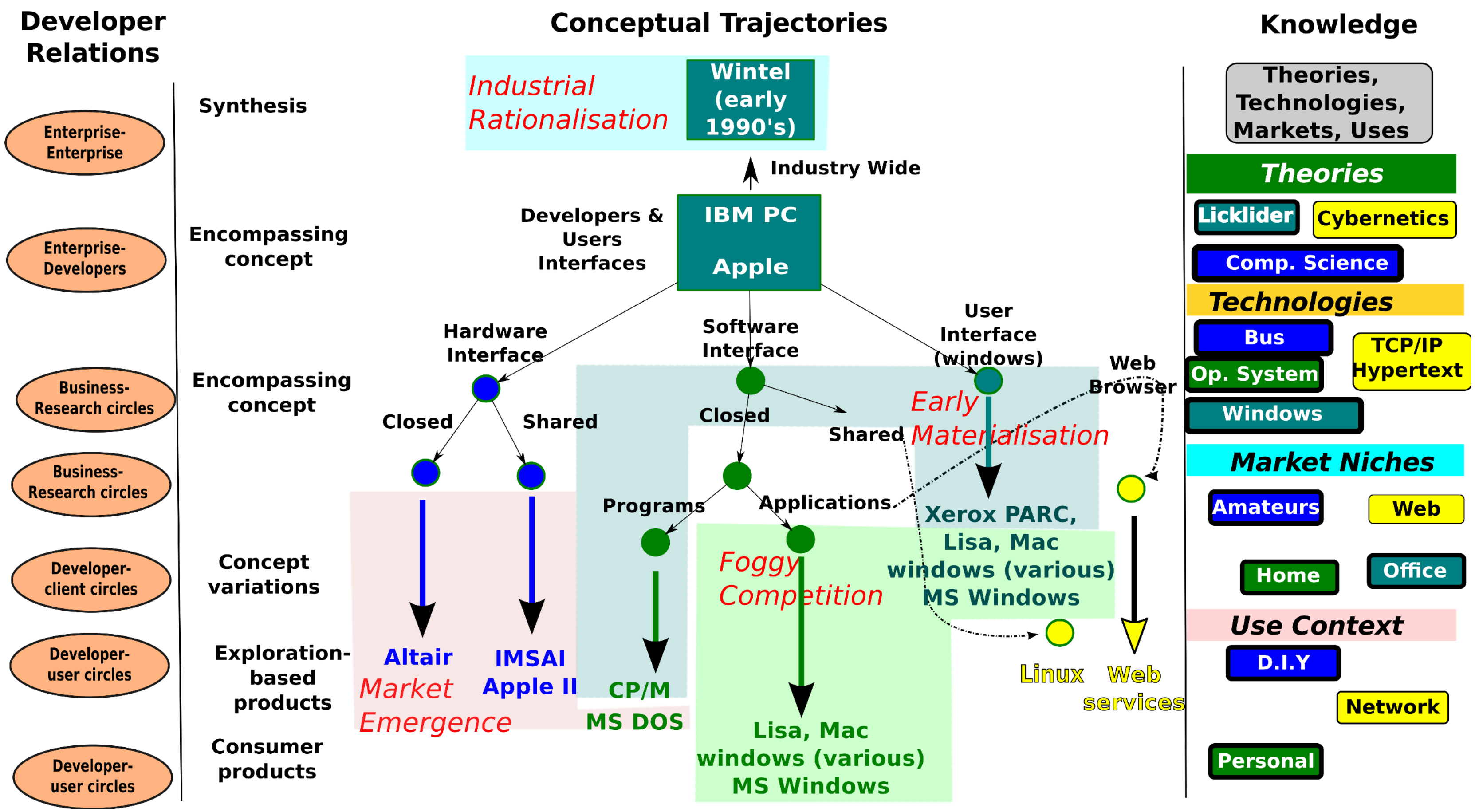


Figure 11.2: A schematic outline of the PC industry development.

"reverse engineering" the *IBM PC*. Those developer-entrepreneurs grew to be businesses that defined the competition in the late 1980s. Then, the commercialisation of "windows" graphical user interfaces (GUI - initially materialised during early 1970s in *Xerox PARC*) facilitated learning processes for end-users, making it possible for the PC to be sold to a non-developer public. From 1984, when the *Macintosh* windows interface appeared, a set of computer manufacturers used the same concepts for their computers. *Microsoft* coupled its windows GUI with its operating system, leading the passage from "software programs" to "applications". *MS Windows* and *MS DOS* interfaces, while closed, enabled the activity of third party developers thanks to the cultivation of a close relationship with software and applications developers. Much later, *Linux* communities would disrupt the PC industry through the provision of shared windows GUI and operating system interfaces (Raymond and Young, 2001; Lee and Cole, 2003; von Krogh et al., 2003a; Benkeltoum, 2008, and others).

During the late 1980s, the "gang of nine" (enterprises that had emerged by cloning the *IBM PC*) questioned again the control of *IBM* over its product, by reverse engineering their hardware interface. At the same time, they "set free" *Intel* and *Microsoft*, who would from then on have a direct collaboration with those enterprises on the basis of the common interface.

IBM's choice to opt for the enterprise market and focus on mainframes computers left the space for *Microsoft* to grow from the growing market of microcomputers. At the same time, applications such as *Office* and *Spreadsheet* gave computers an office character that contributed to their penetration in the business markets, while keeping the architecture open for different kinds of applications and markets. One of such applications, the *Web browser* would later disrupt the PC industry by opening up the way for the Web (Cusumano and Yoffie, 1998).

This reading suggests that a critical challenge for the PC industry has been the design of the means of computer expansion, allowing third parties to transform the computer through their action. More specifically, developers and end-users had been attributed a "design space" (Hatchuel, 2001; Baldwin and Clark, 2006) of the PC through particular innovations:

- Developers were able to "plug" (Gawer and Cusumano, 2002) their components into the device thanks to the *BUS* system, introduced in PCs during early materialisation phase by *Altair* (see paragraph 11.3). Later, developers were able to create "peripherals", thanks to the *USB* system (an "external BUS"), introduced by *Apple* during the Foggy Competition phase. In parallel they could develop and use software for personal computers thanks to innovations that allowed both the use of the same piece of software in different computers (thanks to the *BIOS*) and the use of different software programs on the same computer (thanks to the *Operating System*). Finally, they could develop applications, thanks to the rationalisation of the graphical user interface by the *windows* metaphor.

- End-users were able to transform their computer by choosing different software programs, peripherals, components and applications, when the market offer grew. As the industrial rationalisation advanced, end-users could choose only peripherals and applications, without having, for instance, to "create" their computer by selecting the components one by one.

11.1.1 Chapter Overview

Table 11.1 provides a synopsis of the current chapter, using the same categories as Table 10.1 of the Chapter 10.

Section 11.2 studies the *Early Materialisations* phase of the personal computer industry through user-developers experimentations. Paragraph 11.2.1 begins by the study of early "microcomputers" development, computers that tried to "imitate" the previous ones though in a very small scale, using the advancements in semiconductors. Thence, the very early microcomputer operating system (OS), *CP/M*, developed by Gary Kildall (paragraph 11.2.1) embodied the criteria of mainframe computers OS to the micro scale. Still, the commercialisation of *Intellec-8* by *Intel* in 1972 using it was proven too premature. In parallel, the *Micral* computer in France was an ad hoc microcomputer project developed for the INRA institute (paragraph 11.2.1) though was never seen as a personal object nor was it addressed to a user-developer community. *Alto workstation*, developed in *Xerox PARC* by former *ARPA* researchers (paragraph refsec:altoWS). While it was never commercialised, it has been the basis for a set of early materialisations, especially regarding graphical user interfaces, that would largely define business during the *Foggy Competition* phase in the 1980s. The Personal Computer concept was coined by UD-based press, as a computer to develop for one's own use (paragraph 11.2.3). Within this framework, the *Altair* D.I.Y. computer developed and marketed to user-developers by a start-up, would become an experimentation and exploration object for those communities (paragraphs 11.2.5, 11.2.6 and 11.2.6). Many of the concepts developed by this community would later be adopted by the *IBM PC*.

Section 11.3 studies the *Market Emergence* phase for the PC industry through user-developer-entrepreneurs incentives. *Apple I* and *Apple II* computers, of which the designer was part of such a PC user-developer community, would open the way to the PC commercialisation, illustrating that there was a market for such an object. Still, the first buyers of those computers were still the UDs of the beginning. *Apple* collaborated with developers to further extend its computer, while *Xerox*'s investment in the start-up would be the start of the commercialisation of a series of graphical interface materialisations. These two elements (collaboration with developers and creation of an end-user interface) would be later proven to be critical assets for the enterprise as well as the industry as a whole.

Section 11.4 studies the *Foggy Competition* phase for the PC industry. The entry of *IBM* in this early market would be the beginning of the parallel exploration and exploitation of the PC concept (paragraph 11.4.1). As happened in the enterprise computer case, the potential of this new object was yet to be discovered. Soon after the *IBM PC* appearance in the market, in 1981, developers-entrepreneurs would clone it, progressively becoming important enterprises in the new industrial setting. Still, a market expansion to a broad public would not come without a graphical user interface, allowing the separation between developers and end-users, enabling the later to use the computer. To this goal, the contribution of *Apple* was decisive (paragraph 11.4.2). Such a process was not linear, neither for *Apple* nor for the rest of the industry, though the eventual synthesis by *Microsoft* would provide a solution for the greatest part of the new economy.

Section 11.5 studies the eventual *Industrial Rationalisation* of the PC industry by the *Wintel* model. As this period as been widely studied by management literature and specifically Gawer and Cusumano (2002), this section briefly reviews how the *Wintel* model managed to synthesize and rationalize the explorations of the previous period.

Table 11.1: Chapter overview.

Section	Phase	History	Trajectories deployment	Previous phase input			Phase challenges			Figure				Role
				R	C	K	R	C	K	UD	UDE	DE	Enterprises	
11.2	Early Materialisation	From Kildall to *Altair* (1972-1974).	Old: ad hoc, task based materialisations; New: a "workspace" New: a "personal computer"	Many exploratory collectivities.	Potential for a "personal computer".	Theory use.	User-developer collectivity stabilisation.	Old criteria comparison; New criteria creation.	"How to"; previous criteria comparison.	✓				Development, discussion, milieu incorporation.
11.3	Market Emergence	From *Altair* to *Apple II* (1974-1977).	New (DIY PC): a computer kit for UD communities. New (*Apple*): A PC as a product.	From fellows to early clients.	Object architecture.	Object development "how to".	Client support.	Object-task alignment.	From DIY to products.		✓			Client support & "rendering", production systematisation.
11.4	Foggy Competition	1980s.	Parallel exploration & exploitation.	Market existence.	Some trajectories to further explore.	Some development techniques, theory.	A "business ecosystem".	New trajectories emergence.	Wide, empirical potential exploration.			✓		Conceive new uses, create new markets.
11.5	Industrial Rationalisation	*Wintel* (late 1980s, early 1990s).	Most known trajectories	Market niches	Wide empirical exploration	Dispersed know-how	Market & Enterprise coordination structuring	Additional innovations	Common price/performance criteria proposition				✓	Synthesise knowledge, propose common design rules.
??	Phase breaks	*Linux*, *Netscape* (late 1990's)	Trajectory "jump"	>>	>>	>>	UD return	>>	Old concepts challenging	✓				Development, discussion, milieu incorporation.

11.2 Early Materialisation: the possibility of a Personal Computer

In this section we are going to examine two cases that occurred in parallel, the one in the U.S. and the other in France.

It is about two inventions of the "microcomputer" (a term of which was coined by our second inventor, Thi Truong). In the section 11.3.1 we are going to discuss these cases from the perspective of the two conditions we defined in 10.2.4 (page 197) for a dominant design to emerge (the knowledge condition and the market condition).

In later sections by applying this principle in the case of the PC we will see the interference between the two: the market which a product targets re-determines the product as well as the knowledge one has about its identity. In fact, the case of the emergence of the Personal Computer platform illustrates a specific property of a market-to-emerge, that of a *public*.

11.2.1 Old concept: Microcomputers that were not personal

Kildall and the possibility of using software

One of the first micro-computers reported is the one made by Garry Kildall in 1972. Kildall had received a Computer Science Ph.D. from the University of Washington and was teaching at the Naval Postgraduate School in Monterey, California[1]. Along with teaching, he started experimenting with the new chip of *Intel*, the *4004*[2], marketed by *Intel* as "a micro-programmable computer on a chip"[3]. The specificity of this chip was that it was the first "computer on a chip", in the sense that it incorporated the elements of Von Neumann's architecture (*processor*, *memory*, and Input/Output interface[4]), though it lacked one organ, the stored program. It was an "*ENIAC* in a chip".

Kildall used the *System 360* computer at his workplace to "emulate"[5] this microprocessor - as *Intel* only sold the chip at a low price ($25) to volume purchasers[6] and developed a programming language for it[7], with the use of *Intel*'s manual[8]. Active as a consultant in

[1]Freiberger and Swaine (2000, p. 21)Allan (2001, p. 7/1), Ceruzzi (2003, p. 223)

[2]Allan (2001, p. 7/1).

[3]Faggin et al. (1996).

[4]In that case we had a double phenomenon of generic design.

The *4004* chip was developed by *Intel* for a Japanese calculator company, *Busicom*. At the time, calculators had been among the main markets for the maturing semi-conductors industry. A specificity of *Busicom*'s requirements was that the chip should be able to be used in a family of products, not just one. Hence, the final design was what was called a *microprocessor*.

It was generic for *Busicom*, as it allowed them to build not a single, but a family of calculators. At the same time it was generic for *Intel* as well, as it opened the possibility for micro-computers to emerge, far beyond single purpose chips.

Intel's *4004* was as fast as the *ENIAC* (running at 108KHz). At that same time, mainframe computers, such as the PDP-10, ran at 1MHz. Today's smartphones run at 100MHz.

4004 had a 4-bit central processing unit (CPU), a 256 by 8-bit read only memory (ROM), a 320-bit random access memory (RAM) an a shift register for input/output (I/O) (Allan, 2001; Lecuyer, 2006).

[5]*Emulation* was a part of the computer industry rationalisation processes, enabling a computer to behave as a previous model (see section 10.5). While initially conceived to allow interoperability by backwards compatibility (Baldwin and Clark, 2000), it was used here as a method to pass from mainframe computers to what would later be the personal computer.

[6]Freiberger and Swaine (1999, p. 171).

[7]The Programming Language for Microcomputers, PL/M (Allan, 2001, p. 7/1).

[8]Freiberger, op.cit.

parallel with his teaching activities, he was assigned by *Intel* to develop a "stored program" for its next *microprocessor*, the 8008[9].

Intellec-8 Development System appeared on the market by *Intel* in 1973, having the new processor and Kildall's new *Control Program for Microprocessors (CP/M)*. The computer itself was marketed as a complementary product for enterprises, as a debugging tool to be embedded in existing systems and had a very poor market success for *Intel*[10].

However, Kildall's software, as well as his entrepreneurial activity since, was of a decisive role for the years to come, supplying basic modules for the personal computers, namely what became to be known as *BIOS (Basic Input/Output System)* and *DOS (Disk Operating System)*[11].

At that early point, the personal use of a computer was an enigma. It is reported that *Intel* primed the watches as a privileged consumer market for semiconductors[12] , while Kildall's early projects concerned an "astrology machine"[13]. However, the initial capital that Kildall gained from his activity as developer and consultant allowed him to found a startup, *Digital Research* that was to play an important role at the end of the decade and until IBM's entry in the market, in 1982.

The case of MICRAL

Micral was the first micro-computer commercially available. It was built by the enterprise *R2E (Réalisations d'Études Électroniques)* in May 1973. It was designed by Thi Truong and the engineers Francois Grenelle and Ben Chetrite on the basis of an *Intel 8008* microprocessor. The first mini-computer was and ad hoc, task-based materialisation, ordered by the French institute *INRA*, in order to be used in the measurement of evaporative transpiration, and more precisely the measurement of atmospheric water and temperature, an innovative process that was soil-independent.

The computer was conceived as a "complete product". It was characterised by a modular architecture, that included a "bus" component. A "bus", firstly used in the computer industry by *DEC*'s *PDP-8* mainframe computer, was a hardware device that had as its purpose the management of different computer components interfaces. Instead of connecting the different components to one another (as in the case of IBM S/360 family of mainframes), all components were connected to this bus, which then controlled and distributed current, signals and data between the different parts of the system.

In addition, *Micral* was equipped by an operating system, also built by the enterprise, the *SYSMIC*, later renamed *PROLOGUE*. In a similar way to the *bus*, an *Operating System* controlled and distributed data, calculating power and memory between different software programs, giving multitasking capabilities to the micro-computer (that is the ability to execute various programs at the same time).

After satisfying the contract with *INRA*, *R2E* further commercialised the computer. One of the first moves of *R2E* was to apply for patents both in France and in the USA. The patent concerning the bus, the device where peripherals could be connected, was filed in 1976 and granted one year later[14].

[9]Allan (2001, p. 7/1), Ceruzzi (2003, p. 223).
[10]Ceruzzi (2003, p. 223).
[11]Allan op.cit.
[12]Freiberger and Swaine (1999, p. 174).
[13]Ibid.
[14]Patent US4040026 - Channel for exchanging information between a computer and rapid peripheral units,

In the first year of its launch, *R2E* sold 500 *Micrals*. Its clientele was mainly enterprises and administration, not individual users (or user communities). *Micral* was sold at the price of $1950 and was marketed as a low cost alternative to mainframe computers, already in use. In an effort to expand in the US market, *Micral* was presented in the National Computer Conference Exhibition in Chicago in May 1974. The slogan of the advertisement was: "You don't need a hammer to crush a fly", illustrating the purpose of the micro-computer.

The advantages of *Micral*, as perceived by *R2E* and, thus, marketed lay in its low cost and technological integration[15]. Yet, despite the fact that *Micral* was the first micro-computer to be commercially available, as well as the fact that it was fully functional and had integrated innovative concepts in its design (such as a *bus* and an operating system), *Micral* didn't manage to determine the opening PC market. While the technical architecture of a micro-computer and a Personal Computer ended up being very close, the concept of *personal use* was absent from Micral's strategy.

During the micro or personal computer era, the question of the business model was also posed to the first enterprises that tried to commercialise this potential.

On the one hand, *R2E* used its *Micral* computer in a similar way to that first period of computer commercialization, promoting a use for already well-known fields and often renting its equipment, while supporting the installation with a technical service, in a way similar to *Intel*'s *Intellec-8* use scenario[16].

11.2.2 New concept: Alto "workstation" and its interfaces for profanes

A significant computer project that was never commercialised alibeit the source of many innovations to form the personal computer concept, as we encountered it in the 1990s, was the *Alto* "workstation", developed in *Xerox PARC* in Silicon Valley and used by its developers. *Xerox PARC* was characterised by a greater researcher autonomy and was the home for a set of early materialisations that would be explored by entrepreneurs during the foggy competition phase in the 1980s.

Xerox, a copier manufacturing company, decided to develop *Xerox PARC* in 1970, after acquiring a data processing company (*Scientific Data Systems*) as a research centre for

---

inventor: Francois Gernelle, Patent number: 4040026, Filing date: Nov 24, 1976, Issue date: Aug 2, 1977, USPTO.

[15] In *Micral*'s Manual edited by *R2E*, *Micral* is described as follows:

> *MICRAL is the first of a new generation of mini-computer whose principal feature is its very low cost.*
>
> - *Low Hardware cost resulting from:*
>   1. *use of integrated micro-processor (MOS-LSI circuits);*
>   2. *modular structure allowing minimal HARDWARE configuration for a given system;*
>   3. *no need of loading peripheral due to the use of Read Only Memories.*
> - *Low operating cost through the use of non-specialized personnel;*
>
> *The above automatically implies, in the system's basic design, complete integration between SOFTWARE and HARDWARE (Micral Manual, 1974, p.73, 77).*

This manual provided knowledge on hardware configuration operations as well as basic software commands.

[16] For instance, the report of Malet (1984), describes how the library of the University of Sorbonne installed a book - lending program, by the use of rented *Micral* computers with the help of *R2E*'s technicians.

exploring the market of electronic printing (Smith and Alexander, 1988; Chesbrough, 2006). Luckily for *Xerox*, *ARPAnet*, an early materialisation of the internet, was by that time reducing its activities[17]. As a result, many of the pioneers of computer science of the time were recruited in *Xerox PARC*. A key recruitment was the one of Robert Taylor, one of the leaders of the *ARPAnet* project[18], who then recruited many of the project's scientists and developers[19].

The *Alto* computer project was developed and used internally by those scientists and incorporated many of the concepts that would become known in the wider public during the 1980s and had been invented during the late 1960s. Engelbart, a prominent scientist who became distinguished during late 1960s for his concept of human-computer symbiosis, as it was imagined by another prominent scientist, Licklider, materialised these concepts in Xerox PARC.

The mouse, the windows graphical user interface, the first *WYSIWYG (What You See Is What You Get)* editor, the *NLS (oNLine System)* hypertext system - an early materialisation of what would become the Web - were among the materialisations realised in the PARC, most of them based on theoretical concepts developed by researchers of the *ARPAnet* during the 1960's (Myers, 1998). An underpinning idea in all these materialisations was the possibility of computer use by profanes. A very typical example was the *SmallTalk* application - itself introducing many concepts about windows graphical interfaces - conceived with the intention to be so simple that a child could use it to program.

While *Xerox* never commercialised those innovations, they would be a source of inspiration for many developers, for many of the research results of Xerox PARC would be soon diffused in the developers' community. Characteristically, Ted Nelson's pamphlet *"Computer Liberation - Dream Machines"*, edited in 1974 would amongst others, present *Xerox PARC*'s achievements[20]. Representative of the influence of this pamphlet is the fact that it would be re-edited by *Microsoft Press* thirteen years later, in 1987.

However, the most systematic exploitation of Xerox PARC's explorations would be made by Apple (discussed later), through the late 1970s and the early 1980s.

11.2.3 New concept: the Personal Computer

It wouldn't be wrong to claim that, in the case we are going to study, the concept of a personal computer emerged in the press and was developed by its audience. The concept of the personal computer was materialised through a long process of collaboration between user-developers in many U.S. cities. However, *Silicon Valley*, named after its semiconductor industry at the late 1960s proved to be a fertile ground for PCs as well, with one of the most active and creative user communities which illustrated the potential of a Personal Computer.

---

[17] Freiberger and Swaine (2000, p. 306).

[18] Abbate (2000).

[19] Turner (2006, p. 111).

[20] The conception of the term "hypertext" is attributed to Ted Nelson. This term was suggested to describe the "branching" of texts, as a sub-category of "hypermedia". Other "hypermedia" concepts proposed by Nelson have been the "branching" of films ("hyperfims"), audio, music and slide shows (Nelson, 1974, p. 85).

11.2.4 Before Altair: the emergence of a user community for the PC-to-come.

In July 1974, a radio user-developer journal, *Radio-Electronics*, published an article on the design of a kit called *Mark-8*, based on the Intel 8008 micro-processor (as did the microcomputers studied in sections 11.2.1 and 11.2.1)[21]. The design was realised by Jonathan Titus, a graduate student of Virginia Polytechnic University. The complete design was sold apart from the magazine, in a booklet costing $5. *Radio-Electronics* promoted the kit on its first page as follows:

> *"Build the Mark-8: Your Personal Minicomputer."*

To build their personal computers, readers would have to buy the components from different suppliers[22] and write the programs themselves. Characteristically enough for the nature of the design, as well as for its public, the magazine suggested that users use a keyboard and a TV terminal to facilitate use.

Mark-8 was presented as a "complete minicomputer which can be used for a number of purposes", including its complementary use for sending data to a larger computer as well as its use as a teaching tool.

A *Mark-8 user group* emerged using a Newsletter. The newsletter included software code, bug reports, places where the components of the computer could be purchased, as well as a list with the names and addresses of the community members, outlining their particular interest in building the computer (as a small business equipment, leisure activity and other[23]).

Mark-8 illustrated that there was a particular kind of market: people who were ready to pay and to work in order to build their own, personal computer. Moreover, it showed that these people were capable of self-organising to share knowledge on the design given, its materialisation and optimisation, as well as its extension.

It was a free-revealing of the design of a kit, in the sense that it is used in free software (citations), as it was open to everyone. In fact, at that time, the "hacker" culture according to which knowledge should be shared was already diffused in the universities, where Titus came from.

However, *Mark-8* was to be just the prelude for the computer kit that meant to organise the community of hackers until the market emerged.

11.2.5 Altair: a personal computer kit for the community

As a response to the publication of *Mark-8* designs, *Popular Electronics*, another radio magazine, ordered the construction of a Personal Computer from the *MITS* company. *MITS (Micro Instrumentation and Telemetry Systems)* was an enterprise founded by Edward Roberts, a former researcher of the Air Force Weapons Laboratory in Albuquerque[24], and two colleagues of his. The initial interest of this enterprise was in building toy rocket systems for user developers[25]. Yet, as this project didn't work, Roberts bought out the company from the

[21] Radio Electronics, Computer, Jonathan A. Titus, July 1974, pp. 29-33.

[22] Indicative prices listed circuit broads for $ 47.50, as well as the *Intel 8008* chip for $120.

[23] MARK-8 User group newsletter, 29 September 1974.

[24] Allan (2001, p. 4/8).

[25] Rocket technology had been the major innovation trajectory for the U.S. military since the 1950's. Computer evolution had followed that trajectory as well. See Ceruzzi (1989).

other founders who resigned. In 1971 *MITS* launched a calculator (the *MITS 316*), which was featured in *Popular Electronics*. It managed to sell thousands of them, before cheaper ones entered the market. After the publication of *Mark-8*, *Popular Electronics* ordered from MITS a computer to illustrate to the first volume of 1974, given their previous collaboration. Roberts designed and sent the computer to the magazine at the end of 1973. The device was featured in the January issue. Yet, while the photos of the first MITS Personal Computer were published, the prototype never arrived at the magazine's offices, due to a train strike[26]. This accidental fact presented the occasion for the Altair (the name of the PC) to be re-designed.

The new design included a bus. Yet, no software was available with the device. MITS organised a user group, providing additional information for developers to build software and other components.

The editorial of this issue made the following claim:

> *THE HOME COMPUTER IS HERE!*
>
> *For many years we've been hearing about how computers will be one day a household item. Therefore, we're especially proud to present in this issue the first* commercial type *of* minicomputer project *ever published that's priced within reach of many households - the* Altair 8800*, with an under-$400 complete kit cost, including cabinet.*

Other radio and electronics magazines welcomed the new device also with enthusiasm. In Figure 11.3 we see the cover of the first Issue of the *Interface Age* user developer magazine (December 1975) presenting *Altair* as the embodiment of the "People Computer".

Altair used the *Intel 8080* microprocessor, just introduced. While Intel had quoted a price of $360 for small qualities, MITS bought them for $75 each. At a lower cost, with a more effective microprocessor and with all its components put together, *Altair* was accepted with enthusiasm by the radio user developers communities.

MITS was clearly a D.I.Y. kit. Eddie Curie, a friend of Roberts who joined him early on when the start-up couldn't keep up with the demand, described the initial service:

> *One of the nice things about the kit [from MITS' point of view] was you didn't have to test the finished units. You just put all the stuff in envelopes and shipped them. It was left to the poor customer to figure out how to put all those bags of chunk together*[27]*.*

11.2.6 User communities: the case of Homebrew Computer Club.

Many *Altair* user groups arose within the USA. Amongst them, a user group that was to have an historical influence, the *Homebrew Computer Club*, based in the San Francisco Bay Area. This area had already a long history of radio user developers, linked with the computers and semi-conductor industry[28].

The first newsletter of this community referred to the success of the first meeting as well as the contact with MITS, writing:

[26] Freiberger and Swaine (1999).
[27] Quoted in Levy (1984, p. 195).
[28] a expliciter, plein de references la-dessus

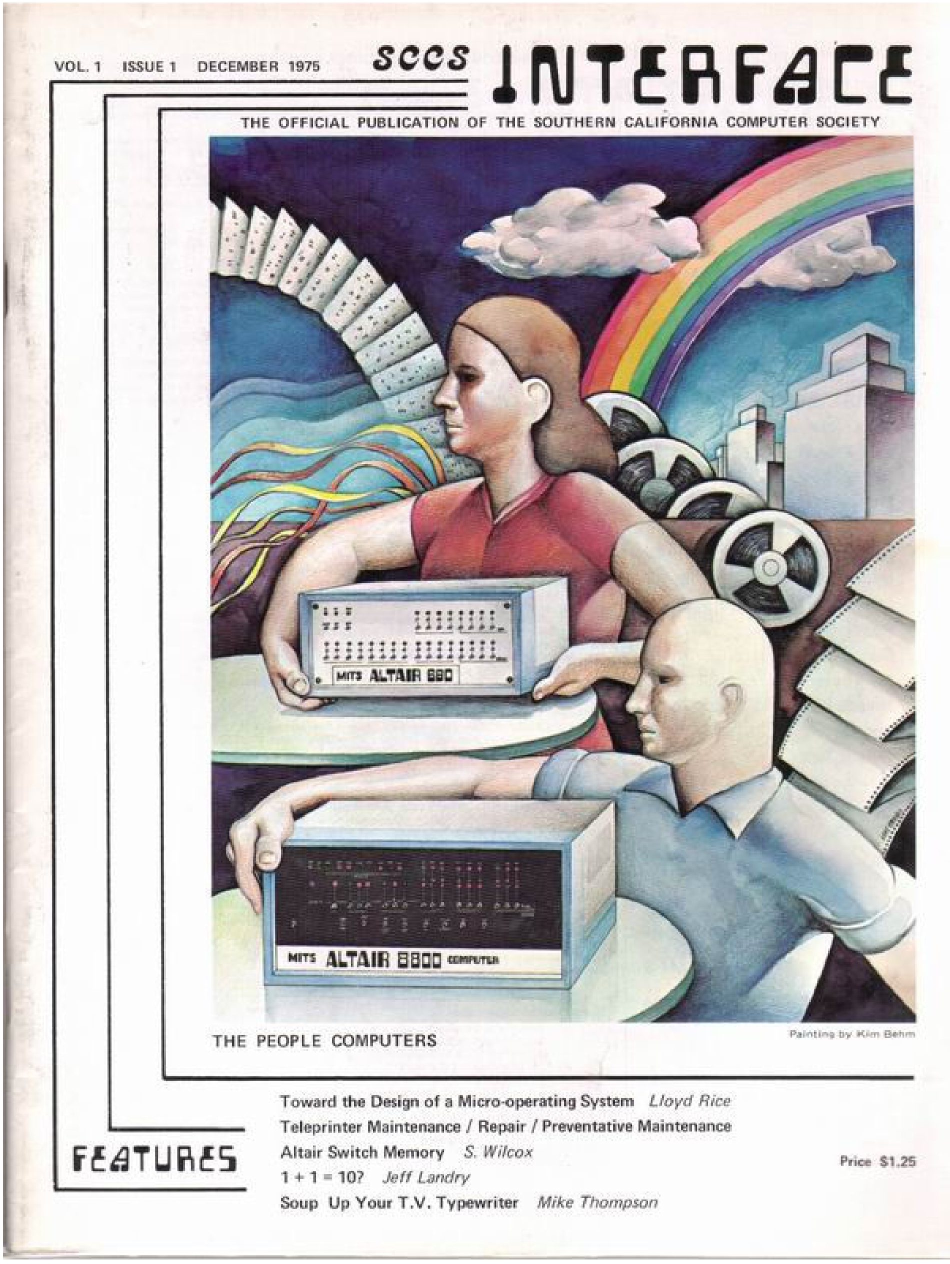


Figure 11.3: Altair computer figuring in *Interface Age* user-developer magazine's first cover as the People Computer (December 1975). *Source*: http://www.vintage-computer.com.

Are you building your own computer? Terminal? TV Typewriter? I/O device? Or some other, digital black-magic box? Or are you buying time on time-sharing service? If so, you might like to come to a gathering of people with likeminded interests. Exchange information, swap ideas, talk shop, help work on a project, whatever ...

This simple announcement brought 32 enthusiastic people together on March 5th at Gordon's garage. (...) The group contained a good cross section of both hardware experts and software programmers.

While many members of this emerging group had already been working on devices based on the 8008 processor, the leader of the group, Steve Dompier, a computer user developer who used to rent time in time-sharing computers[29] in Berkley, had visited MITS' offices:

Steve gave a report on his recent visit to MITS. About 1500 Altairs have been shipped out so far and MITS expects to send out 1100 more this month. No interfaces or peripherals are available until they catch up with the mainframe back orders.

The meeting was organised by Fred Moore and Gordon French, two members of the Popular Computer Company that pre-existed in the region. Amongst the user developers that attended the meeting figured some of the most skilled engineers of the industry (amongst them Stephen Wozniak, who worked at the time at HP). Lee Felsenstein, one other member of the PCC community, who was to invent the first portable computer some years later, commented on the meeting's participants:

a bunch of escapees, at least temporary escapees from industry, and somehow the bosses weren't watching. And we got together and started doing things that didn't matter because that wasn't what the big guys were doing. But we knew this was our chance to do something the way it should be done[30].

The next appointment of the group was given at Stanford's *Artificial Intelligence Laboratory* Conference room.

At the same time, *Homebrew Computer Club* organised its functions on a standard basis, with meetings - usually held in Stanford University - and newsletters. While sharing software was a common practice at that time, PC development was private: each member built his computer at home, while the community was there to provide knowledge, as well as contacts.

This user community wasn't a "large public" community. MITS didn't provide software or an operating system to its users. New comers were always welcome in this community, yet amongst the members of this community one could find experienced computer scientists and designers, working in the industry of the region or researching at its universities.

Regarding community organisation, the club's Newsletter remained the main tool for knowledge and information, as well as for user networking. Overall, it was a skeleton of the hobbyists' market. There, one could find the news about emerging *computer stores* and the emergence of retail outlets, source code for computer programs, workshops organised by platform providers, universities or consultants, access to the club's library.

Moreover, the community undertook initiatives to organise this user developer based market. The "West Coast Computer Faire", starting in April 1977, was co-organised by a

[29]For time-sharing computers, see section ...

[30]Levy (1984, p. 206).

number of UD computer communities, Bay Area chapters of the well-known Association of Computing Machinery (ACM), Stanford University's Electrical Engineering Department[31]. Besides, this is where Steve Wozniak, one of the members of the Homebrew Club, along with his friend Steve Jobs were to present Apple II in 1977, the first market-oriented personal computer.

A user-developer based PC potential exploration

As we've seen in sections 11.2.1 and 11.2.1 on the early microcomputers, vendors (*Intel* and *R2E*) marketed their product assuming a similar use trajectory to previous computers, though a minor one, leading to the identity of a "support equipment" for larger information systems.

Still, when similar devices (mainly the *Altair* kit) came to user communities, there was a far more extensive exploration of possible uses. In fact, the exploration of different trajectories for the microcomputer was one of the founding elements of these communities. The first newsletter of the *Homebrew Computer Club* summarised their discussion as follows[32]:

> *What will people do with a computer in their home? Well, we asked that question and the variety of responses show that the imagination of people has been underestimated. Uses ranged from the private secretary functions: text editing, mass storage, memory, etc., to control of house utilities: heating, alarms, sprinkler system, auto tune-up, cooking, etc., to GAMES: all kinds. TV graphics, x - y plotting, making music, small robots and turtles, and other educational uses, to small business applications and neighbourhood memory networks. I expect home computers will be used in unconventional ways–most of which no one has thought of yet.*

In this small excerpt, we trace a creative exploration of the *Altair* and more generally the Personal Computer potential that was a result of a process that resembles brainstorming (Osborn, 1953), already in use by computer companies (Clark, 1958). This plethora of ideas cited can be categorised as follows:

1. Concepts related to already known instrumental aspects of computing in general (such as text editing, mass storage, memory etc), to be implemented at the level of the microcomputer.

2. Concepts related to domains of productive activity (small business, private secretary).

3. Concepts related to private life, including issues of the house (heating, security) or spare time activities (games, music).

4. A concept of the use by a local collectivity (neighbourhood memory project).

Moreover, the richness of the imagined uses gave the feeling to the writer of the newsletter, and probably to all the participants, that there was a great expansion potential that remained to be discovered.

[31] Homebrew Computer Club Newsletter, Robert Reiling, Editor, Mountain View, CA, vol. 2., Issue 9, September 15, 1976.

[32] NEWSLETTER, AMATEUR COMPUTER USERS GROUP HOMEBREW COMPUTER CLUB, Issue number one, Fred Moore, Editor, 2100 Santa Cruz Ave., Menlo Park, Ca. March 15, 1975.

As the community grew, *MITS* became more systematic, opening retail stores. There, one could find videotapes from the seminars that continued to go on, while qualified personnel in the stores could advise customers[33]. Regarding the computer stores, it became clear since the first one, in California, that users were willing not only to buy a computer, but to spent multiple amounts of money to purchase extensions (extra memory, video terminals, disk drives etc.[34]).

In fact, less than a year after the launch of *BASIC*, *MITS* organised the *World Altair Computer Convention*, attended by several hundred participants, most of them coming from User Groups[35]. Those conferences were attended by what Gawer would call complementors, small enterprises that build peripherals for *Altair*. Some of them did Robert's support, as they were active in new domains, others did not, as they were developing peripherals *MITS* developed as well[36]. Lee Felsenstein, a member of the *HCC*, presented in this Convention a *Visual Display Mode* prototype[37]. This unit permitted the linking of a video display directly to the Altair's RAM, an innovation that, for the first time, made possible obtaining a highly interactive personal computer, and opened the field for uses such as video-games. *Processor Technologies*, built a board that could be connected to Altair's bus and provide additional memory.

From the client support challenge to commercial software

Among the 'tasks' of community support, *MITS* had to extend its platform, the *Altair*, to enable further extensions by the user community. One of the most important was a software language, a feature that was developed outside the enterprise to be quickly integrated in the product.

Bill Gates, student at the time at Harvard, and Paul Allen, working as a programmer at *Honeywell*, developed a version of the *BASIC* programming language[38] for the *Altair*, which was distributed along with the computer for an additional cost of $500. As Kildall had done, Gates used a mainframe computer to "emulate" the *Altair*. Allen became director of software for MITS[39], to found *"Micro Soft"*, later *"Micro-Soft"* retaining the rights for the software[40]. *Microsoft* would later develop the operating system for the *IBM PC*, having already invented a new business model for software as a product[41], on which the software industry would be based for the decades to follow.

Apart from accepting visitors, *MITS* edited a magazine available at a low price, the *Computer Notes*. The content of this resembles to the forums and blogs that platform providers have today, written by engineers:

[33] *MITS mobile caravan seminar*, Computer Notes Volume 1 Issue 11 - April, 1976.
[34] Freiberger and Swaine (2000, p. 232).
[35] Freiberger and Swaine (2000, p. 226).
[36] Freiberger and Swaine (2000, p. 62,63).
[37] Computer Museum Report, Fall 1986.
[38] BASIC programming language had been developed in Dartmouth University and had been diffused through the academic community and the computer enterprises through the time-sharing project (Ceruzzi, 2003, p. 203).
[39] Allan (2001, p. 4/10), Ceruzzi (2003, p. 235).
[40] Ceruzzi (2003, p. 235).
[41] Until then, software was shared among the user developers, as well as among the professionals. On a symbolic level, Gates' "Open Letter to Hobbyists", published in *Homebrew Computer Club's Newsletter* in 1976, stating that software should be ruled by copyright, as with music, marked this conceptual rupture, yet to be implemented some years later.

- Questions & Answers, on the 'right' use of *Altair*'s configuration and problems to be fixed.
- Specialised articles to master specific functionalities, options, tests and configurations of the hardware as well as short introductions to software programming (the latter, *Software Notes*, was often edited by Bill Gates).
- Information about new extensions (such as the floppy disk system or the static memory card).

In addition, user contests were organised by the magazine, constructing a common knowledge base for *Altair*'s User Group, also reflecting the expertise advancement of the community. Concerning a software contest, Gates wrote:

> *We started receiving programs for the Altair Users Library a few months ago, and are getting more and more every day. As the Users Group grows and users become more sophisticated at programming their Altairs, we expect the library to grow at a fast rate and become a valuable resource for Altair users*[42].

In the same journal, its editor would later address *Altair* users trying to convince them to actually pay for the use of Basic[43]. In fact, software had not been commercialised in the previous computer industries, as it was something that either the computer provider company or the using institution would develop. In addition, software sharing had been one of the basic arguments for the development of the university network (the *ARPAnet*), to diminish the costs. In this sense, one can say that *Microsoft* had invented commercial software, by demanding that software royalties should be paid by user developers. The letter Gates addressed to the UDs and was sent out by the *Homebrew Computer Club* in February 1976, had a symbolic value for this "jump". Typically, Gates noted[44]:

> *The feedback we have gotten from the hundreds of people who say they are using BASIC has all been positive. Two surprising things are apparent, however, 1) Most of these "users" never bought BASIC (less than 10% of all Altair owners have bought BASIC), and 2) The amount of royalties we have received from sales to hobbyists makes the time spent on Altair BASIC worth less than $2 an hour.*

In a sense, this letter was the first declaration of the software market's emergence, which would later on develop and wherein *Microsoft* would maintain a dominant position until this day.

11.3 Market Emergence: the possibility of a commercial PC

As time went by, the demand by hobbyists was getting bigger, new start-ups entered the hobbyist kits market and *MITS* continued to be unable to satisfy the demand. One of the

[42] Gates, Bill. *Software Contest Winners Announced*, Computer Notes, Vol. 1, Issue 2, MITS, Inc. Albuquerque, New Mexico, July 1975. Source: New Mexico Museum of Natural History and Science.

[43] Bunnell, David (September 1975). "Across the Editor's Desk". Computer Notes (Albuquerque NM.: MITS) 1 (4): p. 2.

[44] Gates, Bill, AN OPEN LETTER TO HOBBYISTS, February 3, 1976.

major competitors was *IMSAI*, an enterprise found by a former *IBM* engineer, that introduced an *Altair* "clone" and got connected to the hobbyist communities. In addition, it contacted Kildall and agreed with him to use his operating system on the computer (*Altair* didn't have an operating system at the time)[45].

In 1976, Steve Wozniak, an engineer working at the time at HP and a member of the Homebrew Computer Club, along with Steve Jobs, presented their new computer, the *Apple* in the Homebrew Computer Club meeting. Wozniak described the aim of Apple I and Apple II (the latter presented one year later) as follows:

> *The Apple I and II were designed strictly on a hobby, for-fun basis, not to be a product for a company. They were meant to bring down to the club and put on the table during the random access period and demonstrate: Look at this, it uses very few chips. It's got a video screen. You can type stuff on it*[46].

Steve Jobs managed to sell fifty complete boards to the *Byte* computer shop[47] (Apple I did not have keyboard, a box nor a power supply). Moreover, he then managed to raise funds for the next computer, with the aid of a former Intel employee, Mike Markkula, he hired Mike Scott, a former *Fairchild Semiconductor* manager, as a CEO[48]. He also hired Jerry Mannock, a former HP designer, to design a beige plastic case with rounded corners for *Apple II*[49].

Apple II was presented in the West Coast Computer Faire in April 1977[50]. By the end of 1977, the company was making a profit and doubling production every three to four months[51]. It was a computer in a box, for personal or business use, for which users did not have to assemble its parts, though they could "plug in" the desired peripherals.

A year later, Paul Allen had left *MITS* to join Microsoft, that started licensing its *BASIC* to a series of computer enterprises, having a revenue for its first year of over one hundred dollars[52]. This caused a controversy between the two enterprises. In early 1977 *MITS* was acquired by Pertec Computer Corporation. However, in late 1977 the arbitration process that started before the acquisition officially accorded *Microsoft* the right to sell *BASIC* to other suppliers as well, thus denying *Pertec* an exclusive right[53].

An article in the *IEEE* in 1978 (Doerr, 1978) observed that *Altair*'s skeleton (the S-100 bus), that allowed the product's expansion had been used by a great number of small enterprises building peripherals and had been copied by competitors. In 1983, IEEE would extensively describe and expand the specifications of the S-100 bus having as an objective, among others:

- *To provide the microprocessor computer-system user with compatible device families which will communicate in an unambiguous way without modification, from which a modularly expandable computer system may be constructed.*

[45] Ceruzzi (2003, p. 239).

[46] Wozniak, S (1984). Homebrew and how the Apple came to be. In Ditlea, S., *Digital Deli*, Retrieved July 18, 2012, from http://www.atariarchives.org/deli/homebrew_and_how_the_apple.php

[47] Laing (2004).

[48] Levy (1984, p. 257).

[49] Laing (2004).

[50] Freiberger and Swaine (2000, pp. 276-277).

[51] Freiberger and Swaine (2000, p. 282).

[52] Allan (2001, p. 6/10).

[53] Allan (2001, p. 6/10).

- *To enable the interconnection of independently manufactured devices into a single system*[54].

One of the successful and decisive extensions of the Apple II computer was the *VisiCalc* spreadsheet and the *WordStar* word processor. These two programs enabled users to utilize the computer for office tasks. The low cost of the computer (at $800 dollards, compared to the $10000 *Xerox Alto* production cost) made it affordable for a large base of customers[55]. Moreover, they indicated the trajectory where a market for Personal Computers could be directed, a principal use: the personal computers of the eighties would focus on the document trajectory, not the applications one. Applications would have an auxiliary function for document processing[56].

The contribution of Jobs to the history of the personal computer was not only the PC commercialisation, but also the exploration since of the early materialisations of *Apple* computers. During the passage from the *Apple I* computer board to the *Apple II*, there had to be a power supply. However, Jobs made an additional requirement: a power supply that didn't require a cooling fan, for the computer to be silent in operation, compared to the competitors' noisy fans[57]. For this design task, he hired an expert as a consultant[58]. Noise reduction was not within the requirements of the computer industry bat the time, and one could say that PCs continued to be noisy compared to the ones built by *Apple* for decades.

Jobs' capacity of inventing new requirements and recruiting the people that had the necessary knowledge to undertake the design, led *Apple*'s steps from being a start-up to becoming a large enterprise. In this endeavour, Jobs made the best of the tradition of the Silicon Valley, regarding ties between industry, university, venture capital and technological communities. Even during the 1980s, when *Apple* was one of the leading enterprises of the domain, it preferred its suppliers to be installed in the region, "next door", as the materials director had said[59]

One of the fundamental knowledge bases that Jobs managed to exploit was the one in the *Xerox Palo Alto Research Center (PARC)*, regarding *Human-Computer Interaction* [60]

At the time, *Xerox*, a leading enterprise in printers had established a price/performance model, based on speed, per-copy costs and printer size (Christensen and Rosenbloom, 1995, p. 241). However, they had a serious problem in the commercialisation of the PC. The *Alto* computer, possessing many innovations that would be introduced in the market by other computers later, was too costly for users to buy.

Hence, *Xerox* invested in *Apple*, which had proven to be a dynamic start-up. Because of the important demand from investors, Jobs managed to negotiate beyond the $1 million

[54] IEEE, *IEEE Standard, 696 Interface Devices*, 1983.

[55] Moggridge (2006, p. 80).

[56] Moggridge (2006, p. 81).

[57] Allan (2001, p. 5/8).

[58] Ibid.

[59] Jim Bilodeau, *Apple* materials director, quoted in Saxenian (2000, p. 152).

[60] For Chesbrough (2003a) the case of of *Xerox* is a typical case of closed innovation, the following question is posed:

> *How could a company that possessed the resources and vision to launch a brillian research center - not to mention the patience to fund the center for more than thirty years, and the savvy to incorporate important technologies from it - let so many good ideas get away(Chesbrough, 2003a, p. 1)?*

A series of innovations in *Xerox PARC* were commercialised by spin-offs.

Xerox offered to license some key technologies developed in *Xerox PARC*. Moreover, many *Xerox PARC* joined *Apple*, where they contributed their knowledge and experience.

The *Apple II* computer, while serving the ambition of a mass-consumer market, continued to be addressed to experienced users, as did the *Altair*, though more advanced and following a different architecture (mainly because it was based on a different chip than *Intel*'s one). Jef Raskin, the designer of the Macintosh computer, in an internal *Apple* report in 1981, referred to a memo written by another *Apple* employee, "Too many variations on a theme" by Brien Howard (1980), was charged to write the Apple II manual:

> *In writing [a manual] I have, once again, come across an exasperating problem: just what IS an "Apple Computer"? Apples now come in a bewildering variety of flavours*[61].

In addition, *Apple* cultivated a close relationship with user-developer-entrepreneurs, as *"it encouraged independent software developers to write programs for its machine".* (Moore, 1993, p. 77).

11.3.1 Discussion: the distance between the microcomputer, workstation and the personal computer

For both inventors-entrepreneurs and their start-ups (Kildall with the *Digital Research* and Thi Truong with *R2E*) their inventions allowed them to be among the first to enter a new market-in-emergence, both on the level of competencies developed and the one of knowledge acquisition and development.

In this early stage, we observe that for all above mentioned actors, the market assumed was identical with the one that pre-existed. A *microcomputer* was considered as a very small computer, thus performing tasks already known. Hence, their devices were conceived and marketed as auxiliary instruments for larger, professional information systems and not as a product that could be used for personal use. Thus, we could speak about a *fixation effect* (Hatchuel et al., 2011) in this early phase, a phenomenon often observed in innovation processes, restricting the perception of the field of possible to pre-existing concepts.

On the level of knowledge we've seen that a series of notions and methods developed during the era of enterprise computers were mobilised by the two entrepreneurs in the context of the development of a really small computer, a micro-computer. The use of *emulation* by Kildall, Von Neumann's architecture for the two, as well as the bus-architecture for Truong were some of the core competencies that were transposed from the existing computer platform to the emerging ones. That was a design and development accomplishment that one should not underestimate, as it demanded the re-designing of the whole frameworks to fit the specific parameters of the new microprocessor. It is not by chance that the two were scientists working in some of the finest research institutions of their country.

On the level of the market though, their initial concepts were not addressed towards something that could be described as a personal computer. In the case of *R2E*, while *Micral*'s design was very close to what a personal computer would ultimately be, it was neither conceived or marketed as such. Like *Intellec-8*, it was conceived and marketed either as a small and cheap computer for institutional use or as a complementary machine for existing

[61] Quoted in the internal Apple report Raskin, Jef, *Macinthosh project genesis and history*, Feb. 1981, p. 1/5.

computing installations. On the other hand, Kildall's initial intuition on a "astrological machine" didn't meet a market.

However, we should note that a personal computer was not an intuitive concept at that time. Computers had a radically different status, as we've seen in the previous chapter, being either a scientific, a military or an enterprise machine. Moreover, computer science was still an early discipline, and users would have to acquire new competencies in order to use a computer. Thus, the question of knowledge diffusion should be considered a barrier to the new market emergence, as technical manuals were addressed and conceived for engineers, and personal use was not in their paradigms or philosophy.

This knowledge - market gap was to be progressively covered by a user developer public that sprang up.

Going back to Table 10.3 on page 203 regarding the two first phases of a dominant design emergence, we can see that converning *Micral* and *R2E* we can make some important observations.

First of all, the same challenges were posed in the first phase, the one of concept materialisation and *R2E* did manage to respond to most of these challenges. *Micral* did have a generic design as well as the tools (software, operating system) to restrict this design to specific applications. The project was initially financed by a single client, *INRA*, and the computer could be configured to operate market specific tasks.

At the same time though, *Micral* was not based on a new theory about computers. It was conceived as a really small computer (a *microcomputer*), addressing the same tasks with the pre-existing mainframe and minicomputers, as well as the same market (ex. administration).

The PC wave, that started almost simultaneously as we are going to see, while based on the same knowledge on computers, it was carried by a projection of a different concept of a computer, that is a computer that could be personal. In the next section we will see how this 'little' difference (from an engineering point of view) caused a major transformation of business rationales, in the path to concept materialisation, creating a new trajectory for innovation.

11.3.2 Conversation. The emergence of new first phase collective rationales, replacing traditional ones.

The case of the PC concept emergence and the attempts of its materialisation created new collective rationales that radically differed from the traditional ones. The concept-oriented challenges, as well as the market-oriented ones that were address until now with traditional business methods (see Table 10.2 on page 198), where now undertaken by "hacker communities".

Pre-existing knowledge on computer science had been diffused to the public either by specialised magazines (such as the *Radio Electronics Popular Electronics*), or by user developers themselves (such as Jonathan Titus).

We've described two ways in which the question of generic design was addressed in this early phase. On the one hand, *Mark-8* was just the diffusion of a design by a radio user developer magazine. A diffusion that created a dynamic of communities that were self-organised and networked by a newsletter. Thus, in place of the market potential exploration we've seen in the case of Mauchly and Eckert, we have a community vision emerging and moving towards the materialisation of the concept.

For this development, instead of having single client, targeted projects, as both *EMMC* and *M2E* did during their first steps, we have a *Do It Yourself (D.Y.I)* approach, where user

Traditional Business Rationale		Community-based rationale
Marketing by potential	→	Community vision
Single client, targeted projects	→	D.I.Y
Businessmen - University intimacy	→	Hacker communities

Table 11.2: Hackers' rational replacing traditional business rationale. The case of *MITS* firm.

developers create their own, personal computer, based on the common knowledge.

Later, we had the appearance of business in the form of *MITS* start-up. *Popular Electronics*, representing what we could call the "public will' of its readers, ordered the computer, without Roberts having to perform any market research. Since then, and until its end, his start-up generated profit for him, though he never managed to satisfy the growing demand.

With *Altair* there was a combination of the D.I.Y. approach and the product one. It was a computer addressed to a specific public, hacker communities, that could obtain the necessary knowledge to use it as well as to extend it.

The Businessmen-University intimacy we've encountered in the case of *EMMC*, that played a crucial role in knowledge diffusion, demand creation and uses exploration, was now replaced by the "hacker community intimacy", with the communities acting within the universities that advanced computer science (the case of the Homebrew Computer Club being emblematic), the industry experts being part of them.

This 'hybrid' form of a product, this 'half way solution' (Hatchuel and Weil, 1999) or 'not-yet-complete design' (Baldwin and Clark, 2006), put *MITS* in the position to continue to the second phase, the one of market emergence. Here, the Engineering Design Department for *Altair* was the user communities, with the most advanced of the users building their own (complementary or competitive) start-ups.

In the Table 11.2 we outline the replacement of the elements of the traditional rationale met in section 10 with the one met in the current section.

Thus, the Table 10.2 (page 198) is transformed to the Table 11.3 (page 246).

Here, the theoretical object description concerns more the potential of the object on the basis of community ethics or interests. Further design development concerns personal use, as we are in a user innovation realm (von Hippel, 1986). For the enterprise launching the semi-designed product, communities constitute networks of potential clients as well as contractors. Given the shared view amongst community and provider of the emerging phase of the product, there is also a grace period, similar to the "beta version" we observe in software business (Windrum, 2004) or Web services (Evans and Schmalensee, 2007; Evans, 2010). Here, environmental changes and firm strategy co-evolve, as happened in the case of the *Flash memory* invention at the late 1980's (Burgelman, 1991) through platform potential exploration.

We should note that at this point, the business model only concerned the device (Altair computer) and not the software. Concerning the latter, one can trace here the origins of free and open source software, of which the development is still undertaken in a similar mode (Hars and Ou, 2001; Bergquist and Ljungberg, 2001; von Krogh et al., 2003a; Lakhani and von Hippel, 2003; von Hippel and von Krogh, 2003; Osterloh and Rota, 2007; O'Mahony, 2007; Benkeltoum, 2008, and others)

Hence, this different rationale determined the way in which *MITS* entered the second phase, the one of market emergence, when the first signs of a mass market possibility for

	Community-based Rationale		
Challenges	Community vision	DIY	Hacker communities
Design oriented			
Can a generic design be reached?		Intermediary products	
Can the object be described theoretically?		“visionary” object description.	Knowledge diffusion
Can the generic design be restricted to applications?			Computer for the self
Market oriented			
Can there be a client demand?	A network of potential clients		University - community ties
Can the project be financed?		A network of contractors	Demand creation
Can we configure the product for specific tasks?			Extensions demand
Can we scale production for demand?			“Grace” period
		Business environment transformations	

Phase 1 : From a concept to a product

Table 11.3: Phase 1: concept materialisation. The different rationale answers the same challenges in a different way, creating a “paradoxical” business environment.

personal computers started to appear.

11.4 Foggy Competition: the case of the *IBM PC* and the *Apple Macintosh*.

While the early phase of concept emergence, studied in the previous section, is practically ignored by management scholars, as it has never been a study field for the discipline, what followed next has been extensively examined. Hence, we are going to retrace the genealogy based on existing management research.

11.4.1 IBM PC

Like in the case of the System 360 modularisation (section 10.5), the IBM PC was designed and developed by a team working apart from the rest of the organisation. IBM built a separate shop at the IBM plant in Florida, and focussed on getting the new product ready fast[62]. For this, IBM chose to use components from actors that were already in the market. Having as a first choice the CP/M operating system (see section 11.2.1), they eventually decided on a collaboration with *Microsoft*, which developed *MS-DOS* for that reason, having many similarities with the CP/M[63]. The first IBM PC, launched in the market in 1981, also used an *Intel* processor.

At the same time, *IBM* tried to control the architecture of its computer by patenting the 'skeleton', the internal device where other components could be 'plugged' into, the *bus* or *PCI*. In order for third party developers to extend the IBM PC, they needed to obtain a license from the company.

However, both *Microsoft* and *Intel* were allowed to sell their products to other manufacturers as well. Beginning with *COMPAQ*, a series of *IBM - clones* flooded the market during the 1980s. In 1989, the IBM clones sales began surpassing the IBM ones.

Meanwhile, throughout the 1980s, the design of the computer was not stabilised. Many products did not include monitors, while the mouse - already invented in *Xerox PARC* during the 1970s - was to be introduced on a commercial scale in the mid-eighties.

As described above, by the mid-eighties, the elements that described the dominant design of the *Desktop Computer* since the nineties were set. On the one hand, *IBM*'s entry continued the trajectory of the *Homebrew Computer Club*'s computers. On the other hand, *Apple* survived competition through the importation of features that were unknown to the computer user developers, thanks to the strategic alliance with *Xerox*.

However, these were not the only designs that arose during the eighties, neither was a linear trajectory visible for the entrepreneurs of the time. Dozens of computers were developed by various competing companies[64]. Among them, the Osborne 'transportable' computer in 1981 (also using the *CP/M* operating system and costing less than $2000)[65], *Radio Schack*'s *TRS - 80 Pocket Computer* in 1980 and the *TRS - 80 Model 100* early

[62] Moggridge (2006, p. 78).

[63] An *Operating System* is a "spinal cord" for different software applications. In a way similar to the *bus* for hardware, the *DOS* enabled the sharing of computer resources (such as memory and computing power) by different programs. In fact, the *DOS* was mainly a set of interfaces, the *Application Programming Interfaces (APIs)*. As the scholars note, these interfaces enabled the growth Microsoft ecosystem, that is developers producing applications for the Windows.

[64] An extensive list of the main computers of this era can be found at Laing (2004).

[65] Ceruzzi (2003, p. 278).

'laptop'[66] and the *Compaq Portable* in 1982[67]. However, of all these actors of the new market, *Compaq* was one of the few that managed to obtain an ecosystem for extending its platform, and did so by following the design of the *IBM PC*.

11.4.2 Lisa VS Macintosh: the challenge of windows GUI materialisation

Lisa and *Macintosh* were two computers that were launched in early 1984 and marked the computer industry trajectories for the eighties. Both were document-oriented, rather than application-oriented. Earlier, the enterprise was re-structured, following a product-oriented organisation (instead of a functional one)[68]

In 1979, when *Xerox* invested in *Apple*, it allowed Jobs and his team to visit *Xerox PARC* and learn about their inventions on the *Graphical User Interface (GUI)*[69]. The concept of the mouse and the metaphor of the Desktop, intially invented by Douglas Engelbart in Xerox, was re-considered, by the requriements of being twenty times cheaper, "being able to use it on Formica and my blue jeans", as well as to enable actions on the window environment, such as drag'n'drop or smooth scrolling (documents that did not lurch line by line as users scrolled through them, but instead flowed smoothly).

However, despite the innovations *Lisa* computer embodied, it was not a commercial success: on the one hand, its files were not compatible with either *IBM* or *Macintosh*. On the other hand, its price was too high. Hence, *Lisa* was discontinued in 1985[70]. However, many of those innovations would be embodied in later *Macintosh* computers and become elements of the dominant design of the computer of the nineties.

While *Lisa* was a computer that explored the potential of things invented in *Xerox PARC* by introducing them to a *PC* platform, the *Macintosh* computer was the computer that gave *Apple* its character. Initially designed by Jef Raskin, professor in San Diego University, the *Macintosh* was designed to be a mass market product. Its final price ($ 2 495, more than the $1 000 Raskin was aiming for) made it more expensive than an IBM PC, yet affordable, given the fact that the innovations introduced did not enter any of the trajectories on which the PC had been developed.

For *Macintosh*, a re-organisation occured in *Apple*, putting Jobs and the head-designers of *Lisa* computer in charge of the product development[71]. While initially projected to appear on the market in 1982, it was eventually released in 1984. The main reason for the delay was the fact that the product was to embody a rationalisation of the extension capabilities: extensions were to take place only by software, thus the hardware was 'closed' in a box, unlike *Altair*'s case.

For that, a set of programming platforms were developed for *Apple* under Job's supervision: *Microsoft* developed a spreadsheet program (*Multiplan*) and a *Microsoft BASIC* interpreter for the *Macintosh*. *Apple*'s employees developed an operating system, a word processing program (*MacWrite*), a graphics program (*QuickDraw*) as well as other functional programs[72].

[66] Allan (2001, p. 11/2).
[67] Ibid p. 11/10
[68] Allan (2001, p. 10/2).
[69] Isaacson (2011, p. 96,97).
[70] Allan (2001, p. 10/18).
[71] Allan (2001, p. 10/15).
[72] Allan (2001, p. 10/10-10/12).

In the following years, *Apple* organised the community of the developers around meetings, the *AppleWorld* conferences, where it announced the new products.

As the initial concept of *Macintosh* closure regarding hardware had limited the expansion capabilities of the computer, a few years later, in 1987, the *Macintosh SE - (System Expansion)* - targeted to business markets - would embody an *expansion slot*[73] and the *Apple Desktop Bus*, both enabling the plug-in of third party hardware, in the way *Homebrew Computer Club* members could do on the *Bus* of the *Altair* computer, or with *Intel*'s *PCI* and *USB* devices later (for the latter see Gawer and Cusumano (2002)). Provided the 'closure' of the computer within the box, the first interface addressed 'internal' extensions of the computer (such as hard drives) while the second one 'external' extensions (such as keyboards or mice). Later, the transposition of these two features in the *IBM PC* ecosystem - including its clones - and their rationalisation from *Intel* would provide the latter the with platform leadership for the Desktop Computer, as described by Gawer and Cusumano (2002).

11.5 Industrial rationalisation: the dominance of the *Wintel* model.

Industrial rationalisation in the PC industry did not come from a single enterprise. The synthesis of the late 1980s to early 1990s that included foggy competition explorations was also one that opened the way for future explorations. *Microsoft* and *Intel* were its protagonists, though the field was not the same as when they joined the early *IBM PC* development back in 1981. The rationalisation regarded the way a computer could be a common ground for developers (now enterprises) and users (now not necessarily developers).

A first condition has been the "emancipation" of "*IBM* clones" manufacturers from *IBM*. Concretely, this meant the opening up of the interface allowing the fabrication of a computer through the combination of different components, the *BUS*. *IBM* had kept a firm control of this device allowing the connection of the (*Intel*) microprocessor to the other components of the computer. Through forming a consortium and proposing industry-wide specifications for the *Bus*, the "Gang of the nine" opened the way for a new form of enterprise collaboration. As Gawer and Cusumano (2002) contend, *Intel* practised a more open method for the *Bus* management, making sure that a roadmap based consensus was developed with the computer manufacturers, ensuring that the whole ecosystem went together with the advancement of the microprocessor (Gawer and Cusumano, 2002, 2012).

A second condition was the ability of different software enterprises to develop programs for the PC. *Microsoft* had covered this space with *MS-DOS* early on. In addition, it developed a practive of discussing with its developer community the specifications and the changes to the operating system, so that they could exploit its potential.

Thirdly, the windows graphical user interface as developed by *Microsoft* went along with the evolution of the computer as a whole, while facilitating computer use for users beyond developers. In addition, it helped software developers to pass from "programs" to "applications", proposing common design rules (Baldwin and Clark, 2000) for end user interaction. As Andy Grove, CEO of *Intel*, put it later, graphical interfaces *"brought with them a uniformity for all the application programs that were based on them: You learned one and you learned them all"* (Grove, 1997, p. 113).

Fourth, workstation applications, such as text and spreadsheet editors, have attributed to

[73] Allan (2001, p. 10/24).

the microcomputer the "workstation" identity, ensuring income from enterprise users and for providers. Ceruzzi reports that by 1991, 50% of *Microsoft* revenue came from applications, especially *Office* (Ceruzzi, 2003, p. 313). Still, the use of applications instead of a specific workstation product line, rendered the computer capable of addressing different market niches in parallel.

Gawer and Cusumano (2002) study the emergence of the PC dominant design, which became known as the *Wintel* model. According to this design and following the fundamental division between hardware and software, *Intel* built a platform for the hardware architecture, while *Microsoft* the one for the software.

The emergence of *Intel*'s leadership came along with the realisation of the need to establish a close relation with those complementors, rendering the *bus* a common ground of reference for the industry:

> *The willingness of firms to adopt this architectural change, moving from one bus (IBM's ISA) to another (Intel's PCI), and from following one firm (IBM) to another (Intel), would mark the beginning of a change in leadership in the computer industry. Until then, who designed or made different parts of the PC pretty much followed the template IBM had introduced with the original PC in 1981. But IBM failed to continue as the platform leader for all PCs being manufactured in the industry. IBM's weak and unsuccessful attempt in the mid-1980s to recapture the architectural lead by trying to get the industry to switch to yet another new proprietary bus technology, MicroChannel, may have contributed to the decline in its influence.*
>
> *Intel, by contrast, did not try to benefit from proprietary architectural interfaces for the PC. Instead, the company made sure that the new PCI specification was free and open to everyone (Gawer and Cusumano, 2002, p. 29).*

As we've seen in Section 11.3, the *bus* skeleton of the initial *Altair* platform - which became the prototype for IBM PC - had been the actual platform for third party entrepreneurs to build additional features of for this early PC. Both *IBM* and *Intel* inherited its structure. Hence, it becomes interesting from an architectural point of view to look into the properties of this feature, as related to the quest for platform leadership.

In a similar mode, *Microsoft DOS* (and later Windows) controlled the developer and the user interfaces through the PC platform. On the business level, it meant governing the terms of innovation for 4.5 million developers at the beginning of the millennium (Gawer and Cusumano, 2002, p. 135). Regarding the case of *Microsoft*, Gawer and Cusumano describe it as a case of an "open but not open" platform. Based on the initial materialisation of the operating system *CP/M* (see section 11.2.1), *MS DOS (Disk Operating System)*

These features are the predecessors of the Web-based *APIs*, that we met in Part I. As we've also seen in the case of Web 2.0, providing clear manuals on their functionalities (inputs-outputs) is a crucial issue for third party developers to be able to exploit the resources of a computer (or the network, in the case of the Web) and manufacture compatible applications. For Baldwin and Clark, these manuals constitute "visible information" on the interfaces (Baldwin and Clark, 2000, p. 290)

Gawer and Cusumano note:

> *The industry as a whole had a shared, specific problem - the insufficient capacity of the bus - but only the solution proposed by Intel was capable of solving that*

common problem while also creating a technical and industrial environment in which Intel's future innovations could find the PC system as a convenient cradle. The PCI and chip set designs introduced a local modular architecture into this part of the PC that decoupled Intel's zone of innovation from the rest of the computer (Gawer and Cusumano, 2002, p. 31).

On the other hand, Gawer and Cusumano, describing how *Microsoft* used these interfaces to exercise the other two leviers of platform leadership, "product technology" and "relationships with external complementors", as they note that knowledge on interfaces wasn't always visible to everyone.

Product technology refers to the architecture of the product, the degree of openness of the interfaces and how much information about the platform and its interfaces to disclose to outside firms (Gawer and Cusumano, 2002, p. 8, 9). Relationships with external complementors refer to how collaborative versus competitive the relationship should be between the platform leader or contender and the complementors, since achieving a consensus with their partners includes dealing with potential conflicts (Gawer and Cusumano, 2002, p. 9). They refer to a controversy within the *Microsoft* ecosystem, which illustrates the use of these two levers by the firm:

Not surprisingly, the possibility that Microsoft used these levers to create an unfair advantage in applications development was a major issue in the antitrust trial. Some competitors had access not only to Windows engineers but also to "undocumented calls" or low-level APIs found in Windows, which they might use when writing applications (Gawer and Cusumano, 2002, p. 144).

Hence, the control of interfac related information is strongly related with the attribution of design space to third parties. This information is necessary for developing extensions, or applications, a field that the platform provider may also reserve for himself, as in the case of Microsoft *Office* or *Internet Explorer* (Cusumano and Yoffie, 1998; Benkeltoum, 2008).

11.6 Cycle breaks: the Web browser and Linux

Two innovations introduced the breaking of the cycle of the PC industry.

In April 1994, Marc Andreessen, a leader of the hacker group in the University of Illinois that had developed *Mosaic*, the first "universal" browser operating for different devices, along with Jim Clark founded *Netscape*. At the end of the year, they released *Navigator*, their first product, which quickly became the preferred browser for the early Internet user community. A few months later, *Microsoft* responded by giving away its *Internet Explorer* for free along with *Windows*. Not being able to compete, *Netscape* was acquired by *America On Line* in 1998 (Cusumano and Yoffie, 1999). Some months that, *Netscape* would "liberate" its *Navigator*, turning it into *free software* (Benkeltoum, 2008).

Hence, while the browser market was still under emergence, both *Microsoft* and *Netscape* returned to a previous phase, the one of non-commercial use of software. The Web browser became thus a "feature" of the computer, while Web market emergence would deploy with "portals" during the *dot.com* bubble and in recent years with Web-based applications.

A second cycle break was that of *Linux*. As Gawer and Cusumano (2002) mention, tight control of *Application Programming Interfaces* by *Microsoft* caused a dissatisfaction of developers, who returned to a previous phase of non-commercialisation, especially in what regards software development interfaces. *Linux* would give birth to a variety of user-developer communities (Benkeltoum, 2008), while it would become a basis for developer-entrepreneurs to create Web services.

11.7 Conclusion

The invention of the PC ended up being a disruption (Christensen, 1997) for the business of enterprise computers. Still, this disruption possibility was very difficult to anticipate before the early 1990's, when the progressive rationalisation of what once was considered to be a "home" computer contributed to the deployment of the potential for enterprise use. In parallel, this rationalisation was largely based on a conception of *the computers further extension*, since it has been largely based on both the design and the sharing of developers and users' interfaces.

Early computer materialisations were undertaken by user-developer communities during the 1970s. While many of the later commercial PC concepts find their origin in this period, market emergence has not simply been an issue of innovation diffusion and adoption, as proposed by Rogers (1962, 2003). Beyond the scaling of production - itself being a crucial challenge for early developers-entrepreneurs - additional innovations were required for the PC to form an expanding market. The case of a graphical user interface, initially introduced into the market by *Apple* to be later re-framed by *Microsoft* for the entire industrial setting, has been a critical innovation for end users to be able to appropriate the new object and, thus, for diffusion to penetrate to a public other than that of its initial developers. As was also the case after the enterprise computer market emergence for IBM engineers, the question "what can our computer do?" also preoccupied *Apple* designers for years after their PC commercialisation.

Early user-developer innovations were decisive for the exploration of the PC concept. Still, user innovation norms (von Hippel and von Krogh, 2003, 2006) had to be questioned for the market to emerge. In the PC case, this rupture came "from inside" of the early UD communities and not from the enterprises. On the one hand, PC device commercialisation was widely welcomed by the community, as computers had already been commercialised before. On the other hand, software commercialisation was more controversial, since it had not been envisioned as a separate market before.

Wintel, the eventual rationalisation model, required a "platform leadership" (Gawer and Cusumano, 2002) based on a division of "scope" between the different enterprises that had emerged from early developers-entrepreneurs during foggy competition.

Phase breaks came through the questioning of the dominant design either by questioning the Wintel rationalisation (in the case of *Linux* shared programming interfaces) or by using it to question the boundaries of the PC (as in the case of the Web browser application).

Chapter 12

The invention of the Radio

Contents

12.1 Introduction

12.1.1 Chapter overview

The current chapter examines the development of the radio communications industry. Table 12.1 provides a synoptic overview of the chapter.

Section 12.2 studies the Early Materialisation phase and its input from the previous research communities. The potential of a "radio telegraph" concept was projected within the discussion of a scientific community exploring the possibilities of electromagnetic theory use. Marconi, who could be characterised as a user-developer-entrepreuner, had participated in the discussion of this milieu and the development of a device, solving the problem of long distance transmission. Using family contacts, Marconi managed to access the early British exploratory collectivity and to find his first client, for whom the early materialisation would be undertaken, the British Navy. The Early Materialisation phase would last until the World War I. Despite the efforts at a market emergence, initiated by national governments, the

Table 12.1: Chapter overview.

Section	Phase	History	Trajectories deployment	Previous phase input			Phase challenges			Figure				Role
				R	C	K	R	C	K	UD	UDE	DE	Enterprise	
12.2	Early Materialisation	From Marconi to WW I (1889-1914).	Old (Marconi): ad hoc, task-based materialisation Old (AT&T): radio telephone. New (de Forrest, Herrold): radio broadcasting.	An exploratory collectivity.	Potential for a "radio-telegraph"	Theory use.	Collectivity stabilisation, client search.	New criteria creation.	"How to"; previous criteria comparison.	✓				Development, discussion, milieu incorporation.
12.3	Market Emergence	After war.	New (UDs, RCA): radio communication.	From UDE to early adopters.	Object architecture.	Object development "how to".	Client support.	Object-task alignment.	From "ad hoc" to multiple products manufacturing.		✓			Further client search & support, production systematisation.
12.4	Foggy Competition	Until NBC, CBS.	Parallel exploration & exploitation.	Market existence.	Some trajectories to further explore.	Some development techniques, theory.	A "business ecosystem".	New trajectories emergence.	Wide, empirical potential exploration.			✓		Conceive new uses, create new markets.
12.5	Industrial Rationalisation	NBC, CBS.	Most known trajectories	Market niches	Wide empirical exploration	Dispersed know-how	Market structuring	Additional innovations	Common price/performance criteria proposition				✓	Synthesise knowledge, propose common design rules.

radio system remained confined to ad hoc settings for specific uses. Marconi exploited his patents to block others from exploiting the radio telegraph. Different innovations during this phase opened up the potential of cheap, continuous signal transmission and reception, though many developers-entrepreneurs would not manage to compete with Marconi. Radio communications would also be explored during the same period by user developer clubs, as knowledge of radio transmission had been diffused by technical magazines as well as by Marconi's Press.

After the war, the US would meet the expansion of radio user developer communities. Market Emergence (Section 12.3) would come as a surprise, when *RCA* (an alliance taking the place of American Marconi) would realise that the sales of its radio receiver 'side-products' exceeded every prediction. A business ecosystem was created, exploring the trajectory of radio broadcasting and radio programming.

Eventually, the advertisement-based economic model and the creation of broadcasting associations, along with the rationalisation of radio production and radio frequency use would provide the principles of radio broadcasting as it was eventually diffused (Section 12.5).

Disruption of the radio broadcasting industry would come from the radar, developed before the WWII.

12.2 Early Materialisation

12.2.1 Intermittent signal transmission: the "radio telegraph" concept

Marconi, a young developer-entrepreneur, managed to materialise the radio telegraph by exploiting the theory and the concepts that had been elaborated and discussed by the scientific community, as Table 12.2 briefly summarizes.

A crucial point in the history of radio waves exploitation was the invention of Hertz. Heinrich Rudolf Hertz, a German physicist, managed through his experimental device not only to prove Maxwell's theory, but to actually develop a way to produce, transmit and receive electromagnetic - or *Hertzian* - waves, in 1887[1]. Thereafter, scientists from different countries would immediately begin to produce alternative materialisations of radio wave transmitters and receivers. Still, much of this research would remain in the framework of academic circles, until Marconi created an early materialisation of a "radio telegraph" for the English Navy in the late 1890s.

Just as in the the early phase of the mainframe computer[2], as well as in the case of the personal computer[3], the new object was developed and sold on a phenomenologically

[1]Hong (2001).
[2]See Section 10.2.
[3]See Section 11.2.

Key telegraph functions	Technologies	First used by/in
Signal Generation	Spark gap	Hertz (1887)
Signal Transmission	By grounding	Marconi (1894), (telegraphy)
Signal Reception	Coherer	Branly (1890)

Table 12.2: Key architectural elements of the radio telegraph materialisation by Marconi.

"old identity" of the object ("the telegraph"), until developer communities further explored the value of the new object. Some years after Hertz's invention, the concept of a *"wireless telegraph"* was projected by the British physicist William Crookes. In his 1892 article for the *Fortnightly Review*[4], Crookes reviewed the undergone recent experimentations about radio transmission and forecast the concept of a telegraphy without wires as follows:

> *"Here, then, is revealed the bewildering possibility of telegraphy without wires, posts, cables, or any of our present costly appliances."*

Crookes claimed to having assisted in home experiments where messages were transmitted from one place to another[5] and went on to describe the requirements of a radio telegraph: the ability to generate signals of any desired wave-length, the ability to receive and respond to those messages between certain defined limits, the ability to direct the signal. While Crookes' article did not attract the interest of the general public before the activity of Marconi deploys, it did attract the interest of early inventors[6], who began experimenting.

Another British physicist, Oliver Lodge, studying lightning, formulated in the same period the theory of a possibility to produce an electromagnetic oscillation through a flash[7]. The presentation of his ideas caused a public controversy with the "electricians", led by William Preece of the Post Office in 1888[8]. Lodge further continued his work and in 1894, during a memorial lecture at the Royal Institution a few months after Hertz's death, he presented the results of his research leading to the creation of a simple though efficient way to create a radio receiver (or "coherer"). The design of the coherer (generally constituted by fine copper filings in a glass tube) had been outlined earlier, in 1891 by the British journal *The electrician* that had published the translation of a related article of the French physicist Edouard Branly describing the setting[9].

A few months later, when invited to Oxford to give the same speech, Lodge illustrated experimentally a way to transmit "sparks"[10]. However, the transmission of Morse signals was experimentally undertaken by a Russian physicist, Aleksandr Stepanovich Popov, a bit later and was published in the *Journal of the Russian Physico-Chemical Society* in January 1896[11]. Nevertheless, the challenge of the possibility of long distance radio waves transmission remained opened.

Guglielmo Marconi, an Italian 20-year-old user-developer, developed a radio telegraph in his parent's farm, managing to transmit a signal over a two mile distance. In 1886, when he was 22, after his parents failed to convince the Italian government to exploit his invention, Marconi went to London. There, using his mother's Irish-born connections in London, he was introduced to Preece of the British Post Office and subsequently to the English government[12]. Regarding the "how to", Marconi solved the long distance transmission issue, by using the same principle that had enabled the transmission of wired telegraph signals earlier: through

[4]"Some possibilities of Electricity", William Crookes, Fortnightly Review, February 1, 1892, pages 174-176.

[5]According to Garratt (1994) this early experimenter has been David Edward Hughes, in 1879 and demonstrated to a number of distinguished scientists, including Crookes in 1880. Still, the discouraging comments he received did not help him advance his exploration.

[6]Sterling and Kittross (2002a, p. 28).

[7]Garratt (1994, pp. 51-58).

[8]Ibid.

[9]Hong (2001, p. 4).

[10]Ibid. 65-66

[11]Ibid. pp.71-74

[12]Sterling and Kittross (2002b, p. 28).

grounding the one scale of the circuit to the earth - and using radio waves for the other[13]. A design that surprised the community by its simplicity, once revealed by his patent application in 1897, as many researchers had tried to solve it.

Marconi is often described as a "practician", although he was awarded the Nobel Prize for Physics in 1909, along with German physicist Karl Ferdinand Braun. Here is how Marconi described his association with radiotelegraphy in his Nobel lecture[14]:

> *In sketching the history of my association with radiotelegraphy, I might mention that I never studied physics or electrotechnics in the regular manner, although as a boy I was deeply interested in those subjects.*
>
> *I did, however, attend one course of lectures on physics under the late Professor Rosa at Livorno, and I was, I think I might say, fairly well acquainted with the publications of that time dealing with scientific subjects.*

He became aware of the research progress on the concept by attending University of Bologna courses on electromagnetic theory,[15] and by joining a community of scientists and electricians of the time, through talks and articles[16].

While not being an academic was one of the distinctive characteristics of the young Marconi in relation to the early exploration community, having an entrepreneurial orientation was the most important. In fact, he founded his enterprise (the *Wireless Telegraph & Signal Company*) in 1987, and by exploiting family contacts, he collaborated with British Navy officials and managed to have a first order in 1897. Marconi's system would be a first successful materialisation, in the sense that it provided a use value for a specific task.

Some years later he signed a full contract with the navy for a telegraph system (in 1903), having already installed more than a hundred new stations[17].

Marconi had inherited from previous researchers a theory, an exploratory collectivity and a projection of the potential of a radio telegraph. He had yet to develop a manner of how to, compared to previous criteria, stabilize a collectivity and create new criteria. A great challenge was the transantlantic transmission, which experimentally succeeded in 1901. The criteria for radio telegraphs evolved to be the distance of transmission and the quality of reception.

Still, Marconi's telegraph has been the de facto leader of a new emerging market and was thus widely discussed in the press, while he had close relationships to most of the scientists of his time, recruiting scientific radio pioneers to his enterprise[18].

Radio telegraph was further developed until WWI and is often described as a competitive technology to submarine cables. Still, it introduced a use value that was radically new, the possibility for maritime communication, either to shore stations or ship-to-ship. After the war, Marconi's technology was a standard for the domain. This success triggered the involvement of the USA. Action was taken by the US government, leading Marconi to abdicate ownership of his American division as well as the related rights[19]. Hence, a new era began in the US, with a multitude of actors taking over market exploitation.

---

[13] Hong (2001, p. 23).

[14] GUGLIELMO MARCONI Wireless telegraphic communication Nobel Lecture, December 11, 1909, http://www.nobelprize.org/nobel_prizes/physics/laureates/1909/marconi-lecture.pdf .

[15] Flichy (1995a); Aitken (1985).

[16] Flichy (1995a).

[17] Flichy (1995a)

[18] Hong (2001, p. 27).

[19] Hanson (1998).

12.2.2 Continuous signal transmission: from "radio telegraph" to "radio telephone"

In the beginnings of the 20^{th} century radio was still used for telegraphs by Marconi's stations using *Morse* signals. The passing to voice transmission required additional innovations. The invention of the "audion" enabled the transmission and reception of a continuous wave and, thus, voice and music. The key elements of the radio-telephone are outlined in Table 12.3. With the radio telephone most of the elements of radio broadcasting have been there for radio broadcasting, except for a public equipped with receivers.

Key telephone functions	**Technologies**	**First used by/in**
Signal Generation	Alternator	Fesseden & GE (1903)
	Audion	De Forest (1906)
Signal Transmission	By grounding	Marconi (1894), (telegraphy)
	By stronger signal (Alternator)	Fesseden & GE (1903)
	By signal enforcement (Audion feedback)	De Forest (1906)
Signal Reception	Coherer	Branly (1890)
	Audion	De Forest (property discovered later)

Table 12.3: Key architectural elements of the radio telephone materialisation by de Forest.

From 1902 to 1906 a set of inventions made voice transmission possible. In 1900, Lodge, a renowned British scientist, while presenting the advances in radio telegraphy in his book *"Signalling through space without wires: Being a description of the work of Hertz & his successors"* noted that they had observed in the laboratory that a telephone device could be used as a receiver of telegraph waves[20]. One year later, Reginald A. Fessenden, university teacher and a former partner of Edison[21], who was working for the *U.S. Weather Bureau*, experimentally demonstrated that voice transmission could be done. In 1902 he founded the *National Electric Signalling Company* for the development of this concept, claiming that this could be achieved by a continuous wave, on which modulations would be superimposed, instead of a wave interrupted with intermittent impositions, as in the case of the radio telegraph[22]. He had asked *General Electrics* to build an *alternator*, a big and expensive device that could however produce an almost continuous signal. In Christmas 1906 he would make the first voice and music broadcast to those in Massachusetts who had a receiver mainly ships and newspapers)[23]. *General Electrics* would thus become an actor in the radio-telephony industry for the next years.

[20] Lodge, Oliver. Signalling across space without wires: being a description of the work of Hertz and his successors, 1900, p.23.
[21] Flichy (2004, p. 163).
[22] Hilliard (2005, pp.7-8).
[23] Sterling and Kittross (2002b, p. 30-31).

In 1904 in England, the engineer Sir John A. Fleming, also a scientific advisor of Marconi, showed that a vacuum tube (a *diode*) could be used as a receiver[24]. In 1906, an American researcher who did his PhD on Hertzian waves, Lee de Forrest, would introduce a third element to the diode, creating the *audion* (or triode). While it was initially created for signal generation (*oscillator*), some years later it was discovered that it could make a suitable reception device (*detector*), as it managed to amplify the signal and enable long distance voice transmission[25].

In 1908, de Forest travelled to Paris, where he organized a spectacular presentation of the radio from the Eiffel Tower, receivable at a distance of 800 km. He made a transmission programme along with his wife, who played the piano[26]. Hence, this early materialisation of De Forest's concept to *"distribute sweet melody broadcast over the city and the sea so that in time even the mariner far out across the silent waves may hear the music of his homeland"*, as he described it in his diary[27], illustrated the potential for radio broadcasting. De Forest's invention would be proven fundamental for the further development of the radio, and despite the fact that he had patented his audion early on, he did not manage to succeed in the commercial steps. Financial problems led him to sell some of his rights to *AT&T*.

AT&T then used audio as a components of its telephone network to transmit telephone signals over long distances.

12.2.3 Silicon detectors and the UD-based exploration of radio communication

A less recognised invention was the one of Greenleaf W. Pickard. Pickard, after having studied in Harvard and in MIT, worked for AT&T from 1902 to 1906, where he experimented with radio telephony. He engaged in experimentations for constructing an alternative receiver (reportedly, he tested more than 30,000 combinations of materials for detectors). Pickard ended up with a design of a crystal detector using silicon. Unlike other detectors, this was very cheap, light and easy to fabricate. Soon, it became the basis for user-developers to join the radio waves[28].

The first radio transmission closer to radio broadcasting as we know it today was the radio station operated by Charles Herrold in San Jose, California. Herrold had set up a radio station and distributed crystal receivers to the town inhabitants in 1909, mainly in hotels, and operated it until the outbreak of WWI, when all user developer radio transmissions were prohibited. For Herrold it was a "side-project", as his main activity had been the "Herrold College". During the war, the college became one of the places were radio technicians were trained[29].

Around the same period, the first UD radio clubs began to spring up. In 1909, five adolescents created in New York one of the earliest radio clubs, the *Junior Wireless Club Ltd*, two members of which would some years later create one of the earliest radio stations[30]. Similar groups would spread across the USA, with the ones in San Francisco to become more

[24] Flichy, Hilliard, op.cit.

[25] Flichy, op.cit, Hong (2001, p. 119).

[26] Flichy (1995a, p. 107), Hijiya (1992, p. 126).

[27] Barnouw (1966, p. 20).

[28] Sterling and Kittross (2002b, p. 37); Greenleaf W. Pickard - Biography, IEEE Global History Network, http://www.ieeeghn.org/wiki/index.php/Greenleaf_W._Pickard

[29] Sterling and Kittross (2002b, p. 45), Sturgeon (2000).

[30] Walker (2001, p. 13).

important as the new technology was particularly useful to the sailors of the region. Before that, electrical user developer magazines had provided the know-how for developing wireless telegraphs at a small cost[31].

Marconi's neglected contribution: a privileged relationship with UDs

Having a second reading of the history or radio industry one can identify similarities between the cases studied on computer industry and the radio one, rethinking the role of the Marconi and his enterprise.

While aggressive towards potential competitors, American Marconi cared a lot about providing users with the necessary knowledge not only to use its equipment, but also to experiment with new uses. Thus, they founded a dedicated company, *Marconi Publishing Corporation* to undertake this challenge. It is from these editions that the organisation of radio clubs, embryos of the radio professional and business class, would be further organised.

The Wireless Age Magazine

Wireless Age magazine was first published by Marconi Publishing Corporation and specifically its subsidiary, Wireless Press, in 1913[32]. After the war, during the *RCA's* emergence, when Marconi was led to sell his US enterprise as well as the intellectual property related to it, the magazine continued being published by Wireless Press, though under the ownership of *RCA* in 1920[33], to eventually stop being issued in 1925.

The initial title of the magazine was [34]:

> *"The Wireless Age. An illustrated Monthly Magazine of RADIO COMMUNICATION".*

Incorporating the Marconigraph

By the time Wireless Age magazine was published, the user developer public had begun to obtain visibility, mainly through electrical engineering magazines[35]. The term "Marconigraph",

---

[31] See for instance the article "HOW TO CONSTRUCT AN EFFICIENT WIRELESS TELEGRAPH APPARATUS AT A SMALL COST", Frederic Collins,Scientific American Supplement, February 15, 1902, pages 21,849-21,850. Available in the following URL: http://earlyradiohistory.us/1902cons.htm

[32] Beyond Wireless Age magazine, Wireless Press also published a plead of radio user developer handbooks.

[33] (Wenaas, 2007, p. 32).

[34] Wireless Age, April 1914, Vol. 1, no7.

[35] Thomas White, editor of the Wireless Age, recalls in his book:

> *In mid-1908, Modern Electrics notified its readers that it was preparing a "Wireless Registry" of amateurs, and was planning to publish an annual national "Blue Book" listing - its July, 1908 review of the Wireless Registry listed the first ten members. A few months later, the January, 1909 issue of Modern Electrics announced its formation of a free "Wireless Association of America" - by January, 1910 the W.A.O.A., now claiming 3,000 members, was rallying its membership to fight the proposed Roberts bill, warning that "Congress threatens to pass a law licensing all amateurs". Meanwhile, in its September, 1908 issue, Electrician and Mechanic reviewed the 114 charter members of its own free organization in The Wireless Club, which promoted both national and local groups of amateurs. The magazine's first locally affiliated group, "Wireless Club 1", was formed in Chicago, Illinois, and beginning with its October, 1908 issue, a new monthly Wireless Club column featured news of interest to amateurs and experimenters.*

Thomas White, United States Early Radio History, Pioneering Amateurs (1900-1917). URL: http://earlyradiohistory.us/sec012.htm Retrieved on 2/12/2012.

not referred in the literature, was however an intermediary term, reflecting the richness of Marconi's devices, beyond the concept of the radio telegraph, for which Marconi's equipment was, at the same time, the standard. After WWI, the orientation of the magazine towards user developers of radio broadcasting became more evident, as radio broadcasting illustrations figured on most of its cover pages.

In this magazine, one could find articles on the components as well as use cases (initially related with the role of radio telegraph to sea communication). Specific tribute was given to user developers themselves, where they published their own designs improving or modifying *"Marconigraphs"*, hosted in the specific column *"From and for those who help themselves, Experimenters' Experiences"*, their comments in the column *"Comments and Criticism"* and had their questions answered in *"Queries Answered"*. While generally focussed on Marconi's platform, the magazine presented equipment from other enterprises as well, either in the form of paid advertisement or in the form of review articles.

An interesting figure for this magazine - though neglected by the scholars of the period - was Elmer Butcher, a Marconi engineer who passed to *RCA* along with other Marconi executives. Like Bill Gates for the *Alto User Group*, Butcher was responsible for supporting the community of UDEs by providing them twith he necessary knowledge. This knowledge included detailed specifications on Marconi's platform structure, a know-how that Butcher had from being a Marconi engineer[36]. However, the knowledge base that Butcher diffused to the public of the magazine through his articles was not limited to technical specifications. Butcher became a supporter and an advocate of Radio Clubs providing advice on their animation. His pamphlet on how to conduct a radio club was promoted by Wireless Magazine and appeared in multiple editions (Figure 12.1).

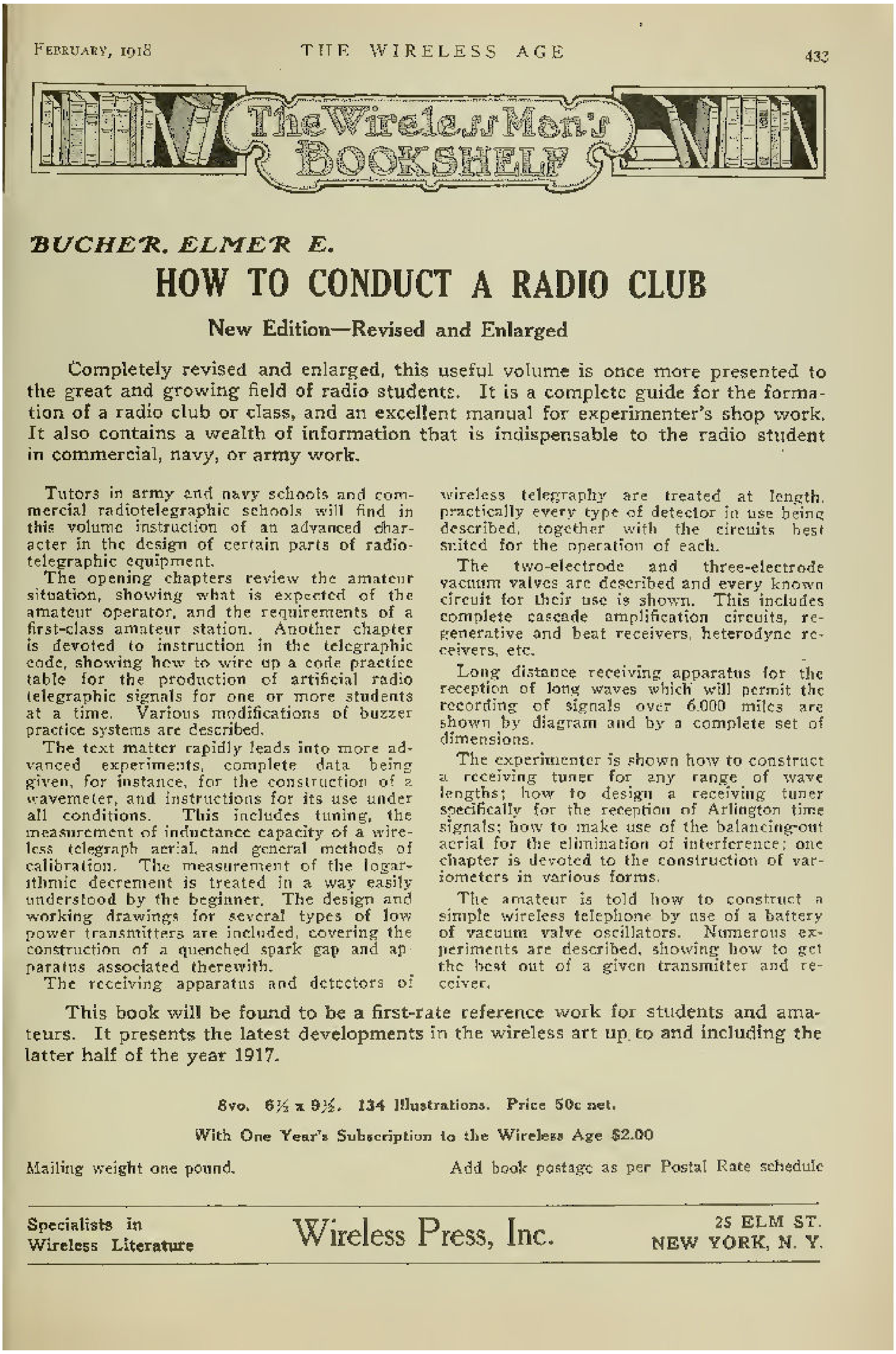
Developers-Entrepreneurs and their "parallel economy"

In fact, Butcher preferred the term *"radio experimenter"* to the one of *"amateur"*. Besides, as indicated by late 20th century institutionalist researchers' approaches for the "amateur problem" (Leblebic et al., 1991), the status of the amateurs at the time polarised public opinion, in a similar fashion to the way the term "hacker" does today. Butcher characteristically mentions in the author's note to the 1920 edition[37]:

> *It is an error of statement to characterize the whole army of wireless experimenters in the United States as mere "amateurs." Hundreds of men, young and old, who have been classed as wireless amateurs are in fact physicists of the highest calibre. They find experimentation in radio not only an instructive mode of recreation, but many of them engage in the work with serious intentions hoping thereby to contribute their mite towards the general progress of the art. That many have made good in that respect is a matter of historical record.*
>
> *The amateur wireless experimenter is no longer considered a menace. His status is now settled. He proved his worth in the recent European conflict as Government officials have publicly acknowledged. The amateur radio expert today is*

[36] In the framework in his engineering work in Marconi Wireless Telegraph Company of America, Butcher has been issued some patents for radio technologies. See for instance patents in USPTO Number 1263204, Issue date: Apr 16, 1918 and Patent number: 1257672, Issue date: Feb 26, 1918. Source: Google Patents, http://www.google.com/patents/US1263204 and http://www.google.com/patents/US1257672 . Retrieved on 2/12/2012.

[37] Butcher 1920, p. III.



FEBRUARY, 1918 THE WIRELESS AGE 433

The Wireless Man's BOOKSHELF

BUCHER, ELMER E.

HOW TO CONDUCT A RADIO CLUB

New Edition—Revised and Enlarged

Completely revised and enlarged, this useful volume is once more presented to the great and growing field of radio students. It is a complete guide for the formation of a radio club or class, and an excellent manual for experimenter's shop work. It also contains a wealth of information that is indispensable to the radio student in commercial, navy, or army work.

Tutors in army and navy schools and commercial radiotelegraphic schools will find in this volume instruction of an advanced character in the design of certain parts of radiotelegraphic equipment.

The opening chapters review the amateur situation, showing what is expected of the amateur operator, and the requirements of a first-class amateur station. Another chapter is devoted to instruction in the telegraphic code, showing how to wire up a code practice table for the production of artificial radio telegraphic signals for one or more students at a time. Various modifications of buzzer practice systems are described.

The text matter rapidly leads into more advanced experiments, complete data being given, for instance, for the construction of a wavemeter, and instructions for its use under all conditions. This includes tuning, the measurement of inductance capacity of a wireless telegraph aerial, and general methods of calibration. The measurement of the logarithmic decrement is treated in a way easily understood by the beginner. The design and working drawings for several types of low power transmitters are included, covering the construction of a quenched spark gap and apparatus associated therewith.

The receiving apparatus and detectors of wireless telegraphy are treated at length, practically every type of detector in use being described, together with the circuits best suited for the operation of each.

The two-electrode and three-electrode vacuum valves are described and every known circuit for their use is shown. This includes complete cascade amplification circuits, regenerative and beat receivers, heterodyne receivers, etc.

Long distance receiving apparatus for the reception of long waves which will permit the recording of signals over 6,000 miles are shown by diagram and by a complete set of dimensions.

The experimenter is shown how to construct a receiving tuner for any range of wave lengths; how to design a receiving tuner specifically for the reception of Arlington time signals; how to make use of the balancing-out aerial for the elimination of interference; one chapter is devoted to the construction of variometers in various forms.

The amateur is told how to construct a simple wireless telephone by use of a battery of vacuum valve oscillators. Numerous experiments are described, showing how to get the best out of a given transmitter and receiver.

This book will be found to be a first-rate reference work for students and amateurs. It presents the latest developments in the wireless art up to and including the latter half of the year 1917.

8vo. 6½ x 9½. 134 Illustrations. Price 50c net.

With One Year's Subscription to the Wireless Age $2.00

Mailing weight one pound. Add book postage as per Postal Rate schedule

Specialists in Wireless Literature — Wireless Press, Inc. — 25 ELM ST. NEW YORK, N. Y.

Figure 12.1: How to conduct a radio club. A promoting article of the book in the Wireless Age. *Source: The Wireless Age (February, 1918)*, 5, no.5, p. 433. Available online at the URL: http://archive.org/stream/n05wirelessage05nyne#page/432/mode/2up

recognized as a safeguard against possible future emergencies. He has the backing of America's foremost scientists, the good will of the Army and Navy, and the commercial company sees in him a potential engineer or expert operator.

The above mentioned testimonial goes along with the observation of Barnouw (1966) that amateur communities became the pool of radio professional recruitment, an observation which questions a view that opposes user-developers and enterprises. In fact, the

By the time of the *RCA's* emergence through government regulations, the thousands of radio user-developer-entrepreneurs were forming a parallel economy, not limited to being the first clientele for radio receiver components before the RCA's product, but also engaging in commercial activity:

All over the country amateurs - they now numbered tens of thousands - were buying parts and putting sets together. People wanted to buy these, so the amateurs sold them and got more parts and made more sets. Amateurs were also making transmitters, often using one or more parts sold by RCA. (...) The amateur-made transmitters were leaving garages and attics and were, in many cases, appearing up the buildings of newspapers, department stores, hotels. AT&T was up in arms. These transmitters were not being used for amateur purposes but - said AT&T- for telephony as a service, in many cases with a business purpose (Barnouw, 1966, p. 86).

So called "attic radio manufacturers" provided the public the components and the books to build their own radio transmitter or receiver. Moreover, the early UDE public became a pool for the emergence of radio professionals. According to Barnouw, *"many of those who started and directed radio broadcasting in the 1920 were 'amateurs' in the fertile time before the WWI"*[38].

12.3 Market Emergence: the radio receivers sales surprise

By the time of their spectacular market emergence in 1922, UDEs had expanded across the USA. In 1917 the government had issued 8500 emission licences, while the number of radio stations was estimated at 123000[39]. Table compares the situtation of radio broadcasting during market emergence and after the rationalisation of the 1930s. During the early 1920s, user-developers-entrepreneurs assured the entire process of radio broadcasting, from both device and content production to distribution. Early adopters would buy the devices and listen to the programs. After the rationalisation, discussed in the following sections, device and content production were made by enterprises, while radio was no longer used for conversation.

However, the role of UDEs is not equally valued by all scholars. Typically enough, Leblebic et al. (1991), studying the history of American broadcasting, saw amateurs as a "pirates" creating a "problem" for "stability", which the *RCA* sought to solve:

Such pirating posed serious threats to stability, since equipment sales were the only economic transactions available to participants in the earliest days of radio. In 1921 the problems were addressed by a pooling agreement between the major

[38]Barnouw (1966, p. 28).
[39]Barnouw (1966, p. 55).

Radio Elements		Market Emergence: UDE	Market Emergence: Early Adopters	Industrial Rationalisation: Manu-facturers	Industrial Rationalisation: Media	Industrial Rationalisation: Users
Device	**Transmitter**	✓	☐	✓	**$**	☐
	Receiver	✓	**$**	✓	☐	**$**
Content	**Program**	✓	listen only	☐	✓	listen only (Ads)
	Conversation	✓	☐	☐	☐	☐

Table 12.4: The state of radio broadcasting during Market Emergence and Industrial rationalisation.
Symbols:
✓ : *Production undertaken by actor,*
$: *Good purchased by actor,*
☐ : *Actor not interested in the feature.*

> *patent holders, including GE (General Electric), AT&T (American Telephone & Telegraph), and International Radio & Telegraph. Westinghouse, together with its own radio station, KDKA, joined the pool several months later (Leblebic et al., 1991, p. 341).*

Nevertheless, as we have already mentioned, the merchant value of radio broadcasting became evident for enterprises once it had been explored and expanded by numerous UDE communities. Besides, the very debate on patents had prohibited a market emergence before the war, rendering exploitation and exploration processes extremely precarious. The war ended these controversies, as all knowledge on radio was nationalised.

Still, as Flichy notes, the performance of those radio amateur broadcasts was mediocre, in terms of signal clarity and bandwidth[40]. For Flichy, in order for the radio to become a general public medium, an industrial production as well as a large scale commercialisation of receivers was necessary, an initiative that was first undertaken by *Westinghouse* [41], who had acquired Fessenden's patents after the war[42].

The *RCA (Radio Corporation of America)* came about as a result of government intervention. It was a particular kind of consortium company, putting together different electrical and telecommunication companies, *General Electric*, *AT&T*, *Westinghouse*, the latter having been previously active in the telegraphy and radio components business[43]. Still, we should note that Marconi's knowledge transfer to the *RCA* might nothave been as successful without the recruitment by the *RCA* of a number of Marconi employees, such as Sarnoff[44], a

[40] Flichy (2004, p. 169).
[41] Ibid. p. 171.
[42] Sterling and Kittross (2002b, p. 31-32).
[43] Hanson (1998).
[44] Sarnoff started in his early years as an "office boy" in American Marconi enterprise, to become a telegraph operator a few years later in the same enterprise. He became known in the US public when he was 21 years old, as he was the operator that received the SOS message from the Titanic in 1912, diffusing the information to the media and organising the rescue operation through the ships that were close[45]. During the WWI he was Commercial Manager of Marconi Telegraph Company of America (Wireless Age, July 1917, p. 707.

Marconi's executive, who moved to *RCA* and played a crucial role in its business strategy[46].

The *RCA* begun its operations following the already known trajectory and competing on the telegraph market, enforced by the fact that the new technology allowed offering lower prices[47].

As business in wireless telegraph was going well, the emergence of the radio broadcasting market was a surprise for *RCA*. In its 1922 Annual Report, *RCA* mentioned[48]:

> *At the time your Corporation was formed in 1919, for the purpose of building up a world-wide international wireless communication system, wireless telephony had not passed out of the experimental stage, and it was not at that time foreseen that the broadcasting art would ever reach the high point of popularity that it has in the last year. The engineers and scientists had anticipated the development of wireless telephony for communication purposes, but no one had visualized the phenomenal expansion of wireless telephony as used today for broadcasting.*
>
> *In the last year the number of broadcasting stations has grown from less than twenty to almost six hundred. The art itself is advancing very fast, and the ultimate effect of broadcasting upon the economic, social, religious, political, educational life of the country and the world, is comparable only with that of the discovery of printing 500 years ago.*

In fact, the number of in-home radio receivers grew from 5 thousand units in 1920 to 25 million units in 1924[49]. Figure 12.2 shows how the revenues from radio receiver sales outstripped all other commercial activities of the enterprise during 1920s. The comparison between revenues from radio sales and other business models used during the 1920s by the RCA is overwhelming. As Hanson mentions, *"merely the presence of a radio station in a town created a rush to buy radio sets".* .

For economists studying the history of the radio industry (Long, 1987, and others) this is the time when radio business begins. However, the spectacular growth of radio receiver sales indicates that when major enterprises entered, the market was already there, as also happened, for instance, a lot later when IBM entered the PC market in 1981.

[46]For an extensive review on Sarnoff's role in *RCA* and radio industry, see Bilby (1986).

[47]Hanson (1998) reports that RCA's prices for a telegraph to Norway from the USA costed 30% less that with the conventional one.

[48]Cited in Hanson (1998, p. 50).

[49]Sturgeon (2000, p. 16).

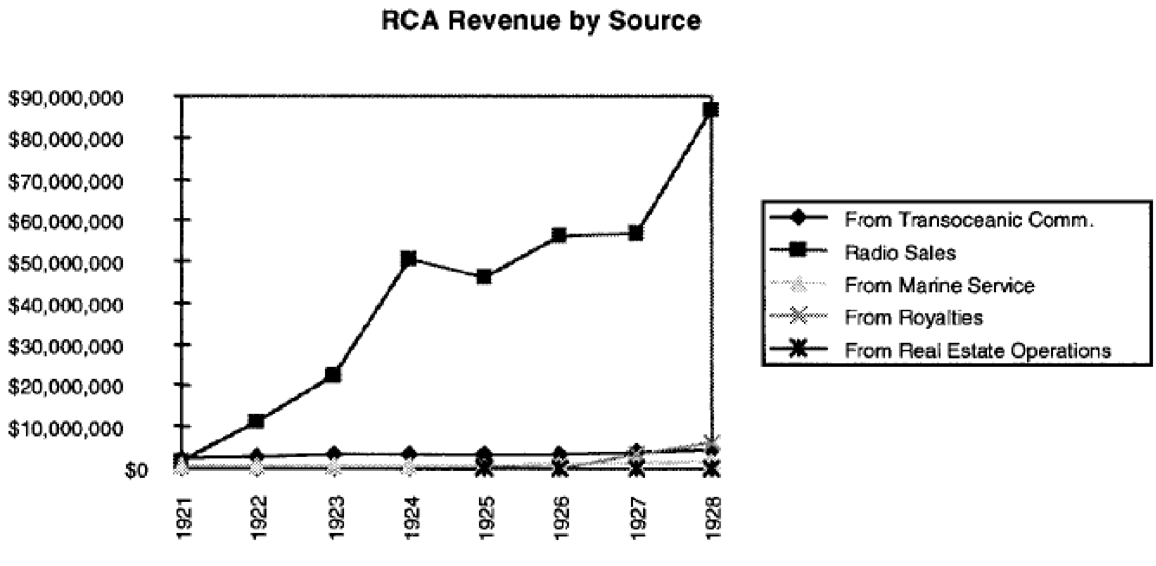


FIGURE 2
Sources of RCA Revenue: 1921–1928

Figure 12.2: RCA revenues during the 1920s. *Source:* Hanson (1998).

12.4 Foggy competition

In the *Popular Science* magazine in November 1921, a user-developer, Armstrong Perry, published an article on his experience on using radio. It was entitled *"How I Listen on the World by Radio"*[50]. Given the experience we have today about radio listening, the author's description may seem awkward:

> *It is not a passive sport, this radio game. With my transmitting apparatus I can exchange messages with neighbours five to ten miles away. And, all the radio adventures that I am having, any man or woman, boy or girl, can duplicate. I began when I was forty and only with a few dollars' worth of apparatus. With just my old mineral detector and a single phone on July the second I heard the description of the Dempsey-Carpentier prize-fight, blow by blow, given by an expert at the ringside.*

In some lines, Perry describes the use of a radio device he developed by mixing components of other devices, to enjoy things we enjoy today through the Web, though ones that have been provided by a set of different services during the previous century. Hence, he exchanged messages with distant neighbours (an activity typically enabled today by telephone or chat services), these messages were copied by other users (an activity commonly observed today in blogging or social networking services) and he listened to sports radio transmissions. The last activity is the only one that is still served by radio broadcasting, while the others are not consistent with the radio as we know it today.

During foggy competition, the division of business domains was not clear. Radio devices' manufacturing as well as radio emission production were undertaken by the same entity,

[50] "How I Listen on the World by Radio", Armstrong Perry, in *Popular Science*, Nov. 1921, Bonnier Corporation, 122 pages, p. 108.

be it UDE or enterprise. Hence this design and work division between radio devices and broadcasting content has since defined the industry, in a similar way that "hardware" and "software" divisions marked the computer industry later. Regarding radio broadcasting, different economical models were proposed for its financing: state tax to the receivers, charging the transmitter, advertisement, the latter model being eventually accepted by the US government[51].

12.5 Broadcasting rationalisation: regulation, professionalisation, engineering.

The rationalisation of the radio broadcasting business came with the foundation of two broadcasting companies, *NBC* and *CBS*. For Patrice Flichy, once *CBS* was created, the fundamental principles of the American radio broadcasting were established and remained unchanged since[52]. *Columbia Broadcasting System (CBS)* came out of the union between different local broadcasters, supported by *RCA, Westinghouse and General Electric*, in 1927 [53]. This initiative came four years after the foundation of *NBC (National Broadcasting Company)*, the latter having emerged from the AT&T initiative to build a centralised, national network, in the image of its telephony one.

Still, knowledge about radio frequencies had to be rationalised as well. A first synthesis regarding the use of radio frequencies came out of members of the radio community. Rudolf Duncan and Charles Drew, engineers engaged both in military use and in radio clubs wrote the book *"Radio telegraphy and telephony, a complete textbook for students of wireless communication"* and was first published in 1929.

A second, scientific synthesis was the work of Stanford professor, Fred Terman. Terman was largely engaged in the entrepreneurial activity in the West Coast, helping his students in their first steps[54]. In 1932 he published a Radio Engineers Handbook, which was later republished including newer knowledge. This Handbook became the reference for the protagonists of the industry. As mentioned in a later edition *"the purpose of the 'Radio Engineers' Handbook' is to provide a reference book summarizing the body of engineering knowledge that is the basis of radio and electronics"*[55]. The Handbook was written in a way in which the reader *"does not need to have an entire library within arm's reach in order to obtain the answer to most problems, and yet knows just where to go for further information"* [56].

This book included knowledge on circuit elements and theory, vacuum tubes and electronics and other radio system components, radio wave processing and transmission theory as well as measurements techniques. Overall though, it was a book written for action, drawing not only from the teaching of Stanford's lab, but from other major corporations as well.

Finally, in 1934, the US government created the Federal Communications Commission, in charge of regulating the use of radio frequencies. At that time more than 60% of the country's homes had radios in 1934, and radio sets could be found in more than 1.5 million automobiles[57].

[51] Flichy (2004, p. 173).
[52] Flichy (2004, p. 174).
[53] Barnouw (1966); Flichy (2004).
[54] His most known students were Hewlett and Packard, who created the homonym enterprise in 1935.
[55] Terman (1943, p. v).
[56] Ibid.
[57] Hilliard (2005, p. 72-73).

12.6 Conclusion

The early materialisation phase of the radio industry was largely ruled by an entrepreneur, Marconi, who managed to use the knowledge of early research exploratory collectivities and add the missing elements to commercialise the 'radio telegraph'. Marconi managed to integrate additional innovations by recruiting a network of scientists as advisors, while he made use of his patents to protect his monopoly. Additional innovations by researchers opened up the possibility for radio broadcasting, though their inventors had a hard time in commercialising them. UDEs, though, managed to advance radio broadcasting, based on the knowledge their magazines provided, once cheaper ways to transmit and receive radio signals were invented.

After the war, the emergence of the radio market came as a surprise. Initially, radio has been introduced as a disruption (Christensen, 1997) of the telegraph market. However, as *RCA* came to realize, the radio broadcasting market was a lot more profitable than the radio telegraph one. Radio has been diffused as an innovation (Rogers, 2003) by UDEs, though the nature of broadcasting did not imply a business model for radio emission. Users had been using radio in an horizontal way (von Hippel and Katz, 2002), while a division of tasks according to use-related and manufacturer related knowledge (von Hippel, 1990) was not yet the case.

The foggy competition phase consisted in the exploitation of radio broadcasting while still exploring ways to produce a program and benefit from it. Early user-developers became the pool where entrepreneurs and the professionals of the new field emerged from. Eventually, rationalisation came by the division of manufacturing, broadcasting and listening - process which required both a knowledge synthesis and a business regulation.

Chapter 13

Conclusion of Part II

While in the first part of my research, user-developer-entrepreneurs (UDEs) appeared to be a peculiarity of Web-based application development, as they did not enter the frameworks of analysis of either user or manufacturer innovations paradigms (von Hippel and von Krogh, 2003, 2006; Raasch and von Hippel, 2012), the study of the history of different industrial settings has shown that the activity of those figures is not peculiar to the field of Web services, as phenomenon-based research suggests.

The "problematization" of UDEs activity though, constructed in the first part, allowed me to revisit the work of historians on three industries (enterprise computer, personal computer and radio), revealing a different reality of industrial emergence than the one already described by prominent scholars such as Rogers (1962), Christensen (1997) and von Hippel (2005).

Still, the fact that other industries had also met such actors does not render my findings irrelevant to the Web service innovation phenomenon. On the contrary, it adds new insights to the way such actors contribute in the parallel exploration and exploitation of the industrial potential, and it brings in new knowledge on pro-active methods which enterprises of the field can use to face potential disruptions. By using such methods, disruption is not solely a "threat" but it is at the same time an opportunity for established actors.

13.1 The role of UDEs in different phases of industrial development

The study of the emergence and the development of three industrial settings has led to the identification of the possibility to analyse industrial development in the following phases: *Early Materialisation*, *Market Emergence*, *Foggy Competition* and *Industrial Rationalisation*.

The *Early Materialisation* phase consists in the construction of the first objects that make use of a new theory illustrating a value potential. Such objects are created by user-developers (UDs), usually close to academic circles, often Ph.Ds (e.g. Mauchly and Eckert), though not necessarily (eg Marconi). These materialisations are more than prototypes, as they are able to satisfy the requirements of a particular use. We can distinguish two types of materialisations, those drawing on an old concept (such as a "calculator" in the case of *ENIAC*) and those drawing on a new concept (such as a "computer" in the case of *UNIVAC*). An "intimate circle" of user-developers is framing this process, discussing the underpinning theories and exploring the means and the goals of the materialisation (such as the *ACM* in the case of *ENIAC* or the *Homebrew Computer Club* in the case of *Altair*). While old concepts are compared to previous performance criteria, proposing a "better embodiment", new concepts introduce new criteria projecting a "new dream". In both cases, the materialisation may be financed by its single user, such as an institution or an organisation (such as the US Army in the case of *ENIAC* or *INRA* in the case of *Micral*).

User-developers don't only have to possess use-context related "sticky information", as theorized in the case of lead users (von Hippel, 1995, 2005), but they also have to possess technology-related "sticky information", traditionally residing within manufacturer's expertise (von Hippel, 1995, 2005). At this level, there is no market, not even insignificant ones, so that such innovations could be characterised as disruptions according to Christensen (1997). Still, even within this very early phase, one can distinguish more than one trajectory, the trajectory based on an older concept proceeding the trajectory based on a newer concept. A design tactic can be identified in this phase, to begin with task-specific innovations and advance toward more abstract and generic materialisations - as did Eckert and Mauchly - as the cost for these first materialisations should be financed, despite the fact that it is difficult to estimate beforehand.

Market emergence comes when an "offer" is formed and the search of potential clients by user-developer-entrepreneurs begins. Typically enough, the delivery of the first *UNIVAC* or the presentation of *Apple II* in a computer fair triggered the interest of both potential buyers and potential manufacturers, the latter being established actors of similar business fields. It can be the same actors of the first phase that continue in the second one (e.g. Eckert and Mauchly) or others (e.g. the passing from *MITS* to *Apple*). Still, it is the same intimate collectivity of user developers that discusses and follows up the developments of the new object, much like user innovation communities are described (von Hippel and Katz, 2002). Yet, once the market potential appears, those early UD collectivities will become an embryo for the potential clients, manufacturers and professionals of the field (e.g. Radio professionals who emerged from the early materialisation phase). User-developer- entrepreneurs continue to have as a point of reference their personal view of the object, while the separation between UDE and DE happens when satisfying potential clients becomes a priority (this typically was been the case with the passage from *Apple II* to *Macintosh*).

However, one of the contributions of this research is the proposition that market emergence is not equal to a development process where market growth is the only parameter. In fact, until a new rationalisation is proposed by a synthesis which "connects the dots", a great number of potential directions are explored in parallel to the exploitation of the new product. The fact that IBM engineers admitted during the 1950s that they did not yet know what a computer was, is very illustrative of this fact. Entrepreneurs and enterprises try to explore neighbouring directions, though their "jump" may be bigger than estimated, or they can be lead to disruptive innovations (Christensen, 1997) for which the market remains to be identified. Thus, "decision making" during this period is very uncertain, not only because of the risks that reside in the challenges, but because of the impossibility of risk estimation. This *Foggy Competition* phase ends with the provision of a new synthesis, establishing a dominant design (Utterback and Abernathy, 1975), on the basis of which market diversity and differentiation can be deployed using the same performance, efficiency and cost criteria. Still, new "jumps", can "turn the clock back" to the *Early Materialisation* phase, where UDs play an important role, as did *DEC* with its *PDP*, opening this way a new cycle, not included in the rationalisation synthesis.

13.2 Lessons from Enterprises that took UDEs into consideration

Throughout the history of the three industries studied, some enterprises did take into account the role of user-developers. This dimension of their activity is largely underestimated even by scholars who study these very cases. Hence, the Marconi Corporation helped in the emergence of radio UDs, by providing through a specific magazine, the necessary knowledge to innovate on devices initially designed for radio telegraph systems. Mauchly and Eckert envisioned early on the provision of tools and support to the early community of *UNIVAC* users, though this move was not further continued by *Remington Rand*. IBM, while often portrayed as a company hostile to "hackers", has had a relationship with user-developers and developers-entrepreneurs since the 1960s, though in a more formalised way: either by providing support as an after sales service, or by leasing its machines to developer-entrepreneurs. Similarly, while *Macintosh* has been a "less open" computer than the *Apple I*, the latter addressed to "hackers", it did invent the *USB*, opening up the opportunity for a PC peripherals market.

Those companies that did take into account the UDE dimension managed to survive the Foggy Competition phase. Typically, IBM took the leasing experience into account during its rationalisation, while Apple managed to survive the competition by IBM. Moreover, the designs as such managed to survive in many cases, even though their initial creators didn't. Thus, many of the elements of the early *Altair* computer are to be found in the first IBM PC, while von Neumann's *"First Draft of a Report on the EDVAC"* has been used as a reference for computer architecture during *Foggy Competition*. Overall, the exploration by UDEs can be beneficial for enterprises, though it has an independent value for industrial development, as it may contribute to the formulation of what a dominant design can end up being.

Overall, those companies while acting in different times and cultural contexts, had something in common: they provided the necessary knowledge for UDEs to further explore the potential of their products, either for own use or for market opportunities. Specific "institutions" were previewed for this activity, including specialised press, developer meetings or conferences, mail lists and clubs fostering the creation of "intimate milieus" of users, sharing their passion, knowledge and ideas on a given technology and its future.

Conclusion de la deuxième partie

Bien que dans la première partie de notre étude, les usagers-développeurs-entrepreneurs apparaissent comme une particularité du champ du développement d'applications Web, comparés aux modèles distingués par Raasch et von Hippel (2012), l'étude de l'histoire des trois cadres industriels entreprise dans cette deuxième partie, a montré que le *modus operandi* en question ne constitue pas une spécificité du Web.

La relecture des histoires des cadres industriels de l'ordinateur entreprise, de l'ordinateur personnel et de la radio sur la base d'une problématisation à partir du *modus operandi* spécifique, nous a permit de réviser le travail des historiens dans ces domaines en mettant en lumière les aspects les moins développés. Il en résulte qu'une discussion concernant les approches du développement industriel faites par Rogers (1962), Christensen (1997) et von Hippel (2005) est possible.

Bien que les aboutissements de cette recherche historique suggèrent que le modus en question ne signifie pas une spécificité du Web, ils ne rendent pas pour autant les résultats de la première partie moins pertinents sur le champ du Web. En revanche, ils ajoutent une perspective dynamique en ce qui concerne la façon dont les acteurs identifiés contribuent à l'exploration et à l'exploitation en parallèle d'un potentiel industriel, et ajoutent du savoir sur les manières dont les entreprises peuvent agir en amont, et faire face à des « *disruptions»* (Christensen, 1997) éventuelles : l'histoire montre que le même phénomène pouvant mettre en cause les entreprises établies peut devenir un avantage compétitif, quand il est mis dans une dynamique d'innovation.

Le rôle des UDEs dans les différentes phases du développement industriel

L'étude historique de la partie actuelle a conduit à l'identification de la possibilité de décrire l'émergence d'un cadre industriel selon les phases suivantes: *« Matérialisation précoce », « Émergence du marché », « Compétition dans le brouillard »* et *« Rationalisation industrielle ».*

La phase de la *« Matérialisation précoce »* se caractérise par la construction des premiers objets utilisant une nouvelle théorie et illustrant un nouveau potentiel de valeur. Tels objets sont construits par des usagers-développeurs (UDs), le plus souvent proches à des milieux académiques, pouvant être des chercheurs (eg. Mauchly et Eckert) ou pas (eg. Marconi). Ces matérialisations sont plus que des prototypes, à partir du moment où elles peuvent satisfaire aux exigences d'un usage spécifique. On peut en distinguer deux types : celles conçues à la base d'un concept ancien (comme le *calculateur* dans le cas de l'*ENIAC),* et celles conçues à la base d'un nouveau concept (comme le *computer* dans le cas de l'*UNIVAC).* Un « cercle intime » des UDs, encadrant ce processus exploratoire, étudie les théories sous-jacentes et explore les moyens et les objectifs possibles de ces matérialisations (comme la *ACM* dans le cas de l'*ENIAC* ou le *Homebrew Computer Club* dans le cas de l'*Altair*). Tandis que les matérialisations à la base des anciens concepts sont comparées à des critères de performance préexistantes, en proposant une « meulière incarnation », celles à la base de nouveaux concepts, en introduisent des nouvelles, en projetant un « nouveau rêve ». Dans les deux cas, c'est souvent une institution ou une organisation qui finance leur construction pour son propre usage (comme l'armée américaine dans le cas de l'*ENIAC* ou *INRA* dans le cas du *Micral*).

Durant cette phase, les UDs n'ont pas que du savoir lié au contexte d'usage (*« use-context sticky information »*) comme c'est le cas pour les usagers (von Hippel, 1995, 2005), mais ils ont également du savoir technologique (*« technology related sticky information »*), attribué par la littérature de l'innovation par les usagers aux entreprises. À ce niveau très en avance du développement industriel, il n'y a pas de marché du tout, donc il ne s'agit pas encore des *« disruptions »* (Christensen, 1997). Cependant, même dans cette phase en amont, on parle déjà de plusieurs trajectoires qui se dessinent, et non pas d'une seule qui reste à être diffusée. Une tactique de conception identifiée lors de cette phase consiste à démarrer le processus de développement ayant comme objective des matérialisations intermédiaires et spécifiques à des tâches dont la valeur est déjà reconnue, et à avancer par la suite à la matérialisation des concepts plus abstraits et génériques, comme on l'a vu dans le cas de Eckert et Mauchly. Une telle tactique permet de répondre au problème du financement de l'exploration, le coût pouvant être couvert par des produits intermédiaires.

L' *« Émergence du marché »* arrive une fois qu'une offre est formulée par des UDEs, durant la recherche de clients potentiels. De façon assez caractéristique, la livraison du premier *UNIVAC* ou l'exhibition de l'*Apple II* dans un salon d'ordinateurs ont déclenché l'intérêt à la fois des acheteurs potentiels et des potentiels fabricants, ces derniers étant des acteurs établis dans des marchés proches. Il se peut que les acteurs de la première phase fassent la transition à la deuxième (comme Eckert et Mauchly) ou qu'elle se fasse par de nouveaux acteurs (comme le passage de *MITS* à *Apple*). Cependant, c'est la même collectivité des UDs qui discute et suit le développement de l'objet nouveau, de manière très semblable à la celle décrite pour le cas des communautés d'usagers innovateurs (von Hippel et Katz, 2002). Cela dit, une fois que le potentiel d'un marché apparaît, ces communautés d'UD précoces deviendront l'embryon des futurs clients, fabricants et professionnels (comme par exemple les professionnels de la radio, qui ont émergé des collectivités de la phase de Matérialisation précoce). Par la suite, la vision personnelle du potentiel de l'objet demeure une référence pour l'activité novatrice des UDEs, tandis que la séparation entre UDEs et DEs prend lieu une fois que la satisfaction des exigences des clients devienne prioritaire (comme dans le cas typique du passage de l'*Apple II* au *Macintosh).*

Néanmoins, une des propositions de cette recherche est que l'émergence d'un marché n'équivaut pas à un processus de développement caractérisé exclusivement par la taille du marché et de la production. En effet, avant qu'une nouvelle rationalisation soit proposée en « connectant les points », une pléthore de directions d'innovations potentielles est explorée, en parallèle de l'exploitation des nouveaux produits. Le fait que les ingénieurs d'*IBM* reconnaissaient dans les années 1950 ne pas savoir encore ce qu'est un ordinateur, illustre bien ce propos. Des entrepreneurs et des entreprises essaient d'explorer des directions voisines, bien que le « saut » d'une direction à l'autre puisse s'avérer plus grand que prévu, pouvant également conduire à des « disruptions » (Christensen, 1997), pour lesquelles un marché reste à identifier. Par conséquent, la prise des décisions au long de cette phase demeure incertaine, non seulement à cause des risques compris dans les enjeux technologiques et commerciaux, mais surtout à cause de l'incapacité de spécifier les risques à rencontrer en amont. La phase de la *« Compétition dans le brouillard »* finit avec la conception et la mise en place d'une nouvelle synthèse, établissant un « *dominant design »* (Utterback et Abernathy, 1975), sur la base duquel la segmentation et la différentiation du marché puissent être déployées, en faisant référence à des critères de performance, d'efficacité et de prix comparables. Cependant, même dans cette phase de maturité, de nouveaux « *sauts* » peuvent prendre lieu, en « *retournant l'horloge à l'arrière* », à la phase de la *« Matérialisation précoce »* , où des UDs jouent de nouveau un rôle important, comme l'a fait *DEC* avec son *PDP*, ouvrant ainsi un nouveau cycle, basés sur un concept qui n'a pas été compris dans la synthèse rationalisante.

Ce qu'on apprend par les entreprises qui ont pris en compte les UDEs

Au long des histoires des trois cadres industriels étudiés, quelques entreprises ont pris en considération dans leur action les figures des usagers-développeurs-entrepreneurs. Cette dimension de leur activité est largement sous-estimée par des chercheurs qui ont étudié ces cas. Comme on l'a vu, la *Marconi Corporation* a contribué à l'émergence des UDs de la radio, en éditant un journal spécialisé fournissant à ses lecteurs le savoir nécessaire pour innover sur la base des appareils initialement conçus en tant que télégraphes radio. Par ailleurs, Mauchly et Eckert on envisagé très en amont la mise en disposition aux premiers usagers-développeurs des ordinateurs; des outils d'extension des appareils, même si cette démarche n'a pas été poursuivie par *Remington Rand*, rachetant leur start-up. *IBM*, même s'il s'agit d'une entreprise souvent décrite comme hostile aux « *hackers* », a eu en fait des rapports proches avec des UDs et des DEs depuis les années 1960, bien que ce rapport soit plus formel : il consistait soit en la provision de support en tant que services après vente, soit au *leasing* d'ordinateurs à des développeurs - entrepreneurs. De façon similaire, bien que le *Macintosh* soit un ordinateur « moins ouvert » que l'*Apple I*, ce dernier s'adressant à des *hackers,* le *Macintosh* a en fait introduit le précurseur de la clef *USB,* ouvrant ainsi l'opportunité aux DEs de construire un marché de périphériques.

Ces entreprises qui ont pris en compte la dimension des UDEs ont réussi à survivre la phase de la *Compétition dans le Brouillard.* Typiquement, *IBM* a pris en compte l'expérience du *leasing* pour la rationalisation de l'industrie par le *System 360*, tandis que *Apple* a réussi à survivre à la compétition d'*IBM* en se posant à des DEs. En outre, les *designs* de leurs produits ont réussi à avoir une vie longue, souvent plus grande de celle de leurs concepteurs. Ainsi, plusieurs éléments de l'ordinateur *Altair* peuvent être retrouvés au premier *IBM PC,* tandis que le *First Draft of a Report on the EDVAC* de von Neumann fut utilisé comme référence architecturale pour l'ensemble des acteurs de la phase de la *Compétition dans le Brouillard* de l'ordinateur entreprise.

Dans l'ensemble, ces entreprises, bien que leur activité se soit étendue dans des périodes historiques et des contextes culturels différents, ont eu une caractéristique en commun : elles ont envisagé la provision du savoir nécessaire aux UDEs, afin que ces derniers aient les conditions de possibilité d'une exploration plus approfondie des produits initiaux, soit à leur propre usage, soit à des fins commerciaux. Des « institutions » spécifiques ont été conçues à ce propos, y compris de la presse spécialisée, des listes de diffusion, des conférences de développeurs ou des réunions, et des clubs encourageant l'émergence des lieux d'intimité parmi les UDEs, permettant d'y partager leur passion, leurs idées et leur savoir sur l'avenir d'une technologie donnée.

Part III

Harnessing UDE activity. Exploration and exploitation methods

Gérer l'activité des UDEs. Méthodes d'exploration et d'exploitation. Introduction à la Partie III

La première partie a exploré le *modus operandi* du développement des applications Web, largement basé sur la capacité individuelle d'innovation des développeurs, permise par un ensemble de technologies disposées soit par de fournisseurs de services, soit par des communautés *open source.* En ce qui concerne la distinction entre les paradigmes d'innovation par l'usager et par le industriel (Raasch et von Hippel, 2012), les normes d'action (Argyris et Schon, 1978) de ces développeurs peuvent être décrites par le modèle d'innovation par les usagers (modèle « collectif-privatif » selon von Hippel et von Krogh), dans le cas des usagers-développeurs (UDs), par un modèle d'investissement privé dans le cas des développeurs-entrepreneurs (DEs), et par un modèle entre les deux dans le cas des usagers-développeurs-entrepreneurs (UDEs).

La deuxième partie a placé ces trois figures d'acteurs (UD, UDE et DE) dans la dynamique du développement industriel, à la base de l'étude des l'histoire des trois cadres industriels, celui de l'ordinateur entreprise, celui de l'ordinateur personnel et celui de la radio. Cette étude a suggéré que les figures identifiées ne sont pas spécifiques au contexte du Web, et qu'elles jouent un rôle important dans les phases en amont d'une rationalisation industrielle dans les cadres industriels où elles sont retrouvées.

Bien que l'exploration de la Partie I ait utilisé davantage des entrées au terrain privilégiant les aspects cognitives, en étudiant les conditions de possibilité d'action de ces acteurs dans le champ particulier du Web par des biais comme les technologies particulières et les « livres de cuisine », la Partie II a montré également l'importance de dispositifs de conversation sur les technologies, les usages et les marchés entre les acteurs d'un champ industriel émergent. Cette troisième partie rentre au champ du développement des applications Web et, tout en demeurant au même niveau d'analyse, elle approfondit l'exploration des dispositifs de conversation particulières, en proposant trois méthodes de gestion du rapport entre les UDEs, ainsi qu'entre les UDEs et les entreprises.

Nous allons donc explorer les conditions et les manières possibles de faire face à ces interactions, pouvant servir aux besoins d'exploration et d'exploitation en parallèle des nouvelles technologies, leurs usages et leurs marchés. Le Chapitre 15 pose la question de la possibilité d'émergence des communautés et des réseaux de développeurs, et explore un dispositif de conversation, les *« Barcamps »,* qui crée les conditions à la réalisation de ce but. Par la suite, les deux autres chapitres proposent quelques réponses à la question plus spécifique de l'exploitation de l'activité des UDEs par les entreprises. Le Chapitre 16 aborde la question de l'usage des UDEs pour l'exploration du potentiel d'une nouvelle technologie, en étudiant un mode original d'organisation de l'activité d'exploration, les *« Hackathons ».* Enfin, le

Chapitre 17 aborde la question de l'usage des UDEs pour l'exploitation du potentiel d'une technologie, en étudiant une manière permettant que les UDEs discutent leurs problèmes avec les entreprises, les forums de support de développeurs.

Le Chapitre 14 qui suit juste après, examine quelques concepts théoriques qui seront utiles par la suite, et fournit une vue d'ensemble de cette dernière partie. La Section 14.2 discutera l'ambigüité que l'on retrouve dans la littérature de gestion quant à l'usage des notions de « réseaux » et de « communautés », surtout quand elles sont abordées du point de vue de l'innovation. Cette section suggère une synthèse permettant a) la distinction des deux notions et b) l'exploration des conditions de possibilité de leur émergence. Elle conclue à la proposition d'un cadre d'analyse d'interactions informelles unifié, qui place la conversation et l'action collective entre les deux notions. Ensuite, la Section 14.3 décrira l'approche méthodologique à utiliser dans la présente partie. En utilisant une stratégie de *« phenomenon-based research »* von Krogh et al. (2012), nous étudierons les trois dispositifs sur la base de deux méthodes: la participation observante, déjà utilisée dans la première partie, et l'*analyse de monuments d'interactions*. Enfin, la Section 14.4 fournira une vue d'ensemble de cette partie.

Chapter 14

Part introduction

Contents

14.1 Introduction

Part I explored the innovation capacity of individual developers within the framework of Web services, enabled by a set of different technologies offered either by service providers or by open source communities. Regarding the distinction between user and manufacturer innovation paradigms (Raasch and von Hippel, 2012), the action norms (Argyris and Schon, 1978) of those developers can be described by the "private-collective" model (von Hippel and von Krogh, 2003, 2006), in the case of user-developers (UDs), by the "private investment model" (von Hippel and von Krogh, 2003, 2006), in the case of developer-entrepreneurs (DEs) or by a mix of the two, in the case of user-developer-entrepreneurs (UDEs).

Part II positioned these three actor figures (UDs, UDEs and DEs) in the dynamics of industrial development through the study of the history of three industrial settings, 1) the enterprise computer, 2) the personal computer and 3) the radio. The outcome of this study was the suggestion that these figures play an important role particularly in phases preceding market emergence or industrial rationalisation.

While Part I focussed mainly on the cognitive conditions (technologies and "cookbooks") for the possibility of the existence of those actors in the particular field of Web services development, Part II has additionally shown the importance of settings enabling a conversation amongst UDEs, as well as between UDEs and enterprises, about technologies, uses and markets of an emerging industrial field. The current part returns to the field of Web services development and, while keeping the same level of analysis, it now focusses on the particular settings enabling this conversation, proposing three different configurations.

The current part explores the conditions and the possible ways of interaction between service providers and user-developer-entrepreneurs, serving the need for parallel exploration and exploitation of new technologies, their uses and their markets. Chapter 15 addresses the question of the possibility of emergence of developer communities and networks and explores a conversational setting, the *Barcamps*, which can provide this possibility. The following chapters provide some responses to the question of how enterprises can harness communities and networks for innovation. Chapter 16 addresses the question of the use of developers-entrepreneurs for new platform potential exploration, exploring an original mode of exploratory action organisation, the *Hackatons*. Finally, Chapter 17 addresses the question of the use of developer-entrepreneurs for platform potential exploitation, exploring a way in which providers and developers can discuss the problems the latter meet and the ways to solve them, through developer support forums.

The current chapter reviews some theoretical concepts that will be useful for the chapters to follow and provides a part overview. Section 14.2 will discuss the ambiguity existing in management literature regarding the use of the "community" and "networks" notions, specifically when addressed by an innovation perspective. This section suggests a synthesis which allows a) the distinction between the two notions and b) the exploration of the conditions of their possibility of emergence. It concludes with a joint analytical framework which puts collective action and conversation in-between the two notions. Section 14.3 will describe the methodological approach that is to be used in the current part. Using a phenomenon-based research strategy (von Krogh et al., 2012), I will study three different fields based on two different methodologies: observant participation, already used in the Part I, and "monument analysis", which will be discussed in this section as well. Finally, Section 14.4 will provide an overview of the current part.

14.2 Literature review: how do informal collectivities emerge and how can they be harnessed?

This section initially reviews the use of the terms "networks" and "communities" by the Open and User innovation literature. Identifying an ambiguity in this use, it then proceeds to a second-order review of the literature specifically addressing these terms. It concludes with the proposition of a joint framework of analysis, valuing the notions of collective action and conversation in both distinguishing and using the two terms.

14.2.1 Innovation beyond firm boundaries : the blurred frontiers between networks and communities

Regarding innovation beyond the frameworks of a specific firm, two "schools" have been the reference point for management scholars. On the one hand, the "Open Innovation" (Chesbrough, 2003b; Chesbrough et al., 2006) framework, on the other hand, the "User Innovation" (von Hippel, 1976, 2005) one. For both, the questions of how networks or communities emerge as well as how enterprises may harness the benefits of their activity remain largely open. Both approaches value the role of "communities" and "networks" for the design and the diffusion of innovative goods. Still, both sets of literature use these two notions depending on the context without really making a distinction between the underpinning action norms corresponding to each. Furthermore, they often assume that communities and networks already exist, while generally recognising that the relationship between enterprises and organisations on the one hand, and communities and networks on the other are far from being clear.

Thus, the following questions remain open:

1. How can knowledge and ideas be informally shared when no networks or communities exist?

2. How can enterprises harness the benefits of user-developer-entrepreneurs' activity to

   - explore the potential of their own service and
   - exploit the potential of their own service.

The first question will be explored in chapter 15, while the second will be explored in chapters 16 and 17 for the specific case of developers in the field of Web services.

Open Innovation and the value networks concept exploration

In a debate animated by *Technovation* journal on the question *Is open innovation a field of study or a communication barrier to theory development?* Chesbrough's approach is challenged by the question on whether or not it is different to the supply chain management approach (von Hippel, 2010; von Krogh, 2011), were inwards and outwards knowledge exchanges also occur. Consequently this critic challenges whether or not an "open innovation era" is something new. In parallel, Gassmann, Enkel and Chesbrough stressed that one of the challenges lies in the fact that *"the operational functioning of open innovation depends on firms' ability to manage decentralized innovation processes and often includes participants who are not even on the company's payroll"* (Gassmann et al., 2010, p. 7), for which the management modes remain to be explored. Therefore, a current of literature has focussed in further exploring the concept of a *"value network"* (Chesbrough, 2003a) that is not a supply chain.

Simard and West (2006) tackle the issue of the role of the firm's external context, building upon prior research on networks. They formulate thus the concept of *"knowledge networks"*, to describe contexts *"such as the fabric of Silicon Valley"* (Simard and West, 2006, p. 220). Of course, the Silicon Valley innovation phenomena have been studied by many scholars, as they are based on numerous, historically constructed, parameters (Saxenian, 1994; Lewis, 2000a; Sturgeon, 2000; Saxenian, 2000; Lecuyer, 2006, and others). Still, the knowledge network concept is proposed as a way to address relationships beyond the enterprise boundaries, either with other enterprises or with individuals. Of course, this concept does not refer to networks within a knowledge field, such as semantic networks. The term is borrowed by Hansen (2002), who claimed that *"knowledge transfers and synergies in multiunit firms should pursue new perspectives that combine the concepts of network connections and relatedness in knowledge content"*(Hansen, 2002, p. 232). Thus, it concerns the research into a combination between network connections and knowledge fields. Building on the work of (Granovetter, 1983) advocating for "the strength of weak ties", Simard and West (2006) argue that 'wide' ties privilege innovation, as they provide access to diversified knowledge, though they imply difficulties in coordination, when with other firms, or to compare to the firm's activities.

Always in the same question, West and Lakhani (2008) in their article *"Getting clear about communities in open innovation"* note that there is a need for a better definition of the community construct in open innovation and operate a first typology. Their typology begins with the insightful question *"communities of what?"*. However, their reply rather refers to an answer to the question *"communities of whom?"*, as they focus on the subject (the actors participating in a community) and not the object of a community (e.g. the *"knowledge domain"* and the *"practice"* for the communities of practice (Wenger et al., 2002)). Thus, the authors propose a framework for community analysis distinguishing different levels in relationship to the ties between communities and enterprises: a) communities in which enterprises participate as simple members, b) communities which are sponsored by an enterprise, c) interactions between the members belonging in the same community (member types and organisational forms).

While the issue of communities and networks puzzles open innovation scholars, user innovation literature also shares this problematization.

User innovation and the mix of communities and networks

Von Hippel supported that user innovation can be undertaken by horizontal user networks. By user network he referred to *"user nodes interconnected by information transfer links which may involve face-to-face, electronic or any other form of communication. User networks can exist within the boundaries of a membership group but need not"* (von Hippel, 2007, p. 295). The attributes of such networks were defined as follows: *"(1) at least some users have sufficient incentive to innovate, (2) at least some users have an incentive to voluntarily reveal information sufficient to enable others to reproduce their innovations, and (3) user-self production can compete with commercial production and distribution".* In the cases of such networks, user innovation can be deployed entirely independently of manufacturers. Von Hippel contrasts the concept of user networks to the one of user communities, the latter characterised by a sense of belonging, and calls for further research on the specificities of user communities, as opposed to user networks. He gives the example of the open source software *Apache* server user network as a typical case of effective competition of users to manufacturers.

This emphasis on the concept of networks rather than communities seems to contradict previous conceptualisations of user innovation. Lakhani and von Hippel (2003) studying the *Apache* server user community showed how the sense of belonging in a community animates the collective action of developers. The importance of a community as the basis of sharing and its related processes have been the object of interest in a great range of user innovation studies (von Krogh et al., 2003b; Franke and Shah, 2003; Auray, 2004; Baldwin et al., 2006; Osterloh and Rota, 2007; Benkeltoum, 2008, and others).

In fact, despite the great amount of research on user innovation scholars have undertaken over recent decades, literature on user innovation does not share a common understanding of what a community or a network is, how it emerges and what its relationship to enterprises is, even though user organisation has a fundamental position in the common understanding. In addition, West et al. (2006), in their article *"Open innovation: a research agenda"* evoke the issue of whether or not there exist *"factors that explain the differences in the ability of firms to utilize user-generated external innovations"* (West et al., 2006, p. 289).

More generally, business studies value differently relationships between the enterprises and external developers, according to the emphasis given by the researcher to the object produced and marketed. Studies emphasizing the importance of the attributes of the object in common (e.g. a platform), tend to highlight the cognitive aspects, while studies interested in the diffusion aspects tend to highlight the relational ones. A typical example is the open question of platform organisation. On the one hand, Gawer (2010b) observes a "Chinese Wall" between the developers of the core of a platform and the developers of the peripheral elements. Benavent (2011) on the other hand, addressing the issue from a marketing perspective, observes "no Chinese Walls", but rather porous boundaries between an enterprise and its clients.

Hence, in order to be able to use the terms of user networks and communities (in our particular case referring to developers), a clarification of these two notions should be attempted.

14.2.2 A return to the basics: networks and communities conceptualisations

Both communities and networks are socially constructed and their activity is carried out in social contexts. Thus, a distinction in absolute terms, *in vitro*, would not have a specific meaning. These notions describe a part of the social activity, highlighting different aspects and, thus, enable the further investigation of those aspects. The current paragraph aims at the identification of the particular aspects of each, as studied by recognised scholars of each approach. This identification will then help me to identify the factors of correlation between the two notions, providing in this way a better understanding of networks and communities. Then, this understanding will be used in the current part to explore the ways in which user-developer-entrepreneur networks and communities emerge, as well as the possible modes of management of their relationship to enterprises.

Social networks and connected individuals

In their more abstract form, networks are chains of connections. Individuals connected do not need to share any other thing beyond their connection, to constitute elements of a network analysis.

The conceptualisation used by (von Hippel, 2007) of networks as *"user nodes interconnected by information transfer links"* is common to the greatest part of the literature in social networks. He also notes that the user network concept applies independently of the technical support for the connection, to interactions operated *"face-to-face, electronic or any other form of communication"*.

Some of the most used concepts in network analysis were introduced by Granovetter (1973, 1983). Granovetter defined three general natures of ties that are widely accepted and used in social networks analysis: strong, weak or absent (Granovetter, 1973, p. 1361). He specified that *"the strength of a tie is a (probably linear) combination of the amount of time, the emotional intensity, the intimacy (mutual confiding), and the reciprocal services which characterize the tie"* (Granovetter, 1973, p. 1361). Granovetter argued that the "strength of weak ties", that is the value of connections that do not belong to one's close circle, lies in their *"likelihood of being bridges"* between different social groups and are very likely to become the channels of new information, knowledge as well as innovations throughout a network. On the contrary, people with few weak ties will become isolated from receiving new information from outside circles, constrained to hear the same, re-circulated information within their own clique comprised of close friends. This approach is useful to the Open Innovation approach, which focusses on "inflows and outflows of information" through the company boundaries (Chesbrough, 2003a). Hence, the value of a network, as opposed to the value of a group having strong ties, resides in this very potential, which Simard and West (2006) also name "wide" ties. Both the genericness and the limitations of this approach is its independence to the specific conditions of knowledge creation, "measuring" its diffusion. For instance, if an individual learns something but never shares it, keeps it a secret, his knowledge will be invisible through the analytical concepts of network analysis.

This theoretical approach becomes clearer in the action of sociological research. Lazega (1994) provides a rigorous methodology for the analysis of networks according to precise procedures. In what regards the editing of social networks representations in graphs, before analysing them, as met in research articles, software tools such as *Pajek* (Batagelj and Mrvar, 1998) or, more recently *Gephi* (Bastian et al., 2009) enable sociologists and other social scientists to visualise and analyse large actor networks. Hence, White and Harary (2001) use *Pajek* to illustrate their methodological proposition regarding a way to measure the cohesiveness of a group by the number of independent paths that connect every pair of nodes in the network, Casper (2007) uses *Pajek* to visualise and analyse social network formation and inter-firm mobility within the San Diego biotechnology cluster, Benkler et al. (2013) uses *Gephi* to study a public debate over proposed legislation in the United States. The use of such tools satisfies (and imposes the satisfaction of) the elementary requirements of network theories: the researcher should decide whether or not two nodes (actors) are connected (e.g. whether the actors collaborate or not). Hence, for a network of n actors, the researcher should provide the information of whether or not each actor is connected with the other $n-1$ actors. The resulting graph can then be used for a network analysis, looking into more sophisticated factors, such as the place of each actor in the network or the local density of relationships.

Therefore, social network research in action, while being very rigorous on the criterion of whether or not two individuals are connected, is more relativist on the criterion of what those individuals know, stating that if they are "weak ties" they will possibly know very different things. According the lines of this approach, a researcher cannot, for instance, state that two physicians - one being in San Francisco, the other being in Marseilles - are connected by the discipline of medicine. To state that they are connected, he should illustrate that they know each other (first degree) or the have somebody in common (second degree) and so on. The paradox of "non relationship" between physicians can however be answered within the frameworks of the same approach by the "small world problem" (Milgram, 1967), which claims that all people are connected within six degrees of separation. For social network analysis, a group is identified not by what is in common, but by the density of the connections footnoteSee for instance the study of Pissard and Prieur (2007) on *Flickr* web service groups identification..

In addition, while being very attentive to the relational part, network theories do not provide an understanding of how people get to know each other, how network connections emerge, that is before a network analysis can take place. Such an understanding could be useful for management, as many scholars have underlined the difficulty of enterprises to develop and cultivate innovation networks. More over, it does not provide an answer to my problem, that is understanding how developer collectivities emerge and how service providers may harness such collectivities for the sake of their exploration and exploitation needs.

Communities of . . . : the common as social fabric

The approach to communities generally stresses *something* that its members *share*. Thus, communities are built around a common and take their meaning from it.

A well-known case of such communities is the one of communities of practice (CoP, Wenger, 1998; Wenger et al., 2002). In management literature there is a strong tendency to use the term "Communities of Practice" (CoP) to describe whatever cannot be included in a general organisational schema rendering the term far too wide a category to be useful in the distinction of informal organisational phenomena.

Wenger et al. (2002, p. 27-29) had however defined three structural elements for communities of practice:

- *The domain*, denoting the topic the community focuses on and creating a common ground and a sense of common identity inspiring members to contribute,
- *the practice*, consisting in the specific knowledge the community develops, maintains and shares and
- *the community*, being the "social fabric" of learning.

Hence, "cultivating" communities of practice means acting on those elements, by different means, as by specifying a topic, introducing a new practice or providing interaction tools. In fact, as communities of practices have been introduced by the study of relatively stable work environments, as Amin and Roberts (2008a) note, a long-term relationship is to be found in their structure[1].

By considering the relations stable, much of the research on CoPs has been able to focus on the ways such relational settings guarantee knowledge on practices sharing, maintenance and development. Regarding relationships themselves, the informal, non-biased nature of relations is important. Hence, for Wenger and Snyder (2000) project teams are to be excluded from the notion of CoP, as the authors have a different view on the kind of relationships at stake. For them, while project teams are held together by *"the project's milestones and goals"*, the linking power of CoP is *"passion, commitment and identification with the group's expertise"*. In the same work, they nuance the dimension of time, as the duration of CoPs is proposed to be for *"as long as there is an interest in maintaining the group"* (Wenger and Snyder, 2000, p. 142). Hence, when the interest of maintaining the group stops being

[1] Amin and Roberts characteristically quote the early work of Lave and Wenger (1991), where the notion of the *time* is the only specific description of the type of relationships built, by quoting the following definition:

> *a system of relationships between people, activities, and the world; developing with time, and in relation to other tangential and overlapping communities of practice (Lave and Wenger, 1991, p. 98).*

present, when the topic or the practice of the community stops being relevant, as in the case of successful disruptions (Christensen, 1997), CoPs also stop being maintained.

Amin and Roberts (2008b) provides a list of the different types of communities of practice identified by the literature (*craft-task-based, professional, expert or high creativity, virtual*). As observed by this list, while the different types do not share any common type of social ties - they vary from long-lived to short-lived - this is not the case for knowledge: every community type shares a specific type of *"knowledge practice"*.

A second approach to communities comes from the sharing of resources. The case of open source software is exemplary in the sense that its intellectual property license is designed to guarantee the sharing (O'Mahony, 2003; von Hippel, 2010). Von Krogh et al. (2003a) thus names "communal resources" the rewards of participating in a community, which in the case of open source are reputation, control over technology, and learning opportunities. Still, in the specific field of my research, resources are not communal, as the service provider maintains the control over his technology, though its used by developers to create applications.

A third category of communities focusses on a specific commercial good, which structures around it a community of its own users. This wide category of client communities varying from fashion consumers (Di Maria and Vladi, 2008) to video gamers (Haefliger et al., 2009) highlights how a product becomes the intermediary between enterprises and users, enabling a feedback acquisition or user-self-regulation. In some cases, such activity may lead to user's entrepreneurial steps, as well (Haefliger et al., 2010). However, those steps enter rather in the model of "accidental entrepreneurship" (Shah and Tripsas, 2007) than a purposive process of both UDEs and enterprises to innovate.

In all forms, communities have a way to define their identity and their processes of interaction. Community members can be distinguished in old and new ones and organising a community means assuring the common life, making sure that the common will be maintained while the newcomers will be initiated to the community values and advance their identification with their group, along the process of engagement and participation. As we will discuss more in detail in Section 17.2.1, when exploring the possibility of a community-based exploration of a service's potential, while the community has the tendency to welcome new end-users, the selection of developers is more slow, the community having to reassure the expertise and the engagement of new developers before accepting them to the team. Auray proposes that the two extremities, a very popular community and a not at all popular community are the situations where community is less *"vital"*. At the same time, in-between, there are two modes of existence of a *"vital"* community: on the one hand, a "supported" rhythm of expansion, on the other hand a "patrimonial" mode of existence, where most members are ancient ones.

14.2.3 A joint approach : collective action and conversation

A specific reading of the theoretical discussion on management research epistemology can provide an analytical framework for the joint analysis of informal collectivities, useful for my particular research problem of identifying how developer communities and networks emerge and the ways in which enterprises can benefit from them.

A central notion that indicates the limits between the division of networks and communities notions on the basis of their relational and cognitive content correspondingly, is the one of the inseparability between knowledge and relations, to wit the impossibility to dissociate knowledge and relations between individuals that acquire and develop the knowledge in the long term (Hatchuel, 2005c). Hence, the distinction between relation-based and cognition-based social constructions, such as networks and communities as reviewed previously, can only be temporal. In fact, what binds together cognition and relationships is collective action. Hatchuel (1997, p. 185) argues that every mode of coordination is inevitably based on a more or less coherent interaction between a relational model and a particular distribution of knowledge. Hence, management as a discipline refers to an "epistemology of action", where commanding is inseparable from the relationships between *subjects-objects* and *subjects-subjects* Hatchuel (2005b, p. 86). Epistemology of action, by addressing collective action as a "central enigma", takes a distance from the classical epistemology of universal scientific laws on the on hand, and from critical approaches that *"support the relativism of every knowledge project (post-modernism), its dependence on social consensus (dialogue relativism) and its contiguous historical character (constructivism)"* (Hatchuel, 2005b, pp. 73-74).

As a consequence, in the particular case of informal settings, structures such as communities and social networks (the first being based on *subjects-objects* relationships, the second being on *subjects-subjects*, as reviewed previously), should practically meet in action. Such an approach may spare the confusion of the reader of open source studies, for instance. The *Apache* server, which is illustrated either as an exemplary case of user *networks* (von Hippel, 2007) or as an exemplary case of user *communities* (Lakhani and von Hippel, 2003) is actually a meeting point of the two, since it is a result of a collective action. Thus, both network elements (such as information diffusion) and community elements (such as expertise development) are needed for such a complex project to be undertaken. Hence, in such settings, the imperative of collective action "forces" in a way the alignment of relations and knowledge domains to the objectives of the specific development process.

Beyond collective action, though, there also is a way of putting together networks and communities that will be explored in the next chapter. It is simply by providing the possibility of a conversation. In fact this has historically been the case for the emergence of the "*bourgeois public sphere*" according to Habermas. While his work is usually used to illustrate the opposition between an intimate sphere and public sphere (Cammaerts, 2008, and others), a more careful look into Habermas' work indicates that intimate conversation initially acted more as an operator, a catalyst, between private and public spheres, than as an opposition to the latter. The typical cases examined refer to the coffee shops, the salons and the clubs of the period of the industrial revolution, as well as to the relation between the personal diary and the literature of the time.

Habermas goes through the transformation of the intimacy at the court of Louis XVI and "*the social gatherings*" that achieved "*the character of a private party*" to the *salons* of the eighteenth century, characterized as the "*cultural heirs of the court*". For Habermas, the early stages of the "*bourgeois public sphere*" are to be found in those transformations in what used to be the court's intimate circle:

> *For inasmuch the "town" took over its cultural functions, the public sphere itself was transformed (Habermas, 1991, p. 31).*

Habermas provides the example of the Academy of Art in Paris, which in 1677 opened its first *salon* to the public, while in the first half of the eighteenth century the *amateurs éclairés* formed the inner circle of the new art public (Habermas, 1992, p. 40).

Borrowing this notion of the potential for a place where intimate conversations occur to create a public, I will use it not for the study of the "*bourgeois public sphere*", but for the possibility of emergence of a developers public, in the particular context of Web services.

Besides, in the previous part we've seen how such a public emerged in the cases of the industrial settings studies by user-developers groups, often united by an amateur press. The advantage of such an approach is to be proven, to the extent that a topic or a place of discussion manages to bring together people that do not belong to the same community and are not connected beforehand.

Of course, there can be other types of synthesis, including an object within a network or refusing the interaction of communities, that can enable an analysis only on a relational or cognitive level. Bruno Latour (1996), for instance, by considering objects as actors is able to deduce every activity at the network level. Michel Gensollen (2003) on the other hand, by considering that online forum users relations are insignificant, is able to analyse informational goods and online communities at the level of the common informational corpus constructed by users' activity. Still, considering action and conversation as a unifying point of networks and communities enables me to study a case where no community or network pre-exists, while valuing both approaches.

14.2.4 An analytical framework for communities and networks meeting points

Figure 14.1 synthesises the above reviewed forms of informal association. For this, I use two axes, the cognitive and the relational one. The cognitive refers to the object of interaction, which can be a practice (Wenger, 1998) or a common resource (von Krogh et al., 2003a), for the cases of CoPs and open source communities correspondingly. The relational axis refers to the connections between individuals, which, as described by (von Hippel, 2007) may use various technical substrates. There is no association that is "purely relational" or "purely cognitive", thus the regions close to the axis do not correspond to any kind of collectivity. Communities, as they emphasise the object that people have in common and which forms their identity, are more cognitive than relational. Networks, as the emphasis is on the connections people have with each other more than their knowledge on objects are more relational than cognitive. In addition, there is a uniting field between communities and networks. I identified two ways in which those two social constructions meet: a) collective action, b) conversation.

Regarding collective action, in Chapter 16 I will explore a rather unconventional mode of bringing together individual UDEs that is used by Web services providers, called *"hackathon"*. More particularly, the study of the three days application development contest on the topic of *HTML5* sponsored by *Google* is in fact a way to "make emerge" communities and networks of developers through the proposition of a specific *action*. The output of this process is a wide exploration of the topic, useful for business in foggy competition.

Still, there is another type of collectivity identified, which brings together individuals and opens up the potential of networks or communities creation, without really organising a collective action. It is the case of the Barcamps, explored in the Chapter 15. In this original case, there is no collective action at stake. The joining of communities and networks as well as the deployment of the possibility for the emergence of new ones is operated through *conversation*. This conversation however does not aim at the development of a consensus (hence is not threatened by controversies) as the collectivities in question are ad hoc, ephemeral. People who meet do not necessarily belong to the same network or community, though a future synergy is always possible. Therefore, they constitute original forms of interaction and sharing which are not identical to communities and networks as reviewed previously. Thus, I use the term *"conversational settings"* to describe places where

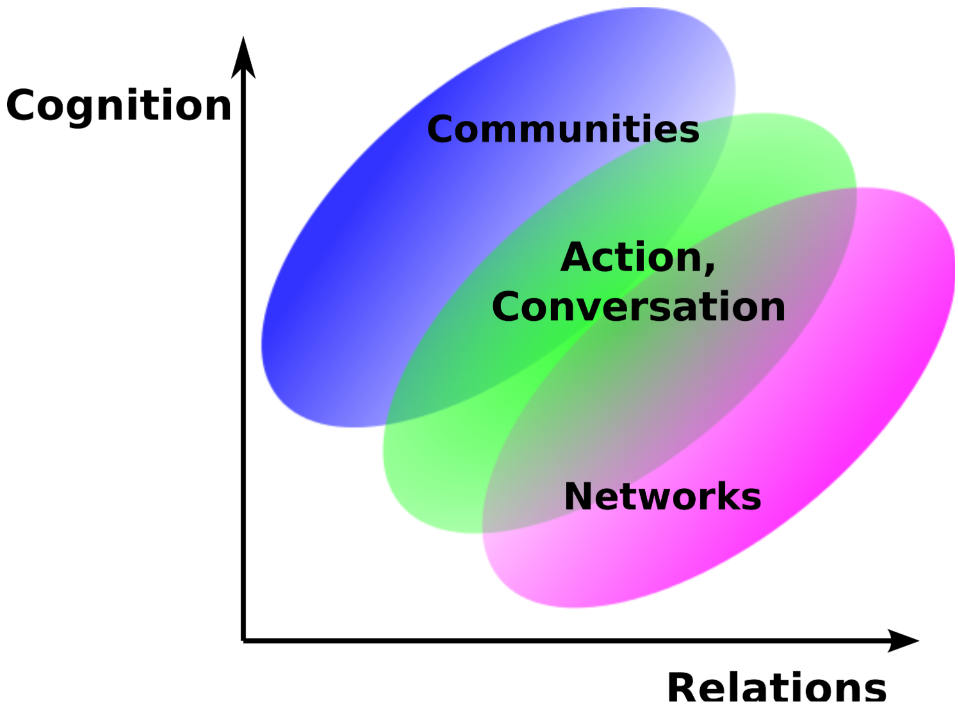


Figure 14.1: An analytical framework for communities and networks meeting points. The uniting field can be either collective action or conversation.

conversation may occur between people who could potentially share common things or get connected, without such an evolution needing to be a prerequisite. Thus, such a mode refers more to the kind of conversations strangers may have in a coffee shop than the discussion between partners, community members or already connected individuals.

Finally, in Chapter 17 I will look into a nowadays more familiar form of interaction, online forums. The online forums I will be interested in examining correspond to the needs of service potential exploitation, and organise a *conversation on the action* of the developers, namely regarding the problems they face while using a specific service's *Application Programming Interfaces* to create their own applications.

14.3 Methodology used: enrolling different strategies of phenomenon-based research.

Von Krogh et al. (2012) in replying to a demand of the *Long Range Planning* editors for more phenomenon-based research proposed a research strategy described by some methodological guidelines for researchers, drawing from the literature corpus studying the open-source phenomenon. Regarding the tradition of the phenomenon-based research in the Management discipline, they refer to the work of Bartlett and Ghoshal (2002), the latter acknowledging that they don't believe *"the transnational corporation really came out of any . . . literature"* (Bartlett and Ghoshal, 2002, p. 13) - cited in von Krogh et al. (2012, p. 278). They also refer to the suggestion of Edmondson and McManus (2007) that qualitative approaches should be used when addressing new phenomena with little or no previous theorisation.

Besides, Mintzberg et al. (2003) when discussing strategy process concerning teaching, share a similar spirit, which challenges the common research practice of relying on a study fn a literature-issued question, in the particular case of the study of new phenomena:

> *Asking a right question in strategy is analogous to an explorer's finding his or her bearing before starting the journey. There is no standard methodology for coming up with questions: intuition and experience play far too important a role in the process (Mintzberg et al., 2003, p. xii-xiii).*

Von Krogh et al. identify five different strategies for these kinds of studies, proposing exemplary works for each one. These strategies are: *distinguish*, *explore*, *design*, *theorize* and *synthesize*. Regarding the first strategy, narratives, ethnographic and conceptual methodologies, followed in the case of the open source software studies to distinguish it from traditional research fields (Raymond, 1999; Stallman, 2002; von Krogh et al., 2003a; O'Mahony, 2007, are some exemplar studies on the field of OSS), are the methods suggested to researchers willing to distinguish a new phenomenon. *Exploration* strategies are then based in descriptive and inductive statistics, archival data and surveys (as in Koch and Schneider (2002); West (2003); Roberts et al. (2006)).

The third strategy, the one of research *design* meets an important variety of methods, depending on the research phase of the scientific community: in the initial phase, where the phenomenon needs first to be singled out from other known phenomena (*"embryonic phase"*), the strategy proposed is the opportunistic approach, as in Lakhani and von Hippel (2003). In the phase where the phenomenon becomes more visible to a larger academic community (*"growth phase"*), the shared infrastructure and service to researchers is suggested, as in Van Antwerp and Madey (2008). When the phenomenon reaches a consistency when regularities become predictable (*"mature phase"*), advanced modelling techniques may be used (Giuri et al., 2008).

Finally, the last two strategies proposed by von Krogh et al. are to *theorize* the phenomenon and to *synthesize* the corpus of management theories under its light. In the case of OSS studies, scholars theorized the phenomenon using abduction or deduction (Lerner and Tirole, 2002; Sojer and Henkel, 2010). Synthesizing was then based on reviews and conceptual contributions (Dalle et al., 2008; Feller, 2005, for the case of OSS).

The current chapter uses the suggestions of von Krogh et al. while also making use of the outcomes of Part II suggesting that during foggy competition, a parallel exploitation and exploration of new technologies, markets and uses is required and that this activity can be partly undertaken by UDEs. Thus, we expect to identify some evidence of previous rationalisations (such as technical support) as well as open-ended methods, oriented towards the joint provider-third party developer exploration of the platform potential - and thus the set of possible innovation trajectories.

Table 14.1 shows an overview of the use of the three first steps described by von Krogh et al. (2012) in the chapters of the current part.

14.3.1 Methods

14.3.2 Observant participation

The observant participation methodological approach has already been discussed in Section 6.3.2. It consists in taking part in field challenges, to study the action norms of individuals in the field. In the current part, observant participation will regard my implication as a researcher in both conversational settings (Chapter 15) and ephemeral action settings (Chapter 16). As already discussed in Section 6.3.2, the short temporal horizon of these

Chapter - Field	**Question**	**Distinction Step** *"Identification of peculiarities and concepts for study. Broad cultural terms description"* .	**Exploration Step** *"Data gathering according to focal concepts, phenomenon description"* .	**Design Step** *"Experiment and propose research designs for the dynamics of the phenomenon"* .
15. - *Bar-camps*	1. How can there be informal knowledge & ideas sharing when no networks or communities exist?	Neither a community, nor a network: a conversational setting.	"Regulars" identification.	A design for conversational settings.
16. - *Hackathon*	2. How can enterprises harness UDE activity for the exploration of their own service potential?	Elements for UDE exploratory action organization.	Technological, entrepreneurial and intimate knowledge for trajectories exploration.	Proposition: personality is an innovation factor.
17. - *Developer Support Forums*	2. How can enterprises harness UDE activity for the exploitation of their own service potential?	Not a typical problem-solving setting.	A very low problem-solving rate.	The "curator" expert figure and the UDE-enteprise "empathy" dimension.

Table 14.1: Overview of the use of the three steps of phenomenon-based research in the current part's chapters, based on von Krogh et al. (2012).

settings renders this approach less influenced by cultural differences between researcher and actors of a specific field, as happens in the study of action within the frameworks of a specific organization or other long-lived settings.

14.3.3 Monument study. Investigating the traces of interaction

A particularity of Web interaction, as compared to other modes of communication, such as face-to-face or the telephone, resides in the fact that it leaves traces. This fact, already observed in the sociology of uses (Georges, 2009) provides a research potential that I will exploit in the current chapter. In addition, when these interactions happen in public, the researcher can access these traces being 'invisible' to the actors in the field.

Inspired by Foucault's Archaeology of Knowledge, I face these traces as *interaction monuments*, and use them to reconstitute the rationalities that determine the modes of online discourse, for the case of service-UDE interaction. I use the term "monuments" to describe the traces of human interaction that provide us the opportunity to "verbalise" and "decipher" the context of their creation.

This methodological approach is different from the research strategies reviewed by Yin (2003) - see Table 4.1 on page 70. Particularly, unlike archive study, where data which has been organized and classified by librarians or other experts, in monument analysis data is yet to be created by the researcher, drawing from the traces interactions have left on the technical support on which they took place.

Hence, using a "monument analysis" method, I will examine the traces of interaction between UDEs and service providers aiming to decode and characterise the conditions of their creation.

The kinds of monuments I will study depend on the research field. In the chapter on *Barcamps* (Chapter 15) those monuments will be the registration files, as well as meeting photos and descriptions, all created and published by participants themselves. In the chapter on *Hackathon* (Chapter 16), those monuments will be meeting organisation documents (such as the meeting's program), created by the meeting organisers, as well as descriptions about projects, created by the participants themselves. Finally, in the chapter studying developer support forums (Chapter 17), those monuments will regard the online discussions between UDEs and service provider staff, produced during the discussion and remaining public thereafter, as well as a "Cookbook" on developer relations management created by a *Google* employee to help her colleagues on their work.

14.4 Part Overview

Table 14.2 shows a short overview of the current part. Chapter 15 will address the question of how there can be informal knowledge and ideas sharing when no networks or communities exist. As we will discuss in Section 14.2, the Open and User innovation literature suggests that innovations can occur beyond the boundaries of an enterprise, through user networks or communities. Still, the question of how such networks emerge and, consequently, how an enterprise may harness them remains open. *Barcamps* are conversational settings were discussion can take place between strangers, in a self-organised mode, on new technologies, uses or markets. To study them I used a phenomenon based approach (von Krogh et al., 2012), drawing on material from "observant participation" and "monument" analysis. I conclude this part with a design proposition for conversational settings analysis and creation.

This design consists of three modules (*open invitation*, *focus* and *networking*), as well as a methodological distinction between participant public and regulars.

Chapter	Question	Field	Methodology (phenomenon-based approach)		Outcome
			Partici-patory Observa-tion	*Monu-ments analysis*	*(Alternative research and deployment design proposal)*
15.	*1. How can there be informal knowledge & ideas sharing when no networks or communities exist?*	Barcamps: ad hoc conversational settings where UDEs meet to discuss about new technologies, uses and markets and which enterprises "infiltrate"	✔	✔	Using conversational settings for networking, environment monitoring and exploration.
	2. How can enterprises harness UDE activity for				
16.	*i) the exploration of their own service potential?*	Hackathon: ephemeral action setting, where UDEs explore the potential of a new service or technology.	✔	✔	Ephemeral exploratory action settings organization and the importance of participants personality.
17.	*ii) the exploitation of their own service potential?*	Developer support forums: conversational settings, where UDEs can express their concerns about issues they face during service potential exploitation.		✔	Beyond problem-solving: "curator" and UDE-enterprise "empathy".

Table 14.2: Part III overview

Chapter 16 will address the question of how enterprises can harness User-Developer-Entrepreneurs (UDEs) activity for the exploration of their own service potential. The *Hackathon* I will study took place in *Google*'s headquarters in Mountain View, California, and consisted of three days of exploration of a new technology, the *HTML5 protocol*. Using a phenomenon-based approach and using material produced by observant participation and monuments study, I propose a design for exploratory settings, consisting in two basic elements: a) knowledge sharing from the provider to the UDEs on the new technology b)

UDE exploratory activity, based on the new knowledge and their own, previous knowledge.

Chapter 17 will address the question of how enterprises can harness UDE activity for the exploitation of their own service potential. Two developer support forums will examined, *Facebook* and *Google Maps*, and common patterns will be identified. Through the study of monuments, I will propose a mode for "leading the lead users", when they are developers.

Chapter 15

Before networks and communities, a conversational setting: the case of the Barcamps in Paris

Contents

15.1 Introduction

In Part I we've seen how Web application developers can be characterised as user-developers, user-developer-entrepreneurs or developer-entrepreneurs depending on the value of their innovations, whether for their own use or commercial value, as perceived before the development process and as perceived once an application is created. In Part II we've seen how such actors emerge in industrial periods in-between market emergence and industrial rationalisation, were exploration of the potential of a new technology goes hand by hand with exploitation of new market niches, having emerged from new uses. As there needs to be a synergy between UDEs and enterprises providing the technologies, sharing "knowledge and ideas" during this process, the resulting situation may point to what Chesbrough calls Open Innovation (Chesbrough, 2004; Chesbrough et al., 2006; Chesbrough and Appleyard, 2007).

The current chapter explores a way for providing the condition of possibility for networking and community emergence between users, developers, entrepreneurs and enterprises. The *Barcamps* conversational setting is such a solution, though it does not actually lead in networks and communities emergence in a deterministic way. Barcamps constitute a meeting point of users, developers, entrepreneurs and service providers. There, participants have the opportunity to discuss and learn about new uses, markets and technologies, what I will call a business environment, as well as their transformations, what I will call an environment shift.

Historically, *Barcamps* begun in 2005 in San Francisco, USA, by Web entrepreneurs that could not participate in an official conference, the *Foocamp*, because of its fees as well as of the fact that invitations to the conference were limited. Since then, their format has spread on an international level, and hundreds of *Barcamps* have assembled service providers, developers, entrepreneurs and "curious" users, interested in innovation in the context of the ICTs.

15.1.1 Chapter overview

Table 15.1 summarises the current chapter. Section 15.2 will review some additional theoretical concepts to the ones already reviewed in Section 14.2, regarding the open question of the possibility of informal knowledge and ideas sharing when no networks or communities exist.

Then, Section 15.3 will discuss the methodological approach that will be used in the current chapter. Adopting a phenomenon-based research strategy (von Krogh et al., 2012), this chapter will undertake its three first steps (distinction, exploration and design), using two methodological approaches: *observant participation* (already discussed in paragraphs 6.3.2 and 14.3.2), and *monuments analysis* (already discussed in paragraph 14.3.3), both applied in the specific context of Barcamps.

Section 15.4 will undertake the exploration step. The peculiarities (von Krogh et al., 2012) of both a Barcamp as a singular setting and the series of Barcamps as a "chain" of conversations will be investigated. The section will propose that those settings are not communities, though we can distinguish those aiming at the exploration of the possibility of a community emergence, of networking and those which just aim to developer-entrepreneurs discussion as such.

Section 15.5 will further explore the focal concept (von Krogh et al., 2012) of participation. It will propose that Barcamps do not form a network, either, while it will advance the concept of the "regulars" for the study of conversational settings.

Question: How can there be informal knowledge & ideas sharing when no networks or communities exist?

	Description	**Methodology**	**Outcome**
15.2 (& 14.2). Literature Review.	Review of informal modes of open and user innovation organisation.		A lack of tools and methods for informal settings establishment, when no communities or networks exist.
15.4. Distinction Step.	Exploration of the peculiarities of a Barcamp and a chain of Barcamps.	Observant participation, monuments analysis.	Field description: not a community. Barcamps for communities, for networks and just for conversation.
15.5. Exploration step.	Further participation analysis.	Observant participation, monuments analysis.	Not a network, either. New concepts proposal: regulars.
15.6. Design step.	A synthesis for a conversational setting study.	Previews data discussion.	Modules of a conversational setting design, topic & participation -based configurations.

Table 15.1: Chapter overview.

Finally, Section 15.5 will perform a second order analysis of the data previously examined, proposing a design for conversational settings research and deployment.

15.2 Literature review

In management studies one can find a broad spectrum of organisation modes for innovation, beyond the level of a single enterprise. Teece and Chesbrough (2002) summarized four typical models of business organisation for innovation: virtual organisation (Alexander, 1997; Wiggins and Crowston, 2010), alliances (Osborn and Hagedoorn, 1997; Soh, 2009), joint venture (Inkpen and Beamish, 1997; Peng and Shenkar, 2002), corporation with autonomous divisions and integrated corporation (March and Simon, 1964; Henderson and Clark, 1990; Nakhla, 2003). More recently, research in New Product Development has focussed on the case of co-development (Chesbrough and Appleyard, 2007; Maniak and Midler, 2008) as well as exploration (Segrestin, 2005; Gillier et al., 2010) partnerships. In research fields where the organisational mode includes multiple firms, a central issue (for both practitioners and academics) is the definition of the *common* boundaries (of the alliance, the venture or the partnership). On the other hand, when innovation is studied at the level of a single firm, the focus of the analysis is on the boundaries between different departments. Moreover, the more exploratory the nature of a project, the more difficult it becomes for the actors to evaluate and share the results of eventual innovations.

On a more informal and less structured level, networks and communities may also be a solution for knowledge and ideas sharing. Still, as we've already discussed in Section 14.2

the question of how communities and networks are created in the first place remains open. I also proposed that two means can be used for "making them emerge": collective action and conversation. Figure 14.1, copied in Figure 15.1 for ease, has synthesized a joint approach for networks and communities, proposing that conversation and collective action

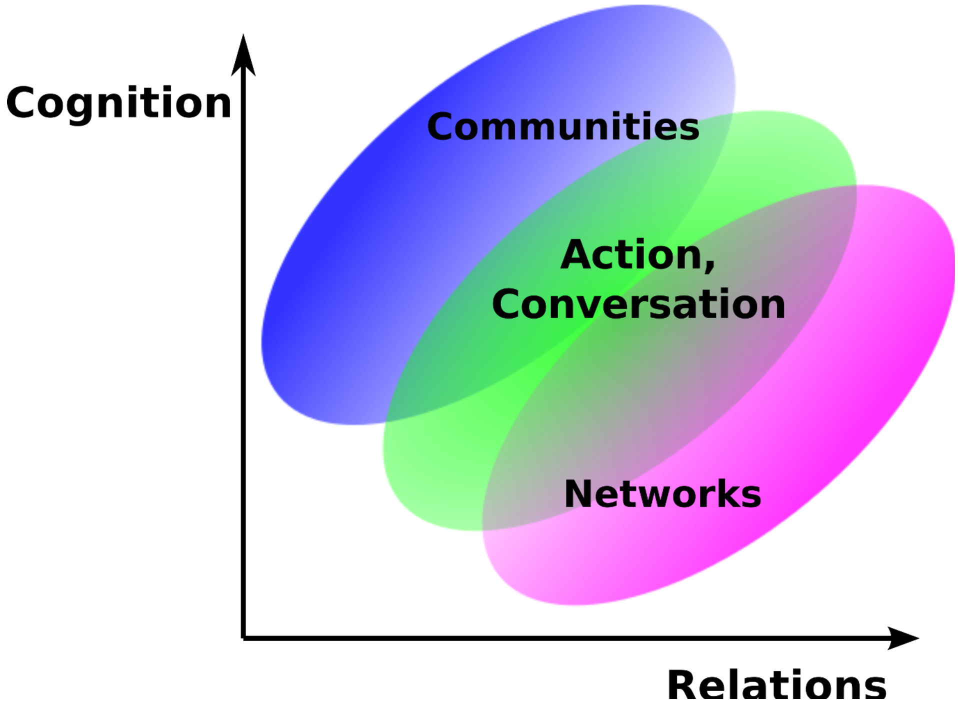


Figure 15.1: An analytical framework for communities and networks meeting points, discussed in Section 14.2. The uniting field can be either collective action or conversation.

Barcamps have been characterised as events that function as "contact generating machines" (Aguiton and Cardon, 2007), a tool to present their ideas or projects, to get to know new ones and meet new people, to have a weak, temporal cooperation without a previously designed strategy. Still, as we will see in the current study, within the framework of foggy competition, these events are in a position to fulfil the strategical requirements both for developers-entrepreneurs to have an overview of the shifting environment and service providers to 'infiltrate' the conversational setting and to promote their service.

From a different perspective, scholars in management have been interested in what are named *field configuring events* (Meyer et al., 2005). Maskell et al. (2006) studying the *Meetings, Conventions and Exhibitions industry*, proposed the term "temporary clusters" to emphasize the linkage between agglomeration behaviour and knowledge effects on an inter-organisational and professional level. Hardy and Maguire (2010) have used the concept of discursive spaces to study the UN conference that resulted in the Stockholm Convention. Using an institutional framework of analysis, they proposed that they can lead to change in an institutional field and in organizations through three mechanisms: domination, interpretation, and translation. Generally speaking, Barcamps could be added to the long list of field configuring events. Still, unlike fairs and conferences, where the state of the art of a market or a profession is exhibited by its official representatives, Barcamps have a far less formal and more exploratory nature. Official representatives are welcome though as individuals, descending in sense to the level of users and developers, not dominating the discourse. In addition, unlike the UN or other institutional fora, no decision is to be taken by the gathered public.

However, a perspective which may be more appropriate to address the Barcamps settings is the one of "coffee-like intimate places". As the political philosopher Habermas (1991) mentions, the notion of "intimate places" appeared during the 18th century, particularly so during the Industrial Revolution, defining the sphere in which scientists, philosophers, engineers and entrepreneurs met in a regular but informal way. In the French case, the "*salons de thé*" were key places where this sphere was developed. The intimate sphere was in interaction with the private, the public and the commercial spheres, constituting a "melting pot" of knowledge and ideas. Thus, as Habermas notes, parts of it were either absorbed or limited by the private sphere (the case of the domestic "*living room*" becoming a "*salon*"), the emergence of large industries (organising the social sphere even beyond work) and the dominant role of media in the public sphere.

The analogy of this type of intimacy, though restricted to the level of UDEs, which is distinguished from the family private sphere, though not entering, yet, into the public sphere, will be useful to gasp the case of the Barcamps.

15.3 Research Methodology

My study on the Barcamp phenomenon will be developed according three steps, following von Krogh et al. (2012): (1) distinction step, (2) exploration step, (3) research design step. In this section we will explain these steps.

The data studied will be drawn from 16 meetings that took place within the period from the 6th of December 2005 to the 5th of April 2008 in the city of Paris, France. My research material will be arranged using two methods:

1. *Observant participation* of the *Barcamps*. As already described in paragraphs 6.3.2 and 14.3.2, this approach consists in observing a group's activity while taking part in it. To have a concrete view of Barcamp's process, I participated in 8 of the 16 events organized in Paris. I took part in the discussions and had numerous talks with the participants during the events, taking notes in the process.

2. *Monument analysis*. As already discussed in paragraph 14.3.3, this approach is based on the study of interaction traces, left on Web sites. The 'monuments of interaction' that will be studied, were produced and published by the participants themselves, and come to complete my own observation material. Quality data (Videos, photos, descriptions of the events), as well as software applications developed by the participants will be examined.

Distinction step.

In Section 14.2 we discussed the *"inadequacy of given body of theory and knowledge in the field"* (von Krogh et al., 2012), to describe and analyse informal innovation related social settings that are neither communities nor networks, as happens in the case where individuals do not belong in the same group and are not interconnected - and are thus *strangers*. I have proposed that one of the ways in which networks or communities can *emerge* is through *conversation*, providing the possibility for individuals to connect or to develop a common ground.

In Section 15.4 I will examine the *"bracket peculiarities"* (von Krogh et al., 2012, p. 290) of the specific conversational settings of *Barcamps*. I will use a narrative approach, aiming at the provision of an explicit description of describing the phenomenon as it appears to its stake-holders.

On the singular level, the elements that will be studied will be organisational (topics of discussion, registration process, participation). These elements suggest that those settings are not built around a specific cognitive domain, as in communities.

Furthermore, on a chain level, and using the model of Figure 15.1, I will distinguish the settings in those aiming at exploring the possibility of a community, of those focussed on networking, and those being content to converse between developer-entrepreneurs as such.

Exploration step.

Advancing my research, I will then proceed to the exploration step, developed in Section 15.5, intensifying data gathering (von Krogh et al., 2012). Individual registration rates will be used as a parameter to filter the list of participants, identify and further analyse the 'regulars' of the Barcamps. The material will be drawn from the 'monuments' of the Barcamps that took place in Paris, particularly the registration files. Operating complex computer data processing required to render these data – existing in various formats – in an exploitable form, I will create a data base and then I will perform a quantitative study.

The information will regard the individual rate of participation in Barcamps (the number of times each individual was registered in a Barcamp) as well as the organizational origin of the participants.

The outcome will re-enforce the indication of them not being a community, as the vast majority of individuals were registered less than two times over the years of the meetings were carried out.

Design step.

Finally, Section 15.6 will be based on the previous two to propose a research design regarding the study of the configurations of conversational settings.

For this, I will use a design approach based on the theory of Axiomatic Design (Suh, 1990; Kim et al., 1991), adapted to the level of conversational setting design. Suh's theory, originally conceived to enable an *ex post* design evaluation, permits the analysis of a given technology by configuring the Design Parameters (DPs) in relation to the Functional Requirements (FRs) of its design. The DPs are understood as the answers to the question "how?", or the inputs to a specific system, while the FRs are understood as the answers to the question "what?", or the outputs of the system. For this adaptation I take into consideration the descriptive model on group behaviour and effectiveness (Hackman, 1987). Hackman's model studies organisational processes also by examining the inputs and the outputs of a work-group design. Yet, in this case the latter model cannot be accurately followed either, as my study field does not concern a workspace.

Regarding the DPs, I will configure the organisational "patterns" that characterize all the settings and remain relatively stable while participants as well as the topics of the settings change. This will be based on an analysis of the following qualitative material: observant participation (at eight of the events), documents, photos and videos published by the participants themselves before and after each event.

Regarding the FRs, I will propose a design of what could be considered the "output" of these events, that is the strategical functions that can potentially be served by the DPs. This design will draw on quantity data coming from the event registration files, available on-line. Conclusions about the participation modalities in the overall environment, will concern the degree of engagement of individuals in this setting, as well as the configuration of the "regular" participants and their characteristics.

Further research can then examine similar phenomena, following the axes proposed (modules of an exploratory business environment).

15.4 Distinction step: characteristic attributes of Barcamps

This section distinguishes the bracket peculiarities of Barcamps. Both the subjects of discussion, as described in the initial call and further specified by the participants themselves, and the participation openness do not refer to a specific community domain or a sense of belonging neither do they presuppose that participants are already connected. In addition, unlike communities of practice, safeguarding and developing a common practice, Barcamps allow to those participants interested in experimenting on *new* or *emerging* practices, potentially interesting to them. The overall setting privileges an egalitarian behaviour, though it attracts individuals of different status, namely users, developers, entrepreneurs and service provider executives.

Table 15.2 shows the general data on the *Barcamps* studied, where we observe an important variation in participation numbers, subjects and places. The following paragraphs describe the specific characteristics of these conversational settings.

15.4.1 Registration process

An event starts with an open call for participation. Through a Web page dedicated to each event, everyone interested can self-register, submitting his/her name, personal Web page, organisation and e-mail, creating in this way the list of participants. There is no secretary for the events and no authorisation is needed in order to register or to organize an event. All these event pages are hosted in a common portal (www.barcamp.org).

The only limit to participation is posed by the capacity of the building. The list of participants is completed publically until their number reaches the threshold of that capacity. Open participation is also a principle widely followed in Web communities, such as Wikipedia or on-line fora, where everyone is able to contribute without authorisation.

This organisation mode may lead to a great variety of participants. For instance, in Barcamp 7, one could meet amongst the participants the product manager of *Gmail* and *Google Maps*, the founder of the *Netvibes* Web service, the president of *APRIL* (a French association for the promotion of free software), a researcher from the San Francisco *France Telecom R&D* department. Even if the most important element of the participants came from the computer science discipline, many also came from different disciplines: management, finance, sociology, psychology. Amongst the registered persons, one could find ten people coming from *Google*, seven coming from *France Telecom R&D (Orange Labs)*, four from *SFR* telecommunication company, as well as many entrepreneurs of *Silicon Sentier* association, members of the *FING (Foundation Internet New Generation)*, and executives of firms like *Mandriva-Linux, Hewlett Packard, SEGA* and *SUN Microsystems*.

The DP of the registration process' openness satisfies the requirement of "outsider's" participation. Each event is open to actors that don't usually take part in known ecosystems or networks. Their presence gives the opportunity to extend one's network beyond a specific industry, as well as to obtain new information and knowledge from other, neighbouring fields.

No	Place	Topic	Partici-pation	Sponsors	Date	Duration
1	Café DUNE	How to organize a"real Barcamp".	7	-	Dec 6, 05	7h
2	Silicon Sentier	No specific subject	74	Bearstech, Domaine Hervé Roumier	Jun 10, 06	7h30min
3	Silicon Valois	Media and proximity	26	IGenerator, ViaBloga	Aug 26, 06	17h
4	Mandriva	No specific subject	105	Mandriva, AF83, FT R&D, Silicon Sentier	Sep 16, 06	7h
5	Canceled	-	-	-	Sep 21, 06	-
6	Hôtel Warwick	Rich Internet APIs	26	People In Action, Adobe	Sep 26, 06	3h
7	Google	No specific subject	144	Google, AF83, Silicon Sentier, FT R&D	Nov 11, 06	12h
8	Cube	No specific subject	40	Cube, AF83, Silicon Sentier, FT R&D	Jan 20, 07	6h30min
9	Silicon Valois	Media and proximity	9	UbiConseil, Explorateurs du Web	Feb 17, 07	8h
10	Café Dune	Co-working Paris or « La Cantine »	18	-	Apr 3, 07	2h
11	Yahoo	No specific subject	151	Yahoo, Ziki, Silicon Sentier, Passage Piéton	May 12, 07	8h
12	Fondation Maison des sciences de l'homme	Migrating population and technology	116	FMSH, Silicon Sentier	Jun 23, 07	9h30min
13	O'Sullivan's Bar	Second Life	27	Community Chest, O'Sullivans, FON	Jul 21, 07	6h
14	ENSCI	Projects in process and upcoming projects in FING.	146	Ecole nationale supérieure de création industrielle, FING	Sep 29, 07	7h
15	Sun Microsystems	Tools 2.0	155	Sun Microsystems, Silicon Sentier, OSSGTP, Ruby France, Supinfo	Dec 8, 07	12h30min
16	La Cantine	No specific subject	73	Silicon Sentier	Feb 2, 08	6h
17	La Cantine	Video	122	Quartier Numérique.tv, SiliconSentier,Faber Novel, Orange, France 24	Apr 5, 08	6h

Table 15.2: The list of the 16 Barcamps that took place in Paris during the examined period (data extracted from registration files).

15.4.2 Self-presentation

The day of the event, participants gather at the programmed place and time. The procedure begins by their self-presentation. Each one uses three tags to describe himself, along with his name and profession. For instance, in the Barcamp organized in the local office of Google in Paris, a participant introduced herself using the tags "innovation", "interdependence" and "chocolate".

As an organisational DP, this kind of introduction prompts the self-classification of the individuals according to their expertise or interests and facilitates social interaction and networking on the basis of specific projects during the event.

15.4.3 Program definition

After the presentations, participants define the subjects of the discussion themselves. For this purpose, a whiteboard is already prepared by the organizers, where the rooms and the time slots of the discussions are already completed, waiting for the subjects to be filled in.

"*Barcamp Paris 11*" took place at *Yahoo*'s local offices and was co-sponsored by *Silicon Sentier* (a start-up association), *Ziki* (a Web market that puts different enterprises in contact) and *Passage Piétons* (an event organising agency). After the self-presentation of the participants, they were invited to fill a whiteboard with the discussion topics they desired. On the table there were already marked the names of the available rooms (*"Montorgueil", "Barbès", "Rue de la Paix", "Lepic", "Trocadero", "Beaubourg", "Grande Arche", "Montmartre")* in columns and the names of the available time slots (*"15:30", "16:30", "break", "18:00", "19:00"*) in rows. As the centre of the common discussion topic was the concept of virtual identity, various related discussion themes were proposed and they either were merged into the same discussion group or constituted a different one. Discussions on digital identity included the following: *"Digital identity by art: the project Skatcha'rt", "Being and appearance Twitter + RSS + Blog = Communication = identity for a lifetime, as of last week where and how to define the Being", "Todeka Project Digital Identity Certification"*. Yet, there were also discussion topics not included in the thematology of digital identity, such as *"Coworking Paris and Web site", "Netvibes UWA*[1] *and Widgets: widgets engines, widgetisation of the web".*

This division of the participating "crowd" into many thematic discussions, serves in having more concrete and specific discussion groups, according to their own interests. This way, an exploration of the environment which is more focussed becomes possible, during which useful knowledge and concepts can emerge and be shared.

From the networking perspective, discussion in small, interest-oriented groups (usually from five to twelve individuals) reinforces the capabilities to expand one's network, as well as the scope of exploration. These discussions are very useful for small enterprises allowing them to trace the important evolutions of their field as well as to present their projects and skills to eventual clients or partners. For platform providers, it gives a chance to spot future innovation trajectories and use this feedback to adjust aspects of their platforms.

[1] Netvibes UWA:Netvibes Universal Widget API. A platform launched by Netvibes Web service as a design tool-kit for external developers.

15.4.4 Collective exploration and creative learning.

After the talks, an experimentation process, called *"Mashpit"* usually takes place using the concepts and knowledge that emerged during the discussions. During a *Mashpit* participants create prototypes of new applications and services. A *Mashpit* can often be a multidisciplinary process, as individuals coming from other disciplines other than the one of computer programming (such as design, marketing or social sciences) can also participate.

In the case of the event organised in Google's local offices, a *Mashpit* took place during the second day, where 48 individuals were registered and where ten projects were proposed on the basis of *Google Maps*[2]. When participants met, they voted for three out of the ten projects to be developed. After a coffee break, they were divided in three project groups and started working. Finally, each team presented its project.

I will present here one of the three innovative projects. It is called *"Interactive Map"*. This application had a very simple functionality: the user could mark points on a map of Paris by a single click. Beginning from the first marked point, a path was automatically created to the next points marked. Finally, a course was designed by the user in some clicks. He could execute the application as many times as he wanted to design more courses on the map. There was no indication of specific uses of this prototype. Nevertheless, we can imagine a way in which this application could be used for a new Web service: such as a site where tourist guides, before exiting the hotel with their clients for a walk in Paris, can demonstrate the course that will be followed.

Nevertheless, whether or not the prototype will end up being a commercialized service or application is not a direct objective of the exercise (today one can find applications and features of *Google Maps* a lot more sophisticated than the "Interactive Map"). What is important, for both developers/entrepreneurs and platform providers, is to experiment with the innovation potential of a new platform (as well as of a combination of already existing platforms), check the functionalities and documentation provided and, eventually, expand the platform via the entrepreneurial activity after the event by adopting its tools.

15.4.5 Topics and places.

Table 15.3 proposes a categorisation of the conversation topics under which the Barcamps were organised, drawing on Table 15.2. While for some meetings, usually the earlier ones, there is often no title other than "Barcamp" (*Barcamps No 2, 4, 7, 8, 11*), other meetings have a topic, although still broad (*3, 9, 12, 15, 17*). Six of the meetings were more specific (*1, 6, 10, 13, 14, 16*), indicating a field to discuss and not a specific technology or service. The places are usually different, as the discussions take place either in a cafe or a bar, or in the premises of a services provider, sponsoring the event. The location's capacity also defines the maximum number of participants accepted. An event can be sponsored by multiple organisations (corporations, start-ups or institutes) and no authorisation is needed for its organisation.

More specifically, the Barcamps that took place in service providers' premises have no title. For instance, in the case of the Barcamp organized in *Google*'s local offices in Paris, its title was "Barcamp Paris 7". Yet, the discussions were strongly influenced by the launch of the new platform, *Google Maps*. In a similar way, the event organized in *Yahoo*'s local offices (*"Barcamp Paris 11"*), was oriented towards the general topic of "virtual identity", an issue that Yahoo, as well as most Web services, was discussing internally at the time when

[2]URL visited on 1/11/2008. http://mashpit.pbwiki.com/MashPitParis

social networking platforms arose. An exception is the case of a later meeting sponsored by *SUN Microsystems* which addressed a specific field, the topic of *"Tools 2.0"* - that is tools for Web 2.0 services[3].

Topic	Barcamp No	Total
None	2, 4, 7, 8, 11	5
Wide	3, 9, 12, 15, 17	5
Field specified	1, 6, 10, 13, 14,16	6

Table 15.3: Barcamp topics according to expertise. Barcamp topics were mosty wide or even not predefined at all. When more precise, a general field was specified.

When topics are specified, they remain broad. They can regard uses, be developer oriented (*Barcamps No 6, 15*) or not (*12, 13, 17*), or they can regard broader reflections, such as *media and proximity* (*3,9*). Two meetings regarded the organisation of the conversational setting itself. The first one was on *"how to organise a Barcamp"*, while the 10^{th} was on the development of a more permanent place were UDEs can meet and discuss.

Compared to the organisational forms met on the Web, the construction of networks around a broad topic is a general characteristic of on-line interaction (Chakrabarti et al., 2002; Highfield et al., 2010).

Seen as an organisational Design Parameter, a wide topic satisfies the requirement of attracting individuals that can be potentially interested in a specific service, when it takes place in the provider's chosen location, or by a specific use and the technologies that can be utilised for it. Therefore, it provides an opportunity for participants to monitor technologies, uses and emerging markets.

15.4.6 Discussion

The self-organising nature of the Barcamps is in opposition to the methods used by the Meetings, Conventions and Exhibitions industry (Maskell et al., 2006), as well as by institutional interlocutors Hardy and Maguire (2010). No common decisions are made, while it is not a place where enterprises are supposed to market their products.

Being too informal and temporally ephemeral to enter into the broad spectrum of organisation modes for innovation beyond the level of a single enterprise proposed by Teece and Chesbrough (2002), Barcamps are not networks or communities either. Beyond their ephemeral nature, their topics are too wide, people meeting there are not necessarily connected, while those who participate in those Barcamps that include a mashpit do not safeguard a given practice, but experiment on new or emerging practices, potentially interesting for them.

Figure 15.2 synthesises the deployment of Barcamps' conversational settings as a whole, using two criteria: topic specificity (from no topic to specified field, as summarised in Table 15.3) and networking opportunities (on the basis of the global number of registered individuals, as listed in Table 15.2). This synthesis makes used of the framework putting conversation in-between informal relational models, namely communities and networks, as discussed in Section 14.2 and modelled in Figure 14.1.

[3]These tools are more often developed by programmers using *Java* or *AJAX*, programming languages animated by *SUN Microsystems*.

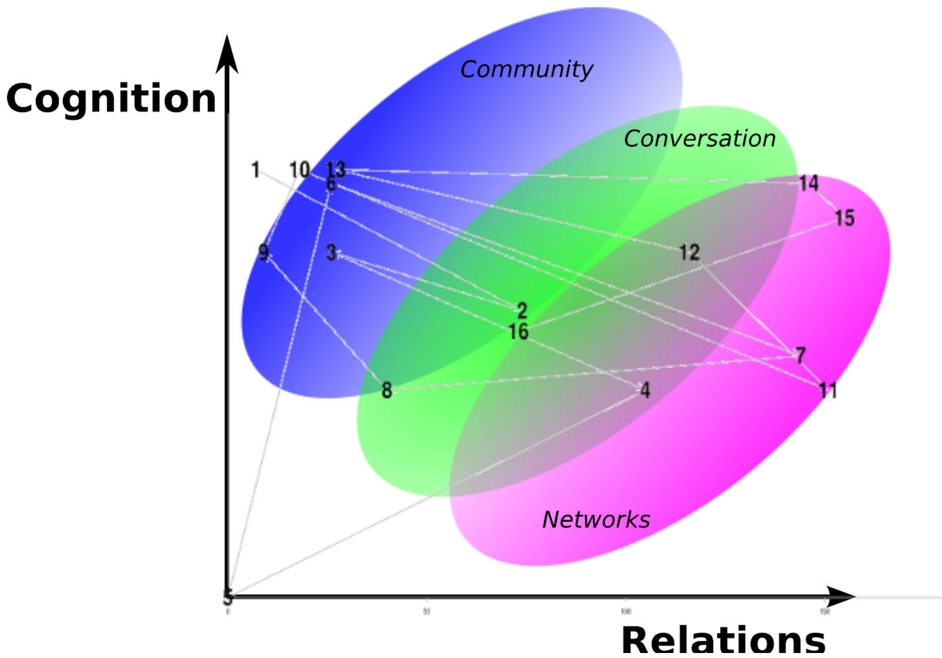


Figure 15.2: A synthesis of the whole set of Barcamps studied, as a conversational setting.

As the schema suggests, there have been Barcamps closer to the notion of community building, as their topic interested a small amount of participants, others that have been closer to networks, as no specific topic was prescribed, and a third category, more clearly oriented towards conversation.

More specifically, the Barcamps 1, 3, 6, 9, 10, and 14 attracted a smaller number of participants and their topic was generally less broad. They were set on the basis of exploring the possibility of emergence of a community on a specific topic. On the other hand, Barcamps 4, 7, 11, 12, 14 and 15 were less narrowly defined, and more focussed on networking. Barcamps 2, 8, 16 were in-between the two categories, and conversation was largely influenced by the place itself (*Silicon Sentier*, *Cube*, *La Cantine*), all those places being themselves Developer-Entrepreneur hangouts.

15.5 Exploration step: participation patterns and regulars

In this section we are interested in further exploring these conversational setting events, by identifying the *"regulars"* in these discussions. Following a quantitative methodology, we configure the qualitative characteristics of participation in the whole environment as well as the specific characteristics of the "core" of this milieu, focussing on the last latterly.

15.5.1 The overall public

From the elaboration of the registration files to 16 Barcamps organized in Paris, we found that 713 individuals were registered in these events. 3% of this population had been registered for more than six times, 10% had participated between three to six times, while 87% had participated only once or twice (as shown in Figure 15.3).

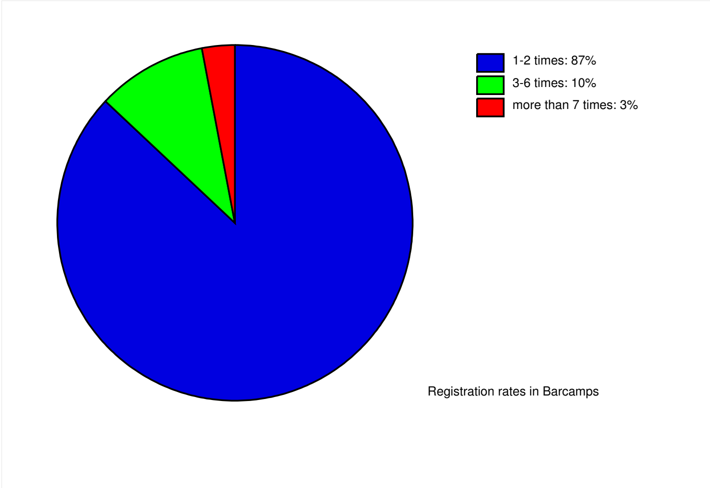


Figure 15.3: Registration rates in Barcamps that took place in Paris from December 6th, 2005 to April 5th, 2008.

While participants came from different professional environments, there were two categories that actively formed this milieu: UDEs of Information and Communication Technologies start-ups or small enterprises on the one hand, and executives of platform providers, usually in charge of technology or community affairs, on the other hand.

What the participation rates suggest is that this milieu managed to be very open, attracting individuals that were interested in specific "wide" topics.

15.5.2 The Barcamp regulars

As Barcamps are informal settings, many of which were organized by "ad-hoc" sponsors, the participation rates being the criterion to determine the regulars. Hence, I am going to analyse the group of people that belong to the 3% of the overall public, having been registered for more than six times during the period studied. Another way to determine the regulars of these events could just be to list the sponsors of all the events (presented in Table 15.2). Nevertheless, since Barcamps are "ad-hoc" events, this method could lead us to false conclusions, as their sponsors are not engaged in participation to future ones.

Once the most active individuals are identified, I configure the organizations to which these individuals belong. This information is often available in the registration files, as participants usually state the organization for which they work during the registration phase. Complementary data drawn from informal discussions with some of the regulars were also used. Finally, I conclude with a model schematically represented in Figure 15.4.

Figure 15.4 illustrates the "group" of Barcamps regulars, during the examined period, consisting of the most regular participants. Each participant is represented by the letter *A* and a number (*i= 1 . . . 23*). Each group is represented by an ellipse. To graphically represent these individuals' participation in Barcamps, I used the model discussed in Section 14.2. The space of communities is where cognition is more important than relations, while the space of networks is where relations are more important than cognition. Barcamps, as a conversational setting, is placed in-between. Individuals are positioned in the schema according to the identity they declared during the registration process, usually professional. Relations-based professions (e.g. PR or marketing) are put on the network side. Expertise-based professions (e.g. developers) are put on the cognitive side.

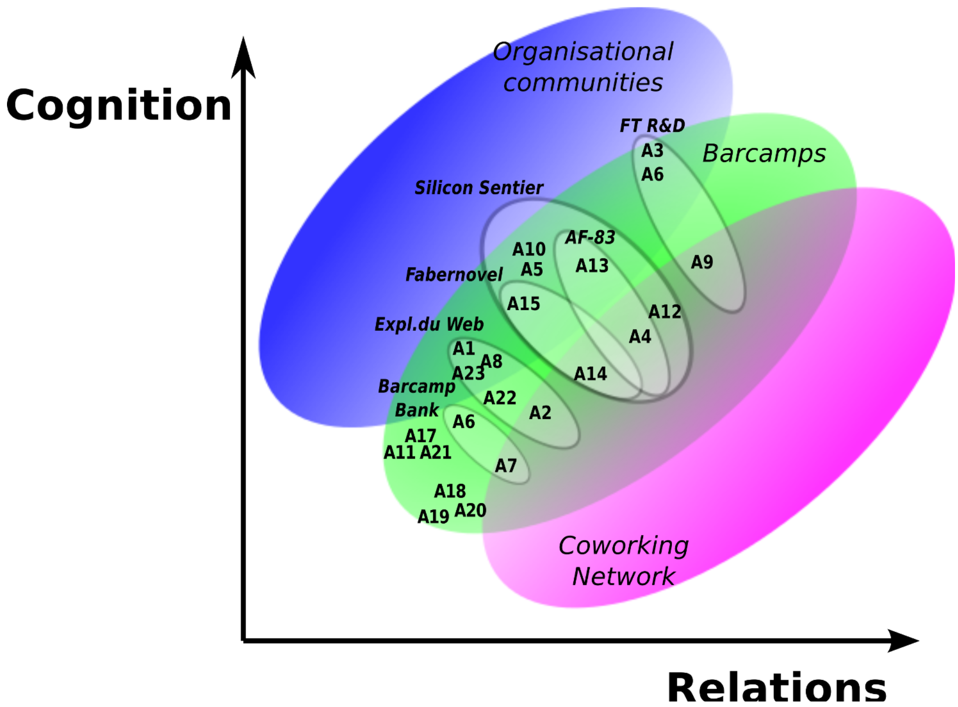


Figure 15.4: Graphical representation of Barcamp "regulars".

The first remark we can make is that the core of Barcamps in Paris is quite structured. The great majority of people frequenting the Barcamps went there along with colleagues of theirs. Only six out of twenty three persons have frequenting Barcamps without being accompanied by a colleague of theirs, and they have been principally marketing professionals or bloggers. Another fact that we can easily observe, is that the Co-working network plays a central role in the constitution of this core, as it connects like a "bridge" (Granovetter, 1973) all the participating groups (at least one individual of each group is inside the Co-working ellipse).

As it transpired from the discussions with the actors, collaboration amongst participants constitutes an eventuality but not a certainty. Both for the regulars and the less regular participants, actors are correlated but not necessarily connected.

15.5.3 The regulars' groups

A look into the regulars' groups reveals a developer-entrepreneur identity, belonging to organisations that are engaged in Web business. Two groups, the *Barcamp Bank* and the *Explorateurs du Web* emerged through the process of conversations. Individuals being regulars without other colleagues participating as much as them either occupied a communication or marketing related post in a service provider, were consultants or bloggers.

Co-working is an international network, united on the basis of the construction of informal collaboration spaces, where entrepreneurs can meet, work and exchange ideas, as well as social niches. During the examined period this network had 734 members on an international level (measured on September 2006). As can be seen in the Figure 15.4, six members of the core of Barcamps participate in this network, representing (formally or informally) their group in this network.

The start-ups association Silicon Sentier were the most regular attendants to these meetings. Financially supported by the local authorities, it emerged as a sort of local "networking bureaucracy", creating and exploiting value through the networking as an activity in itself and playing a role of an intermediary amongst users, developers, entrepreneurs and enterprises. Seven members of *Silicon Sentier* (*A4, A5, A10, A12, A13, A14* and *A15*) and two start-ups (*AF83* and *Fabernovel*) were involved in animating Barcamps. Silicon Sentier is an association of a hundred new technologies start-ups, active in Paris and supported by the City of Paris. In the last part of the period examined, the association managed to extend its network of members and construct a permanent entrepreneurial milieu (called La Cantine), where it organises a great variety of events (from *Barcamps* to seminars and other meetings).

AF83 and *Fabernovel* followed an important development during the period between 2006 (when they were founded) and 2009 following this strategy. *Fabernovel* is an innovation management consulting company, which managed to multiply its turnover by four during this period. AF83 is an enterprise co-founded and co-organised by Fabernovel and Beartech (a technology start-up) for the "agile development" of web applications and services as well as for communities organisation. During the same period, *Beartech* multiplied its turnover by 4.5.

France Telecom R&D also participated in the core of *Barcamps* in Paris, with three individuals (A9, A3 and A6) participating in the core of Barcamps and one in the *Co-working Network*. *France Telecom* (named *Orange* in between) is the largest telecommunication firm of France. Since it employs more than 3,700 researchers, the participation in this milieu could not be understood as a major environment monitoring and exploring strategy. Yet, it managed to establish a relation with developers/entrepreneurs that was useful during the launch of its platform, "*Orange Partner*", in 2008. That platform allowed SMEs to build upon the company's network, in a way similar to the ecosystems of Web platforms.

"*Les Explorateurs du Web*" ("Web explorers") is a consulting group, created in the Barcamp process, specialized in Web services and Web applications. Five members of this group (A1, A2, A8, A22 and A23) took part in the core of Barcamps in Paris. Beyond its participation to this milieu, this group also organised events ("*Explorcamps*") that borrowed DPs from the Barcamp design while modifying others (for instance participation is charged for professional events).

Barcamp Bank is another group which emerged in the Barcamp process. Having also international links, this group organized events that aim to link financial investment with entrepreneurs' projects. Two members of this group (A6 and A7) participated in the core of the milieu in Paris.

After the examined period, different event formats emerged modifying Barcamps' DPs, such as events within large organisations or with topics that weren't related at all to ICTs. In addition, the founding of a permanent "co-working" space by *Silicon Sentier* intensified the activity of this entrepreneurial milieu beyond the organisation of Barcamps.

15.6 Design step: proposition of three modules for conversational settings deployment and analysis

Drawing from the previous sections, I suggest a design for further research and deployment of conversational settings. My design proposal has two levels: the one of a conversational setting as an ad hoc meeting and the one of conversational settings in general, as compared to networks and communities.

		DPs				
		Wide Topic	**Open Registration**	**Self Presentation**	**Topic/group partitionning**	**Experi-mentation**
	Monitor Environment	X	X			
	Attract "outsiders"	X	X			
FRs	**Explore environment**	X	X		X	X
	Creative learning				X	X
	Enable networking	X	X	X	X	X

Table 15.4: Organisational DPs – FRs matrix

15.6.1 On the design of a conversational setting

Unlike the typical setting of an organisation, I found that an active observation of a shifting environment can be performed by the use of conversational settings. The format of Barcamps, which borrows elements from the design of Web communities, can be described by two categories of DPs: *open* and *self-restrictive*. By open DPs I refer to the *broad topic* as well as the *open registration* parameters, while by self-restricting DPs I refer to the *self-presentation*, the *topic/group partitioning* and the *experimentation* parameters. The Functional Requirements of these events are not the same with a typical organisation either. I could describe the FRs of this format as the need to *monitor a business environment*, *attract outsiders*, *explore the environment*, *enable creative learning* and *enable networking*.

Making a synthesis of the preceding analysis, I propose the following DP - FR shown in the Table 15.4.

According to the synthesis proposed, three modules emerge in the analysed format:

1. The *open invitation* module (on the upper-left side of the matrix). This module uses the open DPs and aims to the attraction of a representative sample of a wide business environment. The definition of a wide topic and an open registration process address the requirement of outsiders' attraction in order to monitor and explore shifts on a given business environment.

2. The *focus* module (on the lower-right side of the matrix). This module uses the self-restrictive DPs allowing the focus on specialized topics and working groups, enabling a deeper exploration of practices, knowledge and concepts related to existing or emerging business ecosystems.

3. The *networking* module (on the bottom of the matrix). This module which traverses all DPs, enables social networking by providing the possibility for participants to iden-

tify people and topics related to their own interests and extend their (personal or organisational) social network.

This design is the most beneficial for DEs, as it provides a fast and cheap way to represent the business environment in which they act (first module). In addition, they are in a position to acquire more specialised knowledge in their field from neighbour actors and test new tools and practices while exploring innovative concepts (second module). Finally, they are able to identify the existing actors, networks and communities at a local level and deploy a networking strategy, including potential partners and clients (third module).

For platform suppliers, such as *Google* or *Yahoo*, this design contributes in acquiring a speedy feedback on innovation trajectories that are deployed by entrepreneurs as well as on the documentation and the use modalities of the tools they provide (second module). In addition, as they are able to "meet in person" a sample of their external developer communities and they are facilitated to extend and structure their ecosystem on a local level (third module), taking into the account the specificities of the local market.

At the same time, these events are catalysed by the presence of occasional participants, using specific technologies as end users or user-developers, usually making "surprising" remarks that may open new application fields.

15.6.2 On conversational settings as related to networks and communities

As already discussed in Section 14.2, there are two dominant theoretical approaches to analysing social interaction when leaving the boundaries of a specific organisation: networks, generally reduced to weak and strong ties (Granovetter, 1973), and communities, generally characterised by what they share. While these two notions are often used in order to analyse similar fields, there are distinct characteristics to them: while the notion of social networks focusses on the nature and strength of existing ties between individuals, the notion of communities focusses on common practices, interests and knowledge being more flexible on the qualification of members' ties.

The study of the Barcamps phenomenon suggests that there can be a third way, in between networks and communities. In the Parisian case and for the period studied, Barcamps have explored the emergence possibilities for both networks and communities, by calling for conversation on new topics and bringing together users, developers, entrepreneurs and enterprises that had previously been strangers. To the extent in which an entrepreneurial "milieu" emerged throughout this whole process, from which the "regulars" had been in position to benefit more than the "occasional" participants, Barcamps constituted a conversational setting providing the conditions of possibility for the creation of new networks and communities.

Overall, beyond composing a concrete case of a conversational setting, Barcamps also illustrate the importance of conversation itself, as distinguished from collective action, sense of belonging or establishing long-term relationships.

15.7 Conclusion

The issue of networks and communities emergence possibility has long been puzzling Open and User Innovation studies, since management research has no explanation for informal settings establishment when no social connections or common ground exist.

The case of Barcamps revealed a potential for networks and communities emergence through the establishment of conversational settings. The conditions for this emergence were created thanks to a Barcamp design enabling the exploration of potential topics of interest while providing participants the opportunity to network.

Within these settings, developers-entrepreneurs (DEs) had the chance to monitor the appearance of new technologies, markets and uses, while enterprises had the occasion to infiltrate those settings and informally promote their own services. At the same time, Barcamps created the conditions for both DEs and enterprises to get connected and share their considerations through conversation.

Chapter 16

Organising ephemeral action for service potential exploration. The case of Hackathon

Contents

16.1 Introduction

In an interview taken two years after my field research, Mark Zuckerberg, the founder of *Facebook* admitted the "biggest mistake" the company had made[1]:

> *The biggest mistake that we made as company is betting too much on HTML5 as opposed to native. Because, it just wasn't there. It's not that HTML5 is bad, I'm - long term - really excited about it.*

In the same interview, Zuckerberg commented that the enterprise lost precious time (two years) in focussing exclusively on *HTML5* and was to change its strategy towards redesigning their service.

This chapter explores a way for the harnessing of UDE activity to the requirements of technological potential exploration, which is called *"Hackathon"*. In essence, UDEs are invited to "hack" a specific technology and are for this reason provided with advice by experts, during a three days meeting. While such an invitation may appear paradoxical for other industrial contexts, it is still coherent with a strategy of platform extension through third-party entrepreneurial activity and can provide some answers to the broader question of "how to harness non informal communities and networks for innovation", which remains open in the Open and User Innovation literature, as already reviewed in Section 14.2.

The case I will study took place in *Google*'s headquarters in Mountain View, Silicon Valley and lasted three days[2] and concerned *HTML5*, a projected advancement of the Web services technologies[3].

To distinguish and explore this phenomenon, as well as to propose a research design for its further investigation (von Krogh et al., 2012), I will use literature studying innovation at the group level. Then, different methodological entries (participatory observation and monument study) will be used to draw data from the field and analyse them, according to what is known according to the literature.

This study will provide insights and suggestions for ephemeral group exploratory action organisation, as well as for participants personality consideration while organising such actions.

16.1.1 Chapter Overview

Table 16.1 shows an overview of the current chapter. The question addressed will be how to harness UDE activity for service potential exploration.

Section 16.2 will review some of the most important approaches to the use of groups for innovative action, namely the approaches on user innovation (von Hippel, 2005), on knowledge groups (Nonaka et al., 2000; Erden et al., 2008), on creative groups, as well as

[1] The interview was taken during the *TechChrunch Disrupt*, an event organized by a popular to developer-entrepreneurs news site. Mark Zuckerberg: Our Biggest Mistake Was Betting Too Much On HTML,5 Tuesday, September 11th, 2012, http://techcrunch.com/2012/09/11/mark-zuckerberg-our-biggest-mistake-with-mobile-was-betting-too-much-on-html5/ Retrieved on October 12, 2012.

[2] From 13 to 15/08/2010.

[3] The objective of *HTML5* can be resumed to the use of what in the 1970's was conceived as *hypermedia* for service development reasons. Back in the 1970's, Ted Nelson, who is often attributed the conception of the term "hypertext", had considered the latter as only an attribute of "hypermedia". Nelson's concept of Hypermedia was to include movies ("hyperfilms"), texts ("hypertexts"), audio, music and slide-shows (Nelson, 1974, p. 85) shared in a computer network.

Question: How to harness UDE activity for service potential exploration?

Section	[redacted]	Outcome
16.2. Literature Review.	Different approaches on user, cognition and creativity groups use.	A draft description of the object to explore.
16.3. Research Methodology.	A phenomenon-based approach.	Research steps to follow.
16.4. Distinction Step.	General characteristics of a "Hackathon".	A distinction of roles and phases of a "Hackathon".
16.5. Exploration Step.	Focusing on the cognitive aspects.	The importance of personality for innovation.
16.6. Design Step.	Taking personality into consideration for exploratory action organisation.	Design elements for "personalised collective action" deployment and study.

Table 16.1: Chapter Overview.

synthetic approaches (Hatchuel and Weil, 2009). The section will conclude with a preliminary description of a *Hackathon*, as resulting from the literature review.

For the study of the Hackathon, I will use a phenomenon-based approach (von Krogh et al., 2012) that will be described in Section 16.3. I will use two main methodological entries: participatory observation, as I will use my own participation to the *Google* Hackathon as a first entry, and monuments study, as I will use the traces left by the participants interactions, principally the projects' description, as a means to further explore the field.

Section 16.4 will describe the general characteristics of the Hackathon, establishing it as different to the types of groups already identified by the literature. More specifically, Hackathons appear to cross all levels of group tacit knowledge quality (Erden et al., 2008) in a three day period. At the same time, its different phases as well as the roles undertaken by organisers and participants, generally coincide with what the literature review has suggested.

Then, in Section 16.5 I will further explore those characteristics, specifically in what regards their cognitive aspects as observed in the field. The seminars revealed a "foggy competition" phase for Web apps development, as new performance criteria appear and the potential of a disruption was a topic of discussion. Development and entrepreneurial "tips and tricks" come to support UDEs in their activity. Surprisingly, the study of participants creations, revealing as to personal identity, goes beyond developing and entrepreneurial knowledge, as the "intimate" knowledge, personality related knowledge, appears to have had an important place in the development process.

Section 16.6 revisits the structure of a Hackathon, proposing a way to take into consideration the personal dimension in ephemeral collective exploration activities, and suggests some potential paths for the study and the deployment of "personalised collective action" settings.

16.2 Literature Review: on the use of user, cognition and creativity groups

16.2.1 Harnessing user innovations

An important part of the literature on user innovation is constrained to the noting that user innovation just happens. Harnessing user innovation simply means identifying those lead users that have innovated. This task can be undertaken, for instance, by marketers who, instead of doing research on standard categories, search for users that have modified the characteristics and the functions of a specific product (von Hippel, 1978c). Another approach is to identify lead users through market research and then organise a joint workshop with product managers (Herstatt and von Hippel, 1992), also referred to as the "lead user method". Various limitations have been identified by scholars, such as the inability of expert enterprise staff to communicate and collaborate with users, because of a communication gap due to technical language leading to a lack of motivation (Olson and Bakke, 2001), by a lack of user's ideas feasibility (Magnusson et al., 2003) or by a need to further explore those ideas in order to be able to take them into strategical consideration (Le Masson and Magnusson, 2003). Finally, another approach is to assign users a well-specified design space where they can innovate (von Hippel, 1990, 1994), such as the case of *Threadless*, printed T-shirts with user-designed prints (Piller, 2010).

In fact, the above approaches address different issues: how to exploit spontaneous user ideas for new product development, how to identify ready-made user innovations to produce industrially, how to exploit user creativity on a given product. This diversity of approaches is also related with the fact that users are "strange" to the enterprises rules, and their participation is likely to be surprising (Olson and Bakke, 2001; Magnusson et al., 2003).

Adopting a broader scope, in order to include "enterprise users innovators", we could also mention the case of platforms as a specific one. Gawer and Cusumano (2002), analysing *Intel*'s strategic principles for platform leadership, highlight the case of the *"PlugFests"*, organised by the enterprise for its partners (Gawer and Cusumano, 2002, pp. 57-60). As they note, between 100 and 200 companies usually attended these events, sending two or three of their best engineers. There, visitors tested the prototypes of their products on the new *Intel* platform, which were built on the basis of the specifications the enterprise had published. Moreover, *Intel* had the chance to receive suggestions from the complementors to improve its platform before releasing it. Still, those meetings, as described by the authors, regarded compatibility and specification testing issues, not the exploration of a given (*Intel*'s, in the particular case) technology's potential.

The question of how to harness user-developer-entrepreneur activity to explore the potential of a given service, includes the above mentioned questions, in the broader notion of exploration. Still, a condition is that users *do explore* a potential. For this to be done, the developers need to comply with the requirement of understanding the underpinning technical language. More over, they have the skills to advance from an idea to a prototype, testing the feasibility of their concepts while they explore them.

The current study will explore this possibility by the study of the *Google Technologies User Group* "hackathon" to which I participated. Before passing to the examination of the case, though, a review of the methods, described by the literature, to provide necessary (technical) knowledge to groups will be useful for the further exploration of the case.

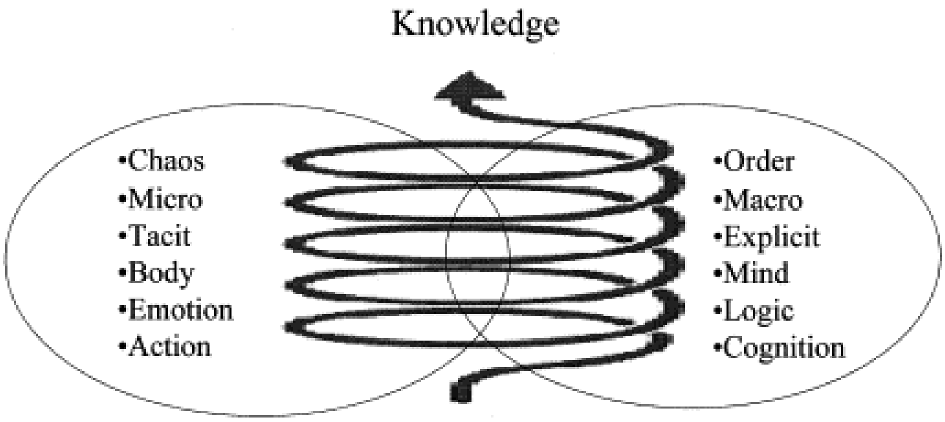


Figure 16.1: Knowledge creation as a spiral process through seemingly antithetical concepts. Quoted from Nonaka et al. (2000).

16.2.2 Knowledge in groups.

Nonaka et al. (2000) criticise the widely encountered confusion in the literature between *"knowledge management"* and *"information management"* that addresses the organisation as an *"information processing machine"*, questioning the problem-solving approach (Nonaka et al., 2000, p. 6), as organisations also *"create and define problems"*. Knowledge creation is thus considered a *'spiral'* process *"through seemingly antithetical concepts"*, such as *'tacit'* and *'explicit'* elements of knowledge, as shown in Figure 16.1.

Reviewing contemporary issues on organizational learning, Nonaka and von Krogh (2009) remark that *"social practices may be necessary, but not sufficient, for understanding organizational knowledge creation"* (Nonaka and von Krogh, 2009, p. 646). Being based on the distinction between 'tacit' and 'explicit' knowledge, they propose that new knowledge as well as new practices may be the outcome of *"knowledge conversion"* from one form to another (Nonaka and von Krogh, 2009, p. 642-647). Hence, they conclude their work by proposing the study of team formation factors and practices that may shed light on the question of the relationship between organisational knowledge and practice:

> *future research on the relationship between organizational knowledge creation and social practice should account for team formation and factors that impact on team performance (Nonaka and von Krogh, 2009, p. 648).*

Sharing a similar framework of analysis and regarding the knowledge created within a group, Erden et al. (2008) propose a hierarchy of group tacit knowledge, as quoted in Figure 16.2.

At the earliest level of the *"Group as assemblages"*, *"group members are as foreigners"* (Erden et al., 2008, p. 10), members are linked by weak ties (Granovetter, 1973), not sharing common memory, understanding or norms.

The advancement at the second level, the one of *"collective action"*, is then performed in three different ways. Firstly, exposure to shared events and experiences contributes to the sharing of the value of "collectively acting". Secondly, through customs, a "tacit memory" of common routines is acquired by the group. Thirdly, a "group culture" emerges, providing a feeling of shared belonging (Erden et al., 2008, p. 11).

At the level of "phronesis", the group obtains some quality characteristics, consolidating its common identity and the shared interests. Erden et al. (2008) describe this level by five

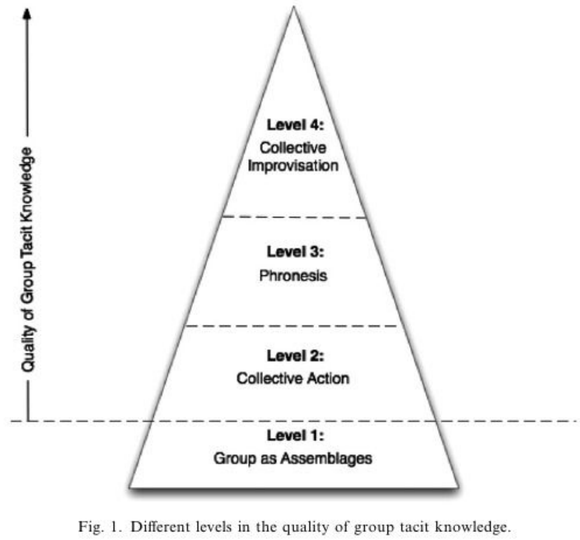


Figure 16.2: Different levels in the quality of group tacit knowledge. Quoted from Erden et al. (2008).

group capabilities, ultimately leading to the capability of the group to manage itself:

- *convert "I" intentions to "we" intentions,*
- *decide on and commit to the "common goodness",*
- *grasp the essence of particular situations,*
- *take the best action for "common goodness" and*
- *manage itself (Erden et al., 2008, p. 12).*

Finally, *"Collective improvisation" is the highest level of tacit knowledge quality a group can achieve.* (Erden et al., 2008, p. 13). At this level, *"there are no predetermined rules and roles"*. Collective action is thus led by collective intuition, based on the groups' "common mind" (Erden et al., 2008, p. 13-14). The scholars share the view of Vera and Crossan (2005) that *"good improvisation is a tool that complements planning efforts but, because of its creative and spontaneous nature, it is not necessarily tied to success, the same way planning is not necessarily associated with success"* (Erden et al., 2008, p. 14)

However, a paradox appears when considering the analysis of Erden et al. in my field, both in the case of the Barcamps (Chapter 15) and the one of the hackathon I will examine in the current chapter: the groups that emerge in such processes seem to immediately cross most of levels mentioned, while in the particular case of a hackathon all participants go from the first one of the *"group as assemblages"* between *"foreigners"* to the fifth one, of *"improvisation"*.

Regarding shared "rituals", an attribute characterizing the second level proposed by Erden et al. (2008), in the cases of group formation we study (in *Barcamps* as well as in *Hackathons*), the procedure to follow is imposed by the events' format from the beginning instead of emerging as a custom in time, as the theory suggests. In other words, it is the procedure which becomes the basis of relationships development and knowledge sharing, and not the opposite.

At the same time, improvisation has to do with creativity. Following Hatchuel (2001) on the distinction between knowledge and concept spaces, we will review the specific literature referring to creative groups.

We will then conclude our review of knowledge-concept groups by the synthesis made through the *KCP® method*.

16.2.3 Creativity in groups

It is highlighted by many management scholars that creativity within a group is a question of rule-breaking, both in what regards the collective processes followed and the object under design (Cunha and Gomes, 2003; Erden et al., 2008; Hatchuel et al., 2011; Le Masson et al., 2011). In what regards business, creativity generally addresses the issue of *exploration* as opposed to the one of *exploitation*, in the sense of *"exploration of new possibilities and the exploitation of old certainties in organizational learning"* (March, 1991).

The most well known method in what concerns collective creativity, used besides early on within the computer industry, has been *brainstorming* (Osborn, 1953; Clark, 1958). While questioning the organisational order and implying intimacy among the participants, brainstorming has been found vulnerable to the 'fixation effect', according to which 'brainstormers' tend to explore a limited scope of concepts, entering the same trajectory (Gillier et al., 2010; Hatchuel et al., 2011; Le Masson et al., 2011, and others).

In Chapter 15 we've seen how the Barcamp conversational setting avoids this effect in practice through the 'bottom-up' division of exploration of a broad topic, while ensuring a focus on particular interests and covering of a wide array of potentially interesting issues.

Still, while creativity literature focusses a lot on the production of "out of the box", original and diverse ideas, it does not provide a solution to a "wide box exploration", that is a creative exploration of a given potential, thus not leading to unrelated ideas, and demanding much work to imagine a possible way of relation to a specific service potential.

16.2.4 KCP®: combining knowledge and creative exploration.

KCP® is an advanced method of creativity, conceived to respond to innovative design issues met during new product development processes. The method aims to structure a collaborative exploration of an innovation field. The expected outcomes are a structured set of innovative concepts for further development and the identification of 'missing' resources and competencies needed to enrich on-going research programmes or external acquisitions (Hatchuel and Weil, 2009; Elmquist and Segrestin, 2009)

The aims of this method, as outlined by the performance criteria defined by its designers, are the following:

- explore the whole conceptual potential of the initial concept,
- involve and support people in a rule-breaking process,
- enable relevant knowledge activation, acquisition and production,
- manage collective acceptance and legitimacy of rules (re) building.

The method is developed in three phases:

1. *K-phase*, where a set of seminars with participants coming from the organisation and its partners 'transfers' knowledge already obtained by the enterprise and indicates potential innovation developments,

2. *C-phase*, where the participant public is divided in subgroups exploring surprising and strongly contracting concepts and

3. *P-phase*, where proposals are issued as a result of the whole process, building a roadmap of immediate solutions, research projects and prototype for the enterprise and its partners.

Interestingly enough, the organisation of this method, applied in various industrial contexts such as the car industry and transportation services, begins by an intense process of learning, as the potential to be explored is specific - related to the enterprise's business. Then, exploration is to be operated through projective concepts, based on this particular knowledge, though projecting a design process into the unknown, obliging participants to search for new knowledge to verify or refute these this concepts. Furthermore, beyond the specific concepts and the 'judgements' on its logical or imaginary state, what is more important is the process itself, as it generates an exploration of trajectories that could not be reviewed if already known trajectories were to be run through.

Unlike Barcamps where various platforms are explored in parallel, the emphasis on a specific enterprise knowledge and potential to be explored at least identifies three conditions for the exploration to succeed. Those conditions, described by the three phases, can be a reference for us in our exploratory study on the means through which platform providers support UDE innovation.

16.2.5 Synthesis: a preliminary description of our object of study.

Specifically in what regards the question of knowledge and information transfer from the platform provider to third parties, a question highlighted in the platform literature (Gawer and Cusumano, 2002, and others), a first observation under the light of organisation learning studies, is that it is not just about information circulation, as the literature on the information economy proposes (Porat, 1977, and others), but it is about a more complex process of collective action's theory and practice.

Using the terms of Nonaka and von Krogh (2009) regarding knowledge, the question becomes one of how service provider's 'tacit knowledge' on a new technology can be 'transferred' to the UDE community on the one hand, and how 'sticky' information UDE's (von Hippel, 1994) is, in turn, 'transferred' back to the provider. Furthermore, regarding creativity and improvisation, unlike the implicit hypothesis of Nonaka and von Krogh (2009) and Erden et al. (2008) that it is undertaken on the basis of the knowledge residing within the group, in the case of UDE innovations this knowledge basis needs to be complemented with a platform-specific one in order for the creativity process to become possible.

Nonaka et al. (2000) and Hatchuel and Weil (2009) while using different entries come to some overlapping conclusions that we will use as a theoretical base to describe the object that we expect to meet when examining the case of provider - UDE interaction *in vivo*.

KCP® method is based on the C-K Theory, distinguishing two major spaces: the knowledge space and the concept space. This theory, initially conceived to describe a design process, can be deductively outlined as the interaction of two spaces, based on Hatchuel and

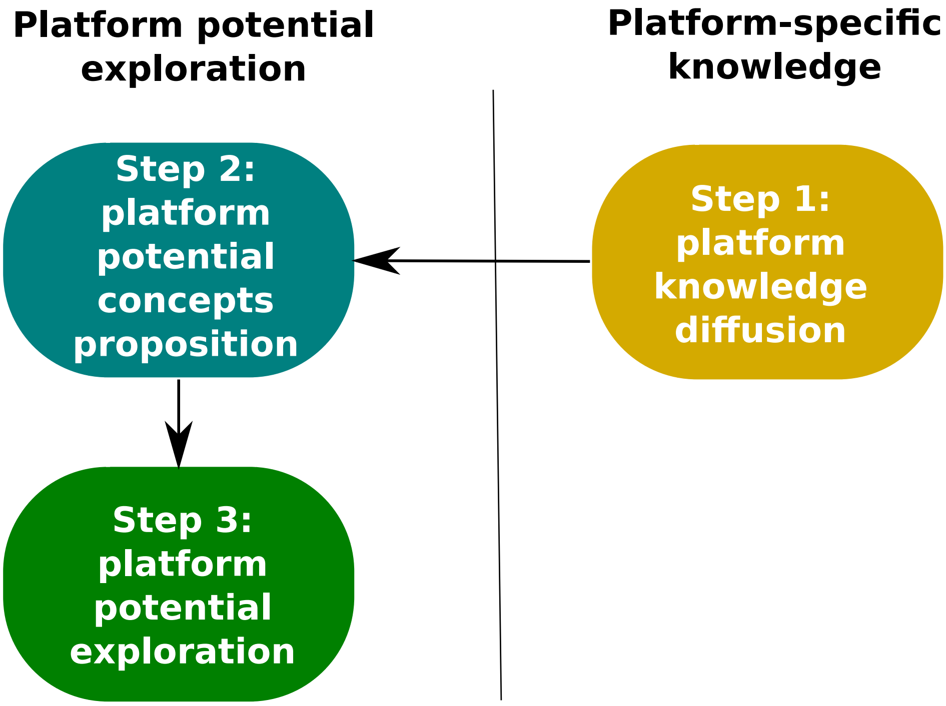


Figure 16.3: The object we wait to meet: a theoretical description (drawing from Nonaka et al. (2000); Hatchuel and Weil (2009)).

Weil (2009) and Hatchuel et al. (2011): the Knowledge space, which consists of propositions that do have logical status (either true or false) and the Concept space, which consists in propositions that may be neither true nor false (*'expansive partitions'*), though demand further exploration in order for the designers to be able to draw a logical conclusion on them.

This abstraction overlaps some of the elements proposed by Nonaka et al. (2000), as outlined in Figure 16.1: "order", "explicit knowledge", "logic" and "cognition" do have a logical status, and thus may be included in the K-space. On the other hand, the space of the "seemingly antithetical" elements may be put in parallel to the C-space, in the sense that one cannot conclude anything about their logical status.

Combining in this way the overlapping fields of the two theoretical approaches within our field of research, we can expect that the nature of the output of the interaction of platform provider and UDE community will be (1) new knowledge, (2) new practices, (3) a roadmap of immediate solutions, research projects and prototypes.

Furthermore, following the pattern of the KCP® method, we can expect that the process will evolve in three steps (shown in Figure):

1. *a platform knowledge diffusion step*, grounded in platform-specific knowledge, where the provider 'initiates' UDEs on the platform specifications.

2. *a platform potential concepts proposition step*, where ideas on 'what to do with the platform' are suggested by UDEs, exploring its potential and

3. *a platform potential exploration step*, where prototypes are proposed or built and a roadmap for immediate solutions and research projects may emerge.

16.3 Research methodology: a phenomenon-based approach

To access this field I will use both observatory participation and monument analysis methodological approaches, already discussed in the first chapter of the current approach. On the one hand, I will use information coming from my own participation in the Hackathon organized in August 2010 at Google's headquarters, in Mountain View, California. I will complement this material with the traces that the participants left, mainly descriptions of their own creations as well as the program of the meeting itself.

Distinction step.

During the first step (Section 16.4), I will study the phenomenon of the 'hackathon', focussing on its peculiarities, by using a descriptive approach (von Krogh et al., 2012). As we will see, this type of event, while also like Barcamps, having an intimate ambiance, is a lot more structured and intense.

Exploration step.

In Section 16.5, I will further explore the 'hackathon' event, expounding the content of both seminars and applications developed. I will perform an qualitative as well as a quantitative analysis upon the applications, proposing a categorisation of the trajectories of User-Developer-Entrepreneur (UDE) innovation in the Web context. I will also use the content of the seminars to test by induction our proposition of *foggy competition* in what regards the state of the Web industry.

Design step.

Provided the results of the Sections 16.4 and 16.5, I proceed to a proposal of a Research design regarding ephemeral action settings, according to which different phases of the process correspond to different roles of the implied actors, in relation to the knowledge and the concepts in use. I will also suggest that, beyond the programming itself, participants' personality appears to play an important role in such settings.

16.4 Distinction step: general characteristics of a "hackathon"

The *Silicon Valley Google Technologies User Group Campout (SVGTUG-Campout)* event was divided in two parts. The first part consisted of a series of *"Technical Talks"* and will be analytically studied in section 16.4.1 and 16.5.1. Globally, it is about a form of open seminars presenting the latest "state of the art" Web development. Some of these talks were broadcasted on the Web during the event (live streaming) while the slides of most of the presentations were also made available online afterwards.

The second part of the event regarded user-conceived and developed projects. This part will be studied in sections 16.4.2 and 16.5.2. Those projects were to be developed using the technology of reference for the event, and especially *Google* platforms, such as *Google Chrome* navigator, the *Android* smart phone operating system and the *OpenSocial* social network development platform. The event registration page characteristically mentioned in the FAQ (Frequently Asked Questions):

> **FAQ**
> **Must my team develop a Google Chrome Extension?**
> No. You are welcome to develop an Android, OpenSocial, etc... application over the weekend. The emphasis is on HTML5, and Google Chrome Extensions are just one easy way to create an HTML5 application. If you can create an application using HTML5 capabilities in other ways, knock yourself out[4].

After the talks, users had 90 seconds to present their concepts to the rest of the participants and recruit a team to work throughout the weekend. The full programme is shown in Table 16.2. Seminar speakers were mainly from three backgrounds: researchers on Web design, Web development "cookbooks" authors and *Google* engineers.

16.4.1 The seminars.

After the registration, participants gathered in the seminar room, arranged like a classroom. The seminars started with a presentation of a Google Programs Engineer on the *HTML5* protocol (*"Tour of HTML5 Features"*, as mentioned in the schedule)[5]. He highlighted the new features supported by it (graphics, video/audio playback, notifications, real-time communication, ability to locally store data on the client's computer) and provided examples of their embodiment in a Web page. He closed his presentation inviting the participants to begin hacking (*"Let's hack!"*).

The second seminar regarded an *"Introduction to the Google Chrome Web Store"*, presented by a member of the *Google Chrome browser* technical team. After going through a short timeline of the Google Chrome project, he emphasised the distinction between end users and developers, noting the entrepreneurial reasoning of the latter and underlining the importance of the question *"how do I make more money?"* for them. Regarding the features of Google Chrome, he emphasized the fact that *"the link to the store is built in to the browser"*. Hence a user base for *Chrome* applications (extensions) was already there, since the navigator had been broadly distributed. Developers could have access to this user base submitting a Web-based application to the *Google Chrome* application market. It was reported that the store addressed 7 million users, according to the latest data. The introduction included some technical advice of business importance, such as *"Ask for the permission to geolocate at the beginning of the app installation"*, an information that is useful when wanting to link a client to a specific market, for instance through advertisement. He closed his introduction by providing the links to the documentation regarding participation in the *Chrome* application market.

The third seminar was introduced by a freelance web developer who also edited handbooks on Web development. Her presentation concerned the development of Web-based applications oriented towards smart phone terminals[7]. The main axis of this talk was on the

[4]svgtugcampout FAQ, URL: https://sites.google.com/site/svgtugcampout/faq Retrieved on October 12, 2012.

[5]Paul Irish, "Tour of HTML5 Features", presentation at GTUG event. Available at the URL: http://html5-gtug-campout2010.appspot.com/ Page retrieved on 8 October, 2012.

[7]"Developing for the Mobile Web", Estelle Weyl. Presentation available online at the following URL: https://sites.google.com/site/svgtugcampout/schedule/mobileweb.pptx?attredirects=0 Retrieved on 8 October 2012.

Gtug Campout Schedule 13-15/08/2010		
Friday: August 13, 2010		
12:30-1:00pm	Registration	
1:00-5:00pm	Technical Talks (broadcast live on ustream)	
	1:00 - 2:00	Tour of HTML5 Features, Paul Irish (slides)
	2:00 - 2:30	Intro to the Google Chrome Web Store, Erik Kay
	2:30 - 3:00	Break
	3:00 - 4:00	Developing for the Mobile Web, Estelle Weyl (slides)
	4:00 - 5:00	Making Your Web Apps Fast, Steve Souders (slides)
6:30-7:00pm	Registration	
7:00-7:30pm	Welcome & Kickoff Talk, Alex Russell	
7:30-8:30pm	Pitch sessions (broadcast live on ustream)	
8:30-onward	Team forming and working	
Saturday: August 14, 2010		
8:00-10:00am	Coffee and muffins	
12:00-1:00pm	lunch available	
	Implementable CSS talk by Estelle Weyl (slides | stream)	
6:00-8:00pm	dinner available	
Sunday: August 15, 2010		
8:00-10:00am	coffee and muffins	
12:00-1:00pm	lunch available	
3:30pm	Registration: Demo Night Ticket Holders	
4:30-6:30pm	Teams demo their apps (broadcast live on ustream)	
6:30-8:00pm	Dinner	
8:00pm	Contest Winners Announced	

Table 16.2: Gtug campout: the schedule of the event[6].

	Native	Web
Cosmetics	✓	✓
Functionality	✓	✓*
Development		✓
Testing		✓
Distribution		✓
Payment		✓
Support		✓

Table 16.3: Seminar presentation table, marking the advantages of Web-based applications.

comparison between the *"Web applications"* and *"native applications"*[8]. This presentation claimed that the Web-based applications were superior than the native ones, as they fulfilled a number of additional requirements, shown in Table 16.3. The presentation table emphasized the ability for developers to update their application, (*Development* and *Testing*), as well as the ability to establish a commercial channel with the end users (*Distribution*, *Payment* and *Support*). Still, user interface parameters are relatively underestimated by the table, as they are described as "Cosmetics", and constitute a field of "native apps" superiority over "Web apps".

The "Technical Talks" session closed with the seminar *"Making Your Web Apps Fast"*[9], which discussed issues about improving the display speed of Web sites by respecting some rules on the design of Web sites. The main argument of the presentation was that even well-known Web sites are slow to display the content to end users. According to the speaker, the economical impact of speed performance is crucial, as the slower the site, the lower the engagement of visitors (a percentage will always not wait and surf to other sites). The speaker commented that there is little emphasis on performance criteria regarding speed in the wider computer science community.

Many of this last seminar's arguments were based on a joint report of a Principal Development Lead Engineer from *Microsoft* and a Decision Support Engineering Analyst from *Google*[10]. The conclusions of that study are shown in the Figure 16.4. As the two engineers had explained in their report, the need for speed has been known in the community, however the trade-off between speed and other parameters, such as revenue or user satisfaction were under-explored. Their joint experiment on *Bing* and *Google Search* user samples provided the correlations shown in the Figure. There, the business performance criteria examined are the number of Queries/User (for instance Web searches), the Revenue/User, the clicks the user

[8]While both types of applications use the Web for the exchange of information, in our view the main difference resides in the user-interfaces. In the first type, the user interface is the Web browser - installed either in the smart phone or in one's PC. In the second type, access on Web information is made by the specific application, without the interference of a browser. We are going to further discuss this topic on the conversation section.

[9]Making Your Web Apps Fast, Steve Souders, presentation in GTUG, Mountain View, Cal. USA, August 13, 2010. Copy available in the URL: stevesouders.com/docs/gtug-20100813.pptx Retrieved on October 8, 2012.

[10]Eric Schurman, Jake Brutlag, Performance related changes and their user impact, Velocity Web Performance and Operations Conference, 2009. Video of the presentation available at: The User and Business Impact of Server Delays, Additional Bytes, and HTTP Chunking in Web Search, 06/23/2009. URL: http://velocityconf.com/velocity2009/public/schedule/detail/8523 . Retrieved on October 9, 2012.

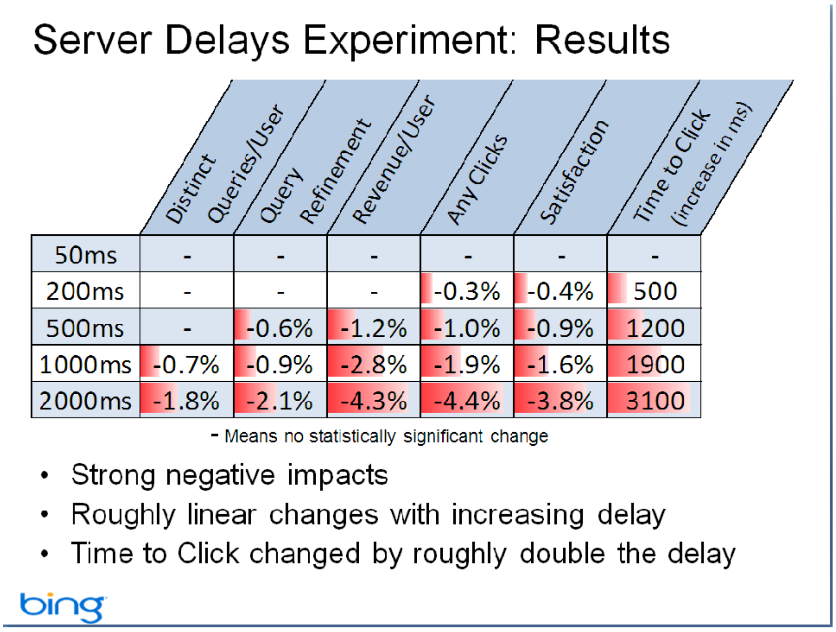


Figure 16.4: Results of experiments on speed as a performance criterion for Web sites. Source: (Schurman and Brutlag, 2009)[12].

operates and the overall user engagement (activity as frequency of clicks and satisfaction).

Hence, the last seminar of the *SVGTUG - Campout* after emphasizing this performance criterion, proposed some design tips that could improve a site's speed (for instance *"Scripts should be at the bottom of the page"*).

16.4.2 "Pitch sessions": entrepreneurs recruiting developers to a concept.

A "*pitch session*" is an event were project leaders should briefly present their concepts to attract volunteers from the public to work with, during the last two days of a "*hackathon*". Unlike the case of the '*'mashpits"* in the "Barcamps", we've examined in Section 15.4.4 on page 308, where concepts emerge after discussions, in this case project leaders should have prepared their proposals in advance, as they only have 90 seconds to talk about them and to convince volunteers to join them. In our case, there were thirty eight such concepts presented to the *"Campout"* participants. Twenty two of them manage to recruit members and build a team to work with, while two were merged because of their concept relevance.

The complete list of the "pitches" proposed is shown in Table B.1, page 397. Each line of the table was completed through a specific form by those proposing concepts to develop ("pitching") on the first day of the event. The fields completed are the following: Name of person to propose the project (*"Who pitched"*), URL were a demo of the application would be available (*"Demo URL"*), names of the participants in the project team (*"Team members"*), a description of the concept , usually in terms of motivation and functionalities (*"Project Description"*), the skills of the developers needed for the project (*"Developers needed (# and skills)"*), additional notes (*"Other special notes to developers"*), the team size (*"Your team size"*) and a self-presentation of the project leader (*"One sentence introduction for yourself"*).

Out of thirty eight project proposals (*"pitches"*), twenty nine were presented during the last day of the *Campout*. Some proposals were merged (*proposals 9, 18, 19, 33*), while the rest were not continued. Globally the criterion of the project advancement was participation.

Table 16.4 lists those projects that were finally presented with a short description, as induced from the complete table.

Table 16.4: List of projects developed and presented on the last day.

Team #	Project name	Short description
20	Allele Interaction	Image sharing within Social Networks.
34	arc.js	A JS access layer for the post message protocol.
5	Copy and Paste	Single feature app for selecting similar items on a page.
14	Day Trader Tool	Display historical data for financial instruments.
24	DoUSeeMe	Temporary Geo location sharing web app.
18,19	Dynamic memory Game	A cross between the Memory Card Game and Tetris
-	Extend Etherpad Collaborative Editor	An Etherpad platform extension using HTML 5.
35	Foreigner	[no description available]
2	HTML 5 Store	A web - based app market.
30	Html 5 Video Klippr	Chrome extension - for content sharing accross Web platforms
10	HTML 5 widgets for DreamFace	Use as much HTML5 as possible in DreamFace platform.
22	Kitlist to cloud	Put a Job postings list into a “cloud” and sort it.
7	LifeAlert	An Androind app sending phone GPS coordinates to a list of contacts
-	LightStalking Library	[no description available]
-	LocalPad	[no description available]
21	meal claim	An Android app for flight meal pay calculation.
-	Merlin Finger	Something magical (graphics)
26	Party Page Partners	A real time event animation app.
36	Pull Prediction	Social game to predict the markets
29	RashoMonty	Extract Data from websites and display a comparative analysis.
11	real time analytics	An analytics tool.
8	Rescue Me	Find out the nearby hospitals around me.
12	Shout shopping browser extension	A browser extension publishing 'shopping related' product pages.

. . . continued in the next page . . .

. . . continues from previous page . . .

Team #	Project name	Short description
15	Slidio	An image slideshow with audio in the background.
17	Soundweaver	HTML- based composition tool.
16	SweetGeo	LinkedIn, Facebook, Foursquare, Meetup and Craigslist mashup
6	Terra Traveler	A "Facebook" for Travelers.
23	TetriNet	Embody TetriNET game in HTML5.
3	Togethr	Event invitation app.

End of table

The winners of the *Campout* were voted by the participant public. The prizes were: to be featured in the Chromium blog/Twitter, App Engine credits for their app, Google swag (advertisement item with the enterprise's logo, usually a T-Shirt), 1 Free *O'Reilly* (well known editions of hackers' handbooks) Ebook per team member. The first winner got in addition an *Adobe Creative Suite* (since Adobe was sponsoring the event) and a lunch with a *Google* engineering team.

The winner of the third place was the *Dynamic Memory Game*, described by the person who coined it as follows (listed in the Table B.1 on page 402):

> *This is a cross between the Memory Card Game and TetrisCards drop face down from the top of the screen and slide to the bottom, stacking edge on edge. The user clicks on two cards temporarily revealing their faces and if identical, the pair is removed. The user must remember the card locations and find matches before the cards pile to the top of the screen. The game ends when one stack of cards reaches the top.*

The second winner was *SoundWaver*, a piano playing application on *HTML5* developed by two young participants.

The first winner was the *SweetGeo* application, which was a *mashup*, as it made used of *APIs* from different Web services platforms, Social Networking ones in particular. The description of the application (listed on page 402) was the following:

> *[SweetGeo] Takes the concept of LinkedIn, Facebook, Foursquare, Meetup, Craigslist and mashes them together. Check in to location, post profile with interests, needs etc. As people check in, they can view and post profiles, post ads etc. The person you [are] looking for or [the] person who can help you might be [the] person who just arrived.*

16.4.3 Discussion: hackathon exploration roles and phases

The hackathon to which I participated managed to go through all four levels described by Erden et al. (2008) in three days. "Groups as foreigners" were constituted, they acted collectively, and developed a common, temporary identity as a concept-based group, improvising on this concept. In addition, given the audience was constituted by developers, communication between *Google* staff and participants was not a difficult issue, as in the cases of non-developer lead users (Olson and Bakke, 2001), as UDEs not only were skilled developers, but were also interested in learning about a new technology, potentially useful to them as well. In parallel, the construction of specific application, implied by the hackathon setting, also provided a solution to the problem of idea formulation (Magnusson et al., 2003; Le Masson and Magnusson, 2003), as it forced UDEs not only to have a specific concept to propose, but also that the concept be attractive to others and eventually be developed into an application in two days - and nights - time.

The hackathon setting included requirements similar to the *KCP* method. There was a phase where knowledge was transferred and a phase where subgroups explored concepts and a roadmap with immediate solutions could be built at the end, drawing on participants materialisations. Some differences existed regarding a more specific role division between provider and UDEs as well as a more advanced kind of proposals through elaborated and often very functional prototypes. Even more importantly, the design of the hackathon process takes into account what Akrich et al. (2002) call *"the art of interessement"*, as attractiveness is a requirement for both seminars and projects.

More specifically, we can describe the process of the hackathon in three phases: 1) an "updated state of the art" transfer, 2) an "attractive concept exploration" and 3) an "overall concept evaluation and sharing". Figure 16.5 summarises schematically those phases, while these phases are further described in the following paragraphs.

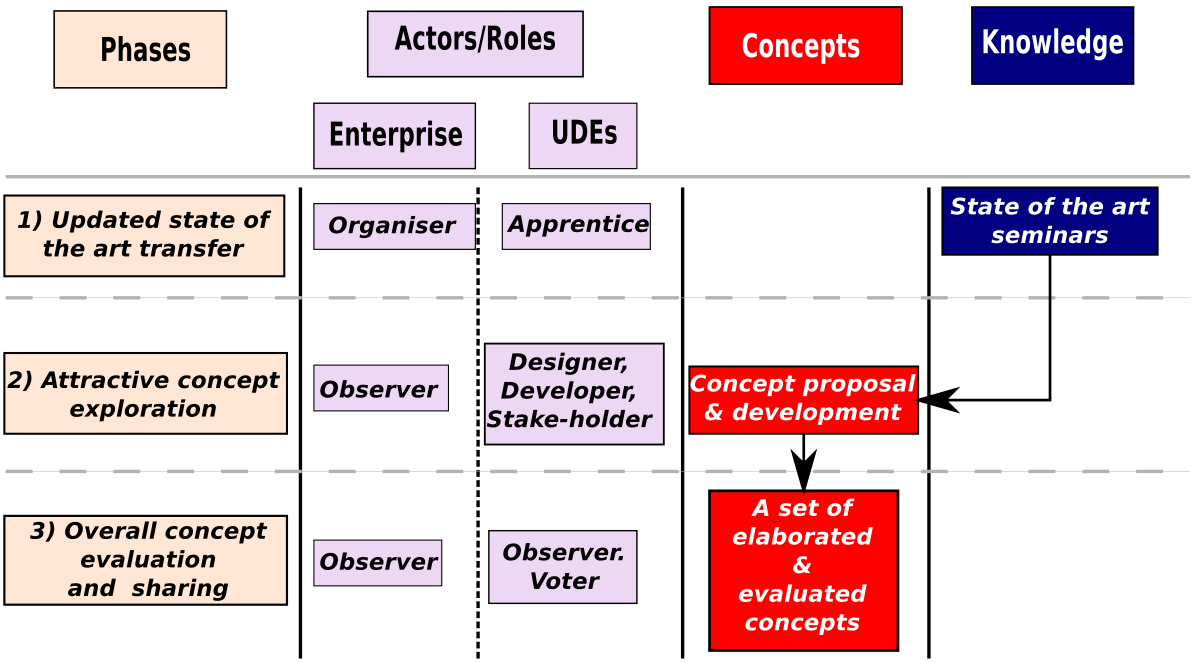


Figure 16.5: Attributes of a hackathon. Three phases for UDE-based exploration.

Phase 1: Updated state of the art transfer

During the first phase, where an updated state of the art is transferred to participants through the initial seminars, UDEs have the opportunity to learn about the latest technological evolutions, while the service provider has the opportunity to "seduce" them, highlighting the new opportunities these evolutions open up for developers. During the hackathon where I participated, the new features of *HTML5* were presented (*graphics, video/audio playback etc.*), while new performance criteria were introduced, namely the one of Web site speed.

In addition, many presentations talked about the business opportunities opening up for developers (e.g. *Chrome* browser applications), as well as the competitive advantages this technology may have against other (e.g. *HTML5 applications* versus *native applications*). Moreover, participants had access to some business related “tips and tricks” (such as asking users for permission to locate their geographical position for them to be more interesting to advertisers).

Therefore, this phase addressed both the developer and the entrepreneurial identity of participants, while providing them with a clear view of the “foggy competition” field, with the latest updates on performance criteria and features to exploit.

Phase 2: Attractive concept exploration

During the second phase, where some UDEs proposed to their peers concepts to develop, an early evaluation of the concepts took place by the personal interest expressed by the public to participate or not in a project. Then, during development process, participants had the opportunity to test the new practices implied by the new technology. Service providers watched the process and were available for developers questions. This way, developers could learn more, while the service provider could also record the questions and the problems UDEs face in action (for instance documentation clarity).

Hence, service provider staff had not only the opportunity to receive ideas from users, but to actually have a sample of the action of developers, usually undertaken individually or in small groups in a dispersed manner.

Phase 3: Overall exploration evaluation and concept sharing

The final project presentation gave the opportunity to all participants to have an overview of the concepts explored and to vote for the best. This added a democratic dimension to the competition process for UDEs, while it also provided an evaluation of the concepts, as presented by the teams.

This phase also added a sense of community to the participants, having been part of a common experience.

16.5 Exploration step: a focus on cognitive aspects

The current section advances the exploration, intensifying data gathering about initial focal concepts. It further reflects on the particular knowledge taught in the seminars and further explores the particular innovation trajectories, as resulted in participant's creations. It concludes with the following propositions:

- During my study period, Web applications were considered as a potential disruption, which the hackathon aimed to explore. Thus, a consideration of the current period as a “foggy competition” phase for Web services, where exploitation goes hand to hand with exploration, justifies the use of UDEs as explorers.

- A closer study of the resulting applications reveals that

  - Hackathon exploration is not only based on the knowledge transferred by the provider, it is also based on the previous knowledge of the UDEs.

- The resulting trajectories explored can be categorised according to the knowledge used by the developers, as well as by two guiding concepts. Developer's knowledge was either enterprise related or intimate, while the guiding concepts were either the computer as an instrument or the Web. I compare those results to a *"broad range of literature"* (Eisenhardt, 1989, p. 544), which enables me to propose four types of trajectories: a) Tools, b) Tools of the self, c) Channels and d) Collective intimacy.

In the discussion part, the trajectories explored will supported by a broader literature than previously, particularly useful to the understanding of the stakes with the specific fields.

16.5.1 Seminars: the state of the art of HTML5.

The overall emphasis of the seminar was to support Web development as a major innovation trajectory and promote *Google* platforms, getting speedy feedback from the side of the developers.

The promise of *HTML5* is to enable what Nelson had already advocated back in 1974, i.e. to go beyond application forms and imagine other types of end user interfaces, now in the framework of Web services. From this perspective, games applications were very illustrative of this new potential given by the new protocols. The strategical question for platform providers as well as the community of developers was whether the Web (as understood as the answer to the question 'what can end-users access using a navigator?') can be reliable, both on the technical and the entrepreneurial level, for developers/entrepreneurs to base their business on. In this context, *Google* promoted its Web store for applications, embedded within its *Chrome* browser.

This type of development was often opposed to the trajectory of 'native applications'. Those applications being the ones that are particular to a device, like the PC or smart phones.

However, the content of the seminars themselves revealed that Web-based applications are not yet rationalized. The most typical example was that of the results of the study on performance criteria on the speed parameter, which illustrated that the industry is yet working for a rationalisation-to-come.

16.5.2 Final projects: knowledge bases and conceptual trajectories

The final projects presented at the *Campout*, listed in table 16.4 (page 331), are issued from UDE groups' action during the three days of the event. Using the theory on user innovation, we should assume that these creations were based on users' "sticky" knowledge (von Hippel, 1994). Thus, the project list provides us with a capacity to look into the specificities of user knowledge in the framework of our field and render it less "sticky", identifying the important categories of use-related knowledge as well as the main conceptual trajectories explored by the participants.

Figure 16.6 analyses the data that are found in the previous section regarding user projects. Judging from the concepts' analytical description (annexed in Table B.1 on page 397) we identify two major knowledge bases used: knowledge about the enterprise and knowledge about intimacy. Both knowledge bases are rather implicit or empirical and not necessarily scientific. In addition, we identify two major conceptual trajectories: the computer as an instrument and the Web.

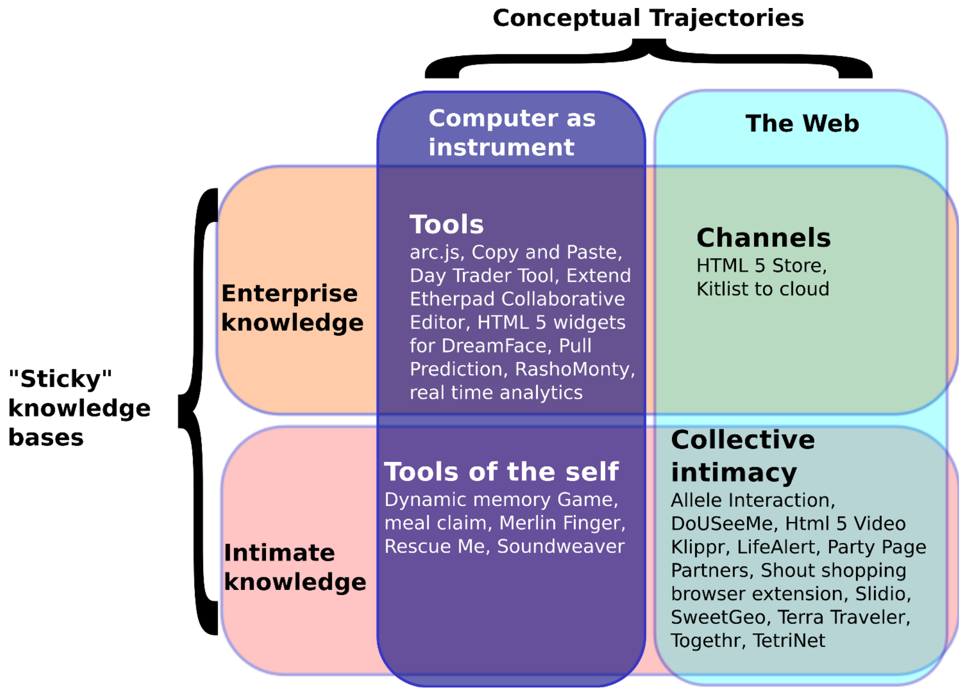


Figure 16.6: Data analysis: cognitive and conceptual categorization of users' innovation in the *Google Campout* hackathon.

Tools

The conceptual trajectory of enterprise and operations instruments (or tools) is one of the oldest ones in the computer industry. It includes concepts of the Operation Research (OR) approach[13], as well as "Traditional Decision-Making Techniques" (Simon, 1965, p. 62).

Among the projects listed, *"Pull Prediction"*, *"RashoMonty"*, *"real time analytics"* and the *"Day Trader Tool"*, all enter into the framework of *Operations Research*. The projects *arc.js*, *"Copy and Paste"* and *"Etherpad Collaborative Editor"*, enter into the category of *Electronic data processing*. The latter two applications specifically enter into what Simon would call "Traditional Decision-Making Techniques", including a '*'programmed part" ("Clerical routine")* and *"non-programmed"* one (*"Judgement, intuition and creativity"*) (Simon, 1965, p. 62).

Channels

The concept of *channels* is related to the "information economy's" conceptual trajectory, emphasizing the importance of transaction (Wallis and North, 1986) and information (Porat, 1977) costs reduction as well as network externalities (Katz and Shapiro, 1985; Farrell and Klemperer, 2007) and two-sided markets (Caillaud and Jullien, 2003; Rochet and Tirole, 2004). Hence, I use the concept of channels to summarise those economical approaches

[13] *OR* initially consisted in the use of mathematical analysis as a tool for understanding how new weapons could enhance U.S. security (Ceruzzi, 2008, p. 23). Yet, *OR* also opened the way for consulting and organisation openness in the basis of knowledge, as it implied the participation of non-military persons to participate in operational decisions, something that was "heretical" to the military hierarchy of the time (Morse, 1977). *OR* introduced in management methods such as the *queuing theory* (applied for instance in air traffic control, telephone centres, assembly-line manufacturing), *simulation* (applied in economy as the first step of its general diffusion as a method) and in logistics.

that, despite their diversity, emerge when it comes to Web services study, since they all value network capacities.

On the one hand, in the case of the *"HTML5 Store"* (later named *HTML Mart*), we already analysed in the part I, enters into the category of markets, generally following the approach of transaction costs diminution (Wallis and North, 1986). On the other hand, *"Kitlist to Cloud"*, referring to the diffusion and elaboration of a work position announcement list to a decentralized network of computers ("cloud"), enters the general approach of a channel as a means of "information costs reduction" (Porat, 1977).

Intimacy-based concepts

The concept of intimacy comes to encompass a set of different approaches to identity[14]. In the specific field of information technologies, I distinguish two forms of intimacy: a) the one that users develop with their computer, seeing it as a "second self" (Turkle, 1984), and I describe as "tools for the self"[15] and b) the one that users develop through a computer mediated interaction with other users, that I call "collective intimacy' '[16].

Regarding the intimacy knowledge base, user developers exploit implicit knowledge residing in the culture of "hacker communities". Regarding the instrumental trajectory of computers, we identify a set of projects elaborated during the event that we characterise as "instruments of the self". These tools are conceived for 'solitary' use and include the applications *"Dynamic memory Game"*, *"mail claim"*, *"Rescue Me"*, *"Soundwaver"* and *"Merlin Finger"*.

The last category is the one of "collective intimacy", which is also is the most popular one among the projects presented. All applications refer to a collective intimacy, usually represented in the technical specifications by the acquisition of a "list of friends" from a major Web platform (*Gmail*, *Facebook*, *Twitter* and others) through their APIs. The applications entering in this category are the following: *"Allele Interaction"*, *"DoUSeeMe"*, *"Html 5 Video Klippr"*, *"LifeAlert"*, *"Party Page Partners"*, *"Shout shopping browser extension"*, *"Slidio"*, *"SweetGeo"*, *"Terra Traveler"*, *"Togethr"*, *"TetriNet"*.

---

[14] I would like to warmly thank Philippe Lefebvre and Anne-François Schmid for our very stimulating discussions on the concept of "collective intimacy", of which the richness goes beyond the scope of the current study.

[15] Turkle (1984) operated an ethnographic study on how computers influence how "we think about our own". Studying children, adolescent and professional programmers, she observed their experience with the computer and proposed that 'a computer program is a reflection of its programmer's mind' (Turkle, 1984, p. 19). Hence, this type of human-computer interaction which is self-referential projects the computer as a tool for the self.

[16] In a recent article, Hounkpatin et al. (2011) describe a case where a therapist chooses to intervene in order to solve an identity based problem of Alexandra, one of their clients. The therapists managed to help her by establishing a "collective intimacy" between her, her family and the therapists. As the authors note, *"this intimacy isn't private at all, but clearly collective (. . .) [and is] capable to nourish the persons in their transformations"*.

The concept of "collective intimacy" is coherent with sociological research on the use of Web services. Particularly in what regards the field of contemporary Web services, *Réseaux* (Networks) Journal. Synthesising the research on the field, Cardon (2008) argues that "Web 2.0" services can be categorised through the "identity formats" dividing Web 2.0 services to *"civil identity"*, *"active identity"*, *"virtual identity"* and *"narrative identity"* ones. Cardon and Delaunay-Téterel (2006a) in one of the early studies of this "school", explored the *"production of the self"* through a user interaction within the *blogosphere*. They analyse how, through using the features of a *blogging* platform (such as *posting, linking, commenting* etc.), users build their own identity while interacting with "their public".

16.5.3 Discussion. "Un-sticking" UDE knowledge: the importance of personality

As analysed in paragraph 16.5.1, the early seminars on *HTML5* illustrated an on-going, though not yet complete, rationalisation process of *HTML5* technology. The potential of a disruption (Christensen, 1997) of applications business by entirely Web-based ones was the question of exploration for the service provider. Moreover, what the study of the eventual creations revealed is that, beyond markets and technologies, the personal dimension played a crucial role for an important part of the projects. Commercial and instrumental projects (as positioned in the trajectories of *channels* and *tools*) were somehow expected, as it was a common ground between the UDEs and the service provider. Yet, the intimacy-based projects (as positioned in the trajectories of *collective intimacy* and *tools of the self*) have been rather surprising. These projects, though, are not completely covered by the user innovation literature, as one might have expected.

Figure 16.7 further synthesises the trajectories explored during the hackathon, already discussed in the previous paragraph. The exploration was based on three main knowledge bases: technological, entrepreneurial and intimate knowledge. While the seminars provided knowledge on technology and entrepreneurship, intimate knowledge determined an important part of what developers eventually wanted to develop as an application during those three days.

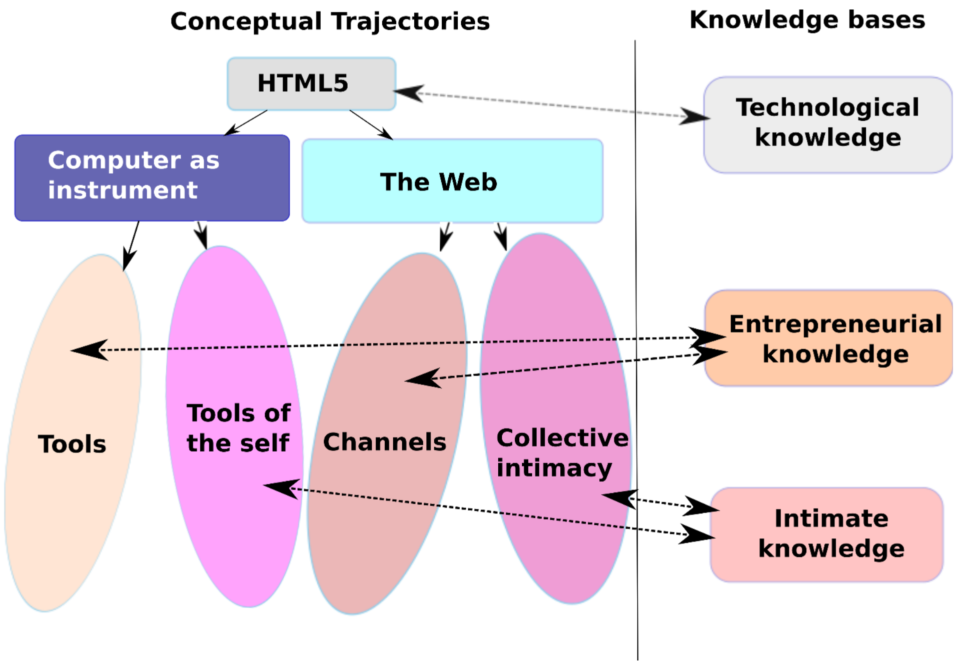


Figure 16.7: Data synthesis: trajectories explored by UDEs during the hackathon and the related knowledge bases.

The following paragraph will further discuss the personal dimension, less highlighted by the literature.

On the personal dimension of innovation

Two of the types of innovations identified are related to the intimate sphere of their developers, the *tools of the self* and the *collective intimacy* ones. Still, the literature on knowledge and creativity in groups, reviewed at the beginning of this chapter (March, 1991; Nonaka et al., 2000; Erden et al., 2008; Hatchuel, 2001; Gillier et al., 2010; Hatchuel and Weil, 2009; Le Masson et al., 2011) does not dedicate any particular attention to the personal dimension. It is not clear whether those studies, when based on a specific field investigation, have encountered phenomena of personality influence on innovative projects or if the authors did not chose to evaluate such phenomena. In any case, personality is less an issue in management research, as actor identity and its influence on innovation is studied mainly on the basis of organisational roles.

At the same time, this dimension is not covered by the literature on user innovation, either. The notion of "use-related sticky information" (von Hippel, 1998, 2005), while being broad enough to describe the innovative activity of both individual and enterprise users, is too broad to address the specific issue of individual personality.

This finding thus calls for further research on an unexplored field of "personalised collective action", where the attributes of the intimate sphere of each participant should be taken into account.

16.6 Design step: taking personality into consideration

Previous sections have shown how hackathons produce an "ad hoc" collective action, creating groups that reach to the ultimate level of collectivity, as proposed by (Erden et al., 2008) in three days. Paragraph 16.4.3 proposed a three phase analysis of hackathons, during which the organiser becomes observer, while the UDEs change roles according to each phase. The three phases (updated state of the art transfer, attractive concept exploration, overall concept evaluation and sharing) are coherent with the phases as synthesised in the literature review part (paragraph 16.2.5), giving a more specific emphasis to seminar and concept to explore attractiveness.

Furthermore, the previous paragraph highlighted a more subtle dimension of the process, noting the role of developers' personality, as expressed by those projects that make used of intimacy-based concepts. However, the personal dimension, while found to be important to my specific field, does not enter into most analytical frameworks in management. Therefore, a research design is proposed, as shown in Figure 16.8.

As suggested by the schema, the personal dimension could be taken into account during the first phase, by proposing "state of the self seminars". Like the technology and market oriented seminars, such seminars could address the personality of participants, which is found to be a resource for innovation. That would also imply a new role for UDEs as well, the one of the "self-exponent", expressing himself or herself during this seminar. Regarding the second phase, the role of "inventive exponent" would not change, since an "intimate-collective" process is already observed as taking place in these events. However, the meaning of a specific component in the first phase would be to reinforce this role.

This proposal is rather counter-intuitive for management studies, as "state of the self seminars" rather refer to spare time activities of a specific public. This attribute is proposed as an element of research design for further study (von Krogh et al., 2012). Some of the directions of such a study are already indicated by other scholars, though their analysis is undertaken at a different level. More precisely, studies exploring the effects of "social software" instruments adoption within the enterprise context highlight self-expression as an important and challenging side effect (Denyer et al., 2011, and others). While from a work division perspective this effect is rather problematic, my study proposes to face it

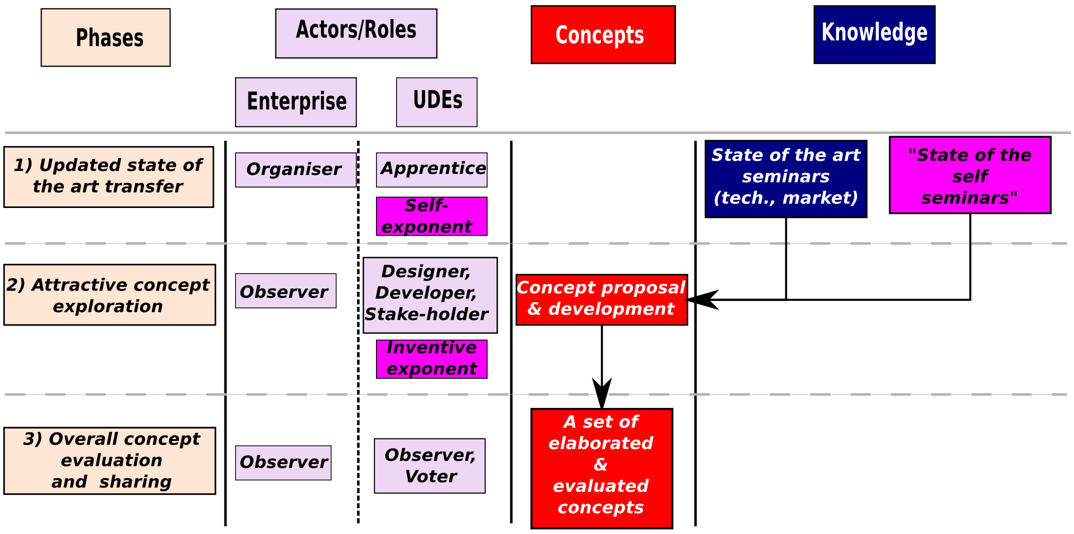


Figure 16.8: A hackathon structure revisited: taking into account the personality dimension.

differently, as an attribute useful for potential exploration. Even though methods fostering and cultivating self-expression are rather rare within the enterprise context, other contexts of collective action, such as role playing, may provide more in-depth insights for the further study of this element.

16.7 Conclusion

The Hackathon case revealed a possibility to harness UDE activity for a rapid service potential exploration. Unlike user innovation literature, UDE innovative activity was "provoked" by the meeting itself, instead of being produced and diffused within a user community.

The meeting format fostered a "rapid maturity" of groups' knowledge, as all levels in the quality of group tacit knowledge described by Erden et al. (2008) were crossed during three days for an important number of groups. Compared to other forms of knowledge and creativity groups management, the design of the event advantaged the evaluation of the concepts by the participant public throughout the whole process.

The study of the content of both seminars and UDE creations has confirmed a "foggy competition" state of the Web applications domain. In addition, the process itself has revealed an important dimension of subjectivity, namely a strong influence of developers' intimate knowledge, in the outcome of their creations, something that has not been identified before by the literature studying groups and innovation.

Chapter 17

Harnessing UDEs activity for service potential exploitation: the cases of developer support forums

Contents

17.1 Introduction

As already discussed in the beginning of the current Part, the relationship between enterprises and developer communities remains an open question within the innovation management literature. The current chapter explores the question of how enterprises can harness UDE activity for their service potential exploitation, by studying online *Facebook* and *Google Maps* developer support forums.

As these forums appear as places where developers can report "bugs", I begin my exploration by a qualitative study of the field as a problem-solving case and compare it to other approaches proposed by the literature. Unlike other models, in this field it is the user-developer-entrepreneurs (UDEs) that set the problems, and the enterprise that undertakes their resolution.

By further exploring this field through the application of problem-solving performance criteria, I find that they are characterised by a low problem-solving rate, thus I project the hypothesis of another function to these forums.

Then, I return to the forum conversations, aiming at characterizing the expertise of the enterprise employees in charge of the process. I propose that an original expertise emerges, the one of the "curator", of which the activity is broader from problem-solving, suggesting that UDE activity exploitation may require the management of Enterprise-UDE "empathy".

17.1.1 Chapter Overview

Table 17.1 provides an overview of the current chapter. In order to configure the ways in which enterprises can harness UDE activity for their service exploitation, I will explore the field of developers support using a phenomenon-based approach (von Krogh et al., 2012). The advantage of this field lies in the fact that the common element of people participating in those forums is the development of applications on the basis of the specific service. Hence, a UDE exploitation activity already takes place during the discussions I will study.

Distinction step

The first step of the current investigation (Section 17.2) will have as an objective the identification of these forums' peculiarities (von Krogh et al., 2012) as compared to other problem-solving settings and will use a narrative methodological approach. This narration will be based on the study of both *Facebook* and *Google Maps* developer support forums as "monuments" of interaction (see Section 14.3.3) between the participants.

The comparison will take into account the phases of a problem-solving process, as well as the actors involved in each phase for the different models proposed by the literature, namely community-based, enterprise-based and mixed actors problem-solving.

This section will conclude with the identification of an uncommon process, where the problems to be resolved are posed by UDEs, while their resolution is to be undertaken by the enterprise. According to this mode of processing, a collaboration between enterprise employees and UDEs only exists in the beginning, where enterprise employees and UDEs try to formulate the issue in question.

Exploration step

During the second step of my study (Section 17.3), I will *"review and evaluate research designs used"* (von Krogh et al., 2012) in the study of problem-solving processes by man-

Section	Description	Problem	Methodology	Outcome
17.2	Distinction step: Are developer support forums just another case of problem-solving?	Identification and characterisation of the peculiarities of developer support forums as compared to community-based, enterprise-based and mixes problem-solving approaches.	Broad cultural terms description, common peculiarities recognition for *Facebook* and *Google Maps* developers support forums through the study of conversations' monuments.	Peculiarity: A UDE-driven problem formulation, an enteprise-based resolution
17.3	Exploration step: Is problem-solving really the problem?	Test the relevance of problem-solving performance criteria for the specific field.	Quantitative study: calculation of problems reported and solved within a six months period.	Rates of problem solving are particularly low for both forums.
17.4	Design step: Alternative design proposal. Characterisation of developer support expertise.	Compare the case of an enteprise employee supporting developers with known types of expertise.	Qualitative study: Modelling the forum conversation process and the employees activity, through the study of both forums and "cookbooks" for developer supporting.	"Curator" and "empathy". An alternative proposal to study and deploy UDE support process.

Table 17.1: Chapter Overview.

agement scholars.

More specifically, I will use a quantitative approach to measure problem solving performance for the two forums, based on the "bugs" reported and resolved within a period of six months.

The findings reveal a very low rate of problems solved (14% for *Facebook* and 10% for *Google Maps*). These findings enable me to question the relevance of this metric for the actual use of the forum, from an enterprise perspective, and formulate the hypothesis of a different kind of process taking place in these forums, beyond problem-solving itself.

Design step

Finally, Section 17.4 will take an in-depth view in the conversations occurring in the forums and will propose an alternative research design in order for the phenomenon to be better reached (von Krogh et al., 2012), beyond problem-solving.

More specifically, I will explore the activity of enterprise employees supporting developers as a specific kind of expertise, based on the expertise typology proposed by Hatchuel and Weil (1992).

For this investigation, in addition to the forum conversation, I will use two early versions of a "cookbook" written by a *Google* employee active in this field and addressed to her colleagues, as well as information from two interviews I had with her.

The section will conclude with the identification of a new type of expertise, the *"curator"*, whose action aims at "taking care" of the UDEs. This activity requires particular know-how, while it also suggests that, before problem solving, a kind of *"empathy"* between UDEs and enterprise needs to be developed.

17.2 Distinction Step. Developer support forums: Another case of problem solving?

17.2.1 Literature review: whose problem, who solves it? Different models for problem-solving

Problem-solving has been one of the most influential approaches in management for the description of innovation processes. For instance, for von Hippel *"the proper organization of innovation work"* is to be undertaken and examined *"with respect to the requirements of problem-solving"* (von Hippel, 1990, p. 408). Different methodologies have been proposed, while the partitioning of this process between different actors may imply a permanent debate about design modifications (Garel and Midler, 2001). Hence, the authors value the relationship between actors engaged in problem-solving for the organisation of a design process and the exchange of knowledge.

The problem-solving approach on innovation has been widely shared in the management community. Having a broader look at the literature, we can observe that the articles in Management correlating problem-solving and innovation have grown considerably over the last two decades. Figure 17.1 shows the results obtained from the *sciencedirect.com* search engine, using the keywords *"problem-solving innovation"* for the category *Business, Management and Accounting* for the period 1994-2012. We clearly observe an explosion of literature correlating the two notions, passing from 284 publications in 1994 to 948 in 2012. Although those results are not normalized by taking into account the general publication number growth since the 1994 due to a lack of information, this graph makes clear however that the field studying the phenomenon can be characterized as mature (von Krogh et al., 2012).

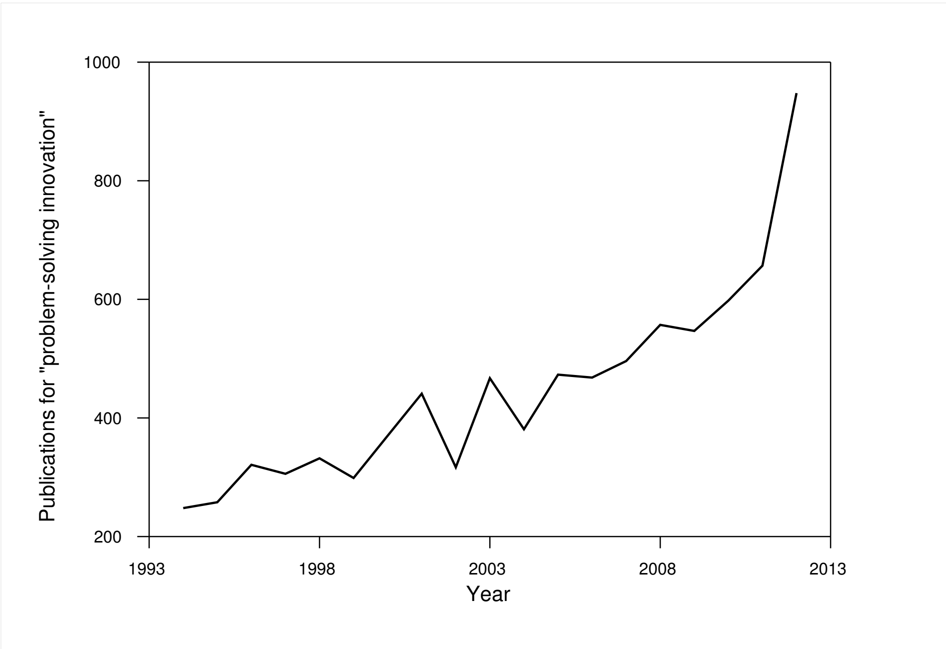


Figure 17.1: Number of articles in the category *Business, Management and Accounting* correlating problem-solving and innovation since 1994. *Source*: sciencedirect.com.

The current section will thus address the question of how enterprises can harness UDE activity to exploit their service potential from a problem-solving perspective. Table 17.2 shows a synthesis of the literature on problem-solving processes, discussed in the following paragraphs, that will be used to distinguish (von Krogh et al., 2012) those processes met in Web services.

While not all scholars attribute a similar emphasis to the different phases of problem solving, the following can be easily induced:

1. *Problem formulation*. This is the first phase of problem solving. While once the problem is formulated, then a decision process can generally be deployed (Simon, 1965), this phase can be by itself tricky. While for Simon it has been described as a starting point for the design process, "ill-defined problems" - and thus the issue of problem formulation - have been an object of debate. On the one hand, they are considered as a user entry to innovation process (through use-related "sticky information" von Hippel, 1990), on the other hand they are viewed as the very limits of the bounded rationality approach (Hatchuel, 2001).

2. *Resolution Process*. This is the second phase of problem solving, once the problem is defined, consisting in actions that aim at the identification of a solution. Scholars (Thomke et al., 1998; Bernoff and Li, 2008) have proposed that the performance of the entire process can be measured by the number of solutions found.

3. *Solution Implementation*. Once a solution is found, this phase consists in operating the actions required for the implementation of its solution.

Thus, according to the actors implied in each phase, I will distinguish in the following paragraphs different models for problem solving: community-based, enterprise-based or mixed ones. As I will examine in Section **??**, Web services processes can be described as "mixed" activities, placed in-between community-based and enterprise-based models. In parallel, they differ from other mixed models, such as crowdsourcing or design division, since they are characterised by a different community-enterprise action configuration. In particularl, it is the enterprise, not the community, that solves the problems, while problem formulation is a joint activity of both UDEs and enterprise staff.

Community-based problem solving processes

Figure 17.2 summarises the model of autonomous community problem-solving management. All three phases are undertaken by the developer community itself, through a self-governance mode.

More specifically, as we have already seen in Section 6.2.1 (on page 98), open source software development is considered to be a case of an autonomous and complete community activity, where the private-collective model (von Hippel and von Krogh, 2003, 2006) is fully deployed, and where innovation "by and for user networks" (von Hippel and Katz, 2002) is undertaken. In Section 14.2.2 we've discussed how communities are defined by the object they share and maintain. Maintenance in particular, is the process in which community-based problem solving principally takes place.

Those communities' action has been extensively studied from an individual motivation perspective (Lakhani et al., 2002; Lakhani and Wolf, 2003; Bagozzi and Dholakia, 2006; Roberts et al., 2006; Shah, 2006) as well as the use value one (Lüthje et al., 2005), while the question of "how they do it?" is less explored. Some studies focus on the diffusion norms,

Model	Actors	Process phases			Indicative Literature
		Problem Formulation	Resolution Process	Solution Implementation	
Community-based	Dev. Community	✓	✓	✓	von Hippel and von Krogh (2003, 2006); Lakhani and Wolf (2003); von Krogh et al. (2003b); Auray (2004); Shah (2006)
	Enterprise				
Enterprise-based	Dev. Community				Thomke et al. (1998); Jablokow (2005); Hatchuel et al. (2001)
	Enterprise	✓	✓	✓	
Mixed					
Design division	Dev. Community	✓	✓	✓	von Hippel (1990); von Hippel and Katz (2002); Baldwin and Clark (2006)
	Enterprise	✓	✓	✓	
Crowd-sourcing	Dev. Community		✓		Howe (2006a); Chesbrough (2006); Lakhani et al. (2007); Piller and Walcher (2006)
	Enterprise	✓		✓	
Web services (case to study)	Dev. Community	✓			
	Enterprise	✓	✓	✓	

Table 17.2: Different models for problem solving

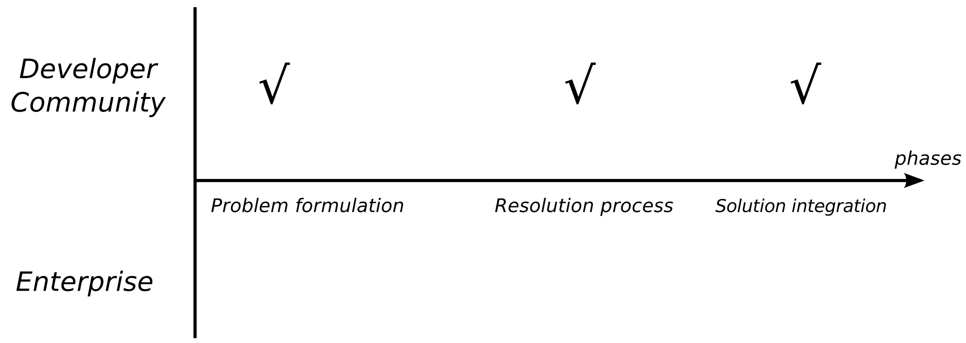


Figure 17.2: Community-based problem-solving. Through self-governance, developer communities manage all phases of problem-solving.

rather than the maintenance ones, as understood by specific intellectual property rights (O'Mahony, 2003) or the network externalities (Bonaccorsi and Rossi, 2003) perspectives.

While the "peer" approach is generally accepted in the above mentioned studies to describe those action norms, more detailed studies revealed both an expertise and a related hierarchy within those communities. Auray (2004), studying *Debian Linux* community has shown that a work division takes place, where only experienced developers undertake maintenance tasks: the community "tests" the new comers before assigning them responsibilities. More specifically in what regards maintenance, Auray argues that, there is the question of the quality of "bug reports". On the one hand, there have to be many reporters, to trace the problems of the platform. On the other hand, these reporters must have a high level of expertise, in order to be able to correctly diagnose the problem and its importance and, eventually, propose and implement a solution. Hence, while the community has the tendency to welcome new end-users, the selection of developers is slower, the community having to be reassured of the expertise and the commitment of new developers before accepting them into the team. Therefore, open source communities are characterised by "self-governance" processes, enabling the maintenance and further development of the common source code.

Von Krogh et al. (2003b), studying the *Freenet* community have also suggested similar findings, where newcomers do not always contribute to the development, while there are cognitive barriers to the participation of the actual development. A self-governance model assures a role attribution and a common process establishment, too. Giuri et al. (2008) proposed that, leadership in such communities emerges through assuring a continuity of the whole process, from problem formulation to eventual problem resolution.

Overall, as summarised in Figure , community-based problem solving processes, such as those in the OSS field, are entirely undertaken by the community, in what regards problem formulation, resolution process and solution implementation.

The term "developer community" is here used as an encompassing notion of the community members, the division of their expertise and the related self-governance model.

Enterprise-based problem solving processes

Figure 17.3 outlines the model of enterprise-based problem solving. Through its organisational structure and instruments, an enterprise here undertakes the entire problem-solving process.

Important research has already been based on the implicit assumption that "more problem solving means more innovation". Thomke (1998) studying simulation of experiments

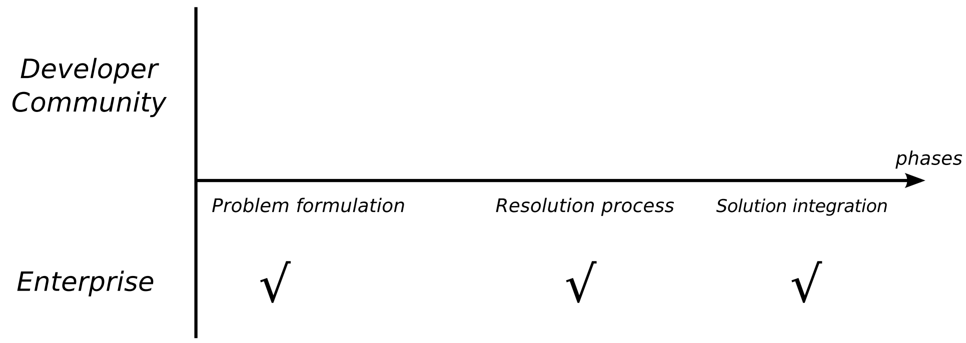


Figure 17.3: Enterprise-based problem-solving. Through its organisational structure and instruments, an enterprise manages all phases of problem-solving.

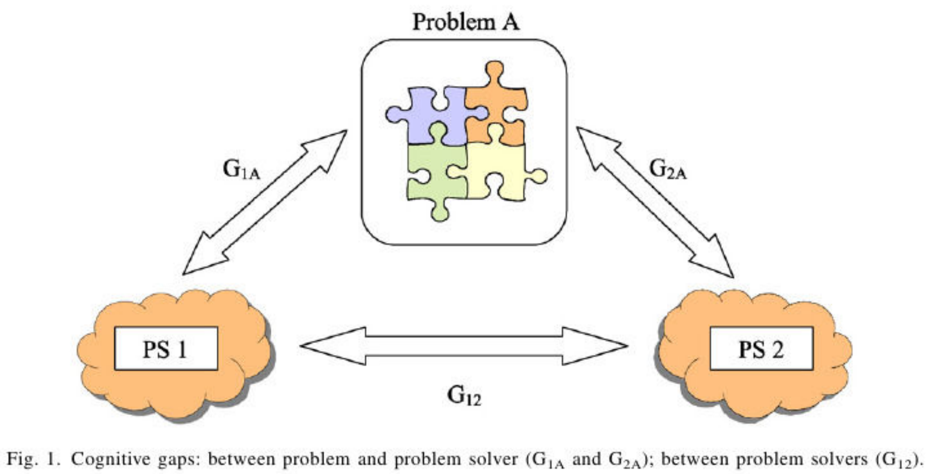


Figure 17.4: Cognitive gaps in problem-solving processes (Jablokow and Booth, 2006).

in the car industry, used the criteria of 'problem-solving cycles', as well as 'diversity' to measure performance in problem solving in the case of R&D. Similarly, in the context of the pharmaceutical industry, also with experimentation, computer simulation experimentation performance was examined as a problem-solving process, of which the efficiency was measured by the chemical compounds found (Thomke et al., 1998). Both industrial settings involve highly skilled engineers who work in a project-based enterprise environment. Moreover, in both cases, experimentation is guided by the pre-established scientific requirements of the domain (security in the former case, affinity of compounds for human and bovine isozyme for the latter), as they refer to a specific dominant design (Utterback and Abernathy, 1975; Abernathy and Townsend, 1975).

In the particular case where the problem 'seeker' and the problem 'solver' is not the same person or team, a problem of a "cognitive gap" appears, beyond communication issues. Jablokow and Booth (2006), also in the setting of an integrated organisation and the field of pharmaceutical industry, identify the issue of a cognitive gap. This gap takes two forms (as seen in the Figure 17.4 quoted from Jablokow and Booth (2006)): a first form resides between the problem and its solvers, while a second form resides between the solvers themselves. Jablokow and Booth propose that more organisational integration contributes in bridging cognitive gaps, though imposing difficulties in human resource management, because of the human and cognitive diversities of the individuals that have to collaborate.

From a business strategy standpoint, discussing the importance of R&D departments in organisations, as well as the limits these structures face through the imperative of intensive innovation, Hatchuel et al. (2001, 2002) go beyond problem-solving by proposing the introduction of a new function in the enterprise, the function of *innovation*. Regarding the 'design reasoning' of innovation processes, the scholars emphasise the value of formulating the right questions, even before problems become visible. The resulting organisational schema (R-I-D) consists in the management of questions to address to Research and Development departments by this new structure that can thus lead enterprise innovation processes.

Overall, enterprise-based problem solving is entirely undertaken by an enterprise, which coordinates the process using activating different employees and departments according to its internal processes.

Design division and problem-solving

Figure 17.5 summarises the model of design division between manufacturer and lead users. Through a modular design, manufacturers can separate technological and use-related problem solving processes. Then, the two can be undertaken autonomously.

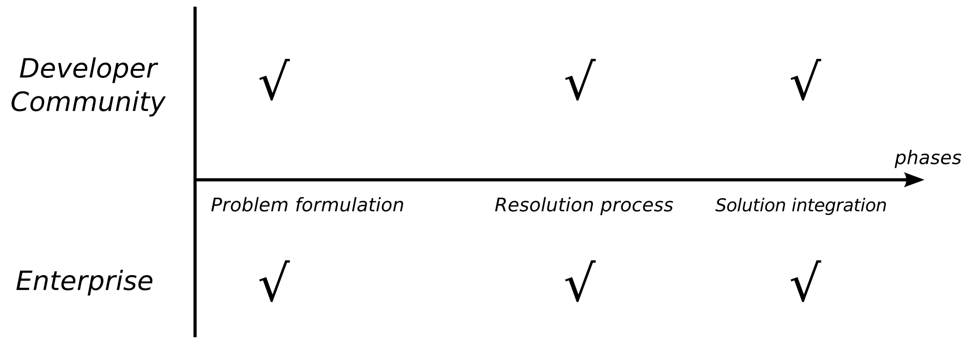


Figure 17.5: Problem-solving through enterprise-user design partitioning. Through a modular design, manufacturers can separate technological and use-related problem solving processes. Then, the two can be undertaken autonomously.

In previous chapters we have already discussed the approach on design space division between manufacturers and users. According to it, ill-defined problems are due to *"sticky information"*, the latter being decomposed to *"technological"* and *"use-context"* ones (von Hippel, 2005, and others). Manufacturers can thus partition the design (von Hippel, 1990) to different design spaces (Baldwin and Clark, 2006; Baldwin et al., 2006), one for the manufacturer, another for user innovators. Thus, there can be two parallel problem solving processes, each one deploying in an autonomous manner. The modularisation (Baldwin and Clark, 2000) of the design leads to the division of innovation process into tasks, a design requiring less effort for the achievement of *"cross-boundary communication and coordination"*, and thus augment problem-solving efficiency (von Hippel, 1990). In other words, the more a project is partitioned in smaller elements, the less communication effort is needed between teams in charge for each element and problem-solving becomes more efficient. Even more, such elements can be delegated to the lead users.

Crowdsourcing as task distribution.

Figure 17.6 outlines the crowdsourcing approach, according to which an enterprise formulates a problem and addresses it to a community of developers to find a solution. Different crowdsourcing configurations have been proposed by the literature, depending on the tools or relational settings that mediate this process.

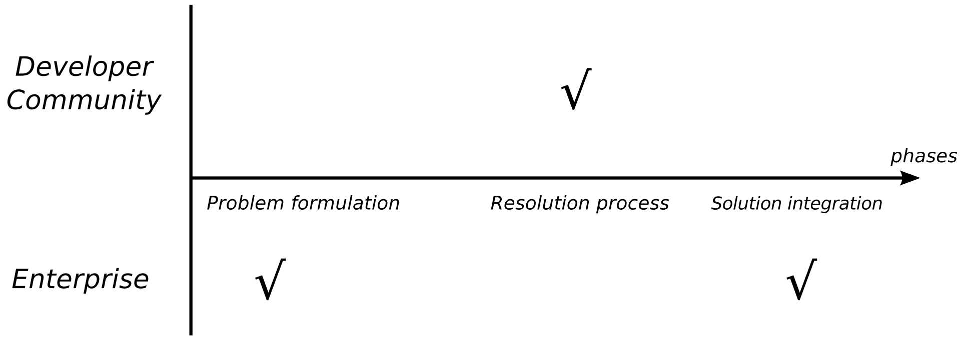


Figure 17.6: Problem-solving through crowdsourcing. Enterprises address their questions to a community of developers to propose a solution.

The term was coined by Jeff Howe (2006a), editor of the *Wired* magazine. In his article, *5 Rules of the New Labor Pool* he outlined five common principles of general application[1]: potential for remote task performing, need for work division in really small tasks, potential of community-based result evaluation.

Hence, this initial, empirical definition of crowdsourcing is based on a division of labour into small tasks, easily performed at a distance. A typical case of this dimension has been the "Clickworkers" project launched by *NASA* (Benkler, 2006, p. 69)[2]

The emergence of this possibility of task modularisation and distributed work had been innovative at the time. Yet, we should note that the users themselves did not innovate: they followed some very specific instructions on elementary tasks presuming elementary knowledge and skills as well (click on a screen using the mouse when identifying a crater in the image), where very little creativity was needed and the results were within a limited scope of solutions (craters spotted).

Regarding the specific operation of community-based evaluation, different methods have been proposed. For instance, marking comments Haefliger et al. (2009) is one of the popular ways to enable user-generated content evaluation. More sophisticated methods, such as the *Web-based Delphi* (Benkeltoum, 2008), where experts are invited to evaluate innovations, can also enter in the broad definition of crowdsourcing in the trajectory of distributed filtering, taking advantage of the Web infrastructure.

A different view on crowdsourcing is developed by scholars focussing on problem-solving processes. A well-known case is the one of *Innocentive*.

[1] Howe, Jeff. *5 Rules of the New Labor Pool*, Wired Magazine, June 2006. Retrieved: October 1, 2012. URL:http://www.wired.com/wired/archive/14.06/labor.html

[2] NASA developed a Web site where Web users could mark crater they spot on the surface of Mars, by clicking on them. As Benkler notes, the tasks performed were discrete, each easily performed in a matter of minutes. He comments:

> *The clickworkers project was a particularly clear example of how a complex professional task that requires a number of highly trained individuals on full-time salaries can be reorganized so as to be performed by tens of thousands of volunteers in increments so minute that the tasks could be performed on a much lower budget. The low budget would be devoted to coordinating the volunteer effort. However, the raw human capital needed would be contributed for the fun of it. The professionalism of the original scientists was replaced by a combination of high modularization of the task (Benkler, 2006, p. 69).*

Within the Open Innovation literature current, this case has been of major interest as an exemplary field of application of the approach. Characterized as an "innovation intermediary", *Innocentive* acts in the intersection between the "seeker", that is the enterprise formulating the problem, and the "solvers", the members of the community that undertake the problem-solving process to get the award, via the mediation of *Innocentive*'s staff members (Chesbrough, 2006, pp. 143-144). The resulting business model is the one of a market between seekers and solvers.

Always regarding the same service, Lakhani et al. (2007), through an experimental investigation methodology, found a 29.5% resolution rate for scientific problems that had previously remained unsolved inside the R&D laboratories of well-known science-driven firms (Lakhani et al., 2007, p. 4). A similar albeit older case is ideas competitions, well known to the architect community, where creativity and originality are requested while the 'deliverables' can be more ill-defined (Ebner et al., 2009).

Addressing the problem of idea competition organisation, Piller and Walcher (2006) have proposed the disposal to the users of *toolkits for idea competition*. Lead by the principle of users' knowledge developed by von Hippel, Piller and Walcher propose the provision to self-selected lead users of toolkits for prototype building. Prototype building is thus considered another way for lead users to develop and explain their ideas, beyond verbal or informational formulations. The aspect of "user innovation horizontal networks" (von Hippel, 2007) is translated in the requirements of these tools by Piller and Walcher (2006) as the communication possibility among users and the enterprise organising the contest. Regarding participants' relations configuration, it has also been proposed that *collective* competitions, where participants also collaborate with each other, may advance the quality of solutions proposed (Blohm I et al., 2011).

Overall, in crowdsourcing, as regards problem-solving, the enterprise formulates a problem, the community solves it and finally the enterprise implements the solution. There can be different configurations of crowdsourcing, according to the tools or the organisational settings that mediate this process.

17.2.2 Research methodology: identification of actors and roles in the process

In order to study the way in which enterprises harness UDE activity for their service potential exploitation, I study two developer support forums, *Google Maps* and the *Facebook* ones. I will study the forums of those two services as both services are among the exemplary ones of the field, while - in addition - both enterprises are "new", in the sense that they both emerged and grew within the industrial context of online services. Those elements suggest that an identification of common methods is very likely to be representative of the action norms used in the specific industrial settings.

My research goal is to distinguish the peculiarities of this field (von Krogh et al., 2012) as compared to the approaches studied by the literature (reviewed in the previous paragraphs). In order to do this, I study the "monuments" of interaction between the participants in the forum conversation, as already described in Section 14.3.3.

The identification of the Websites where forums themselves are situated is easy, as a Web search with the name of the service and the words "developer support" is sufficient to identify the addresses of the corresponding forums. In addition, both sites provide the option to actually download the entire discussion files[3]. Then, the study of the conversations requires a specific familiarity with the very technical language utilised, which can be the study of the specific terms employed. To this, the use of online documentation, which can be found by a Web search of each technical term, is required during the early research period.

After the study of a number of conversations, a conversation pattern is revealed, which is independent from the specific technologies under discussion.

Figure 17.7 shows the general schema of discussion in both forums, as induced from the reading of forum conversations.

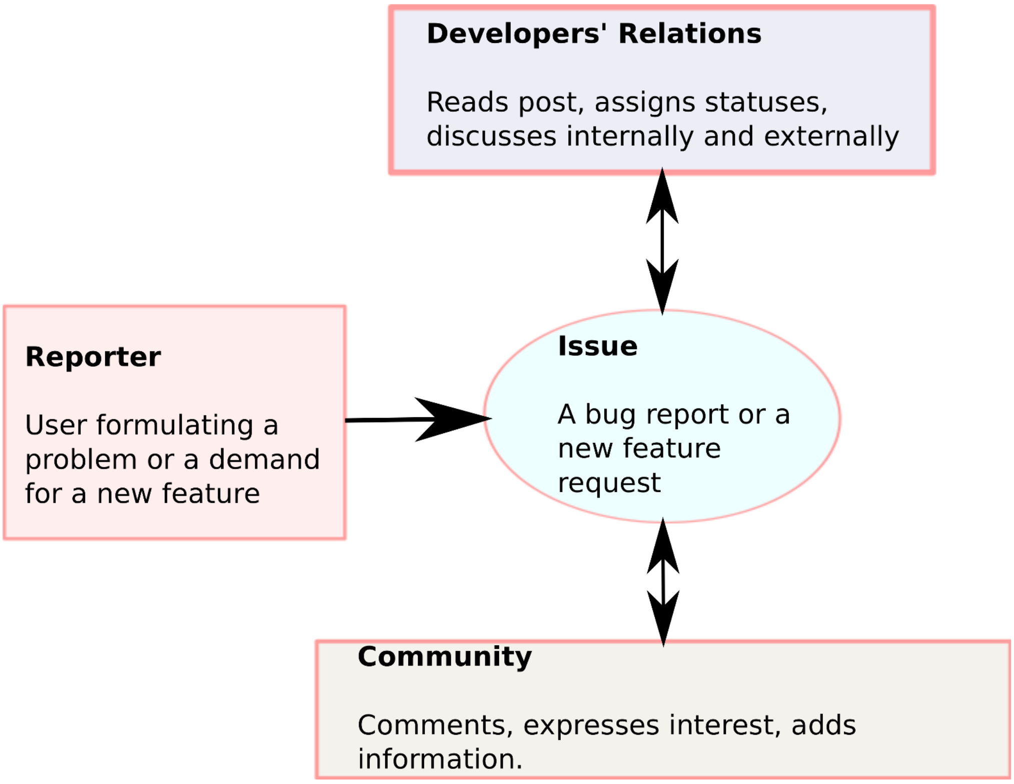


Figure 17.7: The actors engaged in a discussion on the developer support forum: reporter, developer relations, community.

According to this pattern, each individual participates in the discussion with a personal nick-name and an e-mail, from where it is possible to have first indications on whether or not they are service staff or external developers. User profiles, also available in the forum platform, provide additional information on this question.

A *Reporter* formulates a problem report or a new feature request. Reporters are User-Developer-Entrepreneurs and their reports regard issues they personally face. The report becomes an "issue", that is a specific "thread" or conversation in the forum. The forum being public, other UDEs may comment on the issue, usually saying whether or not they also face the same problem or desire the same feature. In some cases, other UDEs propose solutions to the problem, either by directly suggesting the solution or by referring to the documentation. In addition, a provider employee follows up the discussion. In case he or she considers that more information should be provided, she asks for it. In this case, UDEs are likely to refer to their own web site or provide a copy of the problem effects (a "screen-shot"

[3]This possibility was the case during the period of study, from 1/1/2010 to 31/5/2010.

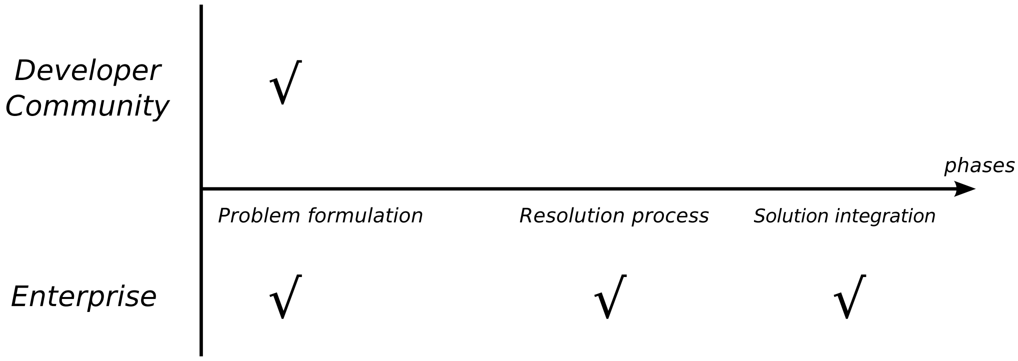


Figure 17.8: Web services problem-solving. While problem formulation is a joint process, problem resolution and solution integration are undertaken by the enterprise solely.

in the case of a graphical problem or the output code in case of data problems).

Once the problem is well defined, the provider employee lets the UDEs know that the report or request has been acknowledged and that an internal process has been undertaken for its resolution. Eventually, when the bug is fixed or the feature integrated in the platform, the employee informs the community.

17.2.3 Outcome: UDE-driven problem formulation, enterprise-based resolution

The problem-solving process identified is shown in Figure 17.8. This configuration is different from all the configurations studied by the literature.

In both UDE support services studied (the *Google Maps* and the *Facebook* developer support forums) the agenda of the topics to be discussed is defined by the developers, not the enterprise. Hence, regarding the initial formulation of the questions to be answered, we observe a reverse direction from crowdsourcing techniques (Howe, 2006a,b), where questions are addressed from the enterprise to the community. The problem to be solved is one presented by a UDE, while the one to solve it is the enterprise.

Still, problem formulation is not entirely a community issue neither. While it becomes obvious for a non-expert that those using the forum share a common technical expertise on programming, it is also clear that this fact does not lead to a direct transmission of "sticky information" (von Hippel, 1994): *An important role is to be undertaken by platform developer staff, and namely the Developer Relations Department (also called DevRel), to help users formulate their problems, in order for the correspondence of the latter to a specific know-how to become evident.* Therefore, provider employees have an important role and an authority in this process, as they are the ones to say when a problem is well-defined, and thus can be accepted as an issue for resolution by the enterprise.

Overall, while for reporters problem resolution is the challenge to address, for provider employees there are other preoccupations before that, while even when they do recognise a problem, they never engage that its resolution will take place.

However, in the case where problem resolution is not the provider's preoccupation, what is? The next section will further explore this question.

17.3 Exploration step: Testing the relevance of problem-solving criteria

The current section explores the problem-solving process described previously by intensifying data gathering inside focal concepts (von Krogh et al., 2012). More specifically, I explore whether or not what appears as a problem-solving process serves indeed in solving problems. I find that the forums studied are characterised by a very low rate of problems solved, thus I advance the proposition that their main function resides beyond problem-solving as such.

17.3.1 Methodology

It has been proposed that the rate of problem resolution can be used to qualify the performance of a problem solving activity (Thomke et al., 1998; Bernoff and Li, 2008). Using an opportunistic design (Lakhani and von Hippel, 2003; von Krogh et al., 2012), I will measure problem solving performance for two different platform support forums which follow similar processes, *Google Maps* and *Facebook*, studying the simple question of whether or not the problems (*"defections"*) reported in the forums by the users are eventually solved. To this end, I will use the following indicator:

$$\text{Problem solving performance} = \frac{\text{Problems solved}}{\text{Problems reported}}$$

Regarding the data, I will exploit the fact that discussions are publicly available. My measurement will be limited to the first semester of 2010 (January 1st to May 31st) for both platforms and will take into account only the issues reported and resolved within this period. For the calculation of the number of the problems reported, I will subtract the number of duplicated issues, that is issues that appear twice or more in the forum and thus have been merged.

Technically, to operate these calculations, I used the *CVS* files of the discussion provided by the forums and I developed a program using *Perl* programming language to process these data.

The number of UDEs actually creating applications with *Google Maps* and *Facebook* would be helpful information to further explore this data. However, I was not in position to obtain this information. A reason for this is the openness of the process itself: UDEs do not need to provide their personal information to develop applications with those services. An anonymous account is sufficient for application development, while each user may create as many accounts as he/she desires. In fact, Google's Developer Relations department once attempted to calculate the number of the developers using their platforms, although they gave up the effort[4].

17.3.2 Findings: a low rate of problems solved

Figure 17.9 summarises the findings regarding the problems both reported and solved within the examined period.

In the *Facebook* developer forum case, 1821 problems were reported from UDEs to the forum for the period 1/1/2010 to 31/5/2010. This number takes into account the duplicates, as I have already subtracted the 198 issues reported that were merged with

[4]Interview with Google Developer Relations manager, August 12, 2010.

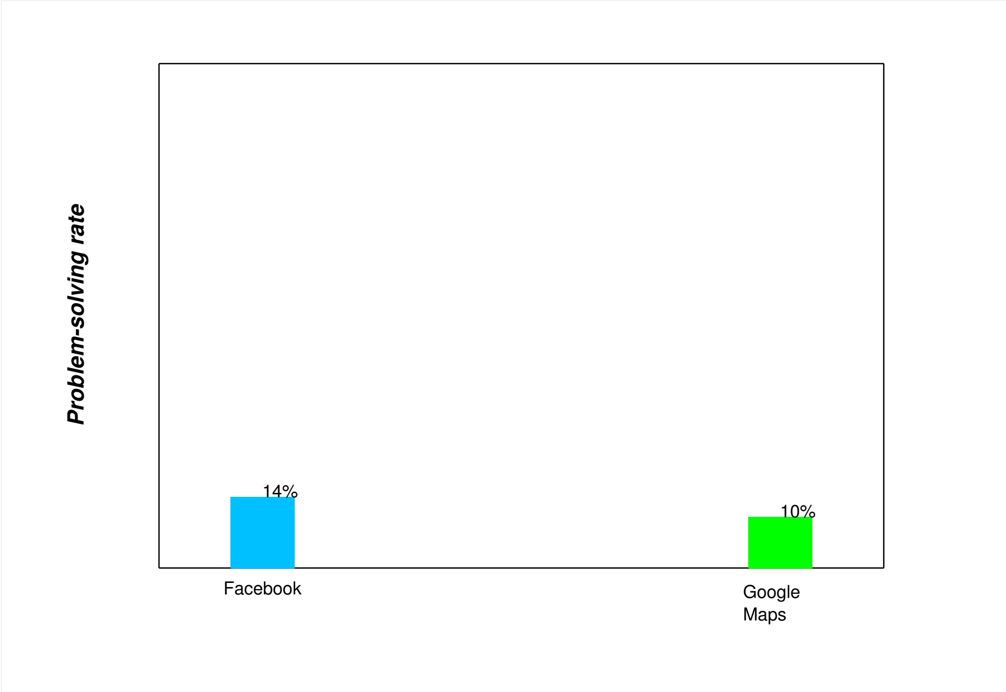


Figure 17.9: Rate of problems (*"defections"*) reported to *Facebook* and *Google Maps* developer forums and solved within the period from 1/1/2010 to 31/5/2010.

previous discussions. From the problems reported, 296 were resolved within the same period. Hence, the problem solving performance, as defined previously, is 14%.

In the *Google Maps* developer forum case, 325 problems were reported from UDEs for the same period. This number takes into account the duplicates, I have already subtracted the 24 issues that were merged with previous discussions. From the problems reported, 32 were resolved within the examined period. Hence, the performance in this case is 10%.

The findings on problem-solving rates for the two Web services platforms indicate a low priority on the resolution of problems, compared to the value problem solving has for enterprises, as proposed by management scholars.

Hence, two propositions can be induced. On the one hand, the role of forums is not limited to problem-solving *per se*. On the other hand, there is a question about the value of a problem reported in the case where no resolving action is taken by the enterprise.

17.3.3 Discussion: is problem-solving the problem?

In section 17.2 I have distinguished the problem-solving process met in Web services developer forums from other settings met in the literature identifying a specific configuration of roles between developer community and enterprise, which is characterised by a joint problem formulating phase and enterprise only resolution and integration phases.

Furthermore, the findings of the current section also add the suggestion that there is a reverse process to the one met in crowdsourcing methods: here, it is actually the UDEs, not the enterprise, that initially identify problems to be solved. Thus, the enterprise is found in a position of "crowdflooding", as developers demand it to take action, formulating a tasks list of problems to solve.

Therefore, my research question, how to harness UDE activity for service potential exploration, has been answered only in part: UDEs should be in position to report the problems they face during their activity. Still, while the use of the forum is clear from a developer perspective, it is less clear from an enterprise one. As the problem-solving rate is particularly low for both services studied, the question remains open regarding the action norms of the enterprise itself, as actually solving the problems does not appear to be the main goal.

The following section will continue the exploration of the same process, this time from an enterprise perspective, proposing a research design to study and deploy the management of such processes.

17.4 Design step: Characterisation of developer support expertise

In the two previous sections, I investigated the developer support forum discussion as a problem-solving process and I found that problem-solving itself is characterised by a very low performance rate. Unlike typical problem-solving life cycles met in enterprise as well as project management go/kill decisions, the two previous sections have revealed an open-ended process, without a provision on a specific closure time: the life-span of an issue can take dimensions that wouldn't be acceptable within an enterprise context.

17.4.1 Theoretical concepts. Beyond problem-solving: three expert figures and one hypothesis

As already evoked in the first section of the current chapter, problem-solving is characterised by a "tricky" phase, the one of problem formulation. This phase has been the one wherein enterprises' problem-solving process is challenged.

From the user perspective, von Hippel (1990, 2005) distinguished two kinds of *"sticky information"*, a technological one corresponding to enterprise expertise, and a use-context one corresponding to lead users knowledge. For von Hippel, possesion of this information is required to innovate either in technologies or in uses. While this notion illustrates the difficulty of enterprises to access lead users' knowledge and vice versa, "sticky information" is at the same time a barrier for further characterisation of each actor's knowledge, due exactly to its stickiness.

From a design perspective, Hatchuel (2001) argued that problem-solving is a special, though limited case of a design situation. Design can also include projects where conception expansions may occur during their undertaking, in contrast to problem-solving, where the concepts are only set in the beginning.

Returning to the particular case I study, on using developer support forums as a means to harness UDEs' activity for service potential exploration, and considering only the first phase, the one of *problem formulation*, we've already seen that it remains an open process, as the majority of the issues reported remain unsolved for a long period. This section will focus on the work of the provider's employee who is in charge of this process for the enterprise.

While this activity was encountered during the two first sections, my focal concepts did not allow its further exploration. Hence, I will propose an alternative research design (von Krogh et al., 2012), appropriate for the study of this specific case.

For that, I will face this activity as a specific kind of expertise. Based on the typology of Hatchuel and Weil (1992) on expert figures, I will project the hypothesis that, to the extent there is indeed an expertise in the employees' work, it is an expertise of a different type.

The following paragraph reviews the work of Hatchuel and Weil and formulates the hypothesis, before proceeding to the field research.

Expert figures: the artisan, the repairer and the strategist

In their influential study, *"L'expert et le système"*, Hatchuel and Weil (1992) identify three different figures of expertise, drawing the study of four different cases of industrial expert systems implementation. Those figures are the *artisan*, the *repairer* and the *strategist*, each representing a different *relationship between knowledge and action*. Table 17.3 summarises those actor figures with the related know-how and expertise.

Actor figure	Know-how	Expertise
Artisan	Doing know-how.	In the way in which certain transformations are obtained by familiar actions.
Repairer	Understanding know-how.	Action and investigation to re-establish the original state of order.
Strategist	Combining know-how.	Knowledge combination, activity planning, new concept and knowledge creation.

Table 17.3: Overview: types of expertise - compiled from Hatchuel and Weil (1992, 1995).

Artisans' know-how is described as *"savoir-faire"*, which is not only limited to knowledge, but also concerns the *doing*. In other words, it is *"the expertise which expresses, whatever the level of detail considered, the way in which certain transformations are obtained by familiar actions"* (Hatchuel and Weil, 1995, p. 31), also referred to as *"doing know-how"*. This knowledge is limited to the application of a variety of rules, ranging from abstract rules, structuring the steps of their reasoning, to more specific ones, aiming at the resolution of well-defined problems (Hatchuel and Weil, 1992, p. 50).

Repairers' know-how is described as *"understanding know-how"*, since - unlike artisan - *"the repairer's expertise is more complex; it cannot be laid out in a straight line, and in each new situation it intermingles action and investigation in an ever-changing pattern"* (Hatchuel and Weil, 1995, p. 36). A repairer has to establish the original state of an order that has been disturbed, deviated from or deformed (Hatchuel and Weil, 1992, p. 52). Hence, there is a part of the knowledge that is strictly related to the *repairing experience*, which cannot be fully anticipated (Hatchuel and Weil, 1992, p. 53), though a reference state is always known.

Finally, strategists' know-how relies on the *combination of knowledge*, while his action aims at the coordination and the planning on an enterprise level (Hatchuel and Weil, 1995, p. 44). This type of action requires the invention of *new concepts* as well as the creation of *new knowledge* (Hatchuel and Weil, 1992, p. 74).

To the above, Hatchuel et al. (2002) also added the role of the organisation itself, being the encompassing setting for both collective learning and action. When faced by innovation challenges, organisations are called to re-configure their division of labour, as well as the corresponding professions and tasks, and create new ones.

17.4.2 Hypothesis. A specific expertise?

The process described in the two previous sections implies a series of questions, where all three figures of experts should be implied for them to be answered in the field, as they address doing, understanding and planning issues. A non-exhaustive list of the questions that can be faced in the field is the following:

- What exactly is the problem? How can distant interlocutors share the 'sticky information' von Hippel (1994) on users' problems, as, by definition, it is *"costly to acquire, transfer, and use in a new location"* (von Hippel, 1994, p. 429), while, in addition, the source code of the platform and the third party innovations is not shared?

- How can platform providers and user-entrepreneurs share the same vision of the "doing know-how" and the "understanding know-how"?
- How can trust be established in this case of a problem-solving process, since providers do not reveal their code?
- What happens if that the "tasks" operated by third party developers using the tools are beyond the scope (Gawer and Cusumano, 2002) of the enterprise, that is they do not correspond in the planned division of design and development tasks?
- What are the strategical interplays of this process, when, for instance, new features are added to the platform?

Hence, based on Hatchuel and Weil (1992, 1995) I propose a configuration of experts as shown in Figure 17.10.

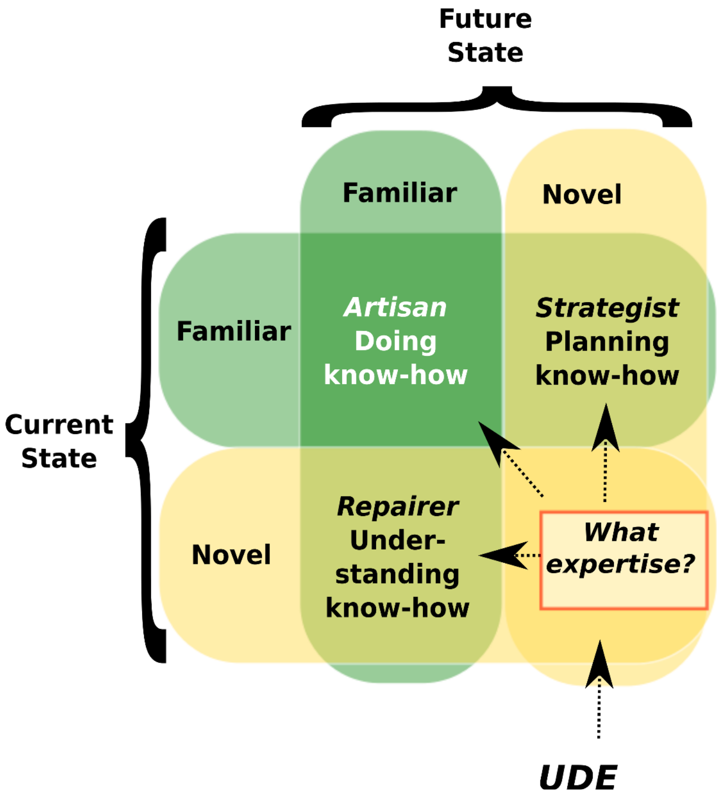


Figure 17.10: Different types of expertise according to the different configurations of the current and future states.

This reformulation uses the argument that expertise is a *relationship between knowledge and action* (Hatchuel and Weil, 1992, 1995), as well as the observation that we can distinguish two times for actions: when action begins and when its effects are produced. Hence, I can distinguish the current state (of the service technology in my case), when action begins and the future state, when the action effects occur. I also distinguish two types of states, a *familiar* and a *novel* one.

The three types of expertise fit in three of the four spaces created: doing know-how corresponds to an action from a known state to another known state, understanding know-how corresponds to an action from a novel state to a familiar one, while planning know-how corresponds to a projection from a known state to a novel one.

Hence, a gap appears, in what regards the action from a novel state to another novel state, I advance the hypothesis that this type of expertise may correspond to the provider employees in charge of the developer support forum.

More specifically, my hypothesis is that this situation appears when a problem is yet-to-be formulated, a situation typical to developer support forums. To better seize this situation, let us think of an analogous, more common situation: an agitated stranger enters the building of a school. From the fact that he is agitated, we can understand that he is seeking something or someone. If he asks a random person (say, a visitor exiting the building) he will have less chance to find what he came for. For such situations, there is a specialised service, the reception. By posing a set of questions (*"Are you looking for someone?"; "Are you looking for a course? A seminar"*) or affirmations (*"This is a School here, Sir!"*) receptionists may identify what the person is searching for and help him, providing the necessary information. In a similar way, provider employees appear to have a kind of "receptionist" role, redirecting outsiders (UDEs) and their requests to different sorts of experts, depending on the request.

Therefore, I assume that this person should have both knowledge of the different expertises existing within the enterprises, and knowledge of their action. This "know-how" is then to be used in the conversation with the UDEs.

In fact, research on sponsored communities has drawn similar conclusions. West and O'Mahony (2008) compared autonomous open source communities with cases where enterprises deliberately started to engage in community collaboration, generating or sponsoring open source communities. Regarding the management of this relationship, the authors identify a duality between *"access"* and *"transparency"*. The first refers to the access that community members have to the source code, as well as to decision making processes, community planning and - more generally - higher level platform strategy making (West and O'Mahony, 2008, p. 8). The notion of *transparency* refers to *"the communication of what is happening"*, according to the words of a community manager (West and O'Mahony, 2008, p. 14). To some projects studied, while this initially meant the release of information such as the release cycle and the goals, sponsors were progressively led to include information on the production *process* (West and O'Mahony, 2008, p. 8).

Hence, the work of the expert under investigation seems to lie in-between access to enterprise knowledge and actions and a sort of transparency of both towards UDEs, by the provision of related information.

17.4.3 Research Methodology

The objective of my research methodology will be to propose alternative concepts for the study of the activity of developer support through forum settings.

For that, I will use a monument study methodological approach (already discussed in Section 14.3.3). The material I will examine are of two kinds:

1. Actual discussions on the online forum.

2. A "cookbook" edited by and addressed to provider employees who work at the post of developer support.

3. Additional information acquired by two interviews with the author of the book.

While the forum conversations are available online, the "cookbook" was written by a *Google* employee to transmit her knowledge to her colleagues, and I have been able to access two early versions of the documents, the one written as a draft in February 2010[5] and the other as an early version of a complete *Developer Community Handbook Documentation* in March 2011[6].

Here is how the author described the need addressed by the book[7]:

> *The area of developer support is quite new, and there isn't much written about how to do it - what works, what doesn't. Given the increasing number of APIs, and I hope, the increasing number of people attempting to support API developers, we need to start documenting our field. This handbook is a first attempt.*

As this statement suggests, developer support is a field under rationalisation, where the actors themselves attempt to produce some criteria and 'best practices' on how to manage third-party developer communities.

To explore this setting, I will firstly provide a description of the online conversation process between UDEs and provider employees, as configured by the conversations. This operation requires me to enter the actual conversation, which is of course extremely technical. Then I will induce a model for the employee's activity, independent from the technical language used in the conversations.

A descriptive modelling of UDE - provider employees conversation

Figure 17.11 shows a modelling of the conversational process followed by the participants in the forum discussion. This conversation regards specifically the problem formulation phase, as the resolution and implementation phases are not discussed: while input on the issue is frequently demanded by the employees, what is communicated after problem formulation is indicative information on the internal process steps, through the use of "statuses".

The figure distinguishes two actors: UDEs and enterprise employees. UDEs have the option to express their problems or desires, while their discourse is weakly structured. The initiative to open a conversation comes to the issue reporter. Then, other UDEs may join in, expressing their interest or providing additional information and suggestions.

Issues are highly technical and regard very specific concerns the developers face during the creation or the operation of their own application. An example of the problem formulation is the following[8]:

> *I have found that dragend event in Google Map also triggers the click event in IE. It is OK in Firefox and Chrome. Open IE/FF/Chrome console and see the result of this fiddle. Any workaround will be appreciated.*

The above quoted suggestion was characterised at the time by a "doing know-how" and "understanding know-how", used while trying to configure the current situation.

The second actor is the enterprise employee. Their role is to structure the discussion and render the information provided by users exploitable in a form of a specified problem to solve

[5] Pamela Fox, Issue Tracking: Why & How, Febuary 2010. Copy provided by the author.

[6] Pamela Fox, Developer Community Handbook Documentation, Release 0.9, March 2011. Copy provided by the author.

[7] Pamela Fox, Developer Community Handbook Documentation, p. 3.

[8] Google Maps Developer support forum, Issue 4072: Bug: Dragend Event also trigger Click Event in IE. Issue URL: http://code.google.com/p/gmaps-api-issues/issues/detail?id=4072 . The full conversation is provided in Appendix C.0.1, page 407.

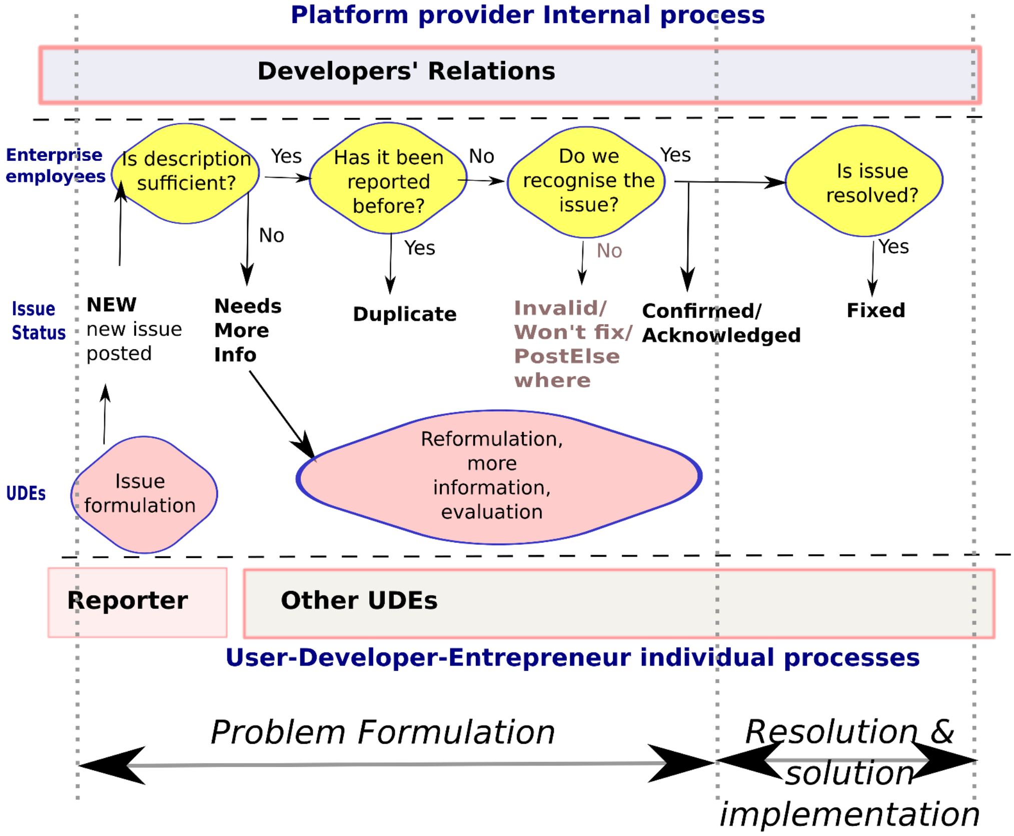


Figure 17.11: The conversational process followed in the developer support forum during an issue resolution. Status indications are providing a level of transparency regarding internal platform resolution process, while the debate among the community aims to clarify and evaluate the issue under review.

or new feature to integrate to the platform. An important tool in this direction is the use of different *statuses*, informing the community on the advancement of the whole process. The most common status is the *"Needs More Info"* one. Here is how the *"Developer Community Handbook"* describes the context of use for this status[9]:

> *Often times, a developer posts what sounds like a valid bug, but doesn't provide enough information to replicate. So, we use this label to indicate that we have reviewed the issue at least once and are awaiting further clarification from the developer. Once they provide enough info, it moves on to one of the other labels.*

Hence, a first task for the employees is the configuration of the current state, as well as the consideration of the desired one. The case of a *"bug"* is relatively simple, as the term indicates a malfunction, thus the future state is known (to *"fix the bug"*). Still, there is work to do in understanding the current state. Hence, beyond a description that *"sounds like a valid bug"*, the employee demands more information in order to be able to *"replicate"* the problem, e.g. on her own computer, which is considered necessary for problem understanding. To this effort, employees often ask reporters to operate specific actions, potential solutions, beyond demanding more information. For instance, the following quote comes from both a demand for more information and a suggestion of an eventual solution from a *Google Maps* employee to an UDE facing a problem with zooming the map[10]:

> *What are you executing against the map clicking a Munich hat? Are you calling fit- Bounds() which kicks off the zoom? What if you simply increase the size of your bounds to minimize the zoom?*

Still, things are more complicated when the issue reported is not a *"bug"* but a request for a new feature. The same handbook notes that in the end of problem formulation phase, there had to be a different status for bugs and new feature requests. While *Google Maps* employees initially used the *"Accepted"* status for both, they changed that to *"Confirmed"* and *"Acknowledged"* for bugs and new features requests respectively. This is how the handbook presents the reasons for this change[11]:

> *Developers thought that "Accepted" meant that we were actually taking on the feature request, and wondered why it was taking so long to fulfil their requests - when in fact, we simply meant to say that it was a valid request, and that we might fulfil it one day, given substantial resources and evidence of its utility to other developers. So, we now use "Confirmed" for bugs, after replicating the bug ourselves, and "Acknowledged" for feature requests, after deciding it is a valid request for the scope of our API, and confirming it isn't fulfilled in some other way (like by an open source library). Anecdotally, I believe this has led to less developers feeling teased.*

In fact, this confusion is related to the different states of a design process, as described by Hatchuel (2001). While a bug refers to a problem-solving configuration, in this particular

[9] Fox, Pamela, Developer Community Handbook Documentation, p. 33 & Issue Tracking: Why & How.

[10] Google Maps Developer support forum, Issue 2122: get-BoundsZoomLevel, URL: http://code.google.com/p/gmaps-api-issues/issues/detail?id=2122 . The full conversation is provided in Appendix C.0.2, page 408.

[11] Fox, Op.cit.

case calling for the "repairing" of the problem, a new feature request refers rather to a, more or less important, conceptual extension, which is a task of a "strategist".

An example of this is the following. On September 24th 2008, a developer requested an enhancement of the Google Maps API. He requested the addition of public transportation itineraries information to the API, in order for developers to be able to provide services based on them. As he commented *"In many cities public transportation is more efficient than going somewhere by car"*. The next day, a Google employee posted a comment saying *"shared with the team"* and marked the issue as *"Accepted"*. This post attracted a great deal of interest from the community. Nevertheless, its implementation required the acquisition of the related transport itinerary data from *Google* for the different cities, which was a strategic action, not an issue of repairing a problem.

What is not illustrated in the Figure, as it cannot be observed by the forum, is the *"triage meetings"*. As described in the handbook, meetings between enterprise colleagues take place once a week to evaluate and share the issues to be answered. Various dimensions of this activity are described there, as the general direction that each issue should be answered within a week, or the suggestion that employees should not monopolise the forum discussion. The issues are distributed according to the particular engineering skills of the employees, while other engineers, members of a specific API team may join their discussion to add their expertise[12]. Eventually, all issues are gathered in a centralised information system. Then, depending on the nature of both the issue and the relationship between the enterprise and the developer, there is a prioritization of the issue resolution[13].

Overall, enterprise employees have the responsibility to reply to the UDE's requests by structuring the conversation in the way that the required level of expertise to address the issue becomes evident for them. The following paragraph will propose a modelling of their activity.

17.4.4 Outcome: a model for the activity of a "curator"

The previous paragraph illustrated that the activity of enterprise employees in charge of replying to UDE requests in the support forum is not exactly problem-solving. Their activity is rather to "take-care" of the UDE concerns, in one way or another, hence, I propose to call them *"curators"*, etymologically coming from the Latin verb *curare* meaning "take care".

Typically enough, Fox, in the developer support "cookbook" provides her colleagues with the following advice, in respect to the relationship with the developers in particular[14]:

- *Care about your developers.*
- *Empathize with your developers.*
- *Keep your developers informed.*
- *Listen to your developers.*
- *Appreciate your developers.*

At the same time, this "empathic" relationship is tied to a very structured, enterprise process. Figure 17.12 suggests an alternative design to the one used in previous sections for the analysis and the exploration of this particular actor. To model a curator's action, I use

[12] Fox, Op.cit.
[13] Interview with Google Maps employee taken in August, 2010.
[14] Pamela Fox, Developer Community Handbook Documentation, March 2011, p. 4.

the typology of Figure 17.10 (discussed on page 360), which I have compiled from Hatchuel and Weil (1992). This compilation is based on an association of the knowledge an actor has for the current and the future states and his ability to act from the one to the other.

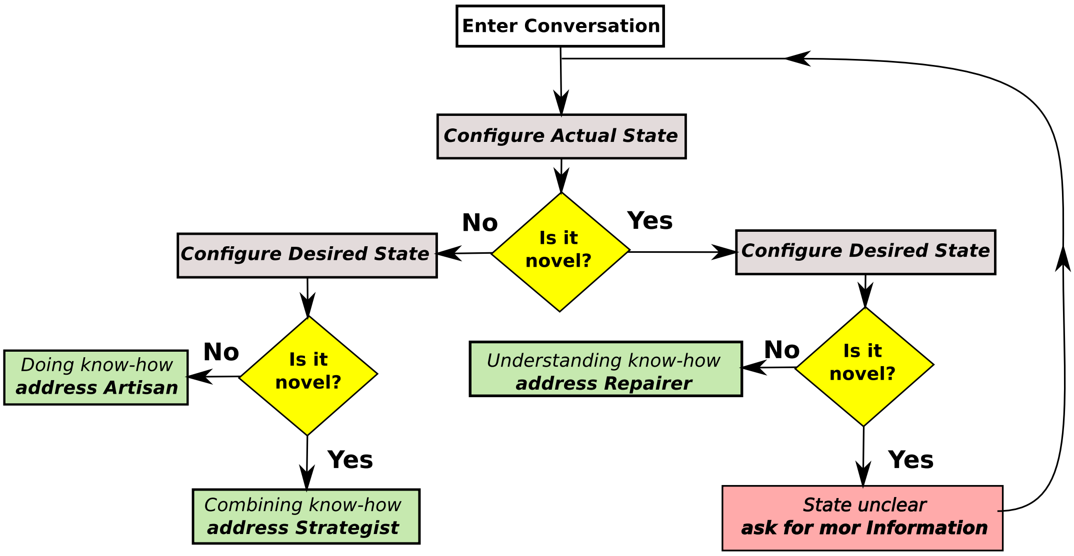


Figure 17.12: A model for the activity of a *"curator"*.

According to this model, a curator enters a discussion trying to determine the current state of the situation. For this a discussion with the reporter is required, which includes aiming at the "transfer of sticky information" from the reporter to the curator. Still, there is a possibility that the opposite is required as well, that is the provision of additional information and instructions from the curator to the reporter, either to directly help him resolve his problem or to enable him to define in a clear way the actual state. Then, once the actual state becomes clear the curator has to configure the desired future state.

The first case is when the current state is familiar, where the expertise required for a reply is either based on *doing know-how* or on *planning know-how*. In this situation, the determination of the future state regards the desires of the reporter.

If the reporter desires to do what the system is designed for, then a *doing know-how* expertise is required for a solution. The curator should address an artisan for a reply. Such an incident illustrates the lack of doing know-how on behalf of the reporter, i.e. his inability to use the service for its own development process as per the service was initially designed. Thus, the "problem" is traced in the reporter's ignorance. Therefore, it is also an indication that the means provided by the service to developers for the latter to acquire the necessary skills may not be sufficient. In such a situation additional actions regarding knowledge transfer from the enterprise to developers may have to be considered at an enterprise level, such as a revision of the documentation[15].

[15]Here is how the Developer Community Handbook evaluates documentation:

> *You can provide the best API in the world, but if you don't document it, the only developers that will use it are the ones that are paid to do it, or that enjoy pain. Most likely, your goal is to attract every type of developer - the hobbyists, the newbies, the freelancers, the paid employees, the student, etc. - and the goal of your documentation is to explain the API in a way that works for all of those types of developers, and all types of learners.*

If the reporter desires to do something the system is not originally designed for, then a *planning know-how* expertise is required. The curator should address a strategist for a reply. Such an incident illustrates the potential of the service to extend to different contexts. Such a potential, though, requires strategical action, which generally demands more resources. The way those requests are addressed is through their internal aggregation, evaluation and prioritization, before strategical action takes place.

The second case is when the current state is novel, in these circumstances the curator response also depends on the configuration of the desired state.

If the desired state is familiar, then the reporter faces a problem which is due to the system. An *understanding know-how* is required to solve this problem, so the curator should contact a repairer. Such an incident indicates that other UDEs may also meet the same problem, though its repair is a question of service robustness.

If the desired state is novel, too, then there is an impossibility to manage the situation. More information should be requested by the curator in order for her to be in position to address the issue. Unless the reporter - or other UDEs - are not in position to provide more information, no action can be undertaken. Hence, while in previous circumstances the responsibility for a resolution lay on the enterprise side, here it lies on the UDE side.

Moreover, as it is often observed, curators as well as some UDEs often have the doing, understanding and planning know-how: they can provide advice on "how to do it", on "why it does not work" or "other technologies to use for the same goal", even though it is not their domain of expertise.

17.4.5 Discussion: the role of the "curator" in service potential exploitation

Table 17.4 summarises the characteristics of the Curator as compared to the ones discussed in the literature review. The curator's know-how resides in taking care of the developer's concerns. Her action requires an expertise on novelty and expertise recognition, communication skills, as well as conversation structuring. Some knowledge on the expertise of other figures is required for the curator to be in position to recognise novelty and the corresponding expertise, as illustrated by Figure 17.13. Besides, this knowledge can occasionally be sufficient to resolve the challenges UDEs face, although it does not enter in a curator's action specificity. The role of this actor figure seems to be coherent with the approach of Garel and Midler (2001) on multiple actors problem solving processes, who value the relationship between the actors for the organisation of design modifications and the exchange of knowledge. Still, in the case of the Web services, this relation takes less formal and more massive dimensions.

In the setting studied, the specificity of a curator's action resided in the ability to configure the current and the desired state of a given system, through a conversational mode, assuring that *"less developers feel teased"*, to use the words of the *Google* employee. Hence, the curator has to be in position to establish an intimacy with the UDE, while having the capacity to structure the conversation for "sticky" information to be codified.

In what regards problem-solving as such, the identification of the curator allows the proposition of alternative performance criteria, beyond problems solved. Discussions that remain open - that is where the curator has not closed the conversation by acknowledging the request and addressing it to an expert - can be explored as well. As previously illustrated by Figure 17.12, during these situations the information provided on an issue by the reporter is not sufficient for the curator to determine either the current or the future states.

Actor figure	Know-how	Expertise
Artisan	Doing know-how.	In the way in which certain transformations are obtained by familiar actions.
Repairer	Understanding know-how.	Action and investigation to re-establish the original state of order.
Strategist	Combining know-how.	Knowledge combination, activity planning, new concept and knowledge creation.
Curator	Taking care know-how	Novelty & expertise recognition, intimacy establishment, conversation structuring.

Table 17.4: Curator: know-how and expertise.

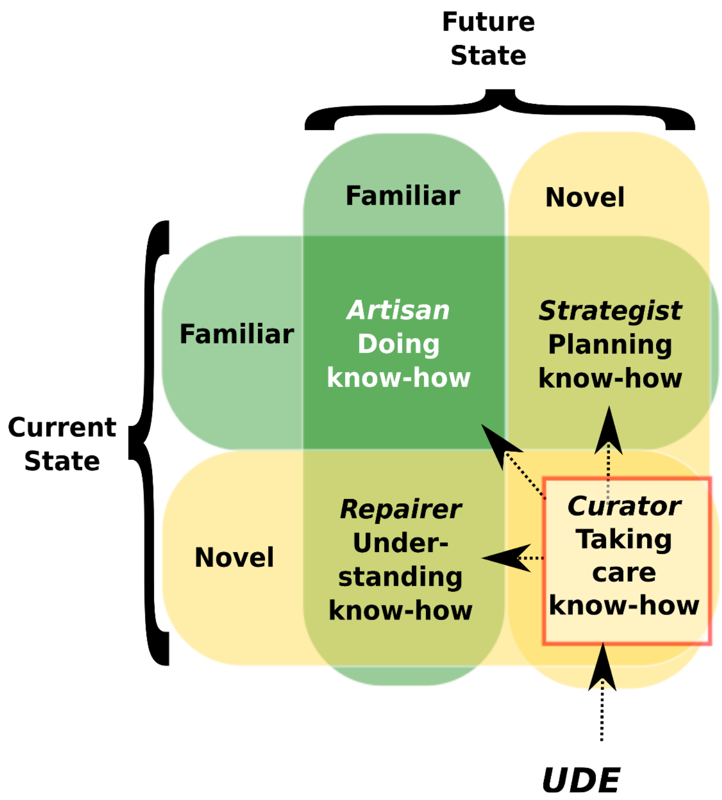


Figure 17.13: Different types of experts: the place of the Curator.

Therefore, an indicator measuring the rate of problems remaining open would address the capacity of a specific community of UDEs to formulate problems and requests in a way that enterprise employees understand them. It can be thus an indicator of the relationship between UDEs and enterprise, a sort of an "empathy indicator". This proposal suggests that the longer problems remain open, the less "empathy" there will be between UDEs and provider and the greater will be the distance between the two.

17.5 Conclusion

This chapter explored the conditions of UDE activity use for service potential exploitation, by studying *Facebook* and *Google Maps* developer support forums.

By examining these forums as problem-solving processes, I initially identified the peculiarity that, unlike community-based, enterprise-based and other mixed problem-solving processes, they are characterised by an uncommon element: it is the UDEs that set the problems to solve, while the enterprise is called to provide a resolution.

Further exploring this setting, I found that problem-solving rates are particularly low. Hence, I suggested that, while some problem-solving occurs, the function of those settings is broader.

Therefore, based on the work of Hatchuel and Weil (1992) on expertise, I identified an original type of expert, the "curator", whose task is to "take care of the developers". This task implies another type of know-how, requiring the ability to identify expertise and novelty, and to structure the conversation, while developing a sort of empathy between UDEs and the enterprise.

Chapter 18

Conclusion of Part III

18.1 The problem addressed

The current part addressed the question of how informal collectivities can emerge and the one of the harnessing of UDE collectivities by enterprises for exploitation and exploration reasons.

The distinction between networks and communities, the question of their emergence as well as the one of their harnessing by enterprises have been open questions in Open and User innovation literature (von Hippel, 2007; West et al., 2006; West and Lakhani, 2008). By constructing a conceptual framework putting *conversation* and *action* in between networks and communities, I was in position to explore the conditions of networks and communities emergence, as well as their harnessing.

For this research, I used the suggestions of Part II as an input, mainly in what regards the challenges of a parallel exploitation and exploration of a new object during the foggy competition phase and the specific role of UDEs in such a phase of industrial development.

18.2 Methods used

For my exploration, I used two main methods: observant participation and interaction monuments analysis. Both referred to ephemeral conversations and actions, where the perception of the researchers distance to the field is not perceived as great by the actors, as - given the temporal framework of the setting - they are also relatively distanced from it. Both approaches have been utilised in different settings, in the framework of a phenomenon-based research strategy (von Krogh et al., 2012), aiming to distinguish and explore the different settings studied, and to advance the proposal of alternative research and deployment designs, more appropriate for settings of this nature.

18.3 Part outcome

The current part has identified and explored three different settings that can be used during a foggy competition phase as such, or through the use of the attributes of the design proposed for further research or settings deployment.

The case of the *Barcamps* (Chapter 15) has illustrated a way in which UDEs can explore the possibility of new networks of communities' emergence. Unlike the literature on communities (Wenger, 1998; Wenger et al., 2002; O'Mahony, 2003; von Krogh et al., 2003b, and others) where a "common ground" and a "sense of belonging" exists among individuals, as well as the case of social networks (Granovetter, 1973, 1983; Simard and West, 2006; von Hippel, 2007, and others), *Barcamps* are conversational setting where no previous common ground or connections are pre-supposed. On the contrary, their design enables the exploration of the possibility for networking or community building on the basis of new or emerging topics. In addition, while they provide UDEs the opportunity to monitor emerging technologies, markets and uses, at the same time they provide the opportunity for enterprises to "infiltrate" this setting and obtain some early ties with the communities and networks still to emerge. Different from the "core" and the "periphery" of network analysis, as well as the "old" and the "news" in communities, the "regulars" of such settings are not necessarily related by personal ties or a common ground. However, they do have the opportunity to seize emerging situations and use their lessons, as well as their relational opportunities for their own purposes.

The case of the *Hackathon* (Chapter 16) has illustrated a more concrete way for enterprise technology or service potential exploration, through the organisation of ephemeral UDE activity. These settings have been found to be very efficient in terms of *"quality of group tacit knowledge"*, as all the levels proposed by Erden et al. (2008) were crossed within three days, for a significant number of groups. Having common elements with approaches that combine knowledge and creativity through collective exploration, such as the KCP (Hatchuel and Weil, 2009), the *Hackathon* also takes advantage of the individual knowledge and preferences of participants, both in developing concepts and evaluating them. This dimension has indicated that personality may play an important role in the innovation processes, especially in such ephemeral exploration settings.

Finally, the cases of *developer support forums* examined initially appeared as original forms of problem-solving, enrolled in the enterprise function of harnessing UDE activity for technology and service exploitation. Unlike problem-solving processes followed within user communities (von Hippel, 1990; Auray, 2004), enterprises (Thomke et al., 1998; Jablokow, 2005) or mixed approaches, such as crowdsourcing (Chesbrough et al., 2006; Lakhani et al., 2007), these forums are characterised by a reverse process: it is the "crowd" that creates the list of uses to be solved, while the enterprise is called to do the resolution work. As shown by the exploration of this field, its problem-solving performance is particularly low. Thus, advancing the hypothesis that they serve for other things, beyond problem-solving, I investigated this function as a specific enterprise expertise. Drawing on the work of Hatchuel and Weil (1992) on experts, I identified yet another type, which I described as a "curator", possessing a "taking care know-how", and addressing cases where both the current and the future circumstances of a system are not familiar. Advancing to a design proposition, I suggested that the contribution of such an expert to a system might be able to get qualified by the criterion of "UDE-enterprise empathy".

Conclusion de la Partie III

Problème de départ

Cette partie s'est posé la question de comment les collectivités informelles peuvent émerger, ainsi que celle de l'exploitation des collectivités des UDEs par des entreprises dans l'intérêt de ces dernières.

La distinction entre réseaux et communautés, la question de leur émergence ainsi que celle de leur exploitation ont été des problèmes ouverts pour la littérature de l'innovation ouverte et de l'innovation par l'utilisateur (West et al., 2006; von Hippel, 2007; West and Lakhani, 2008). À l'aide d'un cadre d'analyse conceptuel mettant l'action collective et la conversation entre les réseaux et les communautés, nous étions en position d'explorer les conditions de leur émergence ainsi que celles de leur exploitation, en ce qui concerne les UDEs qui développent des applications Web.

Dans cette recherche, nous avons utilisé comme point d'entrée les aboutissements de la deuxième partie, suggérant qu'au long d'une phase de *Compétition dans le brouillard* les UDEs jouent un rôle important à la fois en matière d'exploitation et d'exploration d'un potentiel envisageable.

Méthodes utilisées

Deux méthodes ont été utilisées pour les besoins de l'exploration entreprise dans cette partie : la participation observante et l'analyse de monuments d'interaction. Toutes les deux étaient utilisées comme méthodes propres à l'étude des configurations éphémères, où la distance du chercheur des enjeux du terrain, comme aperçue par les acteurs du terrain, n'est pas grande, car, étant donné le cadre temporel court de ces configurations, les acteurs sont eux-mêmes distanciés. Toutes les deux approches ont été utilisées dans des dispositifs différents, dans le cadre d'une stratégie de *« phenomenon-based research »* (von Krogh et al., 2012), visant à distinguer et à explorer ces dispositifs, ainsi qu'à poursuivre la recherche en proposant des concepts de recherche et de déploiement alternatifs, plus appropriés à des dispositifs de nature similaire.

Résultats de la partie

La présente partie a identifié et exploré trois dispositifs pouvant être utilisés lors d'une phase de la *« Compétition dans le brouillard »* en tant que tels, ou par le biais de l'usage des attributs des concepts proposés pour une recherche plus approfondie ou une mise en place plus complexe.

Le cas des *« Barcamps »* (Chapitre 15) a illustré une manière dont les UDEs peuvent explorer la possibilité d'émergence des nouveaux réseaux et des communautés. Contrairement au cas des communautés (Wenger, 1998; Wenger et al., 2002; O'Mahony, 2003; von Krogh et al., 2003b), où des « points communs » ainsi qu'un « sens d'appartenance » existe entre les individus, dans les *« Barcamps »* il n'y a pas quelque chose de partagé en amont par les participants. De plus, contrairement au cas des réseaux sociaux (Granovetter, 1973; 1983; Simard et West, 2006; von Hippel, 2007), les participants ne sont pas nécessairement interconnectés. En revanche, leur conception permet l'exploration de la possibilité de construction des communautés ou de « résautage » sur la base des sujets nouveaux ou émergeant. En outre, au même titre que ces dispositifs de conversation donnent la possibilité aux UDEs de veiller à l'apparition de nouveaux technologies, marchés et usages, ils donnent également l'opportunité à des entreprises d' « infiltrer » ces dispositifs et obtenir des premiers liens avec les communautés et les réseaux en émergence. Autres que le « noyau » des réseaux, ainsi que les « anciens » des communautés, les « réguliers » de ces dispositifs ne sont pas nécessairement connectés et ne partagent pas des points en commun. Néanmoins, ils ont tous l'opportunité de saisir des situations émergeant et d'utiliser leur expérience, ainsi que les opportunités relationnelles, dans leur propre intérêt.

Le cas du « *Hackathon »* (Chapitre 16) a illustré une manière plus concrète pour qu'une entreprise puisse explorer le potentiel de sa technologie ou son service, par le biais d'organisation d'un ensemble d'actions par de groupes des UDEs. Ces dispositifs se sont avérés très performants en matière de « qualité de savoir tacite de groupe », étant donné que tous les niveaux de qualité proposés par Erden et al. (2008) ont été franchis en trois jours, pour un nombre significatif de groupes. Ayant des éléments en commun avec des approches qui permettent une exploration collective des nouveaux concepts et savoirs, comme la KCP (Hatchuel et Weil, 2009), le *« Hackathon »* profite aussi des savoirs et préférences individuels des participants, à la fois pour le développement des concepts et pour leur évaluation. Cette dimension a également montré que la personnalité puisse avoir un rôle important dans les processus d'innovation, du moins quand il s'agit de tels dispositifs éphémères.

Enfin, les cas de forums de développeurs examinés, ont été présentés et étudiés initialement comme des dispositifs originels de résolution des problèmes, s'inscrivant dans une fonction d'entreprise visant à bénéficier de l'action des UDEs pour exploiter son service et sa technologie. Contrairement à d'autres processus de résolution des problèmes utilisés au sein des communautés d'utilisateurs (von Hippel, 1990; Auray, 2004), au sein des entreprises (Thomke et al., 1998; Thomke, 1998; Jablokow, 2005) ou encore des approches combinatoires, telle que le *crowdsourcing* (Chesbrough et al., 2006; Lakhani et al., 2007), ces forums sont caractérisés par un processus inverse : c'est la « foule » qui crée la liste des problèmes à résoudre, tandis que l'entreprise est appelée à faire le travail de résolution. Néanmoins, comme démontré par l'exploration de ce terrain, la performance en matière de résolution de problèmes y est particulièrement faible. Donc, en mettant en avant l'hypothèse que ces dispositifs servent à d'autres choses, au delà de la résolution des problèmes, nous avons étudié cette fonction de support en tant qu'objet d'une expertise particulière au sein de l'entreprise. Faisant appel au travail de Hatchuel et Weil (1992) sur les experts, nous en avons identifié encore un type, que nous avons décrit comme « curateur », ayant un « savoir prendre soin », et capable de faire face à des situations où à la fois les circonstances présentes et futures sont mal connues. En aboutissant à une proposition conceptuelle, nous avons suggéré que la contribution d'un expert de ce type à un système pourrait parvenir à être évaluée suivant le critère de l'« empathie » entre entreprise et UDEs.

Conclusion

Chapter 19

General conclusion

19.1 Problem addressed

The problem addressed by the current study has been the one of the investigation of Web-based applications development as a new phenomenon of *modus operandi*. The importance of Web services has been explored by many disciplines on the level of use: sociological research has focused on personal use (the enabling of personal expression and communication), management research has focussed on enterprise use (the transformation of enterprise organisation by the implementation of similar systems on an internal and external level), marketing research on the new possibilities for message diffusion (viral marketing) and economics on their use by buyers and sellers (network externalities, transaction and information cost reduction). This academic interest goes along with the increasing penetration of services such as *Google*, *Facebook* or *Twitter* in social and economic life.

Still, while the use and diffusion of Web services in general has attracted much interest from researchers, the development of these services has, so far, remained a "purely" engineering issue.

The current the Web application field through a business development perspective, using a phenomenon-based research strategy (von Krogh et al., 2012). once a distinction of the *modus operandi* in comparison with the models proposed by Raasch and von Hippel (2012) and von Hippel and von Krogh (2003, 2006) was made, the question of whether or not this is a Web field particularity owed to be explored, too.

An answer to this question required the exploration of Web-based application development through two different angles:

1. *What* is Web-based application development? This question addressed the need to distinguish and explore (von Krogh et al., 2012) the *modus operandi* (actors, reasons, means) of Web-based application development as compared to the known models, namely the user and the manufacturer innovation paradigms (von Hippel and von Krogh, 2003, 2006; Raasch and von Hippel, 2012). To identify this *modus operandi*, the identification of its actors, reasons and means was required. Thus, the following questions have been also explored:

    (a) *Who* develops Web-based applications?

    (b) *Why* do they he do it?

    (c) *How* is it done?

2. *I*s it only a Web phenomenon? If not, when does this phenomenon appear and *what* are its *effects*? This question addressed the need to specify the conditions of the phenomenon manifestation, as compared to other phenomena, namely the lead-user innovation (von Hippel, 1975) and the disruption (Christensen et al., 1998) ones.

Once these questions are answered, the question of how enterprises can exploit the phenomenon effects was answered by addressing the following questions:

1. How can the actors of the phenomenon emerge?

2. How can an enterprise exploit their action?

To address this question, a configuration of Web-development *modus operandi* has been presented, as compared to the known models. For this comparison, I used the distinction between user and manufacturer innovation paradigms (von Hippel and von Krogh, 2003, 2006; Raasch and von Hippel, 2012) as a reference point for analysing the phenomenon.

19.2 Methodology used

The methods used throughout this study corresponded to the specificities of the questions posed, making use of a phenomenon-based strategy (von Krogh et al., 2012). In Part I, I used methods akin to the distinction and the exploration of the phenomenon. In Part II, I used a method akin to the understanding of the conditions of phenomenon appearance and its effects in business. In Part III, the methods used were akin to the proposition of ways to exploit the phenomenon's effects.

19.2.1 Part I: phenomenon distinction and exploration

Part I addressed the question of the *modus operandi* peculiarities distinction and exploration, that is the questions of who, why and how develops Web-based applications. For this, I used a step-by-step methodological approach, each time considering one factor known (actor, reason or means), while exploring the other two.

The first step of my research investigated some early indications leading from a known *modus operandi* to the target one. Hence, through *interviews* with managers of enterprises active in the Web services sector, I have been able to reconstitute their *discourse* not on their expertise (the actors, reasons and means used within their own enterprise), but on the things they don't master, the things judged as original. This discourse provided me with an entry point to my field regarding the actor and his means, namely the "developer" and the "Application Programming Interfaces (*APIs*)".

Then, considering the actor known (the developer), I joined the action of developers, exploring its reasons and means. For that, I used an *"observant participation"* method, consisting in participating in the field challenges as an actor, while investigating the action norms followed in the field. The specific case studied was an "ad hoc" collective action, consisting in the development of a Web site for the merchandise of applications, created during a "Hackathon" event, in Silicon Valley. There, I observed that developers used both open source devices and APIs (provided by enterprises) to create their own service, while the reasons for this creation appeared to be both use-related and commercial, being thus placed in-between the user and the manufacturer innovation models.

Afterwards, I considered the specific means used as known (the APIs) and explored the underpinning actors and the reasons of their use. For that, I studied the "Cookbooks" describing how to use APIs for Web development. By comparing these action norms to the ones outlined by von Hippel and von Krogh (2003, 2006) regarding the "private-collective" and the "private-investment" models, I identified three actor-figures: the User-Developer (UD), the User-Developer-Entrepreneur (UDE) and the Developer-Entrepreneur (DE).

Finally, considering the specific reason known (own application commercialisation), I investigated the actor and the means. By using a "known reason story identification" method, I studied the case of an application developed by an *eBay* user (a seller) for his own use, and then commercialised towards other users (sellers). I concluded with the proposal for a "personal investment" model, according to which UDEs make use of both lead user and manufacturer's *"sticky information"* (von Hippel, 2005), initially innovating for their own use, while not *"freely revealing"* (von Hippel and von Krogh, 2003) their innovations later.

19.2.2 Part II: conditions of appearance and effects

Part II used a "problematization" methodology, consisting in examining the originality of phenomenon, its conditions of appearance and its effects. More specifically, the *modus operandi* of Web-application development has been the criterion for the re-examination of the history of three industrial settings, namely the enterprise computer, the personal computer and the radio ones.

More specifically, I used the elements of the Web-based application phenomenon (its actors, reasons and means) as a lens for the reading of the emergence of those industries and I found common patterns in what regards the action of UDs, UDEs and DEs, the means they have at their disposal and the reasons why they act.

This study was based on well-known historical works, while some use of original texts was also required for the further exploration of topics not specifically highlighted by historians. It has allowed the positioning of the modus operandi in industrial dynamics, as compared to the user innovation proposal (von Hippel, 2005), the innovation diffusion model of Rogers (2003) and the disruption approach (Christensen et al., 1998).

19.2.3 Part III: phenomenon exploitation

Finally, in Part III, I explored the conditions for the phenomenon exploitation. The methods I used were "observant participation" and "interaction monument analysis".

Initially, using an observant participation approach, I explored the conversational settings of the UDEs themselves, useful to them for the exploration of the possibility of new networks and communities emergence, also being "infiltrated" by the enterprises of the domain. Interaction monuments of these settings were used as complementary data to further explore the phenomenon.

Then, I continued using the observant participation approach to join an UDE action aiming to the exploration of the potential of a specific technology, also completing my observation material with interaction monuments analysis. I identified thus one way for enterprises to exploit UDE action for their own service potential exploration.

Finally, I studied one of the means that enterprises possess for developer support, forums, based on an interaction monuments analysis. I concluded with one of the actors that should be part of an enterprise exploiting UDE action, the "curator".

19.3 Contribution

The contributions of this study come from the exploration of the Web-based application modus operandi, its positioning in relationship to the literature on industrial development, as well as in relationship to other industrial settings and the proposition of methods privileging the harnessing of its benefits. The methods used, while often designed for the exploration of my specific problem in a specific field, could also be tried out in different settings.

19.3.1 On the configuration of the Web-based application development modus operandi

The current thesis original contribution lies in the exploration of a field which admittedly has a major importance for business, Web-based application development, though has not been previously addressed as such by business scholars.

Many business-related disciplines have studied Web services, though all adopting a use perspective. Scholars have stressed the strategical importance of having many end-users - network externalities - (Katz and Shapiro, 1985; Caillaud and Jullien, 2003; Rochet and Tirole, 2003, 2004; Parker and Alstyne, 2005; Eisenmann et al., 2006, and others), though this dimension is generic to networked system and not specific to Web services. Within the market approach, Web services have also been studied as settings allowing the reduction of information and transaction costs for its end-users, both buyers and sellers, (Wallis and North, 1986; Choudhury et al., 1998; Brousseau and Chaves, 2004; Christensen et al., 2005; Baldwin, 2008, and others), though this possibility is met every time a new communication system appears. Management scholars have pointed out the difficulties met by enterprises implementing such settings for internal or external use, where the problem is traced on the use, namely the new expression possibilities given to employees and their implications, amongst others, in roles attribution (Fredberg, 2009; Haefliger et al., 2011; Sutanto et al., 2011; Burger-Helmchen and Cohendet, 2011, and others). Sociologists have explored the use of such settings at an end-user level, proposing that their use value lies in the process of personal identity expression and formation (Cardon and Delaunay-Téterel, 2006a; Cardon, 2008; Beuscart et al., 2009, and others). Still, no understanding of the development process of the services studied is provided by those studies.

The exploration of Web-based application development has revealed a *modus operandi* which is different from both user and manufacturer innovation paradigms (Raasch and von Hippel, 2012; von Hippel, 2005). In respect to the actor, I identified three configurations of developers: the user-developer (UD), the user-developer-entrepreneur (UDE) and the developer-entrepreneur (DE). Unlike the well-known case of open source software (OSS) development (von Krogh et al., 2003a,b; Lakhani and Wolf, 2003; Auray, 2004; Benkeltoum, 2008, and others), the actors of this modus are not limited to the use of OSS: they also have at their disposal an "innovation palette", a set of devices (*Application Programming Interfaces*) furnished by service providers, for them to innovate. Moreover, these actors often possess both *"user sticky information"* and *"manufacturer sticky information"* (von Hippel, 1994, 1998). Regarding the reasons for their action, they can be both use-based and commercial.

More specifically, UDs can be characterised as "skilled lead users", as they use their development capabilities and the means available to them to create applications for their own use. UDEs, have the same starting point with UDs, though they attempt the commercialisation of their creation, instead of freely revealing it, as the user innovation literature suggests (von Hippel and von Krogh, 2003, 2006). Finally, DEs do not innovate for the same reasons that users do: their action directly targets commercial ends, and thus implies the distancing of their own personal preferences and their design goals, as the latter are defined by their projection onto their potential clientele.

19.3.2 On the positioning of a specific *modus operandi* in relationship to industrial dynamics

A second contribution lies in the positioning of this innovation *modus operandi* in industrial dynamics. While user innovation is positioned before innovation diffusion and, thus, before early adoption (von Hippel, 1975; Churchill et al., 2009; Rogers, 2003), UDEs and DEs mark the bridge in-between innovation for own use and commercial innovation. However, this process is not exactly like disruption (Christensen et al., 1998), where a new technology tries to identify an appropriate market. The study of the appearance of the identified *modus operandi* in the three cases of industrial development has suggested that actors, means and reasons are mutually transformed, in parallel to the transformation of technologies, uses, markets, while new enterprise actors are likely to emerge from this process.

The character of this transformation is found to be particular to the phases an industrial setting undergoes during its emergence. Early materialisations cannot come from "unskilled" lead users, as an advanced knowledge of new theories and technologies is required to explore a new potential. In the settings studied, such knowledge came from the intimate circles of users, developers, entrepreneurs and academics. Then, a market emerges usually through successful efforts of UDEs, likely to be materialising an old concept before a new one, that leads to the illustration of the market possibility. Then, until an industrial rationalisation synthesises the technologies, markets and uses explored in some common design rules (Baldwin and Clark, 2000), proposing a new organisational schema, a technical substrate and a management philosophy, DEs, enterprises and early adopters co-exists in business environment characterised by "foggy competition". DEs are then called to prioritize potential market opportunities over their own preferences and to further innovate in order to reach what seems to be a market potential. Established enterprises also join the market, though most usually implementing older rationalisation schemas to develop and market the new object.

19.3.3 On the conditions of UDE action use

A third contribution of this study lies in the exploration of ways to exploit UDE activity. Three settings have been identified and explored for:

- the creation of the conditions for UDE communities and networks emergence,
- the use of UDE activity by enterprises for their technologies and services potential exploration and
- the use of UDE activity by enterprises for their technologies and services potential exploitation.

More specifically, the study of *Barcamps* conversational settings has proposed a design for the deployment and the further research of settings enabling the exploration of the possibility for the emergence of new communities and networks. The elements of such a design are *broad conversation topics*, addressing an emerging issue,*open invitation*, allowing the participation of new or surprising actors, *networking*, enabled by the division of participants in small groups, an intimate ambiance may be created amongst them and common interests may emerge.

Then, the study of the *Hackathon* that took place in *Google*'s headquarters has proposed a design for the deployment and the further research of exploratory settings, based on the organisation of ephemeral, focussed and distributed development of prototypes making use of a new potential. Knowledge on the "updated state of the art" owes to be transferred to participants in order for them to be in a position to explore its potential. Self-selection and voting methods can be applied for the participants to evaluate themselves the concepts explored. As suggested by the study of the specific *Hackathon* outcomes, participants' creations are likely to be influenced by their own personal preferences and their intimate environment knowledge.

Finally, the study of two developer support forums, that of *Facebook* and *Google Maps* has suggested that, unlike usual problem-solving processes, developer support is not exclusively focussed on problem resolution.

19.3.4 On methodology

To access this rather fluid field and be in position to explore my research problem, I used methods that valued the discourse of the actors on the one hand, and my own involvement in ephemeral actions or conversations on the other. While the volatile nature of these data implied a difficulty for me to generalise, it allowed me at the same time to seize the challenges of a very dynamic, though not yet rationalised, field, where actors themselves are often unsure about the "right" methods to use.

The use of different research angles (triangulation) allowed me to distinguish and explore (von Krogh et al., 2012) the *modus operandi* under study and to eventually propose research and deployment designs for specific methods that benefit its harnessing.

"Observant participation" valued my own involvement as a researcher in the field challenges, while the exploratory and ephemeral nature of most fields themselves contributed in narrowing the distance between observant and observed.

"Interaction monuments analysis" exploited the fact that Web interaction leaves traces. Unlike archival study, where an archivist intervenes in a data structure, in this method I had to design the categories myself, to turn "raw material" into "research data". This task was however coherent with an exploratory research perspective, where the categories are to emerge as a research result.

A "limited problematization" was also required for exploring whether or not the *modus operandi* identified was a Web business particularity. Thus, the use of this *modus* as a reference problem for the study of other industrial settings has shown that similar settings have also appeared elsewhere. This exercise has also shown that the distinction in relation to the literature (von Krogh et al., 2012) should be completed with the comparison to other settings, in order to conclude whether or not what appears as a new phenomenon actually is one: there is also a chance that the object of study is just a dimension of business that simply has not been paid much attention to by the literature.

19.4 Further research perspectives

My study has focussed on the Web, and in the second Part, it expanded to similar industrial settings. It would thus be interesting to test the utility of the outcomes of my exploration in contrasted industrial settings.

Moreover, regarding the specific role of UDs, UDEs and DEs, economic research could explore the "development cost" of their activity. Unlike well-known "entry costs", and as Christensen et al. (1998) also notes, often there is no market to enter. Such a cost consideration, should take into account the resources required on an individual level for the acquisition of the expertise and the means necessary for creative experimentation, as well as for early materialisation development.

Regarding the specific means these actors use, *FOSS* and *APIs*, being based on previous research on the open source phenomenon, I did not actually go deep into the categorisation of these means. Both fields can be further explored to propose new categories of such means (possibly in relationship to the different actor figures that use them). Still, such a research would require a profound understanding of the technical stakes, and thus would better be undertaken using an inter-disciplinary approach, bringing together both management and engineering knowledge.

Quantitative research could also use the outcomes of the current study to test and evaluate the concepts proposed. Thus, surveys examining the use of services by UDs, UDEs and DEs, according to the distinction made in the first part, could possibly provide us with a configuration of an emerging sector, according to the phases distinguished in Part II.

Quantitative research could also explore the usefulness of the "enterprise-UDE empathy" concept, and further explore when this empathy should be cultivated and when it should better be avoided.

In addition, the precise measurement of the specific market performance of the enterprises studied in Part II, could further refine - or question - the comparison of my proposal as compared to the model of Rogers (1962). This task would also require an interdisciplinary approach, this time between management and history disciplines, most importantly in what regards archival research methods for data on companies that have ceased existing for decades.

Finally, the empirical use of the notions of *collective action* (Hatchuel, 2005b) and *conversation* as a meeting point between networks and communities has enabled me to study ephemeral settings and examine their design, amidst a shifting business environment. However, the study of the relationship between collective action and conversation may imply important epistemological questions, being able to provide a basis for further problematization.

Conclusion générale

Problème de départ

Le question posée par la présente étude a été celle du *modus operandi* du développement des applications Web comme un phénomène nouveau. L'importance des services Web a déjà été étudié par plusieurs disciplines au niveau de l'usage : des études en sociologie avaient exploré l'usage personnel (en mettant l'accent sur les nouvelles possibilités d'expression personnelle et de communication), des études en gestion avaient exploré l'usage de ces dispositifs au sein de l'entreprise (en mettant l'accent sur les transformations organisationnelles engendrées par leur mise en place, tant au niveau interne qu'au niveau externe), des études en marketing avaient exploré les nouvelles possibilités ouvertes en matière de diffusion des messages commerciaux (*« viral marketing »*) et les économistes avaient mis en lumière leur valeur pour des vendeurs et des acheteurs (externalités de réseaux, réduction des coûts d'information et de transaction). Cet intérêt académique a suivi la pénétration croissante des services comme *Google, Facebook* ou *Twitter* dans la vie sociale et économique.

Cependant, bien que la diffusion des services Web en général ait été mise en valeur par toutes ces disciplines, le développement même de ces dispositifs restait jusqu'à présent une affaire « purement » d'ingénierie.

La présente étude a exploré ce champ en adoptant une perspective de développement d'affaires, utilisant une approche de *« phenomenon-based research »* (von Krogh et al., 2012). Une fois la distinction du *modus operandi* spécifique de ce terrain faite en comparaison à des modèles synthétisés par Raasch et von Hippel (2012) et par von Hippel et von Krogh (2003; 2006), la question de savoir si cette spécificité est particulière au champ étudié ou non, devrait être également explorée.

Par conséquent, la proposition d'une réponse à ce problème complexe d'identification de l'existence ou pas d'un phénomène nouveau, exigeait une investigation à deux entrées, l'une concernant le présent, l'autre le passé :

1. Qu'est-ce le développement des applications Web? Répondre à cette question a exigé la distinction et l'exploration (von Krogh et al., 2012) du *modus operandi* (acteurs, moyens et raisons d'action) du développement des applications Web comme comparé à des modèles connus, à savoir les paradigmes d'innovation par l'usager et par l'industriel (von Hippel et von Krogh, 2003; 2006, Raasch et von Hippel, 2012). Cet objectif de recherche a, donc, imposé l'étude des questions suivantes:

    (a) *Qui* développe des applications Web?

    (b) *Pourquoi* il le fait?

    (c) *Comment* il le fait?

2. Est-ce le *modus operandi* un phénomène propre au développement des applications Web? Si non, quand est-ce qu'il apparaît et quels sont ses effets pour l'industrie?

Bien que la réponse à la première question exige l'étude du contexte actuel et spécifique de ce modus operandi, une réponse à la deuxième question exigeait une prise de distance historique, la dernière étudiant les conditions de l'émergence de la possibilité de son apparition (Lefebvre, 2005), et la comparaison avec d'autres approches proches sur le développement industriel à travers l'innovation (Christensen, 1997; von Hippel, 1977b).

Une fois ces questions répondues, celle de l'exploitation du *modus operandi* en question a été exploré, à l'aide des sous-questions suivantes:

1. Comment ses acteurs peuvent-ils émerger aujourd'hui?

2. Comment une entreprise peut-elle exploiter leur action?

Pour explorer ces dernières questions, une clarification des notions des communautés et des réseaux, utilisées souvent dans la littérature d'innovation sans discrétion (von Krogh et al., 2003b; von Hippel, 2007; West et Lakhani, 2008), a été nécessaire. Un cadre d'analyse plaçant l'action collective (Hatchuel, 2005c) et la conversation entre ces deux notions, m'a aidé à explorer des dispositifs de conversation et d'action éphémères, contribuant à la configuration des conditions d'exploitation de l'activité des UDEs dans le Web.

Méthodologie utilisée

Les méthodes utilisées au long de cette étude correspondaient à des spécificités des questions posées, et faisaient usage d'une stratégie de recherche de phénomène. Dans la Partie I, nous avons utilisé des méthodes conformes à la distinction et l'exploration du phénomène supposé. Dans la Partie II, nous avons utilisé des méthodes conformes au besoin de compréhension des conditions d'apparition du phénomène ainsi que de ses effets dans le monde d'affaires. Dans la Partie III, les méthodes utilisées étaient conformes à l'objectif de proposition des manières d'exploiter les effets du phénomène.

Partie I : les méthodes à distinguer et à explorer un phénomène

La première partie a eu comme objectif la distinction et l'exploration des spécificités du mode opératoire du développement des applications Web, en se posant les questions de *qui, comment* et *pourquoi* les développe. À ce propos, nous avons utilisé une approche méthodologique étape-par-étape, en considérant à chaque fois un des éléments (acteur, moyens, raison) connu pendant que nous explorions les autres deux.

La première étape de notre recherche a étudié quelques indications précoces conduisant d'un *modus operandi* connu à un autre, ciblé. Ainsi, par le biais des entretiens d'experts d'entreprises actives dans le secteur du Web, nous étions en mesure de restituer leur discours, non pas en ce qui concerne leur expertise propre (les acteurs, leurs raisons et moyens d'action au sein de leur propre entreprise), mais sur les choses qu'ils ne savaient pas, celles qu'ils jugeaient eux-mêmes comme originales. Ce discours nous a donné un point d'entrée à notre champ de recherche concernant l'acteur et ses moyens à étudier, à savoir le « développeur » et les « *Application Programming Interfaces (APIs)* ».

Par la suite, tout en considérant l'acteur connu (le « développeur »), nous avons pris partie à l'action des développeurs, pour explorer leurs raisons et leurs moyens d'action. Pour cela, nous avons utilisé une méthode de « participation observante », consistant en notre propre participation aux enjeux du terrain comme un des acteurs, tout en explorant les normes d'action y rencontrées. Le cas particulier étudié était une action collective « ad hoc », visant au développement d'un site Web pour la commercialisation d'applications Web, crée au long d'un *« Hackathon »* dans la Silicon Valley. Là, nous avons observé que les développeurs faisaient usage à la fois des dispositifs *open source* et des *APIs* (ces dernières mises à leur disposition par des fournisseurs des services), afin de créer leur propre service ou application, tant que les raisons de leurs actions semblaient être à la fois liées à leur propre usage et à des fins commerciaux, se plaçant alors parmi les modèles d'innovation par l'usager et par l'entreprise.

Ensuite, nous avons considéré les moyens spécifiques utilisés comme étant connus (les *APIs*), et nous avons exploré les acteurs sous-jacents et leurs raisons d'action. À ce propos, nous avons étudié des « livres de cuisine » décrivant comment utiliser des APIs pour développer des applications Web. En comparant ces normes d'action à celles décrites par von Hippel et von Krogh (2003; 2006), à savoir les modèles d'investissement privé et le collectif-privatif, nous avons identifié trois figures d'acteurs : l'usager-développeur (UD), l'usager-développeur-entrepreneur (UDE) et le développeur-entrepreneur (DE).

Enfin, en considérant la raison connue (la commercialisation de son propre application), nous avons étudié l'acteur et les moyens de développement. En faisant usage d'une méthode « d'identification d'histoire à raison connue », nous avons étudié le cas du développement d'une application pour les vendeurs du service *eBay,* initialement développée pour son propre usage et commercialisée par la suite. Cette étude a conclu à la proposition d'un modèle d'innovation par « l'investissement personnel », selon lequel les UDEs utilisent leur savoir personnel, portant à la fois sur le contexte d'usage et sur la technologie, pour innover.

Partie II : conditions d'apparition et effets du *modus operandi* identifié

La deuxième partie a utilisé une méthodologie de « problématisation limitée », visant à examiner l'originalité du mode opératoire identifié, ses conditions d'apparition et ses effets. Plus précisément, ce modus fut le critère de la relecture de l'histoire des trois cadres industriels, à savoir celui de l'ordinateur entreprise, celui de l'ordinateur personnel et celui de la radio.

À ce propos, nous avons utilisé les éléments du développement des applications Web (ses acteurs, leurs moyens et leurs raisons d'action) comme instrument de lecture de l'histoire de l'émergence de ces trois industries, et nous avons trouvé des points communs en ce qui concerne l'action des UDs, des UDEs et des DEs, ainsi que parmi les moyens d'action qu'ils disposaient dans des cas différents.

Cette étude a été fondée sur des travaux historiques bien connus, bien qu'un recours à des textes originaux soit parfois exigé, pour approfondir l'exploration des aspects indiqués par notre instrument de lecture, quoique moins étudiés par les historiens. Ainsi, un positionnement du *modus operandi* en question dans la dynamique du développement industriel est convergé, ayant des caractéristiques différentes des lectures de l'innovation par l'usager (von Hippel, 2005) et du dilemme de l'innovateur (Christensen, 1997).

Partie III : conditions d'exploitation du phénomène

Enfin, la troisième partie a exploré les conditions d'exploitation du phénomène. Cette étude a fait usage des méthodes de la « participation observante » et de l' « analyse des monuments d'interaction ».

Tout d'abord, en utilisant une approche de participation observante, nous avons exploré les dispositifs conversationnels utilisés par les UDEs eux-mêmes, pouvant leur servir à l'exploration de la possibilité d'émergence de nouvelles communautés ou des nouveaux réseaux, étant par ailleurs « infiltrés » par des entreprises du secteur. Les monuments d'interaction de ces dispositifs ont été utilisés comme des données supplémentaires à nos propres observations, pour soutenir une exploration plus approfondie.

Ensuite, nous avons continué notre recherche en utilisant l'approche de la participation observante pour faire partie d'une action des UDEs, visant à l'exploration du potentiel d'une technologie spécifique, en complétant également nos observations avec une analyse des monuments d'interaction. Il en est résulté l'identification d'une manière pour les entreprises d'explorer le potentiel de leurs technologies.

Enfin, nous avons étudié un des moyens disposés par les entreprises pour le support des développeurs, les forums, en faisant usage d'une analyse des monuments d'interaction. Nous avons conclu cette recherche en proposant un type d'expert qui devrait faire part d'une entreprise Web, le « curateur ».

Contribution

Les contributions de cette étude viennent de l'exploration du *modus operandi* du développement des applications Web, de son positionnement dans la dynamique industrielle, ainsi que de la configuration des méthodes facilitant son exploitation. La méthodologie utilisée, bien que conçue pour les objectifs de notre propre recherche, pourrait également être adoptée pour l'étude des problématiques différentes dans d'autres champs de recherche.

Sur la configuration du *modus operandi* du développement des applications Web

La première contribution de la présente thèse réside dans l'exploration d'un champ d'affaires, dont l'importance a été largement reconnue par plusieurs disciplines, quoique peu exploré au delà des aspects d'usage.

Plusieurs disciplines liées à l'étude du monde d'affaires ont étudié le champ des services Web, en adoptant une perspective d'usage. Des économistes ont souligné l'importance stratégique pour une entreprise du secteur d'avoir un grand nombre d'usagers, plusieurs *externalités de réseau* (Katz et Shapiro, 1985; Caillaud et Jullien, 2003; Rochet et Tirole, 2003; 2004; Parker et Alstyne, 2005; Eisenmann et al., 2006), quoique cette dimension soit rencontrée dans tous les systèmes en réseau et ne soit pas spécifique aux services Web. Au sein des approches marchandes, les services Web ont été également étudiés en tant que dispositifs permettant la réduction des coûts d'information et ceux de transaction pour leurs usagers finaux, à la fois acheteurs et vendeurs (Wallis et North, 1986; Choudhury et al., 1998; Brousseau et Chaves, 2004; Christensen et al., 2005; Baldwin, 2008), quoique cette possibilité soit soulignée par toutes les études de la même discipline portant sur les systèmes de communication. Des chercheurs en Gestion ont souligné les difficultés auxquelles les entreprises font face, lors de la mise en place de tels dispositifs pour usage interne ou externe, où le problème est identifié à l'usage, notamment sur les nouvelles possibilités d'expression s'ouvrant pour les employés, et leurs implications dans l'attribution des rôles, entre autres (Fredberg, 2009; Haefliger et al., 2011; Sutanto et al., 2011; Burger- Helmchen et Cohendet, 2011). Des sociologues ont exploré l'usage de tels dispositifs d'un point de vue d'usager final, en affirmant que leur valeur d'usage réside dans le processus de construction et d'expression de l'identité individuelle (Cardon et Delaunay-Téterel, 2006; Cardon, 2008; Beuscart et al., 2009). Pourtant, ces études ne suggèrent pas une approche particulière quant au développement même de ces dispositifs.

L'exploration du développement des applications Web entreprise dans la présente étude, a révélé un *modus operandi* distingué à la fois par les modèles d'innovation par l'usager et par l'entreprise von Hippel and von Krogh (2003, 2006); Raasch and von Hippel (2012). Concernant son acteur, nous en avons configuré trois figures : l'usager-développeur (UD), l'usager-développeur-entrepreneur (UDE) et le développeur - entrepreneur (DE). Contrairement au cas bien étudié de l'*open source* (von Krogh et al., 2003a,b; Lakhani et von Hippel, 2003; Auray, 2004; Benkeltoum, 2009), les acteurs en question ne sont pas contraints à l'usage de l'open source : ils innovent à la base d'une « palette d'innovation », qui leur est fournie à la fois par des entreprises et par des communautés. En outre, ces acteurs disposent d'un savoir à la fois de la *« user sticky information » et de la « manufacturer sticky information »* von Hippel (1995). Quant aux raisons de leur action, elles sont à la fois liées à l'usage et à la possibilité d'une commercialisation.

Plus précisément, les UDs peuvent être décrites comme des « *lead users* compétents » , car ils utilisent leurs compétences de développement et les moyens disponibles afin de créer des applications pour leur propre usage. Les UDEs partagent la même démarche de développement, sauf qu'ils poursuivent la commercialisation de leur application par la suite, au lieu de la révéler librement, comme le suggérait la littérature sur l'innovation par les usagers (von Hippel et von Krogh, 2003; 2006). Enfin, les DEs n'innovent pas pour les mêmes raisons que les usagers : leur action vise directement à des fins commerciaux, fait qui implique une distanciation entre leurs préférences personnelles et leurs objectifs de conception, car ces derniers sont définis par leurs projections sur leur clientèle potentielle.

Sur le positionnement du *modus operandi* identifié dans la dynamique industrielle

Une deuxième contribution de cette thèse tient au positionnement du mode opératoire identifié dans la dynamique industrielle. Bien que l'innovation par les usagers soit positionnée avant la diffusion et, donc, son adoption est précoce (von Hippel, 1977b; Churchill et al., 2009; Rogers, 2003), les UDEs et les DEs créent un pont entre l'innovation pour son propre usage et l'innovation à des fins commerciaux. Cependant, ce processus n'est pas exactement une « *disruption »* (Christensen, 1997), selon laquelle un marché serait à identifier. L'étude des conditions d'apparition du *modus operandi* en question dans l'histoire des trois cadres industriels, a suggéré une transformation mutuelle des acteurs, leurs moyens et leurs raisons d'action, en parallèle de la transformation des technologies, des marchés et des usages, durant laquelle d'autres acteurs sont susceptibles d'émerger, au delà d'une simple diffusion d'une innovation à un nouveau marché.

Ces transformations sont caractérisées par des conditions bien spécifiques. Des matérialisations précoces ne peuvent pas être entreprises par des « *lead users »,* à moins qu'ils aient des compétences de développent assez pointues, tandis qu'un savoir assez avancé sur les technologies et les théories nouvelles est exigé afin d'être en mesure d'explorer un nouveau potentiel. Dans les cadres industriels étudiés, ce savoir provenait des cercles intimes des usagers, développeurs, entrepreneurs et académiques. Par la suite, un marché émerge souvent à partir des efforts bien aboutis des UDEs, susceptibles de matérialiser d'abord un concept ancien, et révélant un nouveau potentiel de marché. Ensuite, cette activité exploratoire se poursuivra, lorsque des DEs ainsi que des entreprises exploiteront l'objet en question dans un marché « brumeux », où l'identité de l'objet produit et vendu restera mal connue. Enfin, la rationalisation industrielle prend en considération les technologies, les marchés et les usages explorés auparavant, en les synthétisant dans la proposition des règles de conception englobantes (Badwin et Clark, 2000) et en concevant un nouveau schéma organisationnel, une philosophie de gestion et un substrat technique (Hatchuel et Weil, 1992). Une telle rationalisation est susceptible de dominer l'ensemble du marché, sauf si une matérialisation précoce qui n'y soit pas englobée ouvre un nouveau cycle.

Sur les conditions d'exploitation de l'action des UDEs

Une troisième contribution de cette étude réside dans l'exploration des méthodes utiles à l'exploitation de l'action des UDEs. Trois dispositifs ont été configurés, qui permettent de:

- créer les conditions pour l'émergence des réseaux et des communautés des UDEs,
- donner la possibilité aux entreprises d'utiliser l'action des UDEs afin d'explorer le potentiel de leurs propres services et technologies,
- donner la possibilité aux entreprises d'utiliser l'action des UDEs afin d'exploiter le potentiel de leurs propres services et technologies.

Plus précisément, l'étude des dispositifs de conversation des *Barcamps* a conduit à la proposition d'un ensemble de concepts utiles au déploiement des dispositifs permettant l'exploration de la possibilité d'émergence de relations informelles favorisant l'innovation. Ces mêmes concepts pourraient être également utilisés comme base pour un approfondissement de la recherche sur les dispositifs conversationnels eux-mêmes. Ainsi, des *sujets de conversation vastes*, qui visent un thème émergeant, des *invitations ouverte*s, permettant la participation de nouveaux acteurs ou d'acteurs surprenants, peuvent devenir des paramètres de conception et de conduction des dispositifs de conversation, ayant comme objectif de faire du *« réseautage »*, aidé par la division des participants en de petits groupes, ainsi que de privilégier le développement d'une ambiance intime entre les participants où des intérêts communs puissent être identifiés.

Par la suite, l'étude du *Hackathon* qui a pris lieu dans les locaux de *Google,* a abouti à la proposition d'un design pour le déploiement des dispositifs d'action exploratoire et leur étude. Dans ce dispositif éphémère, le savoir d'un « état d'art mis à jour » était transmis aux participants, afin qu'ils puissent explorer le potentiel de la technologie en question. L' « auto-sélection » des participants aux projets, ainsi qu'un vote pour désigner les meilleurs projets ont fourni une évaluation des concepts lors même de leur émergence et leur premiers aboutissements de développement. De plus, comme l'étude de ces projets l'a suggéré, ces concepts sont susceptibles d'exprimer les préférences personnelles de leurs développeurs, observation en prendre également en compte dans l'étude et l'organisation de tels dispositifs exploratoires.

Enfin, l'étude de deux forums de support de développeurs, celui de *Facebook* et celui de *Google Maps,* a suggéré que, contrairement aux processus divers de résolution de problèmes déjà étudiés par la littérature, ils ne visent pas exclusivement à la solution des problèmes. Cette étude a proposé que l'expertise exigée pour la gestion de ces forums est celui d'un « curateur », dont l'activité peut être comprise comme agissante sur l'« empathie » entre l'entreprise et les UDEs. Cette empathie est mesurée par le partage des problèmes, plutôt que par leur solution.

Sur la méthodologie

Afin d'accéder à un terrain plutôt fluide, et se mettre en position d'explorer mon problème de recherche, nous avons utilisé des méthodes qui ont mis en valeur d'une part le discours des acteurs, de l'autre mon implication propre à des dispositifs d'action et de conversation éphémères. Bien que la nature instable des données récoltées par ces méthodes impliquait une difficulté de généralisation des résultats supplémentaires, elle m'a permis en même temps de saisir les enjeux d'un champ très dynamique, quoique pas encore rationalisé, où les acteurs eux-mêmes doutent souvent des la pertinence des méthodes à suivre.

L'usage des angles de recherche différents (triangulation), nous a permis de distinguer et d'explorer (von Krogh et al., 2012) le *modus operandi* en considération, et finir par la proposition des designs de recherche et de déploiement de méthodes spécifiques privilégiant son exploitation.

La méthode de « participation observante », proposant notre implication propre aux enjeux du terrain, nous a permis d'avoir une observation « dans la genèse » des normes d'action de ce *modus operandi*, pendant que la nature éphémère des configurations étudiées nous a rendu « inconnu parmi des inconnus », diminuant ainsi la distance entre chercheur et acteurs du terrain.

La méthode d'« analyse de monuments d'interaction » a exploité le fait que l'interaction Web laisse des traces (Georges, 2009). Contrairement à l'étude d'archives, où le documentaliste intervient dans la structure des données, ma méthode implique que je conçois les catégories d'information structurantes, afin de transformer le « matériel brut » en des données de recherche. Néanmoins, cette tâche était compatible avec la perspective exploratoire de ma recherche, où les catégories sont censées émerger comme résultat de recherche.

Enfin, une « problématisation limitée » était également exigée afin d'explorer si mode opératoire identifié constitue une spécificité des affaires du Web ou non. Ainsi, l'usage des éléments de ce *modus* en tant que problème de référence pour l'étude de l'histoire d'autres cadres industriels, a montré que des modes similaires ont également existé ailleurs. Cet exercice a également montré qu'une distinction d'un phénomène étudié en rapport avec la littérature (von Krogh et al., 2012) devrait être également complétée par une comparaison avec d'autres cadres, afin de conclure sur l'originalité du phénomène en question : il se peut également que l'objet d'étude illustre une dimension des affaires, sur laquelle la littérature, tout simplement, n'avait pas accordé assez d'importance auparavant.

Perspectives de recherche

Notre recherche a été centrée sur le champ du Web et, dans la deuxième partie, son champ a été élargi à des cadres industriels proches. Il serait donc intéressant de mettre à l'épreuve mes résultats de recherche en examinant leur validité dans des cadres industriels plus éloignés.

De plus, en ce qui concerne le rôle spécifique des UDs, des UDEs et des DEs, la recherche économique pourrait explorer le coût de développement à leur échelle. Contrairement à l'approche « coût d'entrée » qui, comme Christensen (1997) le souligne, exige un marché (d'entrée) pour qu'elle puisse être considérée, il arrive souvent que ce marché n'existe pas. Un calcul du coût de développement pour les UDEs devrait tenir compte des ressources nécessaires au niveau individuel pour l'acquisition des moyens de développement et l'expertise correspondante, pour une exploration créative ou pour une matérialisation précoce.

En ce qui concerne les moyens spécifiques identifiés pour le cas du Web, à savoir le logiciel *open source* et les *APIs,* en faisant usage des études précédentes sur le phénomène de l'open source, nous n'avons pas poursuivi une catégorisation plus approfondie. Les deux types de dispositifs peuvent faire objet d'une étude plus approfondie, dessinant leurs grandes catégories, éventuellement en rapport avec les figures d'acteur identifiés. Il est important de noter qu'une telle étude exigerait une compréhension approfondie des enjeux techniques sous-jacents, et, donc, pourrait être menée à bien d'une manière plus rigoureuse en tant qu'étude interdisciplinaire, en mettant ensemble à la fois du savoir en gestion et du savoir en ingénierie.

Des études quantitatives pourraient également exploiter les résultats de la présente étude, afin de tester et d'évaluer les concepts proposés. Ainsi, des enquêtes examinant l'usage des dispositifs de développement (comme les APIs), par des UDs, des UDEs et des DEs, selon les grandes lignes identifiées dans la première partie, pourraient éventuellement proposer une configuration d'un secteur émergeant, selon les phases distinguées dans la deuxième partie.

Des études quantitatives pourraient également explorer l'utilité de la notion d'« empathie » entre entreprise et UDEs, comme indicateur des problèmes partagés au sein d'un processus collectif, et approfondir sur la proposition d'une règle décrivant les conditions dans lesquelles elle doit être poursuivie, et les conditions dans lesquelles elle doit être évitée.

De plus, une mesure précise de la performance marchande des entreprises étudiées dans la deuxième partie, aurait pu affiner - ou mettre en question - la comparaison du modèle proposé par cette étude au modèle de Rogers (1962). Cette tâche exigerait également une approche interdisciplinaire, cette fois entre des chercheurs en Gestion et des chercheurs en Histoire, notamment en ce qui concerne l'étude d'archives d'entreprises disparues depuis plusieurs décennies.

Enfin, l'usage empirique des notions de l'action collective (Hatchuel, 2005c) et de la conversation, comme un point de rencontre entre des réseaux et des communautés, nous a permis d'étudier des dispositifs d'action et de conversation éphémères, et d'étudier leur design dans un contexte général d'un environnement d'affaires mobile. Cependant, l'étude du rapport entre action collective et conversation pourrait conduire à des questionnements épistémologiques importants, générant les conditions d'une problématisation plus approfondie.

Appendix A

Auction Street application Design Analysis

The elementary object of the *eBay* service is an *item*. It represents the item to be offered within the service, that is to be *"listed"*. All *eBay* applications serve to act on these items.

Each item listed on *eBay* is described by an *"ID number"* (automatically attributed by the service), a *"title"*, one or two *"categories"* characterizing it, a *"description"*, a *"photo"* of the item and by the mode in which it is offered: either by auction,or by direct sale (*"buy now"* option). In addition, sellers can use the following additional attributes for their item on sale: *"subtitle"*, *"3rd category"*, *"reserve"* item option. These latter attributes cost sellers extra fees. Table A.1 illustreates these *Design Parameters (DPs-1)* (Suh, 1990).

Auction Street application obtains these information on items by an *eBay API*, which is used to extract a list of an item's transaction history. Subsequently, the application creates a file for every item, where additional information about the item are added by the user. The user is called to complete the following fields for each item:

- *Acquisition information.*
- *Customer information.*
- *Shipping information.*
- *Quantity of items in stock, not yet listed to eBay.*

The *Acquisition information field* includes information related to the way the seller has acquired the item in question, before using the service. By completing this such information to a form, the seller adds the following attributes to the item:

- *Name of supplier.*
- *Date of acquisition.*
- *Type of acquisition.*
- *Cost.*

The *Customer information* field includes the *"name"*, the *"address"* and the *"email"* of the customer. The user is called to manually copy and paste these informations from the confirmation email he receives once his item is sold. The *Shipping information* field

Design Parameters of a listed item in eBay (DPs-1)															
Item ID	Title	Sub-title*	Categories			Item Descrip-tion	Photo		Quan-tity	Price			Auction dura-tion	Sale	
			1st	2nd	3rd*		1st	2nd*		Starting Price	Buy it now*	Reserve Price*		Date	Price

* Extra attributes charged extra fees by eBay.

Table A.1: DPs corresponding to the attributes of an item listed on *eBay*.

includes the *"weight"* of the item and its respective *"handling fee"*. In addition, *Auction Street* application provides the possibility to upload extra item *photos* for free, by using a third-party server owned by Brown's company.

These fields extend the object in question (the item) as well as the services that a seller can use. These new services include the ability to add extra photos of the item without being charged extra fees, to manage contacts (customers and suppliers) information, to calculate profit and loss for the total number of transaction for all items and to print consignment labels for items to be sent to their buyers.

Table A.2 illustrates the extension of the final service the seller gets while using Auction Street software, in terms of *Design Parameters* as well as of *Functional Requirements* (Suh, 1990; Kim et al., 1991).

However, application use is not fully automated. As a consequence end users have to use three different applications in order to use Auction Street:

- *Auction Street* itself.
- Their email client application, from where they had to copy their contact informations and paste it in the respective field in *Auction Street* application.
- eBay interface for listing items (either the standard one, either the automated tool granted by *eBay*, the *TurboLister*).

Thus, the user experience is dispersed to different applications, as the application user ows to master different interfaces to eventually use it.

		DPs-1*	*Additional Design Parameters*														
			Extra Photos	**Customer Info**			**Shipping Info**		**Acquisition Info**				**Transactions History**				
				Name	**Address**	**Email**	**Weight**	**Fee**	**Supplier Name**	**Date**	**Type of Acquisition****	**Cost**	**Account *****	**Date**	**Descri-ption**	**Type ******	**Amount**
	List item	X															
Additional Functional Require-ments	**Add photos**	**X**	**X**														
	Manage contacts	**X**		**X**	**X**	**X**			**X**								
	Calculate Profit/Loss	**X**						**X**		**X**	**X**	**X**	**X**	**X**		**X**	**X**
	Print Labels	**X**		**X**	**X**												
	Review Transactions	**X**		**X**									**X**	**X**	**X**	**X**	**X**

* DPs-1 already presented in Table

** Three types of acquisition are previewed: Purchase, % of Gross, % of Net.

*** The application gets information from multiple *eBay* and *PayPal* accounts belonging to the same user.

**** The following types of transaction are previewed: sold, committed, shipped, adjusted, and received.

Table A.2: Suh Matrix for Auction Street. Extension of the service provided.

Appendix B

Hackathon projects

Complete list of "pitches" at Google Campout.

Table B.1: List of projects "pitched" during the Campout at Google locals.

Timestamp	Status	Team #	Who Pitched	Demo URL	Team Email	Team members	Team Name	Project Description	Developers needed (# and skills)	Other special notes to developers	Your team size	One sentence introducion about yourself
8/13/2010 15:56:15		1	Bosco So		reg. gtug @boscoso .com		Sugar Race	A Social Network game that helps children maintain their Type I Diabetes protocol. Racing against their friends (and other anonymous users around the world) by adhering to blood sugar monitoring schedule, and following dietary & exercise regime, this game will help incentivize children to take better care of themselves.	"1 UI Designer 1 Front End coder (HTML, CSS, JavaScript - particularly JQTouch, Sencha, etc) 1 Rails (or GWT on AppEngine) back-end engineer + me (bit of front end & bit of backend, but no design skillz)"		4	SF Ruby Meetup organizer
8/13/2010 16:11:04	Active	2	alvin wang		alvin @cloud-wizard .com	Alvin Wang, Laura Klemme, Kenneth Ng, Annie Zhao, Paris Chrysos, Suyash Joshi	HTML5 Store	"Iphone and Android store cost 30%. They need to pay AT&T, Verizon, etc. We don't. The prizes for this campout are small. Paypal is offering $10K so if we use adaptive payments, we can hit 2 contests for 1 app. "	"Python or Java for appengine UI Javascript. What payment models will we support¿'		4	appengine developer

. . . continued in the next page . . .

. . . continues from previous page . . .

Time-stamp	Status	Team #	Who Pitched	Demo URL	Team Email	Team members	Team Name	Project Description	Developers needed	Notes	Team size	About you
8/13/2010 16:14:05		3	Luca Candela	togethr .us	luca. candela @gmail. com	Luca Candela, Chris Cinelli, Patrik Chopra, Jonhnson Nakano, James Williams, Patrick Laban, Jon McIntosh	Togethr	“Did you ever want to do something out of the blue, but didn't know anybody to do it with? With instant posse you can 1. create an event 2. tag it, and decide who can see it (your friends or the world) 3. define the radio of where to look for people (in miles) 4. define a minimum number of participantsOnce there's enough people willing to do what you have in mind, the system shoots a message to all willing participants with the contact info (user can define what to share) or the meeting place and provides directions based on the current location of the participant.Simple, easy on-the-fly event organizer.“	“1 Android developer 1 PHP developer 1 Front end developer“	Experience with Latitude would be a great advantage for this project.	4	Product Design Extraordinaire
8/13/2010 16:22:13		4	Alex Ryan		alexander .j.ryan @gmail .com	Chris Cinelli	QuoteShare	“QuoteShare is an app that I am building which enables users to collect and organize their favorite quotes and share them with their friends via social networking apps like facebook. The idea to have a place that you can go to to find the perfect quote when you need it and to find it quickly. The collection of quotes and the tagging and rating is all crowd sourced. It is implemented using MySql, Java/Tomcat and JQuery. I'd like to try converting it to HTML, maybe adding mobile, etc.Voluntaryist Ventures is the un-company. It is an organization without hierarchical power structures where developers can self-organize their own projects and distribute rewards in a meritocratic fashion. The mission of Voluntaryist Ventures is to relegate the bureaucracy-laden corporation to the dustbin of history.“	“MySql Java Tomcat JavaScript JQuery HTML5 Eclipse mobile“	Open to new ideas for functional enhancement.	4	Founder of Voluntaryist Ventures
8/13/2010 16:23:57	Active	5	Jason Laster		Jason. Laster.11 @gmail .com	Patrik Chopra	Copy and Paste	Make copy and paste native to the web. Single feature app for making selecting similar items on a page as simple as copy and paste.	“2 Javascript guys 1 UI guy“	Think refactored selector-gadget for the common man. Should be fun	3	engineer at Ning

. . . continued in the next page . . .

. . . continues from previous page . . .

Time-stamp	Status	Team #	Who Pitched	Demo URL	Team Email	Team members	Team Name	Project Descri-ption	Deve-lopers needed	Notes	Team size	About you
8/13/2010 16:28:49	Active	6	Jonathan Voelm	terratraveler .mobi	jonvoelm @gmail .com	Jonhnson Nakano	Terra Traveler	"Design an App that is like a stripped down mobile App Facebook (Traveler Book) that is for travelers. Users make ""Traveler"" user account profiles with whatever information that they want to put out there to the Traveler community that uses the App. The App couples location based Google maps/navigation and latitude technologies with chat ""waves"" that are location based and also private messaging when users make a handshake.The App will help users meet other Travelers when they are out and about. Users can share tips, meet up for drinks, set up ad-hock events and group-ups.Join the team to learn about Monetization. "	"2 Java devs Sr.-mid level (android SDK familiarity is ""++"") 1 UI dev (android) 1 Web dev (HTML/SQL/PHP etc.)Me - Designer/Researcher "	"Since we will be developing an android app that pulls data from a web-server developers with knowledge about getting android apps to pull data from webserver SQL databases is a big plus. Also knowledge about developing social web features and wave/private messaging is sought. Experiance with GoogleMaps App API is a ""++"" with Latitude ""+++"". Experience with Facebook App API is a ""+""."	5	Cal Grad in EECS, Space Application EEE Parts Engineer, mobile software architect
8/13/2010 16:53:26		7	Anil Pattni		info. wirelessapps @gmail .com	James Williams	LifeAlert	"This is an alert notification system which allows a phones GPS coordinates to be attached to a custom alert and sent to a select list of contacts at any given time and it's triggered by a users input. Will be developed as an app for the android platform. "	"1 JAVA/CSS/HTML for android app 1 PHP mid level for website backend"	"I'm not sure if the coding languages mentioned will be sufficient to develop this app, will need to discuss websites functionalities to figure out which languages and level of expertise would be appropriate.Twitter @WirelessAppsTK"	3	Founder & CEO WirelessApps Non-profit smartphone application development

. . . continued in the next page . . .

. . . continues from previous page . . .

Time-stamp	Status	Team #	Who Pitched	Demo URL	Team Email	Team members	Team Name	Project Descri-ption	Deve-lopers needed	Notes	Team size	About you
8/13/2010 16:32:50	Active	8	Bess Ho		bess.ho @gmail .com	Patrick Laban	Rescue Me	"Find out the nearby hospitals around me using HTML APIs, HTML5 tags and CSS3: Web SQL Database, Micro format, Microdata, Geo location, Semantic tags, Web Forms. This will be deployed to Web Store and Mobile. I will continue to build and take this hack project to enter Health 20 Developer Challenge."	1 serious hacker with experience in HTML5 & CSS3. Google Map and mobile are bonus.	I'd like to find a hacker who would code 2 hours without spending 1 hour chatting. Someone who would embrace clean code and good engineering practice. Someone is willing to write pure HTML5 using all web standards.	2	hacker and cute pie
8/13/2010 16:34:59	Merged with 33	9	Patrick Laban		laban. patrick @gmail .com	Jon McIntosh	Better ways to trade items using HTML5	"Boiled down, its essentially a new Craigslist. I plan to create a site for the trading of physical items ranging from computers and apartments to arranging a ride-share.So how will it be different? Features such as auto-event recognition based on location or using the canvas to manipulate a 3d render of objects for sell will set it apart.Hopefully you can see what I'm aiming for. Talk to me if you want to know more about my idea."	1 - 2 general web developers, 1 UI designer		3	I'm a software engineer 1 year out of college and hopping on to HTML5.
8/13/2010 16:46:07		10	Jerome Calvo		jerome @dreamface .org		HTML5 widgets for DreamFace	"Use as much HTML5 as possible in DreamFace (dreamface.org) widget based plaftorm. Build an app tbd with team, thinking of using BART API (XML live feed) and display info using as much HTML5 new features as possible. JavaScript/JQuery/JSON knowledge required as well (or not...). Presentation optimized for desktop and mobile browser if time permit. "	JavaScript, JQuery (JQTouch if time permit), XML/JSON.		3	Frenchman, serial entrepreneur, technologist, percussionist, DreamFace evangelist
8/13/2010 16:46:14	Active	11	Carlos Cardona		cgcardona @gmail .com	Carlos Cardon, Dylan Clendinin, Mary Zhu	Real Time Analytics	We intend to write an analytics tool using websockets and node.js.	Calling all Javascript Ninjas!	We are using node.js for the backend if anyone is interested in javascript on the server.	3	Front End Web Developer from Santa Cruz, CA.

. . . continued in the next page . . .

. . . continues from previous page . . .

Time-stamp	Status	Team #	Who Pitched	Demo URL	Team Email	Team members	Team Name	Project Descri-ption	Deve-lopers needed	Notes	Team size	About you
8/13/2010 16:48:35		12	S. Sri-ram		ssriram @gmail .com		Shout shopping browser extension	"Build a browser extension that filters and logs the 'shopping related' product pages and searches that a user visits. This activity can than be posted to Facebook/twitter etc. so friends can provide input, vendors can offer deals and experts can offer advice. "	1 emarketr/busdev/-social media guru w/experience in the shopping space	The web service has already been built. I'm looking to build out the browser extension over this weekend. I could use the insight of folks with emarketing/bus-dev experience in the shopping space.	2	Building a shout shopping service, Founder 565labs.com
8/13/2010 16:55:12		13	Alexander Smirnov		alsmirnnov @gmail .com		Photo Story	"Then you show your photos to friends or relatives, there is always a small story behind the shot (who is this guy, how we got there, what's happening and what's going on). There are great project to place photos in internet, or write you story on blog, but no one let's to combine both in convinient way. The project intended to create such ""Illustrated stories"" easy, for any purpose - vacation trip description, corporate event report, or just to share fun from your party with friends who were missed there."	"1 Sr Web designer 1-2 Java mid to sr level. 1 content creator "	"I've prepared project template as Mercurial repository, created Google Appengine account and deployed stub there, so everything is ready to start. It uses Java Server faces for UI, dependency injection (Jboss Weld) and JPA for server-side, and Maven as build system. Although I use Java Server Faces, I'm free to choose any other technology; it would be even more fun to learn some new tricks during campout. "	3	Jboss Richfaces and Jboss Portletbridge projects architect, JSR-314 (Java Server Fices) expert group member

. . . continued in the next page . . .

. . . continues from previous page . . .

Time-stamp	Status	Team #	Who Pitched	Demo URL	Team Email	Team members	Team Name	Project Descri-ption	Deve-lopers needed	Notes	Team size	About you
8/13/2010 17:17:20	Active	14	Peter Harrington	teamdreadtraders .appspot .com	teamdreamtraders @gmail .com	Peter Harrington, Jocelyn Harrington, Venugopal Jidigam, Hoa Long, P J Gupta	Day Trader Tool	"This project will display historical data for financial instruments. These could be stocks, ETFs, etc. The data will update as time progresses. Ideally this will be an app for sale in the Chrome App Store. Depending on the number of people interested and the effort put into this project a number of features can be added. Some potential features are: -Pop up alarms to notify the trader when a security passes a certain value -A list of securities, then clicking on each one will display a chart -Creative ways to display the price/volume information -Synthesized securities: moving averages, combinations of securities -A plug-in framework that allows users to create different ways to view data, and share these. etc. "	Anyone is welcome		2	I am a bad-ass engineer
8/13/2010 17:23:29		15	Harpreet		hsbsitez @gmail .com		Slidio	An image slideshow with audio in the background. Uses CSS3 transitions to provide interactive experience. The images are loaded by dragging and dropping and the web application randomly assigns transition that will be used between the switching of an image and an animation to run while the image is shown.	"1 JavaScript 1 HTML5/CSS3"		3	A Web developer.
8/13/2010 17:28:33	Active	16	Jennifer Peck	SweetGeo.com	jenniferleapeck @gmail .com	Rojat Mahajan, Rohit Surve, Andreas, Nik Hodgkinson, David Phillips, David Phillips, Damien Patton, Jennifer, Peter, Toby Morning	SweetGeo	"Takes the concept of LinkedIn, Facebook, Foursquare, Meetup, Craigslist and mashes them together.Check in to location, post profile with interests, needs etc. As people check in, they can view and post profiles, post ads etc. The person you you looking for or person who can help you might be there person who just arrived. "	"DB 1 - Ruby or Python 1 Front end HTML5/CSS3 "		4	
8/13/2010 17:59:55	Active	17	Alex Hwang		Aykhwang @gmail .com	Panini Raman, Alex Hwang, TBD	Soundweaver	HTML- based staff-oriented composition tool.	2 HTML, JS and CSS savvy developers	Experience in music helpful.	4	Student who loves HTML 5!
8/13/2010 18:06:44	Active	18	Siamak Ashrafi	artfx.appspot.com	biologica @gmail .com		dynamic memory Game	"This is a cross between the Memory Card Game and TetrisCards drop face down from the top of the screen and slide to the bottom, stacking edge on edge. The user clicks on two cards temporarily revealing their faces and if identical, the pair is removed. The user must remember the card locations and find matches before the cards pile to the top of the screen. The game ends when one stack of cards reaches the top. "	"1 Javascript Sr. 1 HTML 5 Canvas - SVG Sr. 1 CSS3 Sr."	I need a lot of help.	3	computation biologist

. . . continued in the next page . . .

. . . continues from previous page . . .

Time-stamp	Status	Team #	Who Pitched	Demo URL	Team Email	Team members	Team Name	Project Descri-ption	Deve-lopers needed	Notes	Team size	About you
8/13/2010 18:06:44	Merged with 33	19	Siamak Ashrafi		biologica @gmail .com		dynamic memory Game	"This is a cross between the Memory Card Game and TetrisCards drop face down from the top of the screen and slide to the bottom, stacking edge on edge. The user clicks on two cards temporarily revealing their faces and if identical, the pair is removed. The user must remember the card locations and find matches before the cards pile to the top of the screen. The game ends when one stack of cards reaches the top. "	"1 Javascript Sr. 1 HTML 5 Canvas - SVG Sr. 1 CSS3 Sr."	I need a lot of help.	3	computation biologist
8/13/2010 18:38:21		20	Kyle B. Koski		kyle2501 @gmail .com	Kyle B. Koski, Joe Saunders	Allele Interaction	Image File System within a Social Network. Packaged as Web App and/or a Chrome Extension and/or a Bookmarks Bar Button. Integrating the Google ContactsManager API, the Tumblr API, the Vark API, maybe other APIs too like Wave and Voice and Earth. Product design goals are in User Experience (Typography & Colour Theory) and to outperform Facebook. Images uploaded to Tumblr are tagged to Contacts/Users via the Google ContactsManager then becomes socialised with the Vark API.	"1.) APIs 2.) AJAX, Javascript, XML, PHP, CSS, HTML 3.) Innovative Thinkers in Theory of User Interaction"	Macintosh users are a huge plus. o.0	5	Independent Interaction Designer from Sun Valley, ID with a family history at Apple Computer.
8/13/2010 18:07:37		21	Henry Chan		hpychan @gmail .com		meal claim	" Create an Android application that help Air Canada flight attendent to have an accuracy calculation for their meal pay. "	2 android java, 1 ui guru	The java back-end has partially implemented in Google app engine.	3	Software architect in Logitech
8/13/2010 18:31:44	Active	22	Adam Glickman		aglick35 @gmail .com	Adam Glickman	Kitlist to Cloud	"Put the LARGE (68k viewers) Kitlist (http://www.kitlist.org) of Job postings into a cloud based list. This can sort jobs by location, general type of job, and by offering company. There can be fb, twitter posts from this list too. The list is currently on Yahoo Groups. http://www.kitlist.org/ and http://groups.yahoo.com/group/KITlist-Tech/ This cloud based list can have anonymous mailings to other lists and RSS posts. Currently the list is on MySQL. "		Currently it is on MySQL.	2	InfoMonger of Technology

. . . continued in the next page . . .

. . . continues from previous page . . .

Time-stamp	Status	Team #	Who Pitched	Demo URL	Team Email	Team members	Team Name	Project Descri-ption	Deve-lopers needed	Notes	Team size	About you
8/13/2010 18:41:21	Active	23	Shahin Saneine-jad		ssaneine @gmail .com	Vincent Malm-rose, Earl Malm-rose, Jake Malm-rose, Kory Mam-rose, Luong Tam, Larry Tu	TetriNET	"Do you remember TetriNET, competitive Tetris from the age of Webvan and Monica Lewinsky? Nothing re-mains but a decade-old Win32 client and a server that's fallen and might never get up.Let's make TetriNET cool again with HTML5. We'll draw it with Canvas and sync a game across clients with XMPP. It'll run on my Mac, your Ubuntu, Ballmer's laptop, and might even give the iPad a proper use case (finally!).Bonus points for cool power-up ideas. Block Bomb, Gravity, and Blockquake need a refresh."	"1 Javascript dev 1 Jab-ber/XMPP or Websockets dev 1 CSS/UI designer"	Should our clients speak XMPP over BOSH or a custom solution over Web-sockets?	4	I'm a Data Analyst by day, a hippie de-veloper by daydream.
8/13/2010 18:42:15		24	Allen Barteld	www. DoUSee. me	abarteld @gmail .com	Allen, Joseph Chow, Nick Mahilani, Vivek Ganes, Shon Saoji	DoUSeeMe	Temporary Geo location sharing web app		Interested in any location based apps.	2	Non-technical serial en-trepreneur recently relocated to Palo Alto
8/13/2010 18:50:09		25	Gopal Gopalakr-ishna		gopal .gopalakr-ishna @gmail .com		Adhoc so-cial	Build a app where people can join based on a topic that they are interested in and have a virtual social meeting	"HTML 5 guru -2 J2EE "	Developers who has experi-enced in building social net-working is given preference	4	Consultant
8/13/2010 19:23:14		26	Gabriel J Her-nan-dez		webspinner .gabriel @gmail .com		Party Page Partners	A real time event app that will hang on top of a client website allowing the "host" to present products or new information to "in-vited guests" in a real time social atmosphere. Lay-ers of real time interac-tion are placed on top of the branded "party page" for multiuser chat and host "announcements". Join friends in through existing social networks.	1 Python Sr., 1 UI Wiz, 1 JavaScript Mid, All should like html5.	The idea is to create an easy to plug social event to a branded website using mostly web sock-ets and dom manipu-lations, some session storage will be used also.	2	Gabriel is a Web Ap-plication Developer working mostly in Google Ap-pEngine Python
8/13/2010 18:59:34		27	Peter Lau-ren-tis		Pmjlus @gmail .com		Trip plan-ner	An HTML5/android app for mobile devices that al-lows the user to program an itinerary, save it, share it by link, email, social media networks, calculate costs, etc	"1 designer 2 android developers 1 chrome devel-oper 1 other programmers "	Someone experi-enced with geoloca-tion	5	Entrepreneur
8/13/2010 19:04:06		28	David Phillips		davjphillips @gmail .com		PeerSonar	PeerSonar allows friends to find their friends any-where. Through geoloca-tion and googlemaps tech-nology, two users can pin-point their locations and guide their friends to them in real time. With Peer-Sonar, you will never get lost again.	DB, ruby on rails or python, ex-perience with API, front end developer (HTML5, CSS3, graphic design)		4	

. . . continued in the next page . . .

. . . continues from previous page . . .

Timestamp	Status	Team #	Who Pitched	Demo URL	Team Email	Team members	Team Name	Project Description	Developers needed	Notes	Team size	About you
8/13/2010 19:12:45		29	Arup Kumar Kabi		arupkumarkabi @gmail .com		RashoMonty	"Extract Data from Review websites and display Content in the form of Websites in tabbed presentation Use Metadata, Map info and Tagging inputs to give user a comparative analysis."	"Java(Regular exp) [Server side Data mining] Android UI Ajax/Chrome API's Chrome/HTML5 exp"	Read the Movie summary of Rashomon, if you have not seen the movie..	3	Like to Program and Campout
8/13/2010 19:22:24	Active	30	Chandramouly Rathinasabathy	Chrome Extension	chandru. mouly @gmail .com	Chandramouly Rathinasabathy, Manoj Govidassamy	html5 Video Klippr	"html5 video klippr - chrome extension - content sharing (facebook, youtube, twitter)"	2 javascript, youtube apis		2	
8/13/2010 19:40:36			Ryan Delucchi		bender @onsrc .com		ActiveDrive	"Active Drive*** Be notified of traffic backups that lay ahead1.) Recieve live information about alternate routes2:) Users can serve as a ""Traffic Enhancer"" for points and improve traffic by following driving instructions that will actually improve the traffic."	"1 Front-End Web Developer (JavaScript) 1 Back-End App Engine Developer (Java/Python - Python Preferred)"	"Visit the following link for more information and to make more suggestions:http://www.ietherpad.com/z0ldHHUKKw"	4	Java and Python Developer at Taleo
8/13/2010 19:45:45		31	Guy Malachi		guymal @gmail .com		Will it work	A system for getting feedback about whether an idea will succeed. You pitch your idea and other users give you feedback-/advice and rating about your idea.	1 client side, 1 php backend developer		4	
8/13/2010 19:57:46			Newton Chan		the1geek @gmail .com	Merlin Finger	Something magical	Graphics Artist			2	The #1 Geek
8/13/2010 19:59:27			Chris Lesner		chris .lesner @gmail .com		Extend Etherpad Collaborative Editor		" == What technologies are involved? == * Javascript ** server side with Rhino https://www.mozilla.org/rhino/ ** client side DOM/HTML manipulation * scala/-java/mysql/linux/etc * HTML/CSS GUI person (AJAX) * mysql "		3	Open Technology Software Developer
8/13/2010 21:28:32	Active	33	Joe Saunders		joe.a .saunders @gmail .com	Joe Saunders, Alan Carwile, Kyle Koski, Jerome Calvo	DataViz	"Dynamic data visualization application that allows some level of configurable widgets. For the system that is feeding the data there are plugins that are downloadable. Initial implementation centeed in the networkik equipment space.Blue sky: command prompt shell, target system upgrade, etc . . . "	1or 2 javascript/css/html5 expert		4	
8/13/2010 23:06:47	Active	34	Daniel Buchner		teamarcjs @gmail .com	Aaron Schaar, Brad Boswell, Dan Buchner, Jethro Larson	arc.js	"We are making a JS access layer for the post message protocol what enables all sorts of x-domain awesomeness. The JS access layer's focus is on DOM manipulation, open method discovery, and solving common x-domain use-cases."			4	I PM stuff and develop thangs

. . . continued in the next page . . .

. . . continues from previous page . . .

Time-stamp	Status	Team #	Who Pitched	Demo URL	Team Email	Team members	Team Name	Project Descri-ption	Deve-lopers needed	Notes	Team size	About you
	active	35	miro be-ran	http://green-plate.mobi	miro.beran@gmail.com	miro	Foreinger	Foreinger				
	active	36	Oksana Yarem-chuk		oksana.yarem-chuk@gmail.com	Oksana Yarem-chuk	Pull Pre-diction	Social game to predict the markets				

End of table

Appendix C

Forum conversations

C.0.1 New bug: Request for more information

Conversation quotes	Other Web page information
	Issue name: *Issue 4072: Bug: Dragend Event also trigger Click Event in IE*
	Issue URL: http://code.google.com/p/gmaps-api-issues/issues/detail?id=4072
	Reported by user960...@gmail.com, Apr 11, 2012
	8 people starred this issue and may be notified of changes.
	Reported by user960...@gmail.com, Apr 11, 2012
I have found that dragend event in Google Map also trigger the click event in IE. It is OK in Firefox and Chrome. Open IE/FF/Chrome console and see the result of this fiddle. Any workaround will be appreciated. http://jsfiddle.net/ABqMH/8/	
	Comment 1 by st...@stevehorn.cc, Jul 2, 2012 Duplicated and confirmed this bug.
	Comment 2 by project member lu...@google.com, Jul 4, 2012
Which version(s) of IE does this affect?	
	Status: NeedsMoreInfo
	Comment 3 by ScottFer...@gmail.com, Jul 4, 2012
I see it in version 9.0.8112.16421	
	Comment 4 by lourol...@gmail.com, Aug 31, 2012
To make this worse : The click event is fired only when dragging a certain distance. If a marker is only dragged for a few pixels, no additional click event is fired. Hard or impossible to find a workaround for this. Issue appears in IE 8 and 9.	
	Status: NeedsMoreInfo
	Owner: —-
	ApiType-Javascript3

Conversation quotes	Other Web page information
	Type-Defect

C.0.2 New feature: request for more information

Conversation quotes	Other Web page information
	URL: http://code.google.com/p/gmaps-api-issues/issues/detail?id=2122
	Issue 2122: getBoundsZoomLevel
	2 people starred this issue and may be notified of changes.
	Reported by erich.sc...@gmail.com, Jan 26, 2010
Is it possible to get this functionality back or emulate it in one way or another. I currently use map.fitBounds(), but this does often zoom too much. Therefore, I'd like to limit the maximum zoom increase to e.g. 4 zoom steps each click. This is easy when I can get the 'optimum' zoom myself and then use panTo(), setZoom(). It would be cool to be able to pass the destination zoom level to panTo, and eventually have a smooth zoom in process. Thank you.	
	Comment 1 by erich.sc...@gmail.com, Jan 26, 2010
FWIW, my toy project can be found here: http://swing.vitavonni.de/dynamic.html It loads placemarks from a KML file, aggregates ("clusters", except that it's so primitive threshold-merging that I wouldn't call it clustering) them dependent on the current zoom level. The data is pulled from a full-text-indexed xapian database, with temporal and geographical constraints. The database is filled from 25+ Google Calendars provided by various swing dancing communitys around the world.	
The starting page, http://swing.vitavonni.de/ is basically the same, but with Maps APIv2, and the data is included during page load instead of coming from KML. It does have more features right now: the search box is working and so is the list view. So if you type "where:USA" into the search box, you'll get results for the US only (not too many yet) (If you do a full text search for "USA" it might find results in Germany, and then zoom the map there instead of the US, this is intentional that any hit in Germany takes precedence. You can see them on the right though. No button to auto-center the map on them though)	

Conversation quotes	Other Web page information
In the "dynamic" version, clicking on the Munich hat will zoom all the way in, basically taking you to a zoom level where it's not easy to tell where in Munich you actually are. In the "old" version, I limit the zoom increase, trying to keep the zoom in process more helpful for the user. This is what I'd like to do with the new API, too.	
	Comment 2 by daniel...@google.com, Feb 4, 2010
What are you executing against the map clicking a Munich hat? Are you calling fitBounds() which kicks off the zoom? What if you simply increase the size of your bounds to minimize the zoom?	
	Status: NeedsMoreInfo Comment 3 by erich.sc...@gmail.com, Feb 5, 2010
I have this group of markers, aggregated to a "cluster marker" (which is the hat). WHen the user clicks the marker, I essentially want to zoom the map to show all the markers in this group. So I setup a new LatLngBounds, and add all the marker positions to it. Then I call fitBounds() on the Map with that bounds. That works, but it can sometimes take you from a global level all the way to a street level, without much indication of where you are. This is why I'd like to limit the zoom increase. If you want to try it yourself, you can zoom the map all the way out. There should be a single marker, showing some number (of clustered markers) over south-east china. Clicking on it takes you all the way into street level of Hong Kong, without much indication to the user where he went. Since the markers are clustered dynamically, I do not have the information that the marker is in China, Hong Kong, Hong Kong Island; so I cannot easily take the user these semantic navigation steps down. I'm not convinced that such a semantic clustering is useful on every zoom level. Grouping markers into supermarkers when their icons would overlap is so far working quite well. Obviously, increasing the bounds is an option, but by how much should I increase the bounds to avoid an overly large zoom - in particular without an option to query the zoom that would occur with a particular bounds. What I essentially had in mind was along the pseudo code: setZoom(min(getZoomForBounds(), getZoom() + 4)) This was possible with the v2 API.	
	Comment 4 by erich.sc...@gmail.com, Feb 8, 2010

FWIW, this is the code I'm now using (via closure, compiler, so verbose unoptimized code):

```
/**
 * @param {google.maps.Map} map Map
 * @param {google.maps.LatLngBounds}
     bounds Destination
 */
var smoothZoomMap = function(map,
    bounds) {
  var is = map.getBounds();
  var issize = is.toSpan();
  var wantsize = bounds.toSpan();
  var delta = Math.min( wantsize.lat()
      / issize.lat(), wantsize.lng() /
issize.lng());
  var maxstep=1./4;
  if (delta < maxstep) {
    var c = bounds.getCenter();
    var nlat = Math.max(maxstep *
        issize.lat(), wantsize.lat()) /
        2;
    var nlng = Math.max(maxstep *
        issize.lng(), wantsize.lng()) /
        2;
    bounds = new
        google.maps.LatLngBounds(new
        google.maps.LatLng(c.lat()-nlat,
c.lng()-nlng), new
    google.maps.LatLng(c.lat()+nlat,
    c.lng()+nlng));
  }
  map.fitBounds(bounds);
}
```

Settings maxstep=1./4 (= 4 fold zoom in = 2 levels?) will result in animated zoom steps; however that takes too many clicks in my situation to be useful. 1./16 is okay to prevent the user from getting lost by too large zoom increases, while also zooming in with reasonably few clicks. Otherwise, the zoom steps would often be in the range of 0.002 to 0.02. In my use cases, 16x zoom increase seems to be useful, 32x is too much. But that may vary, some people will prefer 4x to get the zoom effects.

Comment 5 by daniel...@google.com, May 27, 2010

How would your workaround code look if there was a getBoundsAtZoomLevel() function?

Comment 6 by thor.mit...@gmail.com, Jun 29, 2010

Conversation quotes	Other Web page information
	(No comment was entered for this change.)
	Owner: thor.mitchell
	Comment 7 by thor.mit...@gmail.com, Jul 1, 2010
	(No comment was entered for this change.)
	Owner: t...@google.com
	Comment 8 by erich.sc...@gmail.com, Jan 12, 2011

@daniel: Instead of computing the bounds-that-might-fit-my-intended-zoom myself in the "if" statement and map.fitBounds(), I'd like to do:

```
---
var curzoom = map.getZoom();
var newzoom =
   map.getZoomLevelForBounds(bounds);
map.panTo(center);
map.setZoom(min(curzoom + 4, newzoom));
---
```

Not sure what the parameters for a "getBoundsAtZoomLevel" would be. What I'm right now doing is to artificially increase my bounds to avoid overly big zoom changes.

Comment 9 by alfonsfr...@gmail.com, Jan 25, 2011

I also need a getBoundsZoomLevel as i need to know the optimal zoom to display a collection of polygons whitout changing the actual map zoom

Comment 10 by project member lu...@google.com, Feb 7, 2011

have you tried setting minZoom on the map?

Comment 11 by erich.sc...@gmail.com, Mar 22, 2011

minZoom as I understand *blocks* certain zoom levels.
I do not want to *prevent* the user from zooming to this level. If he deliberate clicks on a certain level on the zoom scale, so be it.
I just want to prevent large unexpected zoom changes triggered by clicking map markers, since they might confuse the user. By doing multiple iterations, he could still zoom all the way in, but this way he'll still actually see where he is zooming into.
It's really all about keeping the zoom *smooth* by preventing *large* changes in the zoom level.
Thank you.

Comment 12 by pholm...@gmail.com, Mar 24, 2011

Conversation quotes	Other Web page information
This would be useful, another way to possibly solve this problem is have fitBounds() return the zoom level its going to use. This way some action can be taken as a result or not. In my application I want take action if the zoom level changed but not if it didn't change.	
	Comment 13 by project member c...@google.com, Mar 18, 2012
> In my application I want take action if the zoom level changed but not if it didn't change. You can use addListenerOnce to do this.	
	Comment 14 by maft.mor...@gmail.com, Apr 18, 2012
Just seen this thread, but my suggestion was for a separate 'zoomTo()' function similar to panTo(): http://code.google.com/p/gmaps-api-issues/issues/detail?id=4084	
	Comment 15 by project member t...@google.com, Jul 30, 2012 (No comment was entered for this change.) Owner: k...@google.com

Bibliography

Abbate, J. (2000). *Inventing the Internet*. Inside technology. MIT Press, Cambridge, Mass, Etats-Unis, abbate2000 edition.

Abernathy, W. J. and Clark, K. B. (1985). Innovation: Mapping the winds of creative destruction. *Research Policy*, 14(1):3–22.

Abernathy, W. J. and Townsend, P. L. (1975). Technology, productivity and process change. *Technological Forecasting and Social Change*, 7(4):379–396.

Abrahamson, E. (2009). Necessary conditions for the study of fads and fashions in science. *Scandinavian Journal of Management*, 25(2):235–239.

Abrahamson, E. and Rosenkopf, L. (1997). Social Network Effects on the Extent of Innovation Diffusion: A Computer Simulation. *Organization Science*, 8(3):289–309.

Adam, F. and Murphy, C. (1995). Information flows amongst executives: their implications for systems development. *The Journal of Strategic Information Systems*, 4(4):341–355.

Adorisio, A. L. M. (2009). *Storytelling in organizations: from theory to empirical research*. Palgrave Macmillan, Basingstoke.

Aggeri, F. and Labatut, J. (2010). La gestion au prisme de ses instruments. Une analyse généalogique des approches théoriques fondées sur les instruments de gestion. *Finance Contrôle Stratégie*, 13(3):5–37.

Aguiton, C. and Cardon, D. (2007). The Strength of Weak Cooperation: An Attempt to Understand the Meaning of Web 2.0. *Communications & Strategies*, 65(1):51–65.

Aitken, H. G. J. (1985). *The Continuous Wave: Technology and American Radio, 1900-1932*. Books on Demand.

Akrich, M. (1993). Les objets techniques et leurs utilisateurs, de la conception à l'action. *Les objets dans l'action*, 4:35–57.

Akrich, M., Callon, M., Latour, B., and MONAGHAN, A. (2002). The key to success in innovation part I: the art of interessement. *International Journal of Innovation Management*, 6(02):187–206.

Alexander, M. (1997). Getting to grips with the virtual organization. *Long Range Planning*, 30(1):122–124.

Allan, R. A. (2001). *A History of the Personal Computer: The People and the Technology*. Allan Publishing, 1st edition.

Amin, A. and Roberts, J. (2008a). Knowing in action: Beyond communities of practice. *Research Policy*, 37(2):353–369.

Amin, A. and Roberts, J. (2008b). The resurgence of community in economic thought and practice. In Amin, A. and Roberts, J., editors, *Community, economic creativity, and organization*, chapter 1, pages 11–36. Oxford University Press, Oxford ; New York.

Anderson, P. and Tushman, M. L. (1990). Technological Discontinuities and Dominant Designs: A Cyclical Model of Technological Change. *Administrative Science Quarterly*, 35(4):604–633.

Andreani, J.-C. and Conchon, F. (2001). *Les études qualitatives en marketing*. ESCP-EAP, Paris.

Arena, R. and Conein, B. (2008). On virtual communities: individual motivations, reciprocity and we-rationality. *International Review of Economics*, 55(1):185–208.

Argyris, C. and Schon, D. A. (1978). *Organizational Learning: A Theory of Action Perspective*. Addison-Wesley.

Aspray, W. and Ceruzzi, P. E. (2008). *The Internet and American Business*. MIT Press, Cambridge, Mass.

Auray, N. (2004). La régulation de la connaissance: arbitrage sur la taille et gestion aux frontières dans la communauté Debian. *Revue d'économie politique*, pages 161–182.

Auray, N. (2007). Folksonomy: the New Way to Serendipity. *Communications and Strategies*, 65:67.

Auray, N., Hurault-Plantet, M., Poudat, C., and Jacquemin, B. (2009). La négociation des points de vue. *Réseaux*, 154(2):15.

Babbage, C. (1826). *A Comparative View of the Various Institutions for the Assurance of Lives*. J. Mawman.

Bagozzi, R. P. and Dholakia, U. M. (2006). Open source software user communities: A study of participation in Linux user groups. *Management Science*, 52(7):1099.

Balderas, M. (2011). *PayPal APIs: up and running*. O'Reilly, Beijing; Sebastopol, CA.

Baldwin, C., Hienerth, C., and von Hippel, E. (2006). How user innovations become commercial products: A theoretical investigation and case study. *Research Policy*, 35(9):1291–1313.

Baldwin, C. and von Hippel, E. (2011). Modeling a paradigm shift: From producer innovation to user and open collaborative innovation. *Organ. Sci. Organization Science*, 22(6):1399–1417.

Baldwin, C. Y. (2008). Where do transactions come from? Modularity, transactions, and the boundaries of firms. *Industrial and Corporate Change*, 17(1):155.

Baldwin, C. Y. and Clark, K. B. (2000). *Design Rules, Vol. 1: The Power of Modularity*. The MIT Press, Cambridge, MA, first edit edition.

Baldwin, C. Y. and Clark, K. B. (2006). The architecture of participation: Does code architecture mitigate free riding in the open source development model? *Management Science*, 52(7):1116.

Baldwin, C. Y. and Woodard, J. (2010). The architecture of platforms: a unified view. In Gawer, A., editor, *Platforms, Markets and Innovation*, pages 19–44. Edward Elgar Publishing, Cheltenham, UK ; Northampton, MA.

Bancel-Charensol, L. and Jougleux, M. (1997). Un modèle d'analyse des systèmes de production dans les services. *Revue française de gestion*, mars-avril(113):71–81.

Baptista, L. O., Durand-Barthez, P., and Lesguillons, H. (1991). *Les associations d'entreprises (joint ventures) dans le commerce international*. FEDUCI : L.G.D.J., Paris.

Barnouw, E. (1966). *A tower in Babel : a history of broadcasting in the United States*. Oxford University Press, New York.

Bartlett, C. A. and Ghoshal, S. (2002). The transnational and beyond: reflections and perspectives at the millennium. *Advances in international management*, 14:3–36.

Bastian, M., Heymann, S., and Jacomy, M. (2009). Gephi: an open source software for exploring and manipulating networks. In *ICWSM*, page 2.

Batagelj, V. and Mrvar, A. (1998). Pajek-program for large network analysis. *Connections*, 21(2):47–57.

Baumard, P. (1999). *Tacit knowledge in organizations*. Sage Publications, London; Thousand Oaks, Calif.

Bausch, P. (2003). *Amazon hacks*. O'Reilly, Sebastopol, CA; Farnham.

Bausch, P. (2006). *Yahoo! hacks*. O'Reilly, Sebastopol, Calif.

Bayart, D., Bonhomme, Y., and Midler, C. (2000). Management Tools for RetD Project Portfolios in Complex Organizations – the case of an international pharmaceutical firm. In Benghozi, P.-J., Charue-Duboc, F., and Midler, C., editors, *Innovation Based Competition et Design Systems Dynamic*, pages 135–158. L'Harmattan, Paris, France.

Benavent, C. (2011). Réseaux sociaux : une question d'organisation. *Prisma.com*.

Benavent, C. and de la Villarmois, O. (2006). CRM, apprentissage et contrôle organisationnel. *Journal of Information Systems and Technology Management*, 3(2):193–210.

Benkeltoum, N. (2008). *Les régimes de l'open source: Management, Innovation, Solidarité et Business Models*. PhD thesis, Mines-ParisTech.

Benkeltoum, N. (2011). *Gérer et comprendre l'open source : Une modélisation en termes de "régimes"*. Presses de l'Ecole des mines.

Benkler, Y. (2006). *The wealth of networks: How social production transforms markets and freedom*. Yale Univ Pr.

Benkler, Y., Roberts, H., Faris, R., Solow-Niederman, A., and Etling, B. (2013). Social Mobilization and the Networked Public Sphere: Mapping the SOPA-PIPA Debate. Technical report, Social Science Research Network, Rochester, NY.

Benraïss-Noailles, L. and Viot, C. (2012). Les médias sociaux dans les stratégies de recrutement. Quelle compatibilité avec la vie privée ? *Revue française de gestion*, 38(224):125–138.

Bergquist, M. and Ljungberg, J. (2001). The power of gifts: organizing social relationships in open source communities. *Information Systems Journal*, 11(4):305–320.

Berkeley, E. C. (1949). *Giant brains or Machines that think*. Chapman & Hall, New York : J. Wiley & sons ; London.

Bernoff, J. and Li, C. (2008). Harnessing the power of the oh-so-social web. *MIT Sloan Management Review*, 49(3):36.

Beuscart, J. S., Cardon, D., Pissard, N., and Prieur, C. (2009). Pourquoi partager mes photos de vacances avec des inconnus ? *Réseaux*, 154(2):91–129.

Bhattacharya, S., Krishnan, V., and Mahajan, V. (1998). Managing New Product Definition in Highly Dynamic Environments. *Management Science*, 44(11-Part-2):S50–S64.

Bijker, W. E., Hughes, T. P., and Pinch, T. J. (1987). *The social construction of technological systems : new directions in the sociology and history of technology*. The MIT press, Cambridge ; London.

Bilby, K. (1986). *The general: David Sarnoff and the rise of the communications industry*. Harper & Row.

Blohm I, Bretschneider U, Marco J.L, and Krcmar H (2011). Does collaboration among participants lead to better ideas in IT-based idea competitions? An empirical investigation. *Int. J. Networking Virtual Organ. International Journal of Networking and Virtual Organisations*, 9(2):106–122.

Boje, D. M. (2002). *Narrative methods for organizational and communication research*. Sage Publications, London [etc.].

Bonaccorsi, A. and Rossi, C. (2003). Why Open Source software can succeed. *Open Source Software Development*, 32(7):1243–1258.

Boullier, D. (1990). *Genèse des modes d'emploi: la mise en scène de l'utilisateur final*. Euristic Media, Cesson-Sévigné (France).

Bourreau, M. and Gensollen, M. (2004). Communautés d'expérience et concurrence entre sites de biens culturels. *Revue d'Économie Politique*, mars 2004:61–89.

Bozeman, B. and Dietz, J. (2001). Strategic research partnerships: Constructing policy-relevant indicators. *Journal of Technology Transfer*, 26(4):385–393.

Brabham, D. C. (2008). Crowdsourcing as a model for problem solving. *Convergence: The International Journal of Research into New Media Technologies*, 14(1):75.

Briscoe, B., Odlyzko, A., and Tilly, B. (2006). Metcalfe's law is wrong-communications networks increase in value as they add members-but by how much? *Spectrum, IEEE*, 43(7):34–39.

Brooks, F. P. (2010). *The design of design : essays from a computer scientist*. Addison-Wesley, Upper Saddle River, NJ.

Brousseau, E. and Chaves, B. (2004). Diffusion and Impact of E-Commerce: The French Specific Path. *Report for the Center for Research on Information Technology and Organizations (CRITO), University of California at Irvine, and the US National Science Foundation (CISE/IIS/CSS), Globalization and E-Commerce Project. Available from: http://crito. uci. edu*.

Brousseau, E. and Penard, T. (2007). The Economics of Digital Business Models: A Framework for Analyzing the Economics of Platforms. *Review of Network Economics*, 6(2):81–114.

Brown, S. L. and Eisenhardt, K. M. (1995). Product Development: Past Research, Present Findings, and Future Directions. *The Academy of Management Review*, 20(2):343–378.

Brown, T. J., Dacin, P. A., Pratt, M. G., and Whetten, D. A. (2006). Identity, Intended Image, Construed Image, and Reputation: An Interdisciplinary Framework and Suggested Terminology. *Journal of the Academy of Marketing Science*, 34(2):99–106.

Bryant, L. (1976). The development of the diesel engine. *Technology and Culture*, 17(3):432–446.

Buganza, T. and Verganti, R. (2006). Life-Cycle Flexibility: How to Measure and Improve the Innovative Capability in Turbulent Environments. *Journal of Product Innovation Management*, 23(5):393–407.

Burgelman, R. A. (1991). Intraorganizational Ecology of Strategy Making and Organizational Adaptation: Theory and Field Research. *Organization Science*, 2(3):239–262.

Burger-Helmchen, T. and Cohendet, P. (2011). User Communities and Social Software in the Video Game Industry. *Long Range Planning*, 44(5 - 6):317 – 343.

Caillaud, B. and Jullien, B. (2003). Chicken & Egg: Competition among Intermediation Service Providers. *The RAND Journal of Economics*, 34(2):309–328.

Calishain, T. and Dornfest, R. (2003). *Google hacks: [100 industrial-strength tips & tools]*. O'Reilly, Sebastopol, CA.

Callon, M. (1992). Variété et irréversibilité dans les réseaux de conception et d'adoption des techniques. In Foray, D. and Freeman, C., editors, *Technologie et richesse des nations:*, pages 276–324. Economica, Paris, France.

Cammaerts, B. (2008). Critiques on the participatory potentials of Web 2.0.

Cardon, D. (2005). De l'innovation ascendante.

Cardon, D. (2008). Le design de la visibilité. *Réseaux*, 152(6):93.

Cardon, D. (2010). *La démocratie Internet : Promesses et limites*. Seuil, Paris.

Cardon, D. (2012). Les origines hippies de la révolution digitale. In Turner, F., editor, *Aux sources de l'utopie numérique: de la contre-culture à la cyberculture*, chapter Préface, pages 11–32. C&F éditions, impr. 2012, Caen, France.

Cardon, D. and Delaunay-Téterel, H. (2006a). La production de soi comme technique relationnelle. *Réseaux*, 4(138):15–71.

Cardon, D. and Delaunay-Téterel, H. (2006b). La production de soi comme technique relationnelle. *Réseaux*, 4(138):15–71.

Carof, J. (1973). Un apport décisif à l'innovation: le consommateur. In Anglade, B. and Cukier, H., editors, *Innovation et produits nouveaux*, chapter 2, page 226. Dunod, Paris, Bruxelles, Montréal.

Casper, S. (2007). How do technology clusters emerge and become sustainable?: Social network formation and inter-firm mobility within the San Diego biotechnology cluster. *Research Policy*, 36(4):438–455.

Castel, R. (1994). "Problematization" as a mode of reading history. In *Foucault and the Writing of History*, chapter 13, pages 237–52. Blackwell Publishers Ltd, Cambridge (Mass.).

Ceruzzi, P. E. (1989). *Beyond the limits: flight enters the computer age*. MIT Press.

Ceruzzi, P. E. (2003). *A history of modern computing*. MIT Press.

Ceruzzi, P. E. (2008). *Internet Alley: High Technology in Tysons Corner, 1945-2005*. MIT Press, Cambridge, Mass.

Chakrabarti, S., Joshi, M. M., Punera, K., and Pennock, D. M. (2002). The structure of broad topics on the web. In *WWW '02 Proceedings of the 11th international conference on World Wide Web*, pages 251 – 262, Honolulu, Hawaii, USA. ACM New York, NY, USA.

Chanal, V. (2004). Innovation management and organizational learning: a discursive approach. *European Journal of Innovation Management European Journal of Innovation Management*, 7(1):56–64.

Chantepie, P. (2010). Révolution(s) numérique(s) des industries culturelle. *Réalités industrielles*, Nov:34–41.

Chase, R. B. (1978). Where does the customer fit in a service operation? *Harvard Business Review*, 56(6):137–142.

Chatzis, K. (2008). Rationalizing Maintenance Activities within French Industry during the Trente Glorieuses (1945-75). *HOST: Journal of History of Science and Technology*, 2:75—-138.

Chatzis, K. and Ribeill, G. (2008). Des périodiques techniques par et pour les ingénieurs. Un panorama suggestif, 1800-1914. In *La presse et les périodiques techniques en Europe (1750-1950)*, chapter 5, pages 115–157. Editions L'Harmattan, Paris.

Chesbrough, H. (2004). Managing open innovation. *Research-Technology Management*, 47(1):23–26.

Chesbrough, H. and Schwartz, K. (2007). Innovating business models with co-development partnerships. *Research-Technology Management*, 50(1):55–59.

Chesbrough, H. W. (2003a). *Open Innovation: The New Imperative for Creating And Profiting from Technology*. Harvard Business Press, Boston, Mass., first trad edition.

Chesbrough, H. W. (2003b). The Era of Open Innovation. *MIT Sloan management review*, 44(3):35–41.

Chesbrough, H. W. (2006). *Open innovation: the new imperative for creating and profiting from technology*. Harvard Business Press.

Chesbrough, H. W. and Appleyard, M. M. (2007). Open innovation and strategy. *California Management Review*, 50(1):57.

Chesbrough, H. W., Vanhaverbeke, W., and West, J. (2006). *Open innovation: researching a new paradigm*. Oxford University Press.

Choudhury, V., Hartzel, K. S., and Konsynski, B. R. (1998). Uses and Consequences of Electronic Markets: An Empirical Investigation in the Aircraft Parts Industry. *MIS Quarterly*, 22(4):471–507.

Christensen, C. M. (1997). *The innovator's dilemma: when new technologies cause great firms to fail*. The management of innovation and change series / ed. Michael L. Tushman and Andrew H. Van de Ven. - Boston Mass : Harvard Business School Press. Harvard Business School Press, Boston (Mass.), Etats-Unis.

Christensen, C. M. and Rosenbloom, R. S. (1995). Explaining the attacker's advantage: Technological paradigms, organizational dynamics, and the value network. *Research Policy*, 24(2):233–257.

Christensen, C. M., Suarez, F. F., and Utterback, J. M. (1998). Strategies for survival in fast-changing industries. *Management Science*, 44(12):S207–S220.

Christensen, J., Olesen, M., and Kjæ r, J. (2005). The industrial dynamics of Open Innovation - Evidence from the transformation of consumer electronics. *Research Policy*, 34(10):1533–1549.

Chrysos, P., Kazakci, A., and Lefebvre, P. (2010). Open products: a study on the open partition of web 2.0 platforms. In *17th International Product Development Management Conference*, page 21, Murcia, Spain.

Churchill, J., von Hippel, E., and Sonnack, M. (2009). Lead User Project Handbook: A practical guide for lead user project teams. Technical report.

Clark, C. H. (1958). *Brainstorming, the dynamic new way to create successful ideas.* Doubleday, Garden City, N.Y.

Clark, K. and Fujimoto, T. (1991). *Product development performance: Strategy, organization, and management in the world auto industry*. Harvard Business Press.

Clark, K. B. (1985). The interaction of design hierarchies and market concepts in technological evolution. *Research Policy Research Policy*, 14(5):235–251.

Codello-Guijarro, P., Jougleux, M., and Bancel-Charensol, L. (2011). Vers une approche enrichie des fronts offices dans les relations de service Réflexion à partir du cas d'une CPAM. (French).

Cook, S. (2008). The contribution revolution. *Harvard Business Review*, 86(10):60–69.

Cooper, R. G. (1987). *Winning at new products*. Gage, Agincourt, Ont.

Cooper, R. G. (2001). *Winning at New Products: Accelerating the Process from Idea to Launch, Third Edition*. Basic Books, 3rd edition.

Cortada, J. (2000). *Before the Computer: IBM, NCR, Burroughs, and Remington Rand and the Industry They Created, 1865-1956*. Princeton University Press, New Jersey, USA.

Cunha, M. P. e. and Gomes, J. F. (2003). Order and Disorder in Product Innovation Models. *Creativity and Innovation Management*, 12(3):174–187.

Cusumano, M. A. (2008). The Changing Software Business: Moving from Products to Services. *Computer*, 41(1):20–27.

Cusumano, M. A. and Yoffie, D. B. (1998). *Competing on Internet time: lessons from Netscape and its battle with Microsoft*. Free Press, New York, NY, USA.

Cusumano, M. A. and Yoffie, D. B. (1999). What Netscape learned from cross-platform software development. *Commun. ACM*, 42(10):72–78.

Czarniawska, B. (1998). *A narrative approach to organization studies*, volume 43 of *Qualitative research methods, v. 43*. Sage.

Dahlander, L. and Magnusson, M. G. (2005). Relationships between open source software companies and communities: Observations from Nordic firms. *Research Policy*, 34(4):481–493.

Dalle, J. M., David, P. A., den Besten, M., and Steinmueller, W. E. (2008). Empirical issues in open source software. *Information Economics and Policy*, 20(4):301–304.

De Bruyn, A. and Lilien, G. L. (2008). A multi-stage model of word-of-mouth influence through viral marketing. *International Journal of Research in Marketing*, 25(3):151–163.

Delaunay, J. C. and Gadrey, J. (1987). *Les enjeux de la société de service*. Presses de la Fondation nationale des sciences politiques, Paris, France.

Demsetz, H. (1967). Toward a Theory of Property Rights. *The American Economic Review*, 57(2):347–359.

Denner, A. (1971). *Principes et pratique du marketing*. Ce qu'il vous faut savoir, ISSN 0750-3431. J. Delmas, Paris.

Denyer, D., Parry, E., and Flowers, P. (2011). "Social", "Open" and "Participative"? Exploring Personal Experiences and Organisational Effects of Enterprise2.0 Use. *Long Range Planning*, 44(5–6):375–396.

Deshayes, C. and Bourguinat, E. (2008). Faut-il prendre au sérieux le web 2.0 ?

Di Maria, E. and Vladi, F. (2008). Communities of Consumption and Made in Italy. *Industry & Innovation*, 15(2):179–197.

Diebold, J. (1959). *Automation: its impact on business and labor*. National Planning Association, Washington.

Dobele, A., Lindgreen, A., Beverland, M., Vanhamme, J., and van Wijk, R. (2007). Why pass on viral messages? Because they connect emotionally. *Business Horizons*, 50(4):291–304.

Dobele, A., Toleman, D., and Beverland, M. (2005). Controlled infection! Spreading the brand message through viral marketing. *Business Horizons*, 48(2):143–149.

Doerr, J. (1978). Low-cost microcomputing: The personal computer and single-board computer revolutions. *Proceedings of the IEEE*, 66(2):117– 130.

Drucker, P. (1998). The discipline of innovation. *Harvard business review*, 76(6).

Dye, R. (2000). The buzz on buzz. *Harvard Business Review*, 78(6):139–146.

Ebner, W., Leimeister, J. M., and Krcmar, H. (2009). Community engineering for innovations: the ideas competition as a method to nurture a virtual community for innovations. *R&D Management*, 39(4):342–356.

Edmondson, A. C. and McManus, S. E. (2007). Methodological Fit in Management Field Research. *ACADEMY OF MANAGEMENT REVIEW*, 32(4):1155–1179.

Eiglier, P. and Langeard, E. (1975). Une approche nouvelle du marketing des services. *Revue Française de Gestion*, 2.

Eisenhardt, K. M. (1989). Building Theories from Case Study Research. *The Academy of Management Review*, 14(4):532–550.

Eisenhardt, K. M. and Galunic, D. C. (2000). Coevolving: At Last, a Way to Make Synergies Work - Synergies have always been notoriously challenging to capture. Here's a corporate strategic process for capturing synergies that resembles the coevolving systems found in nature. *Harvard business review.*, 78(1):91.

Eisenhardt, K. M. and Tabrizi, B. N. (1995). Accelerating Adaptive Processes: Product Innovation in the Global Computer Industry. *Administrative Science Quarterly*, 40(1):84–110.

Eisenmann, T., Parker, G., and Van Alstyne, M. (2011). Platform envelopment. *Strategic Management Journal*, 32(12):1270–1285.

Eisenmann, T., Parker, G., and Van Alstyne, M. W. (2006). Strategies for two-sided markets. *Harvard business review*, 84(10):92.

Ellis, C. (2003). Lessons from online groceries. *MIT Sloan management review*, 44(2):8.

Elmquist, M. and Segrestin, B. (2009). Sustainable development through innovative design: lessons from the KCP method experimented with an automotive firm. *International Journal of Automotive Technology and Management*, 9(2):229–244.

Engelbrecht, H.-J. (1997). A comparison and critical assessment of Porat and Rubin's information economy and Wallis and North's transaction sector. *Information Economics and Policy*, 9(4):271–290.

Enis, B. M. and Sullivan, E. T. (1985). The AT&T settlement: legal summary, economic analysis, and marketing implications. *Journal of Marketing*, 49(1):127.

Erden, Z., von Krogh, G., and Nonaka, I. (2008). The quality of group tacit knowledge. *The Journal of Strategic Information Systems*, 17(1):4–18.

Erle, S. and Gibson, R. (2005). *Google Maps Hacks*. O'Reilly Media, Inc.

Evans, D. S. (2010). How catalysts ignite: the economics of platform-based start-ups. In Gawer, A., editor, *Platforms, Markets and Innovation*, page 99. Edward Elgar Publishing, Cheltenham, UK ; Northampton, MA.

Evans, D. S. and Schmalensee, R. (2007). *Catalyst code : the strategies behind the world's most dynamic companies*. Harvard Business School Press, Boston, Mass.

Fa, L. L., Hippel, E. V., and Ogawa, S. (2011). The Age of the Consumer-Innovator. *MIT Sloan Management Review*, 53(53105):27–35.

Faggin, F., Hoff, M., Mazor, S., and Shima, M. (1996). The history of the 4004. *Micro, IEEE*, 16(6):10–20.

Farrell, J. and Klemperer, P. (2007). Coordination and Lock-In: Competition with Switching Costs and Network Effects. In Armstrong, M. and Porter, R., editors, *Handbook of Industrial Organization*, volume Volume 3 of *Handbook of Industrial Organization*, pages 1967–2072. Elsevier.

Farrell, J. and Saloner, G. (1986). Installed Base and Compatibility: Innovation, Product Preannouncements, and Predation. *The American Economic Review*, 76(5):940–955.

Fauchart, E. and Gruber, M. (2011). Darwinians, communitarians, and missionaries: the role of founder identity in entrepreneurship. *Academy of management journal*, 54(5):935–957.

Fayard, A.-L. and DeSanctis, G. (2010). Enacting language games: the development of a sense of 'we-ness' in online forums. *Information Systems Journal*, 20(4):383–416.

Fein, L. (1959). The Role of the University in Computers, Data Processing, and Related Fields. In *Managing Requirements Knowledge, International Workshop on*, volume 0, page 119.

Fein, L. (1961). The computer-related sciences (synnoetics) at a university in the year 1975. *BIT*, 1(4):227–255.

Feller, J., editor (2005). *Perspectives on free and open source software*. MIT Press, Cambridge (Mass.).

Felzensztein, C., Gimmon, E., and Carter, S. (2010). Geographical Co-Location, Social Networks and Inter-firm Marketing Co-operation: the Case of the Salmon Industry. *Long Range Planning*, 43(5-6):675–690.

Fisher, F. M., McKie, J. W., and Mancke, R. B. (1983). *IBM and the U.S. data processing industry : an economic history*. Praeger studies in select basic industries. Praeger, New York, N.Y., U.S.A.

Fixari, D., Moisdon, J. C., and Weil, B. (1997). A quoi rêvent les logiciels? L'informatisation des services extérieurs du ministère du Travail. *Du mode d'existence des outils de gestion, Seli Arslan*, 1:68–90.

Fleming, L. and Waguespack, D. M. (2007). Brokerage, Boundary Spanning, and Leadership in Open Innovation Communities. *Organization Science*, 18(2):165–180.

Flichy, P. (1991). *Une histoire de la communication moderne*. La Découverte, Paris.

Flichy, P. (1995a). *Dynamics of modern communication : the shaping and impact of new communication technologies*. Media, culture and society series. Sage Publications.

Flichy, P. (1995b). *L'innovation technique: récents développements en sciences sociales, vers une nouvelle théorie de l'innovation*. La découverte.

Flichy, P. (2004). *Η ιστορία της σύγχρονης επικοινωνίας*. ΚΑΤΟΠΤΡΟ, Αθήνα.

Fombelle, P., Jarvis, C., Ward, J., and Ostrom, L. (2012). Leveraging customers' multiple identities: identity synergy as a driver of organizational identification. *Journal of the Academy of Marketing Science*, 40(4):587–604.

Forte, A., Larco, V., and Bruckman, A. (2009). Decentralization in Wikipedia governance. *Journal of Management Information Systems*, 26(1):49–72.

Foucault (1969). *L'Archéologie du savoir*. Gallimard, service de edition.

Foucault, M. (1984). Le souci de la vérité. *Magazine littéraire*, 207.

Franke, N. and Shah, S. (2003). How communities support innovative activities: an exploration of assistance and sharing among end-users. *Research policy.*, 32(1):157.

Franke, N., von Hippel, E., and Schreier, M. (2006). Finding Commercially Attractive User Innovations: A Test of Lead-User Theory*. *Journal of Product Innovation Management*, 23(4):301–315.

Fredberg, T. (2009). Organising Customers: Learning from Big Brother. *Long Range Planning*, 42(3):320–340.

Freiberger, P. and Swaine, M. (1999). *Fire in the Valley: The Making of the Personal Computer*. McGraw-Hill Professional, 2nd edition.

Freiberger, P. and Swaine, M. (2000). Fire in the valley the making of the personal computer.

Frey, K., Lüthje, C., and Haag, S. (2011). Whom Should Firms Attract to Open Innovation Platforms? The Role of Knowledge Diversity and Motivation. *Long Range Planning*, 44(5–6):397–420.

Funk, J. L. (2005). The future of the mobile phone Internet: an analysis of technological trajectories and lead users in the Japanese market. *Technology in Society*, 27(1):69–83.

Gallouj, F. and Weinstein, O. (1997). Innovation in services. *Research Policy*, 26(4-5):537–556.

Gandon, M. and Jacquin, Y. (2001). L'effort de recherche et développement des principaux groupes industriels français. *Note d'information - Direction de la programmation et du développement*, (41):1–5.

Garel, G. (2003). Pour une histoire de la gestion de projet. *Gérer et comprendre*, (77):77–89.

Garel, G. and Midler, C. (2001). Front-loading problem solving in co-development: managing the contractual, organisational and cognitive dimensions. *International Journal of Automotive Technology and Management*, 1(2):236–251.

Garel, G. and Mock, E. (2012). *La fabrique de l'innovation*. DUNOD, Paris (France).

Garel, G. and Rosier, R. (2008). Régimes d'innovation et exploration. *Revue française de gestion*, 7(187):127–144.

Garfinkel, S. L. (2008). Wikipedia and the Meaning of Truth. *Technology Review*, 111(6):84–86.

Garratt, G. (1994). *The early history of radio: from Faraday to Marconi*. IEE history of technology series, ISSN 0266-1721. Institution of Electrical Engineers, in association with the Science Museum, London, UK, Royaume-Uni.

Garud, R., Jain, S., and Tuertscher, P. (2008). Incomplete by Design and Designing for Incompleteness. *Organization Studies*, 29(3):351–371.

Gassmann, O., Enkel, E., and Chesbrough, H. (2010). The future of open innovation. *R&D Management*, 40(3):213–221.

Gault, F. and von Hippel, E. (2009). The prevalence of user innovation and free innovation transfers : implications for statistical indicators and innovation policy.

Gawer, A. (2010a). Platform dynamics and strategies: from products to services. In Gawer, A., editor, *Platforms, Markets and Innovationa*, chapter 3, pages 45–76. Edward Elgar Publishing, Cheltenham, UK ; Northampton, MA.

Gawer, A. (2010b). The organization of technological platforms. In Phillips, N., Griffiths, D. S., and Sewell, G., editors, *Technology and organization : essays in honour of Joan Woodward*, volume 29, pages 287–296. Emerald Group Publishing, Bingley, UK.

Gawer, A. and Cusumano, M. (2012). Industry Platforms and Ecosystem Innovation. In *DRUID 2012*, page 28, Copenhagen, Denmark. DRUID.

Gawer, A. and Cusumano, M. A. (2002). *Platform leadership: how Intel, Microsoft, and Cisco drive industry innovation*. Harvard Business Press, Boston, MA, USA.

Gensollen, M. (2003). Biens informationnels et communautés médiatées. *Revue d'économie politique*, 113:9–42.

Gensollen, M. (2007). Information goods and online communities. In Brousseau, E. and Curien, N., editors, *Internet and Digital Economics Principles Methods and Applications*, pages 173–200. Cambridge University Press, Cambridge, UK.

Georges, F. (2009). Représentation de soi et identité numérique. *Réseaux*, 154(2):165.

Ghosh, R. A., Glott, R., Krieger, B., and Robles, G. (2002). *Free/libre and open source software: Survey and study*. Maastricht Economic Research Institute on Innovation and Technology, University of Maastricht, The Netherlands, June.

Gillier, T., Piat, G., Roussel, B., and Truchot, P. (2010). Managing Innovation Fields in a Cross-Industry Exploratory Partnership with C-K Design Theory*. *Journal of Product Innovation Management*, 27(6):883–896.

Giuri, P., Rullani, F., and Torrisi, S. (2008). Explaining leadership in virtual teams: The case of open source software. *Information Economics and Policy*, 20(4):305–315.

Godin, B. (2006). The Linear Model of Innovation The Historical Construction of an Analytical Framework. *Science, Technology & Human Values*, 31(6):639–667.

Goldman, J. (2009). *Facebook cookbook*. O'Reilly, Beijing; Sebastopol.

Gollwitzer, P. M. and Brandstätter, V. (1997). Implementation intentions and effective goal pursuit. *Journal of Personality and Social Psychology*, 73(1):186–199.

Granovetter, M. (1983). The strength of weak ties: A network theory revisited. *Sociological theory*, 1(1):201–233.

Granovetter, M. S. (1973). The strength of weak ties. *The American journal of sociology*, 78(6):1360–1380.

Gray, G. and Smith, R. (2003). Before the B5000: Burroughs computers, 1951-1963. *Annals of the History of Computing, IEEE*, 25(2):50– 61.

Gray, G. and Smith, R. (2004). Sperry Rand's first-generation computers, 1955-1960: hardware and software. *Annals of the History of Computing, IEEE*, 26(4):20– 34.

Grove, A. S. (1997). *Only the paranoid survive : how to exploit the crisis points that challenge every company and career*. HarperCollins Business, London.

Guha, R. and Garg, A. (2005). Disambiguating people in search. In *WWW 2005*, pages 463–470, Chiba, Japan. International World Wide Web Conference Committee (IW3C2).

Habermas, J. (1991). *The structural transformation of the public sphere: an inquiry into a category of bourgeois society*. MIT Press.

Habermas, J. (1992). Further reflections on the public sphere. In Calhoun, C. J., editor, *Habermas and the public sphere*, chapter 17, pages 421–461. The MIT Press, Cambridge, MA.

Hackman, J. R. (1987). The design of work teams. *Handbook of organizational behavior*, 315:342.

Haddon, L. (2004). *Information and Communication Technologies in Everyday Life: A Concise Introduction and Research Guide*. Berg Publishers, Oxford ; New York.

Haefliger, S., Jäger, P., and von Krogh, G. (2010). Under the radar: Industry entry by user entrepreneurs. *Research Policy*, 39(9):1198–1213.

Haefliger, S., Monteiro, E., Foray, D., and von Krogh, G. (2011). Social Software and Strategy. *Long Range Planning*, 44(5 - 6):297 – 316.

Haefliger, S., Reichen, P., Jäger, P., and von Krogh, G. (2009). Modding as rating behavior in virtual communities: The case of Rooster Teeth Productions. *Lecture Notes in Computer Science*, 5621:197–206.

Hagedoorn, J., Link, A. N., and Vonortas, N. S. (2000). Research partnerships. *Research Policy*, 29(4-5):567–586.

Hansen, M. T. (2002). Knowledge networks: Explaining effective knowledge sharing in multiunit companies. *Organization science*, 13(3):232–248.

Hansen, S., Berente, N., and Lyytinen, K. (2009). Wikipedia, Critical Social Theory, and the Possibility of Rational Discourse 1. *The Information Society*, 25(1):38–59.

Hanson, W. (1998). The original WWW: Web lessons from the early days of radio. *Journal of Interactive Marketing*, 12(3):46–56.

Hara, N. (2009). Online Communities of Practice: Beyond Organizational Boundaries. In *Communities of Practice*, pages 95–111. Springer.

Hardy, C. and Maguire, S. (2010). Discourse, field-configuring events, and change in organizations and institutional fields : narratives of DDT and the Stockholm Convention. *Academy of Management journal : AMJ. - Briarcliff Manor, NJ : Academy of Management, ISSN 0001-4273, ZDB-ID 2218598. - Vol. 53.2010, 6, p. 1365-1392*.

Hars, A. and Ou, S. (2001). Working for Free? - Motivations of Participating in Open Source Projects. In *34th Hawaii International Conference on System Sciences*, volume 7, page 9, Los Alamitos, CA, USA. IEEE Computer Society.

Hatchuel, A. (1997). Fondéments de savoirs et légitimité des règles. In Reynaud, B. C. d. C., editor, *Les figures du collectif, Tome 2*, pages 183–210. La Découverte & Syros, Paris, les colloq edition.

Hatchuel, A. (1998). Comment penser l'action collective? Théorie des mythes rationnels. In Damien, R. and Tosel, A., editors, *L'action collective*, pages 177–202. Presses Univ. Franche-Comté, Besançon, France.

Hatchuel, A. (1999). The Foucauldian Detour: A Rebirth of Organization Theory? *Human Relations*, 52(4):507–519.

Hatchuel, A. (2001). Towards Design Theory and expandable rationality: The unfinished program of Herbert Simon. *Journal of management and governance*, 5(3):260–273.

Hatchuel, A. (2005a). Pour une épistémologie de l'action. L'expérience des sciences de gestion. In Teulier, R. and Lorino, P., editors, *Entre connaissance et organisation: l'activité collective*, chapter 3, pages 72–92. Découverte, Paris, colloque d edition.

Hatchuel, A. (2005b). Pour une épistémologie de l'action. L'expérience des sciences de gestion. In Teulier, R. and Lorino, P., editors, *Entre connaissance et organisation: l'activité collective*, chapter 3, pages 72–92. Découverte, Paris, colloque d edition.

Hatchuel, A. (2005c). Towards an epistemology of collective action: management research as a responsive and actionable discipline. *European Management Review*, 2:36–47.

Hatchuel, A. and Le Masson, P. (2007). Shaping the unknown and the emergent: Design theory as a new source of management knowledge. *European Academy of Management*, 21.

Hatchuel, A., Le Masson, P., and Weil, B. (2001). From R&D to R-I-D: Design strategies and the management of "Innovation Fields". In *8th International Product Development Management Conference*, Enschede.

Hatchuel, A., Le Masson, P., and Weil, B. (2002). From knowledge management to design-oriented organisations. *International Social Science Journal*, 54(171):25–37.

Hatchuel, A., Le Masson, P., and Weil, B. (2011). Teaching Innovative Design Reasoning: How Concept?knowledge Theory Can Help Overcome Fixation Effects. *AI EDAM*, 25(01):77–92.

Hatchuel, A. and Molet, H. (1986). Rational modelling in understanding and aiding human decision-making: About two case studies. *European Journal of Operational Research*, 24(1):178–186.

Hatchuel, A. and Weil, B. (1992). *L'expert et le système: gestion des savoirs et métamorphose des acteurs dans l'entreprise industrielle, suivi de quatre histoires de systèmes-experts*. Economica.

Hatchuel, A. and Weil, B. (1995). *Experts in Organizations: A Knowledge-Based Perspective on Organizational Change*. Walter de Gruyter, Berlin, New York.

Hatchuel, A. and Weil, B. (1999). Design-Oriented Organizations, Towards a Unified Theory of Design Activities. In *6th international product development management conference*, page 28, Cambridge, UK.

Hatchuel, A. and Weil, B. (2009). C-K design theory: an advanced formulation. *Research in Engineering Design*, 19(4):181–192.

Haythornthwaite, C. (2009). Crowds and Communities: Light and Heavyweight Models of Peer Production. In *System Sciences, 2009. HICSS'09. 42nd Hawaii International Conference*, pages 1–10. IEEE.

Henderson, R. M. and Clark, K. B. (1990). Architectural innovation: the reconfiguration of existing product technologies and the failure of established firms. *Administrative science quarterly*, 35(1):9–30.

Hendler, J. (2008). Web 3.0: Chicken Farms on the Semantic Web. *Computer*, 41(1):106 –108.

Hendler, J. and Golbeck, J. (2008). Metcalfe's law, Web 2.0, and the Semantic Web. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(1):14–20.

Henkel, J. (2003). Software Development in Embedded Linux-Informal Collaboration of Competing Firnis. *Wirtschaftsinformatik 2003/Band II: Medien-Markte-Mobilitat*, 2:81.

Herstatt, C. and von Hippel, E. (1992). From experience: Developing new product concepts via the lead user method: A case study in a "low-tech" field. *Journal of Product Innovation Management*, 9(3):213–221.

Hienerth, C., Keinz, P., and Lettl, C. (2011). Exploring the Nature and Implementation Process of User-Centric Business Models. *Long Range Planning*, 44(5–6):344–374.

Highfield, T., Kirchhoff, L., and Nicolai, T. (2010). Challenges of Tracking Topical Discussion Networks Online. *Social Science Computer Review*.

Hijiya, J. A. (1992). *Lee de Forest*. Lehigh University Press.

Hilliard, R. L. (2005). *The broadcast century and beyond: a biography of American broadcasting*. Focal Press, Amsterdam; Boston.

Ho, S.-H. and Huang, C.-H. (2009). Exploring success factors of video game communities in hierarchical linear modeling: The perspectives of members and leaders. *Computers in Human Behavior*, 25(3):761–769.

Hong, S. (2001). *Wireless: from Marconi's black-box to the audion*. Transformations (Cambridge, Mass.). MIT Press, Cambridge (Mass.), Etats-Unis.

Hooge, S. (2010). *Performance de la R&D en rupture et des stratégies d'innovation Organisation, pilotage et modèle d'adhésion*. Centre for management science, Mines-ParisTech.

Hounkpatin, L., Wexler-Czitrom, H., Perez, A., and Courbin, L. (2011). Vers un nouveau paradigme: la clinique de la multiplicité et la fabrication de «l'intime collectif. In Guerraoui, Z. and Pirlot, G., editors, *Comprendre et traiter les situations interculturelles, approches psychodynamiques et psychanalytiques*, volume 67, chapter 3, pages 55–107. Bruxelles, de boeck edition.

Howe, J. (2006a). 5 Rules of the New Labor Pool. *Wired magazine*, 14(6).

Howe, J. (2006b). The rise of crowdsourcing. *Wired magazine*, 14(6):1–4.

Huber, M., Bretschneider, U., Krcmar, H., and Leimeister, J. M. (2009). Leveraging Crowdsourcing - Theory-driven Design, Imple-mentation and Evaluation of Activation-Supporting Components for IT-based Idea Competitions. *Journal of Management Information Systems*, 26(1):1–44.

Hudson, C. (2012). *eBay commerce cookbook: using eBay APIs*. O'Reilly, Farnham.

Husbands, P., Holland, O., and Wheeler, M. (2008). *The mechanical mind in history*. MIT Press, Cambridge (Mass.); London.

Iansiti, M. (1993). *Technology integration : managing technological evolution in a complex environment*. Division of Research, Harvard Business School, Boston, MA, USA.

Iansiti, M. (1998). *Technology integration: making critical choices in a dynamic world*. Harvard Business School Press.

Iansiti, M. and Levien, R. (2004). Strategy as ecology. *Harvard Business Review*, 82(3):68–81.

Inkpen, A. C. and Beamish, P. W. (1997). Knowledge, bargaining power, and the instability of international joint ventures. *The Academy of Management Review*, 22(1):177–202.

Isaacson, W. (2011). *Steve Jobs*. J.-C. Lattès, [Paris].

Jablokow, K. W. (2005). The catalytic nature of science: Implications for scientific problem solving in the 21st century. *Technology in Society*, 27(4):531–549.

Jablokow, K. W. and Booth, D. E. (2006). The impact and management of cognitive gap in high performance product development organizations. *Journal of Engineering and Technology Management*, 23(4):313–336.

Jarvenpaa, S. L. and Lang, K. R. (2011). Boundary Management in Online Communities: Case Studies of the Nine Inch Nails and ccMixter Music Remix Sites. *Long Range Planning*, 44(5–6):440–457.

Johnson, L. R. (1952). Installation of a large electronic computer. In *Proceedings of the 1952 ACM national meeting (Toronto) on - ACM '52*, pages 77–80, New York, New York, USA. ACM Press.

Jouët, J. (2000). Retour critique sur la sociologie des usages. *Réseaux*, 100(18):487–521.

Jougleux, M. (2006). Enrichir l'approche théorique de la qualité dans les services : qualité du service et qualité de service. *Recherche et applications en marketing*, 21(3).

Jouini, S. B. M., Midler, C., and Garel, G. (2004). Time-to-market vs. time-to-delivery: Managing speed in Engineering, Procurement and Construction projects. *International Journal of Project Management*, 22(5):359–367.

Kalashnikov, D. V., Zhaoqi Chen, Mehrotra, S., and Nuray-Turan, R. (2008). Web People Search via Connection Analysis. *IEEE Transactions on Knowledge and Data Engineering*, 20(11):1550–1565.

Kalyanam, K., McIntyre, S., and Masonis, J. T. (2007). Adaptive experimentation in interactive marketing: The case of viral marketing at Plaxo. *Journal of Interactive Marketing*, 21(3):72–85.

Kane, G. (2009). It's a network, not an encyclopedia: A social network perspective on Wikipedia collaboration. In *Best Paper proceedings, Academy of Management 2009*, page 7.

Kaplan, A. M. and Haenlein, M. (2011). Two hearts in three-quarter time: How to waltz the social media/viral marketing dance. *Business Horizons*, 54(3):253–263.

Karakas, F. (2009). Welcome to World 2.0: the new digital ecosystem. *Journal of Business Strategy*, 30(4):23–30.

Karp, D. A. (2003). *eBay hacks: [100 industrial-strength tips & tools]*. O'Reilly, Sebastopol, CA; Farnham.

Katz, M. L. and Shapiro, C. (1985). Network Externalities, Competition, and Compatibility. *The American Economic Review*, 75(3):424–440.

Katz, M. L. and Shapiro, C. (1986). Technology Adoption in the Presence of Network Externalities. *Journal of Political Economy*, 94(4):822–41.

Keelee, B. and Love, P. (2010). *From crisis to recovery: the causes, course and consequences of the great recession*. OECD Publishing, Paris, France, oecd insig edition.

Kendall, G. and Wickham, G. (1999). *Using Foucault's methods*. Sage Publications, London.

Kilkki, K. and Kalervo, M. (2004). KK-law for group forming services. In *XVth International Symposium on Services and Local Access*, pages 21–26, Edinburgh.

Kim, S.-J., Suh, N. P., and Kim, S.-G. (1991). Design of software systems based on axiomatic design. *Robotics and Computer-Integrated Manufacturing*, 8(4):243–255.

Kim, W. C. and Mauborgne, R. (1997). *Value innovation : the strategic logic of high growth*. Harvard Business School Pub., Boston, Mass.

Kirsch, D. and Goldfarb, B. (2008). Small Ideas, Big Ideas, Bad Ideas, Good Ideas:'Get Big Fast'and Dot Com Venture Creation. In Aspray, W. and Ceruzzi, P. E., editors, *The Internet and American Business*, pages 259–276. MIT Press, Cambridge, MA.

Kita, C. (2009). Events and Sightings. *Annals of the History of Computing, IEEE*, 31(2):76–79.

Kittur, A., Chi, E. H., and Suh, B. (2008a). Crowdsourcing user studies with Mechanical Turk. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 453–456. ACM.

Kittur, A., Chi, E. H., and Suh, B. (2008b). Crowdsourcing user studies with Mechanical Turk. page 453, Florence, Italy.

Kittur, A., Suh, B., Pendleton, B. A., and Chi, E. H. (2007). He says, she says: Conflict and coordination in Wikipedia. pages 453–462.

Koch, S. and Schneider, G. (2002). Effort, co-operation and co-ordination in an open source software project: GNOME. *Information Systems Journal*, 12(1):27–42.

Koss, A. M. (2003). Programming at Burroughs and Philco in the 1950s. *IEEE Annals of the History of Computing*, 25(4):40–50.

Krickx, G. A. (1995). Vertical integration in the computer mainframe industry: A transaction cost interpretation. *Journal of Economic Behavior & Organization*, 26(1):75–91.

Krishnan, V. and Ulrich, K. T. (2001). Product Development Decisions: A Review of the Literature. *Management Science*, 47(1):1–21.

Krogh, G. V., Roos, J., and Slocum, K. (1994). An essay on corporate epistemology. *Strategic Management Journal*, 15(S2):53–71.

Lacy, S. (2009). *The Facebook story*. Crimson, Richmond, England.

Lai, L. S. L. and Turban, E. (2008). Groups Formation and Operations in the Web 2.0 Environment and Social Networks. *Group Decision and Negotiation*, 17(5):387–402.

Laing, G. (2004). *Digital retro*. Sybex, Alameda, CA.

Lakhani, K., Wolf, B., Bates, J., and DiBona, C. (2002). The boston consulting group hacker survey. *Boston, The Boston Consulting Group*.

Lakhani, K. and Wolf, R. (2003). Why hackers do what they do: Understanding motivation and effort in free/open source software projects. *MIT Sloan Working Paper No. 4425-03*.

Lakhani, K. R., Jeppesen, L. B., Lohse, P. A., and Panetta, J. A. (2007). The Value of Openness in Scientific Problem Solving. *Biotech Business*, 18S(Spec No 2):5533–464.

Lakhani, K. R. and von Hippel, E. (2003). How open source software works: "free" user-to-user assistance. *Research Policy*, 32(6):923–943.

Lancaster, K. J. (1966). A New Approach to Consumer Theory. *Journal of Political Economy*, 74(2):132–157.

Langeard, E. and Meyer, J. (1975). *Principes et pratique du marketing*. Systèmes-décisions, Section Marketing [Texte imprimé]. - Paris : Presses universitaires de France, 19..- ; 2. Presses universitaires de France, Paris.

Lassila, O. and Hendler, J. (2007). Embracing "Web 3.0". *IEEE Internet Computing*, 11:90–93.

Latour, B. (1996). On actor-network theory: a few clarifications. *Soziale welt*, 47(4):369–381.

Lattemann, C. and Stieglitz, S. (2005). Framework for Governance in Open Source Communities. In *System Sciences, 2005. HICSS '05. Proceedings of the 38th Annual Hawaii International Conference*, volume 7, page 192a, Los Alamitos, CA, USA. IEEE Computer Society.

Lave, J. and Wenger, E. (1991). *Situated learning : legitimate peripheral participation*. Cambridge University Press, Cambridge.

Lazega, E. (1994). Analyse de réseaux et sociologie des organisations. *Revue française de sociologie*, 35(2):293–320.

Le Masson, P., Hatchuel, A., and Weil, B. (2011). The Interplay between Creativity Issues and Design Theories: A New Perspective for Design Management Studies? *Creativity and Innovation Management*, 20(4):217–237.

Le Masson, P. and Magnusson, P. (2003). User involvement: From ideas collection towards a new technique for innovative service design. In *2nd World Conference on Mass Customization and Personalization*, page 23, Munich, Germany.

Le Masson, P., Weil, B., and Hatchuel, A. (2006). *Les processus d'innovation : Conception innovante et croissance des entreprises*. Hermes Science Publications.

Le Masson, P., Weil, B., and Hatchuel, A. (2010). *Strategic Management of Innovation and Design*. Cambridge University Press.

Leblebic, H., Salancik, G. R., Copay, A., and King, T. (1991). Institutional Change and the Transformation of Interorganizational Fields: An Organizational History of the U.S. Radio Broadcasting Industry. *Administrative Science Quarterly*, 36(3):333–363.

Lécuyer, C. (2006). Defiant West. In *Making Silicon Valley : innovation and the growth of high tech, 1930-1970*, chapter 1, pages 13–52. MIT Press, Cambridge, Mass.

Lecuyer, C. (2006). *Making Silicon Valley : innovation and the growth of high tech, 1930-1970*. MIT Press, Cambridge, Mass.

Lee, G. K. and Cole, R. E. (2003). From a Firm-Based to a Community-Based Model of Knowledge Creation: The Case of the Linux Kernel Development. *ORGANIZATION SCIENCE*, 14(6):633–649.

Lee, S. M., Kim, T., Noh, Y., and Lee, B. (2010). Success factors of platform leadership in web 2.0 service business. *Service Business*, 4(2):89–103.

Lefebvre, P. (2005). Foucault et la discipline d'atelier: problèmes soulevés par une contribution originale. In *Gouvernement, organisation et gestion: l'héritage de Michel Foucault*, pages 179–213. Presses de l'Université de Laval.

Lefebvre, P. (2009). Subordination et « révolutions » du travail et du droit du travail (1776-2010). *Entreprises et histoire*, 57(4):45–78.

Lefebvre, P. (2013). Organizing Deliberate Innovation in Knowledge Clusters: From Accidental Brokering to Purposeful Brokering Processes. *International Journal of Technology Management*, page 29.

Leimeister, J. M., Huber, M., Bretschneider, U., and Krcmar, H. (2009). Leveraging Crowdsourcing: Activation-Supporting Components for IT-Based Ideas Competition. *Journal of Management Information Systems*, 26(1):197–224.

Leontief, W. W. (1941). *The structure of American economy, 1919-1929: An empirical application of equilibrium analysis*. Harvard University Press.

Lerner, J. and Tirole, J. (2002). Some Simple Economics of Open Source. *The Journal of Industrial Economics*, 50(2):197–234.

Lerner, J. and Tirole, J. (2004). *The economics of technology sharing: Open source and beyond*. National Bureau of Economic Research Cambridge, Mass., USA.

Lettl, C., Herstatt, C., and Gemuenden, H. G. (2006). Users' contributions to radical innovation: evidence from four cases in the field of medical equipment technology. *R&D Management*, 36(3):251–272.

Levrel, J. (2006). Wikipedia, un dispositif médiatique de publics participants. *Réseaux*, 138(4):185–218.

Levy, M. (2009). WEB 2.0 implications on knowledge management. *Journal of Knowledge Management*, 13(1):120–134.

Levy, S. (1984). *Hackers : heroes of the computer revolution*. Anchor Press/Doubleday, Garden City, N.Y.

Levy, S. (2010). *Hackers: Heroes of the Computer Revolution - 25th Anniversary Edition*. O'Reilly Media, 1 edition.

Lewis, M. (2000a). The New New Thing: A Silicon Valley Story (Book Review). *Harvard Business Review*, 78(6):194.

Lewis, M. W. (2000b). Exploring Paradox: Toward a More Comprehensive Guide. *The Academy of Management Review*, 25(4):760–776.

Long, S. L. (1987). Technological Change and Institutional Response: The Creation of American Broadcasting. *Journal of Economic Issues (Association for Evolutionary Economics)*, 21(2):743.

Lumpkin, G. and Dess, G. G. (2001). Linking two dimensions of entrepreneurial orientation to firm performance: The moderating role of environment and industry life cycle. *Journal of Business Venturing*, 16(5):429–451.

Lumpkin, G. T. and Dess, G. G. (1996). Clarifying the Entrepreneurial Orientation Construct and Linking It to Performance. *The Academy of Management Review*, 21(1):135.

Lüthje, C. and Franke, N. (2003). The 'making'of an entrepreneur: testing a model of entrepreneurial intent among engineering students at MIT. *R&D Management*, 33(2):135–147.

Lüthje, C., Herstatt, C., and von Hippel, E. (2005). User-innovators and "local" information: The case of mountain biking. *Research Policy*, 34(6):951–965.

MacCormack, A., Verganti, R., and Iansiti, M. (2001). Developing Products on "Internet Time": The Anatomy of a Flexible Development Process. *Management Science*, 47(1):133–150.

Magnusson, P. R., Matthing, J., and Kristensson, P. (2003). Managing User Involvement in Service Innovation Experiments with Innovating End Users. *Journal of Service Research*, 6(2):111–124.

Mahr, D. and Lievens, A. (2012). Virtual lead user communities: Drivers of knowledge creation for innovation. *Research Policy*, 41(1):167–177.

Makice, K. (2009). *Twitter API: Up and Running*. O'Reilly Media, Inc., Bejing; Cambridge; Farnham; Koln; Sebastopol; Taipei; Tokyo.

Malet, F. (1984). Mobi-prêt à la bibliothèque de l'Université de Paris I. Technical report, Paris I, Sorbonne, Paris.

Mallard, A. (2007). Une articulation forte entre innovation et usage est-elle possible? Quelques propositions autour du cas des assemblages numériques. In *Expérimenter, éprouver, assembler*, page 20, Paris. CSI, Mines-ParisTech.

Maniak, R. and Midler, C. (2008). Shifting from co-development to co-innovation. *International Journal of Automotive Technology and Management*, 8(4):449–468.

March, J. (1978). Bounded rationality, ambiguity, and the engineering of choice. *The Bell Journal of Economics*, 9(2):587–608.

March, J. G. (1991). Exploration and Exploitation in Organizational Learning. *Organization Science*, 2(1):71–87.

March, J. G. and Simon, H. (1964). *Les organisations*. Dunod.

Maskell, P., Bathelt, H., and Malmberg, A. (2006). Building global knowledge pipelines: The role of temporary clusters. *European Planning Studies*, 14(8):997–1013.

Mcafee, A. (2006). Enterprise 2.0: The Dawn of Emergent Collaboration. *Management of Technology and Innovation*, 47(3).

McKinlay, A. and Starkey, K. (1998). *Foucault, management and organization theory : from Panopticon to technologies of self*. Sage, Thousand Oaks, Ca.

Metcalfe, R. (1995). Metcalfe's law: A network becomes more valuable as it reaches more users. *Infoworld*, 40(17).

Meyer, A. D., Gaba, V., and Colwell, K. A. (2005). Organizing far from equilibrium: Nonlinear change in organizational fields. *Organization Science*, 16(5):456–473.

Midler, C. (1993). *L'auto qui n'existait pas : management des projets et transformation de l'entreprise*. InterEditions, Paris, France.

Midler, C. and Beaume, R. (2010). Project-based learning patterns for dominant design renewal: The case of Electric Vehicle. *International Journal of Project Management*, 28(2):142–150.

Midler, C., Garel, G., and Kesseler, A. (1997). Le codéveloppement : définition, enjeux et problèmes : Le cas de l'industrie automobile. *Education permanente*, (131):95–108.

Midler, C. and Navarre, C. (2007). Project Management in the Automotive Industry. In Morris, P. W. G. and Pinto, J. K., editors, *The Wiley Guide to Managing Projects*, pages 1368–1388. John Wiley & Sons, Inc.

Milgram, S. (1967). The small world problem. *Psychology today*, 2(1):60–67.

Miller, K. D., Fabian, F., and Lin, S. J. (2009). Strategies for online communities. *Strategic Management Journal*, 30(3):305–322.

Millier (1989). *Le marketing des produits "high-tech": Outils d'analyse*. Les Éditions d'Organisation, Paris.

Miltgen, C. L. (2011). Vie privée et marketing. *Réseaux*, 167(3):131–166.

Mintzberg, H. (1979). An Emerging Strategy of "Direct" Research. *Administrative Science Quarterly*, 24(4):582–89.

Mintzberg, H., Lampel, J., Quinn, J. B., and Ghoshal, S. (2003). *The Strategy Process: Concepts, Contexts, Cases*. Pearson Education, Harlow, England, 4, illustr edition.

Mitchell, R. G. (1993). *Secrecy and fieldwork*. Sage, Newbury Park.

Moggridge, B. (2006). *Designing interactions*. MIT Press, Cambridge, Mass.

Moore, J. (1993). Predators and prey: a new ecology of competition. *Harvard Business Review*, 71(3):75–86.

Morse, P. M. (1977). *In at the Beginnings: A Physicist's Life*. MIT Press.

Mowery, D. and Rosenberg, N. (1979). The influence of market demand upon innovation: a critical review of some recent empirical studies. *Research Policy*, 8(2):102–153.

Mueller, J. C. (1953). National Cash Register.

Muniz Jr, A. M. and Schau, H. J. (2011). How to inspire value-laden collaborative consumer-generated content. *Business Horizons*.

Myers, B. A. (1998). A brief history of human-computer interaction technology. *interactions*, 5(2):44–54.

Nakhla, M. (2003). Information, coordination and contractual relations in firms. *International Review of Law and Economics*, 23(1):101–119.

Nambisan, P. and Watt, J. H. (2010). Managing customer experiences in online product communities. *Journal of Business Research*, 64(8):889–895.

Nassimbeni, G. (2001). Technology, innovation capacity, and the export attitude of small manufacturing firms: a logit/tobit model. *Research Policy*, 30(2):245–262.

Neisser, H. P. (1941). Review: The structure of American economy, 1919-1939: an empirical application of equilibrium analysis. *The American Economic Review*, 31(3):608–610.

Nelson, T. (1974). *Computer Lib/Dream Machines*. Hugo's Book Service, Chicago, Illinois, USA.

Nonaka, I., Toyama, R., and Konno, N. (2000). SECI, Ba and Leadership: a Unified Model of Dynamic Knowledge Creation. *Long Range Planning*, 33(1):5–34.

Nonaka, I., Umemoto, K., and Senoo, D. (1996). From information processing to knowledge creation: A Paradigm shift in business management. *Technology in Society*, 18(2):203–218.

Nonaka, I. and von Krogh, G. (2009). Perspective—Tacit Knowledge and Knowledge Conversion: Controversy and Advancement in Organizational Knowledge Creation Theory. *Organization Science*, 20(3):635–652.

Nora, S. and Minc, A. (1978). L'informatisation de la société, rapport à M. le Président de la République. Technical report, Inspection Générale des Finances, Paris, France.

Norberg, A. (2003). Software Development at the Eckert-Mauchly Computer Company Between 1947 and 1955. *Iterations*, 2:1–26.

Nov, O. (2007). What motivates Wikipedians? *Communications of the ACM*, 50:60–64.

Ogawa, S. (1998). Does sticky information affect the locus of innovation? Evidence from the Japanese convenience-store industry. *Research Policy*, 26(7–8):777–790.

Oliveira, P. and von Hippel, E. (2011). Users as service innovators: The case of banking services. *Research Policy*, 40(6):806–818.

Olson, E. L. and Bakke, G. (2001). Implementing the lead user method in a high technology firm: A longitudinal study of intentions versus actions. *Journal of Product Innovation Management*, 18(6):388–395.

O'Mahony, S. (2003). Guarding the commons: how community managed software projects protect their work. *Research Policy*, 32(7):1179–1198.

Osborn, A. F. (1953). *Applied imagination : principles and procedures of creative problem-solving.* C. Scribner's Sons, New York.

Osborn, R. N. and Hagedoorn, J. (1997). The institutionalization and evolutionary dynamics of interorganizational alliances and networks. *The academy of management journal*, 40(2):261–278.

Osterloh, M. and Rota, S. (2007). Open source software development - Just another case of collective invention? *Research Policy*, 36(2):157–171.

Oudshoorn, N. and Pinch, T. (2008). User-Technology Relationships: Some Recent Developments. In Hackett, E. J., Amsterdamska, O., Lynch, M., and Wajcman, J., editors, *The handbook of science and technology studies*, chapter 22, pages 541–565. The MIT Press, Cambridge, USA.

O'Mahony, S. (2007). The governance of open source initiatives: what does it mean to be community managed? *Journal of Management & Governance*, 11(2):139–150.

Pahl, G., Beitz, W., Schulz, H.-J., and Jarecki, U. (2007). *Engineering Design*. Wallace, Ken; Blessing, Lucienne T.M.

Parker, G. G. and Alstyne, M. W. V. (2005). Two-Sided Network Effects: A Theory of Information Product Design. *Management Science*, 51(10):1494–1504.

Patriotta, G. (2003). Sensemaking on the Shop Floor : Narratives of Knowledge in Organizations. *Journal of Management Studies*, 40(2):349–375.

Peng, M. W. and Shenkar, O. (2002). Joint venture dissolution as corporate divorce. *The Academy of Management Executive (1993)*, 16(2):92–105.

Phelps, J., Lewis, R., Mobilio, L., Perry, D., and Raman, N. (2004). Viral Marketing or Electronic Word-of-Mouth Advertising: Examining Consumer Responses and Motivations to Pass Along Email. *Journal of Advertising Research*, 44(04):333–348.

Piller, F. T. (2010). Open Innovation with Customers: Crowdsourcing and Co-Creation at Threadless. *SSRN eLibrary*.

Piller, F. T. and Walcher, D. (2006). Toolkits for idea competitions: a novel method to integrate users in new product development. *R & D Management*, 36(3):307–318.

Pissard, N. and Prieur, C. (2007). Thematic vs. social networks in web 2.0 communities: A case study on Flickr groups. In *9ème Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications*, pages 31–34. INRIA.

Poetz, M. K. and Schreier, M. (2010). The Value of Crowdsourcing: Can Users Really Compete with Professionals in Generating New Product Ideas? *SSRN eLibrary*.

Porat, M. U. (1977). The Information Economy: Definition and Measurement. Technical report, Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402 (Stock No. 003-000-00512-7), Washington, D.C., United States.

Pugh, E. W. (1995). *Building IBM: Shaping an Industry and Its Technology*. MIT Press.

Quesnay, F. (1759). *Tableau économique avec ses explications.* la bibliothèque de l'État de Bavière.

Raasch, C. and von Hippel, E. (2012). Modeling interactions between user and producer innovation: User-contested and user-complemented markets. In *DRUID*, page 37, Copenhagen, Denmark.

Rallet, A. and Rochelandet, F. (2011). La régulation des données personnelles face au web relationnel: une voie sans issue? *Réseaux*, 167(3):17–47.

Raymond, E. S. (1999). *The cathedral and the bazaar: musings on Linux and open source by an accidental revolutionary*. O'Reilly & Associates, Inc., Sebastopol, CA.

Raymond, E. S. and Young, R. M. (2001). *The cathedral and the bazaar : musings on linux and open source by an accidental revolutionary*. O'Reilly, Sebastopol, Calif.

Reed, D. P. (1999). That Sneaky Exponential—Beyond Metcalfe's Law to the Power of Community Building. *Context magazine*, 3.

Reed, D. P. (2001). The law of the pack. *Harvard Business Review*, 79(2):23–24.

Richard, E. (2010). Internet et les nouveaux langages de la relation client. *Réalités industrielles*, Nov:10–14.

Riggs, W. and von Hippel, E. (1994). Incentives to innovate and the sources of innovation: the case of scientific instruments. *Research Policy*, 23(4):459–469.

Roberts, J. A., Hann, I.-H., and Slaughter, S. A. (2006). Understanding the Motivations, Participation, and Performance of Open Source Software Developers: A Longitudinal Study of the Apache Projects. *Management Science*, 52(7):984–999.

Robinson, D. K., Rip, A., and Mangematin, V. (2007). Technological agglomeration and the emergence of clusters and networks in nanotechnology. *Research Policy*, 36(6):871–879.

Rochet, J.-C. and Tirole, J. (2003). Platform Competition in Two-sided Markets. *Journal of the European Economic Association*, 1(4):990–1029.

Rochet, J. C. and Tirole, J. (2004). Two-sided markets: an overview. *Institut d'Economie Industrielle working paper*.

Rogers, E. M. (1962). *Diffusion of innovations*. Free press, New York, Etats-Unis.

Rogers, E. M. (2003). *Diffusion of innovations*. Free Press, New York.

Roman, D. (2009). Crowdsourcing and the question of expertise. *Communications of the ACM*, 52:12–12.

Rosenbloom, R. S. (2000). Leadership, capabilities, and technological change: The transformation of NCR in the electronic era. *Strategic Management Journal*, 21(10/11):1083–1103.

Rosenbloom, R. S. and Spencer, W. J. (1996). *Engines of innovation : U.S. industrial research at the end of an era*. Harvard Business School Press, Boston, Mass.

Rybalko, S. and Seltzer, T. (2010). Dialogic communication in 140 characters or less: How Fortune 500 companies engage stakeholders using Twitter. *Public Relations Review*.

Sahal, D. (1981). Alternative conceptions of technology. *Research Policy*, 10(1):2–24.

Sawhney, M. S. (1998). Leveraged High-Variety Strategies: From Portfolio Thinking to Platform Thinking. *Journal of the Academy of Marketing Science*, 26(1):54–61.

Saxenian, A. (1994). *Regional advantage : culture and competition in Silicon Valley and Route 128*. Harvard University Press, Cambridge, Mass.

Saxenian, A. (2000). Origins and Dynamics of Production Networks in Silicon Valley. In Kenney, M., editor, *Understanding Silicon Valley : the anatomy of an entrepreneurial region*, pages 141–162. Stanford University Press, Stanford, Calif.

Schlombs, C. (2008). Engineering International Expansion: IBM and Remington Rand in European Computer Markets. *Annals of the History of Computing, IEEE*, 30(4):42–58.

Schumpeter, J. A. (1939). *Business Cycles: A Theoretical, Historical and Statistical Analysis of the Capitalist Process*, volume 1939 of *Business Cycles: A Theoretical, Historical, and Statistical Analysis of the Capitalist Process*. McGraw-Hill.

Schurman, E. and Brutlag, J. (2009). Impact, Performance related changes and their user. In *Velocity Web Performance and Operations Conference*.

Segrestin, B. (2005). Partnering to explore: The Renault-Nissan Alliance as a forerunner of new cooperative patterns. *Research Policy*, 34(5):657–672.

Segrestin, B. (2006). *Innovation et coopération interentreprises: comment gérer les partenariats d'exploration?* CNRS Ed., cnrs editi edition.

Segrestin, B., Lefebvre, P., and Weil, B. (2002). The role of design regimes in the coordination of competencies and the conditions for inter-firm cooperation. *International Journal of Automotive Technology and Management*, 2(1):63–83.

Sen, R., Subramaniam, C., and Nelson, M. L. (2008). Determinants of the Choice of Open Source Software License. *Journal of Management Information Systems*, 25(3):207–240.

Shah, S. K. (2003). *Community-based innovation & product development : finding from open source software and consumer sporting goods*. PhD thesis, Massachusetts Institute of Technology.

Shah, S. K. (2006). Motivation, governance, and the viability of hybrid forms in open source software development. *Management Science*, 52(7):1000.

Shah, S. K. and Tripsas, M. (2007). The accidental entrepreneur: the emergent and collective process of user entrepreneurship. *Strategic Entrepreneurship Journal*, 1(1-2):123–140.

Shapiro, C. and Varian, H. R. (1999a). Corporate Strategy - The Art of Standards Wars. *California management review.*, 41(2):8.

Shapiro, C. and Varian, H. R. (1999b). *Information rules: a strategic guide to the network economy*. Harvard Business Press.

Shaw, B. (1985). The Role of the Interaction between the User and the Manufacturer in Medical Equipment Innovation. *R&D Management*, 15(4):283–292.

Shuen, A. (2008). *Web 2.0: A strategy guide*. O'Reilly Media, Inc.

Siggelkow, N. (2007). Persuasion With Case Studies. *Academy of Management Journal*, 50(1):20–24.

Simard, C. and West, J. (2006). Knowledge networks. In *Open Innovation: Researching a new paradigm*, chapter 11, pages 239–240. Oxford University Press, Oxford ; New York.

Simon, H. A. (1965). *The shape of automation for men and management*. Harper & Row, New York.

Singer, J. and Piluso, J. (2010). Five tools to prepare for future discontinuities. *Strategy & Leadership*, 38(1):17–21.

Smith, D. K. and Alexander, R. D. (1988). *Fumbling the future: how Xerox invented, then ignored, the first personal computer, William Morrow & Co.* William Morrow & Co, New York.

Soh, P.-H. (2009). Network patterns and competitive advantage before the emergence of a dominant design. *Strategic Management Journal*, 31(4):438–461.

Sojer, M. and Henkel, J. (2010). Code Reuse in Open Source Software Development: Quantitative Evidence, Drivers, and Impediments. *Economic Policy*, 11(12):868–901.

Soulier, E. (2005). Le storytelling: concepts, outils et applications. *Revue française de gestion*, 159(31):247–266.

Spector, P. (2001). Research Methods in Industrial and Organizational Psychology: Data Collection and Data Analysis with Special Consideration to International Issues. In Anderson, N., Ones, D. S., and Sinangil, H. K., editors, *Handbook of industrial, work and organizational psychology*, chapter 1, pages 10–26. SAGE Publications, London.

Stallman, R. (2002). *Free software, free society : selected essays of Richard M. Stallman*. Free Software Foundation, Boston, MA.

Stay, J. (2008). *FBML essentials: [Facebook Markup Language fundamentals]*. O'Reilly, Beijing.

Sterling, C. H. and Kittross, J. M. (2002a). *Stay tuned: a history of American broadcasting*. Lawrence Erlbaum Assoc. Publ., Mahwah, NJ [u.a.

Sterling, C. H. and Kittross, J. M. (2002b). *Stay tuned: a history of American broadcasting*. Lawrence Erlbaum Assoc. Publ., Mahwah, NJ [u.a.

Stern, N. (1979). The BINAC: A Case Study in the History of Technology. *IEEE Annals of the History of Computing*, 1(1):9–20.

Sturgeon, T. J. (2000). How Silicon Valley came to be. In Kenney, M., editor, *Understanding Silicon Valley : the anatomy of an entrepreneurial region*, chapter 2. Stanford University Press, Stanford, Calif.

Suarez, F. F. and Cusumano, M. A. (2010). The role of services in platform markets. In Gawer, A., editor, *Platforms, Markets and Innovation*, pages 77–98. Edward Elgar Publishing, Cheltenham, UK ; Northampton, MA.

Suárez, F. F. and Utterback, J. M. (1995). Dominant designs and the survival of firms. *Strategic Management Journal*, 16(6):415–430.

Suh, N. P. (1990). *The principles of design*, volume 226. Oxford University Press New York;.

Sutanto, J., Tan, C.-H., Battistini, B., and Phang, C. W. (2011). Emergent Leadership in Virtual Collaboration Settings: A Social Network Analysis Approach. *Long Range Planning*, 44(5 - 6):421 – 439.

Swanepoel, C., Lye, A., and Rugimbana, R. (2009). Virally inspired: A review of the theory of viral stealth marketing. *Australasian Marketing Journal (AMJ)*, 17(1):9–15.

Taleb, N. N. (2007). *The black swan : the impact of the highly improbable*. Random House, New York.

Tanenbaum, M. and Thomas, D. (1956). Diffused emitter and base silicon transistors. *Bell Syst. Tech J.*, 35(1):1–15.

Teece, D. J. (1986). Profiting from technological innovation: Implications for integration, collaboration, licensing and public policy. *Research Policy*, 15(6):285–305.

Teece, D. J. and Chesbrough, H. W. (2002). Organizing for Innovation: When Is Virtual Virtuous? *Harvard Business Review*, 80(1):127–135.

Terman, F. E. (1943). *Radio engineers' handbook*. McGraw-Hill, New York.

Thomke, S., von Hippel, E., and Franke, R. (1998). Modes of experimentation: an innovation process - and competitive - variable. *Research Policy*, 27(3):315–332.

Thomke, S. H. (1998). Simulation, learning and R&D performance: Evidence from automotive development. *Research Policy*, 27(1):55–74.

Tropp, H. (1999). John W. Mauchly (1907-1980) interview: January 10, 1973. page 72. Smithsonian Institution Press.

Tropp, H. S. (1987). The 20th Anniversary Meeting of the Association for Computing Machinery: 30 August 1967. *Annals of the History of Computing*, 9(3):249–270.

Turing, A. M., Gandy, R. O., and Yates, C. E. M. (2001). *Mathematical Logic*. Collected Works of A. M. Turing. Elsevier Science.

Turkle, S. (1984). *The second self : computers and the human spirit*. Simon and Schuster, New York.

Turner, F. (2006). *From counterculture to cyberculture: Stewart Brand, the Whole Earth Network, and the rise of digital utopianism*. Chicago, Ill., Etats-Unis, Royaume-Uni.

Usselman, S. (2007). Learning the Hard Way: IBM and the Sources of Innovation in Early Computing. In Lamoreaux, N. R., Sokoloff, K. L., and Janeway, W., editors, *Financing Innovation in the United States, 1870 to the Present*, pages 317–63. MIT Press, Cambridge (Mass.) ; London.

Utterback, J. M. and Abernathy, W. J. (1975). A dynamic model of process and product innovation. *Omega*, 3(6):639–656.

Utterback, J. M. and Suárez, F. F. (1993). Innovation, competition, and industry structure. *Research Policy*, 22(1):1–21.

Vaast, E. (2003). Recherche en gestion avec TIC et recherche sur la gestion avec TIC. *Revue française de gestion : hommes et techniques*, 29(146):43–58.

Valley Jr, G. E. (1985). How the SAGE Development Began. *IEEE Annals of the History of Computing*, 7(3):196–226.

Van Antwerp, M. and Madey, G. (2008). Advances in the SourceForge Research Data Archive (SRDA). In *Workshop on Public Data about Software Development WoPDaSD at The 4th International Conference on Open Source Systems Milan Italy*. Citeseer.

van de Ven, A. H. and Huber, G. P. (1990). Longitudinal Field Research Methods for Studying Processes of Organizational Change. *Organization Science*, 1(3):213–219.

Venkatraman, N. (2000). Five steps to a dot-com strategy: How to find your footing on the Web. *MIT Sloan Management Review*, 41(3):15–28.

Vera, D. and Crossan, M. (2005). Improvisation and Innovative Performance in Teams. *Organization Science*, 16(3):203–224.

Verganti, R. (1999). Planned Flexibility: Linking Anticipation and Reaction in Product Development Projects. *Journal of Product Innovation Management*, 16(4):363–376.

von Hippel, E. (1975). The dominant role of users in the scientific instrument innovation process. Technical report, Massachusetts Institute of Technology (MIT), Sloan School of Management, Cambridge, MA.

von Hippel, E. (1976). The dominant role of users in the scientific instrument innovation process. *Research policy*, 5(3):212–239.

von Hippel, E. (1977a). Industrial innovation by users: evidence, explanatory hypotheses and implications.

von Hippel, E. (1977b). The dominant role of the user in semiconductor and electronic subassembly process innovation. *Engineering Management, IEEE Transactions on*, (2):60–71.

von Hippel, E. (1978a). A customer-active paradigm for industrial product idea generation. *Research Policy*, 7(3):240–266.

von Hippel, E. (1978b). A review of data bearing on the users role in industrial innovation.

von Hippel, E. (1978c). Successful industrial products from customer ideas. *The Journal of Marketing*, pages 39–49.

von Hippel, E. (1983). *Generation and evaluation of novel product concepts via analysis of experienced users*. Cambridge, Mass. : Massachusetts Institute of Technology, Boston (Mass.).

von Hippel, E. (1986). Lead users: a source of novel product concepts. *Management science*, 32(7):791–805.

von Hippel, E. (1990). Task partitioning: An innovation process variable. *Research Policy*, 19(5):407–418.

von Hippel, E. (1994). "Sticky Information" and the Locus of Problem Solving: Implications for Innovation. *Management Science*, 40(4):429–439.

von Hippel, E. (1995). User learning, "sticky information," and user-based designs.

von Hippel, E. (1998). Economics of Product Development by Users: The Impact of "Sticky" Local Information. *Management Science*, 44(5):629–644.

Von Hippel, E. (2001). Learning from open-source software. *MIT Sloan management review*, 42(4):82–86.

von Hippel, E. (2005). *Democratizing innovation*. The MIT Press, Cambridge, MA.

von Hippel, E. (2007). Horizontal innovation networks - by and for users. *Industrial and corporate change*, 16(2):293–315.

von Hippel, E. (2010). Comment on 'Is open innovation a field of study or a communication barrier to theory development?'. *Technovation*, 30(11-12):555–555.

von Hippel, E. (2011). People Don't Need A Profit Motive To Innovate. *Harvard Business Review*, 89(11):36–37.

von Hippel, E. and Katz, R. (2002). Shifting Innovation to Users via Toolkits. *Management Science*, 48(7):821–833.

von Hippel, E. and von Krogh, G. (2003). Open source software and the private-collective innovation model. *Organization Science*, 14(2):209–223.

von Hippel, E. and von Krogh, G. (2006). Free revealing and the private-collective model for innovation incentives. *R&D Management*, 36(3):295–306.

von Krogh, G. (2011). Is open innovation a field of study or a communication barrier to theory development? A commentary. *Technovation*, 31(7):286.

von Krogh, G., Haefliger, S., and Spaeth, S. (2003a). Collective action and communal resources in open source software development: the case of freenet. *Academy of Management*.

von Krogh, G., Rossi-Lamastra, C., and Haefliger, S. (2012). Phenomenon-based Research in Management and Organisation Science: When is it Rigorous and Does it Matter? *Long Range Planning*, 45(4):277–298.

von Krogh, G., Spaeth, S., and Lakhani, K. R. (2003b). Community, joining, and specialization in open source software innovation: a case study. *Research Policy*, 32(7):1217–1241.

von Krogh, G., von Hippel, E., and von Krogh, G. (2006). The Promise of Research on Open Source Software. *MANAGEMENT SCIENCE*, 52(7):975–983.

Walker, J. (2001). *Rebels on the air: an alternative history of radio in America*. New York University Press, New York.

Wallis, J. J. and North, D. (1986). *Measuring the transaction sector in the American economy, 1870-1970*. University of Chicago Press, Chicago.

Walras, L. (1896). *Éléments d'économie politique pure, ou, Théorie de la richesse sociale*. F. Rouge.

Weil, T. (2010). Des histoires de la Silicon Valley. *Entreprises et histoire*, 58(1):129.

Wenaas, E. P. (2007). *Radiola: The Golden Age of RCA, 1919-1929*. Sonoran Publishing.

Wenger, E. (1998). *Communities of practice : learning, meaning, and identity*. Cambridge University Press, Cambridge, U.K.; New York, N.Y.

Wenger, E., McDermott, R. A., and Snyder, W. (2002). *Cultivating communities of practice : a guide to managing knowledge*. Harvard Business School Press, Boston, Mass.

Wenger, E. C. and Snyder, W. M. (2000). Communities of practice: The organizational frontier. *Harvard Business Review*, 78(1):139–145.

West, J. (2003). How open is open enough? Melding proprietary and open source platform strategies. *Research Policy*, 32(7):1259–1285.

West, J. and Lakhani, K. R. (2008). Getting Clear About Communities in Open Innovation. *Industry & Innovation*, 15(2):223.

West, J. and O'Mahony, S. (2008). The Role of Participation Architecture in Growing Sponsored Open Source Communities. *Industry & Innovation*, 15(2):145.

West, J., Vanhaverbeke, W., and Chesbrough, H. (2006). Open innovation: a research agenda. *Open Innovation: Researching a New Paradigm*, pages 285–507.

White, D. R. and Harary, F. (2001). The cohesiveness of blocks in social networks: Node connectivity and conditional density. *Sociological Methodology*, 31(1):305–359.

Whitla, P. (2009). Crowdsourcing and Its Application in Marketing Activities. *Contemporary Management Research*, 5(1).

Wiggins, A. and Crowston, K. (2010). Developing a conceptual model of virtual organisations for citizen science. *International Journal of Organisational Design and Engineering*, 1(1):148–162.

Willcocks, L. P. and Plant, R. (2001). Getting from bricks to clicks. *MIT Sloan Management Review*, 42(3):50–59.

Windrum, P. (2004). Leveraging technological externalities in complex technologies: Microsoft's exploitation of standards in the browser wars. *Research Policy*, 33(3):385–394.

Wirtz, B. W., Schilke, O., and Ullrich, S. (2010). Strategic Development of Business Models: Implications of the Web 2.0 for Creating Value on the Internet. *Long Range Planning*, 43(2-3):272–290.

Wise, T. (1966). IBM's $5,000,000,000 Gamble. *Fortune*, page 118.

Yates, J. (1997). Early interactions between the life insurance and computer industries: the Prudential's Edmund C. Berkeley. *Annals of the History of Computing, IEEE*, 19(3):60–73.

Yin, R. K. (2003). *Case study research: design and methods*. Applied Social Research Methods Series. Sage Publications, Thousand Oaks (Calif.), Etats-Unis, Royaume-Uni, Inde.

Zeitlyn, D. (2003). Gift economies in the development of open source software: anthropological reflections. *Research Policy*, 32(7):1287–1291.

Zhao, H., Seibert, S. E., and Hills, G. E. (2005). The Mediating Role of Self-Efficacy in the Development of Entrepreneurial Intentions. *Journal of Applied Psychology*, 90(6):1265–1272.

Zott, C. and Amit, R. (2010). Business Model Design: an activity system perspective. *Long Range Planning*, 43(2-3):216–226.

Quand les utilisateurs créent l'industrie : le cas des applications Web

Résumé : La présente étude explore le problème, curieusement peu étudié en gestion, de l'innovation dans le développement des applications Web. En partant de la distinction entre innovation par « l'usager » (U) et innovation par « l'industriel/entrepreneur » (E) et en utilisant une approche de « phenomenon-based research », elle identifie un troisième acteur, le « développeur » (D), dont l'action se positionne entre les deux autres. Trois figures d'acteur sont proposées pour le décrire : celle de l'usager-développeur (UD), de l'usager-développeur-entrepreneur (UDE) et du développeur-entrepreneur (DE).

Cette grille de lecture (U, D, E) est testée, dans la 2ème Partie, sur le cas de trois industries dont on peut retracer l'histoire, depuis leur genèse jusqu'à des stades de maturité avancés : l'industrie des radiocommunications, celle du PC, celle de l'ordinateur d'entreprise. L'importance du rôle joué par l'acteur D identifié se retrouve dans chacune de ces 3 industries. Les étapes de développement de chacune de ces industries peuvent être lues comme une suite d'innovations liées à des acteurs différents - successivement : UDs, UDEs, DEs - jusqu'à l'étape de proposition de rationalisations par des entreprises (Es) puis d'autonomisation de chacun trois acteurs U, D, E.

Dans la phase de maturité intermédiaire que connaît le Web où sont déjà apparues des entreprises spécialisées E mais où les développeurs D continuent à exister sous les formes riches de DEs ou d'UDEs, se pose la question pour les entreprises de savoir comment mobiliser et gérer l'activité de ces développeurs à des fins d'innovation. Trois formes de gestion sont identifiées dans la 3ème Partie. La première méthode consiste à favoriser l'auto-révélation de ces acteurs. La seconde consiste à mobiliser ces acteurs dans des actions éphémères visant l'exploration du potentiel d'un service donné. La dernière consiste à animer une communauté de développeurs qui utilisent déjà la technologie de l'entreprise pour les encourager à développer des applications sur la base de cette technologie.

Mots clés : développement des applications Web, innovation par les usagers, innovation par les industriels, entrepreneurs, développement industriel, potentiel technologique.

When users create industry: the case of Web-based applications

Abstract: The current study explores the curiously not much studied in management problem of innovation of contemporary Web-based applications. Starting from the distinction between user (U) and manufacturer/entrepreneur (E) innovation and using a phenomenon-based research approach, it identifies a third actor, the developer (D), whose action is found to lie in-between the two. Three actor figures are proposed for his description: user-developer (UD), user-developer-entrepreneur (UDE) and developer-entrepreneur (DE).

This interpretative framework (U, D, E) is tested in the second part on the cases of three industries, where it enables a tracing of their history, from their birth to their maturity: radio industry, PC industry and enterprise computer industry. The important role of D actor is identified in all three settings. Their development phases can be read as a sequence of innovations related to different actors, UDs, UDEs and DEs successively, until the proposition of a rationalisation by enterprises (Es) and leading to the independence of the three actors, U, D, E.

During the intermediate maturity phase of the Web, where expert enterprises Es have appeared, yet the developers Ds remain under the forms of DEs or UDEs, the question posed for enterprises is how to harness their activity for innovation. Three management modes are identified in the third part. The first method consists in fostering the self-revelation of these actors. The second consists in harnessing their action using ephemeral settings for the exploration of the potential of a given service. The last consists in animating a community of developers already using the enterprise's technology to encourage them in developing applications on the basis of this technology.

Keywords: user innovation, manufacturer innovation, entrepreneurs, industrial development, technological potential.

MINES ParisTech

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY