



ISSN: 1286-4892

**Editors:**

Martin Evans, *U. of Toronto*

Bernard Forgues, *U. of Paris 12*

Thomas Loilier et Albéric Tellier 2004  
Comment peut-on se faire confiance sans se voir ?  
Le cas du développement des logiciels libres  
*M@n@gement*, 7(3): 275-306.

*M@n@gement* is a double-blind reviewed journal where articles are published in their original language as soon as they have been accepted.

Copies of this article can be made free of charge and without securing permission, for purposes of teaching, research, or library reserve. Consent to other kinds of copying, such as that for creating new works, or for resale, must be obtained from both the journal editor(s) and the author(s).

For a free subscription to *M@n@gement*, and more information:  
<http://www.dmsp.dauphine.fr/Management/>

© 2004 *M@n@gement* and the author(s).

# Comment peut-on se faire confiance sans se voir ? Le cas du développement des logiciels libres

---

Thomas Loilier . Albéric Tellier

Université de Caen Basse-Normandie  
Institut d'Administration des Entreprises  
eMail: loilier@iae.unicaen.fr

Université de Caen Basse-Normandie  
Institut d'Administration des Entreprises  
eMail: tellier@iae.unicaen.fr

Cette recherche porte sur la collaboration au sein des réseaux d'innovation distants. Ces équipes-projets sont constituées d'individus dispersés géographiquement, réunis temporairement, et utilisant les technologies de l'information et de la communication comme support de communication et de mise en place du projet. La littérature consacrée aux relations entre les acteurs d'un réseau fait de la confiance le principal mode de coordination. Or, il semble admis que cette confiance s'appuie essentiellement sur la connaissance de l'autre et le face-à-face. Notre objectif est d'étudier les conditions dans lesquelles la confiance peut être un mode de coordination quand il n'y a pas d'interaction directe sur le même lieu entre les acteurs du projet d'innovation. Pour répondre à cette question, nous analysons le fonctionnement d'équipes de développement de logiciels libres associées au projet Linux. Il apparaît que l'absence d'interactions directes et simultanées limite considérablement la confiance interpersonnelle. Cette absence est compensée par une confiance institutionnelle élevée mais aussi par un dispositif formalisé de contrôle, dont la combinaison permet d'assurer un niveau de performance élevé. Aussi, nous nous éloignons des approches privilégiant la confiance comme une alternative au contrôle pour préférer un point de vue intégrateur. En particulier, le contrôle sanction peut être utilisé sans démotiver les membres de la communauté Linux parce qu'il vient compléter un système global de contrôle qui s'apparente à un contrôle social.

---

## INTRODUCTION

On assiste depuis quelques années à un foisonnement des travaux sur la capacité des technologies de l'information et de la communication (TIC) à permettre des configurations nouvelles de réseau d'innovation (par exemple Howells, 1995). Différents outils permettent aujourd'hui des échanges écrits asynchrones (e-mail, FAQ) et synchrones (chat) mais aussi une communication visuelle synchrone à distance non codifiée (visioconférence, salle de maquettage 3D...). Il devient donc possible à des individus éloignés géographiquement de se coordonner par ajustement mutuel sans passer par une phase préalable de codification<sup>1</sup> (Rallet, 1997 ; Gallié, 2003). Volontiers qualifiés de e-réseaux (Loilier et Tellier, 2003), d'équipes virtuelles (Lipnack et Stamps, 1997 ; Potter, 2000) ou encore d'équipes virtuelles globales (Jarvenpaa et Leidner, 1999), ces équipes-projets sont constituées d'individus dispersés géographiquement, parfois issus d'organisations et de cultures différentes et réunis temporaire-

**1.** Les individus se contentent d'écrire, de parler ou de dessiner.

ment (pour la durée du projet). Ils utilisent principalement les TIC comme support de communication et de mise en place du projet global.

Notre ambition est de contribuer à une meilleure compréhension des modes de coordination des membres des réseaux d'innovation distants. La littérature consacrée aux relations entre les acteurs économiques, et plus particulièrement aux réseaux, fait de la confiance le principal mode de coordination des formes hybrides (Powell, 1987). Or, il semble largement admis que la confiance à l'œuvre dans les réseaux s'appuie au moins pour partie sur la connaissance de l'autre et le face-à-face. Notre objectif est donc de comprendre comment la confiance parvient à se développer au sein des organisations distantes sachant que ces dernières ne peuvent s'appuyer que très épisodiquement sur la proximité géographique—en particulier le face-à-face—au contraire des équipes-projets classiques.

De nombreux travaux ont été menés depuis quelques années sur le développement et le maintien de la confiance dans les équipes-projets dispersées géographiquement (O'Hara-Devereaux et Johansen, 1994 ; Iacono et Weisband, 1997 ; Lipnack et Stamps, 1997 ; Duarte et Snyder, 1999 ; Jarvenpaa et Leidner, 1999 ; Crisp et Jarvenpaa, 2000 ; Maznevski et Chudoba, 2000 ; Bos, Gergle, Olson et Olson, 2001 ; Kanawattanachai et Yoo, 2002 ; Panteli, 2003). Ces recherches s'attachent essentiellement à tester l'hypothèse centrale d'Handy (1995) selon laquelle la confiance nécessite des contacts entre les individus. Dès lors, les auteurs se focalisent sur la confiance entre les individus comme facteur clé de succès du travail à l'intérieur de ces équipes (Kanawattanachai et Yoo, 2002 : 189). En dépit d'apports indéniables, ces travaux nécessitent d'être enrichis sur deux points : les dimensions étudiées de la confiance et la prise en compte d'un possible enchevêtrement des mécanismes de coordination. D'une part, si la confiance peut naître de relations entre des individus, elle peut également se développer à des niveaux supérieurs, notamment des collectifs organisés, et reposer sur des compétences techniques ou une réputation morale. D'autre part, il peut être délicat d'analyser le rôle de la confiance dans ce type d'équipe sans intégrer les autres modes de coordination, notamment les dispositifs de contrôle qui peuvent pallier les difficultés du travail collaboratif distant. Cet article est donc construit autour de la question de la construction de la confiance : dans quelles conditions la confiance peut-elle être un mode de coordination quand il n'y a pas d'interaction directe sur le même lieu entre les acteurs du projet d'innovation ?

Pour y répondre, nous analysons le fonctionnement des équipes de développement des logiciels libres, plus particulièrement celles associées au projet Linux. La conception de ces logiciels libres, dont la réussite n'est aujourd'hui contestée par personne, est fondée sur une communauté de développeurs indépendants disséminés géographiquement et, le plus souvent, non intéressés d'un point de vue commercial. En somme, ces logiciels ont été développés par un réseau de développeurs qui se font confiance sans se voir ni se connaître per-

sonnellement et qui utilisent essentiellement les TIC comme mode de communication. Au sens strict, la communauté des logiciels libres n'est pas une équipe virtuelle globale mais un ensemble d'équipes virtuelles (un réseau d'équipes) travaillant chacune sur des projets distincts au sein d'un objectif partagé : la construction d'une offre libre intégrée en informatique (logiciels, serveurs, système d'exploitation...), alternative au modèle propriétaire développé notamment par Microsoft. Le niveau d'analyse retenu dans cette recherche est donc l'équipe-projet constituée d'individus *a priori* dispersés géographiquement. Au sein de ces équipes, les face-à-face sont quasi-inexistants, les acteurs ne se connaissant que *via* leur réputation au sein de la communauté ("*egoboo*") et leur activité sur les forums de discussion et les listes de diffusion. Il s'agit donc d'un cas extrême au sens d'Eisenhardt (1989) méritant une analyse approfondie de la production de la confiance en son sein.

La première partie de l'article permet de revenir sur la notion de confiance et d'aborder les spécificités du fonctionnement des réseaux d'innovation distants qui incitent à une vision enchevêtrée des mécanismes de coordination. La deuxième partie de ce travail présente le cas choisi et la méthodologie retenue. Enfin, la troisième partie analyse les conditions de production de la confiance dans ces équipes puis discute les résultats mis en évidence.

## **INNOVATION COLLABORATIVE, ORGANISATION DISTANTE ET CONFIANCE**

### UN RETOUR SUR LA NOTION DE CONFIANCE

Qu'est-ce que la confiance ? La question est simple, la réponse plus qu'ardue tant ce concept « est subtil, diffus et difficile à saisir » (Nooteboom, 1996 : 990). Au sein des recherches anglo-saxonnes, cette complexité s'est notamment traduite par une grande diversité sémantique qui ne simplifie pas la prise de contact avec cette notion (notamment Fukuyama, 1995 et Mayer, Davis et Schoorman, 1995). Outre la distinction entre *trust* et *confidence* maintenant largement discutée, de nombreux auteurs ont cherché à imposer des conventions sémantiques pour combattre cette complexité. Nooteboom (2002) prend par exemple le parti de distinguer la confiance au sens large (*reliability*) de la confiance stricte liée au calcul de l'intérêt personnel ("*real*" *trust*). La subtilité de cette notion se retrouve aussi fort logiquement dans l'étude de ce que n'est pas la confiance. Contrairement aux idées reçues, Lewicki, McAllister et Bies (1998) montrent que confiance et défiance ne sont pas opposées mais bien détachées. Autrement dit, au sein d'une même relation, ces deux comportements peuvent tout à fait coexister. La confiance est donc une notion aux multiples facettes difficilement réductible.

Elle peut être définie comme « l'anticipation qu'un partenaire à l'échange ne s'engagera pas dans un comportement opportuniste, même en présence d'incitations compensatrices de court terme et d'une incerti-

tude sur les bénéfiques à long terme » (Chiles et McMackin, 1996 : 85). Cette analyse donne cependant une vision économique de la confiance. Le recours à ce mode de coordination en complément de l'autorité et de la hiérarchie est guidé par des motifs d'efficacité (atteinte des objectifs du réseau) et d'efficience (au moindre coût). Cette approche s'avère primordiale mais pas unique : la confiance n'est pas simplement économique mais aussi sociologique et psychologique. Rooks, Raub, Selten et Tazelaar (2000) ont montré que les caractéristiques intrinsèques d'une transaction (investissement spécifique, ampleur et incertitude) ne suffisent pas à expliquer les efforts de management et la confiance accordée par chacune des deux parties impliquées dans la transaction. Pour les comprendre, il faut tenir compte de l'encastrement social de la transaction dans sa dimension temporelle (répétitivité des échanges passés), réticulaire (relation avec un tiers et/ou d'autres firmes) et institutionnelle (existence d'institutions sociales qui tiennent compte de conventions et d'engagements crédibles).

L'une des distinctions les plus communes concerne ensuite le niveau de la confiance. Elle permet, dans la lignée des travaux de Luhmann (1979) et Giddens (1990), de dissocier la confiance interpersonnelle (*personal trust*) de la confiance systémique (*system trust*). La première caractérise la confiance placée dans les individus, la seconde celle relative à un système dans son ensemble (par exemple le système bancaire). Cette dernière transcende l'expérience personnelle ou les relations de face-à-face et n'implique pas la croyance qu'un individu ou un groupe d'individus soit digne de confiance. Si l'on affine l'analyse, la confiance interpersonnelle peut être décomposée en deux dimensions : intentionnelle et de compétence (Sako, 1991)<sup>2</sup>. La première concerne la croyance qu'un individu respectera ses engagements sans faire preuve d'opportunisme, la seconde qu'il en détient les capacités notamment en termes de formation et d'expérience professionnelle. Enfin, il est fréquent d'opposer deux formes de confiance personnelle : l'une est fondée sur les normes et l'autre sur le calcul et l'intérêt. La première associe la confiance à une fiabilité dans le comportement (l'individu qui fait confiance pense que l'autre respectera les normes). La confiance calculatoire revient à donner à la confiance une dimension rationnelle, chacun cherchant alors à maximiser son utilité espérée. Nous avons déjà souligné que cette approche strictement économique nous semblait réductrice. En outre, elle n'ajoute rien à l'analyse classique de prise de décision en économie<sup>3</sup>.

Finalement, il ressort de ces différents travaux plusieurs distinctions porteuses de sens qui invitent à employer le pluriel : de la confiance, il devient plus judicieux de parler des confiances. Il apparaît cependant que la nature et les caractéristiques de la confiance dépendent de son mode de construction (Mangematin, 1999 : 31). Dans certains cas, ce mode va permettre l'émergence d'une confiance partagée par l'ensemble d'une communauté. Dans d'autres situations, la confiance se limitera à des relations interpersonnelles. Dans un second temps, la nature de la confiance produite par le mode de construction va influencer sur la coordination des activités. En d'autres termes, le rôle de la

**2.** Mayer *et al.* (1995) distinguent trois dimensions : la capacité, la bienveillance et l'intégrité. La première renvoie aux aptitudes et compétences des parties en présence, la deuxième à leurs manifestations de dispositions positives et la dernière à leur qualité de caractère (honnêteté, fidélité, esprit d'ouverture...). Cependant, la proximité d'acceptation des deux dernières dimensions invite assez logiquement à leur rapprochement.

**3.** Nous ne faisons ici que reprendre l'argument développé par Williamson (1993) sur l'inutilité du concept de confiance calculatrice qui n'est jamais que le pendant de l'opportunisme.

confiance dans la coordination des activités innovatrices dépend de sa nature et donc de ses conditions de production. L'analyse de son efficacité dans des réseaux d'innovation ne peut donc pas se faire sans référence à ses modes de production (Mangematin, 1999 : 32).

Si l'on choisit de focaliser son attention sur les différents mécanismes de production de la confiance, la distinction opérée par Zucker (1986) s'avère alors centrale. Elle distingue trois formes de confiance selon leur mode de production : la confiance *intuitu personæ* (*characteristic based trust*), relationnelle<sup>4</sup> (*process-based trust*) et institutionnelle (*institutional based trust*) comme le précise le **Tableau 1**.

La confiance *intuitu personæ* naît des caractéristiques personnelles des individus. Ceux-ci peuvent par exemple appartenir à une même ethnie, famille ou encore religion. Ces caractéristiques, qui ne peuvent être produites à volonté, sont exogènes à la relation des acteurs. La confiance relationnelle est en revanche inséparable de la relation proprement dite. Elle est finalement issue du savoir que l'on peut détenir sur l'autre grâce à des actions répétées (loyauté passée, logique de don/contre don...) ou des informations, provenant d'un tiers, relatives à sa fiabilité (réputation par exemple). Ces deux formes de confiance sont avant tout interpersonnelles.

La confiance institutionnelle est d'une autre nature. Systémique, elle peut exister entre individus sans que ceux-ci ne se connaissent ou n'aient d'interactions directes les uns avec les autres. Cette confiance caractérise celle que l'on place dans les institutions formelles comme par exemple les lois. Selon Zucker (1986), il s'agit d'une reconstruction de la confiance produite localement par les acteurs qui s'avère extérieure au contexte de l'échange. Elle peut prendre deux formes : un ensemble de signaux (par exemple une marque, un diplôme, la norme ISO...) émis par l'un des protagonistes qui réduit le champ de ses comportements possibles ou l'intrusion d'un tiers dans la relation qui peut notamment rassurer les acteurs sur le résultat de cette relation (par exemple une compagnie d'assurance).

Au-delà de ces différents modes de construction, la confiance est un mode de coordination qui, lorsqu'il est utilisé dans les réseaux d'innovation, présente des caractéristiques prononcées.

**4.** Que l'on traduit parfois par confiance interactive.

**Tableau 1.** Les différents modes de production de la confiance\*

Modes de production/ Mécanismes de la confiance	Fondements de la confiance	Exemples
Confiance <i>intuitu personæ</i>	Caractéristiques propres d'un individu (la confiance est donc ici attachée à une personne)	Famille, communauté, ethnie, culture, religion...
Confiance relationnelle	Echanges passés ou attendus, réputation, don/contre don	Loyauté, engagement...
Confiance institutionnelle	Une structure sociale formelle garantissant les attributs d'un individu ou d'une organisation	Règles, code éthique, standards professionnels, normes, marques...

\*: Adapté de Zucker (1986).

## LES MODALITES DE COORDINATION DES ACTEURS DANS LES RESEAUX D'INNOVATION

Le réseau d'innovation peut se définir comme un ensemble coordonné d'acteurs hétérogènes (laboratoires privés ou publics, entreprises, clients, fournisseurs, organismes financiers...) qui participent activement et collectivement à la conception, à l'élaboration, à la fabrication et à la diffusion d'une innovation (d'après Maillat, 1996 : 84). Une des caractéristiques premières du réseau est qu'aucun des acteurs qui le compose ne dispose *a priori* de l'intégralité des actifs indispensables au projet, chacun recherchant alors l'acquisition d'actifs complémentaires (Teece, 1987). Au delà de cette caractéristique majeure maintenant bien connue, les réseaux d'innovation présentent des spécificités qui interpellent la confiance et qui conduisent à introduire des modalités de coordination originales.

Dans le cas du processus d'innovation, un certain nombre d'actifs spécifiques ne préexistent pas à la décision de s'engager dans le projet. Ces actifs spécifiques dits endogènes (Boissin, 1999), se construisent en marchant, au fil du processus d'innovation<sup>5</sup>. Cette co-construction s'observe particulièrement au sein des communautés d'innovation où la mise en commun d'actifs complémentaires donne lieu à un apprentissage collectif (le "faire avec") qui peu à peu devient un actif spécifique de première importance. Le cas le plus radical est sans conteste celui où le volet émergent (Mintzberg et Waters, 1985) du projet d'innovation est si prégnant que la communauté d'innovation ne peut savoir *ex ante* quels seront les actifs spécifiques qu'elle va réellement développer, ce qui peut notamment entraîner des modifications dans la configuration du réseau à mesure que le projet prend forme (l'enrôlement d'un partenaire nouveau par exemple). Accepter de participer à un tel réseau revient à s'engager dans un processus dont on ne peut *a priori* évaluer les coûts et les bénéfices pour chacun des participants puisqu'il s'avère difficile d'imaginer les résultats du travail collaboratif. Planque (1991) et Maillat (1998) notent ainsi que les acteurs d'un réseau d'innovation sont amenés à investir dans le projet avant même d'être certains de réussir et qu'ils procèdent ensuite par essais-erreurs et réorientations successives. Dès lors, il est crucial de pouvoir s'engager avec des partenaires de confiance qui feront de leur mieux pour arriver à des résultats. La logique d'échange entre les acteurs du réseau d'innovation est celle du don/contre don : ce que donne chaque acteur au reste de la communauté (compétence technique, réputation, information stratégique...) ne fait pas l'objet d'une compensation immédiate mais d'une compensation différée dont la nature n'est pas définie au moment de l'échange (Ferrary, 2002). Ce système permet le développement de la confiance si les échanges sont équitables c'est-à-dire s'ils « consistent à aider le partenaire lorsqu'il en exprime le besoin et inversement, à ce qu'il fasse de même lorsque l'occasion s'en présente » (Bouty, 1999 : 10).

L'incertitude qui pèse sur l'aboutissement des projets d'innovation conduit les acteurs à mobiliser les partenaires du réseau par des accords très informels. Mais pour limiter les risques d'appropriation

5. Certaines compétences humaines (routines individuelles ou organisationnelles) ou physiques (nouveaux procédés, nouvelles machines, nouveaux produits...) vont se développer à mesure de l'avancement du processus d'innovation et doivent donc être considérées comme la résultante du travail coopératif.

unilatérale, plus largement les comportements opportunistes, les acteurs du réseau doivent socialiser leurs échanges, c'est-à-dire les inscrire dans un groupe social qui a ses règles de fonctionnement, ses coutumes, ses rites... D'une part, cette socialisation accroît le coût des trahisons en augmentant le coût économique (exclusion des projets futurs) de coûts symboliques et sociaux (exclusion des manifestations propres au groupe social) ; d'autre part, elle assure l'optimalité de la régulation par le don puisque le comportement opportuniste sera sanctionné même s'il n'y a pas de contrat formel. La sanction est dans ce cas sociale et non légale : les informations sur le comportement opportuniste seront diffusées au sein du réseau et inciteront chacun des membres à refuser toute nouvelle collaboration avec le tricheur (Ferrary, 2002).

Ainsi, si la confiance est un mode de coordination possible du réseau, il est néanmoins nécessaire de disposer d'outils de résolution de conflits, de dispositifs de sanction, de définition des engagements.... Même si le réseau utilise plutôt les normes de réciprocité (logique du don/contre don) et les effets de réputation, il mobilise également le contrat et la routine. En effet, tous ces dispositifs de coordination ne sont pas exclusifs. En d'autres termes, une forme de gouvernance (le réseau, le marché et la hiérarchie) utilise la plupart du temps une combinaison d'outils de coordination enchevêtrés (Bradach et Eccles, 1989).

Cet enchevêtrement des mécanismes de coordination se retrouve dans les travaux qui ont mis en avant les limites de la confiance comme mode unique de coordination. Dans les équipes virtuelles, la confiance construite est généralement considérée comme fragile (Crisp et Jarvenpaa, 2000) et des dispositifs de contrôle, même simplifiés, sont indispensables pour éviter les mauvaises surprises. Brulhart et Favoreu (2003 : 10) notent que la faiblesse ou l'absence de contrôle génère des incertitudes qui se traduisent pour les acteurs par une perte de repère et un accroissement de la déviance. Il s'avère ainsi indispensable de mettre en place des sécurités contractuelles qui conforteront chacune des parties dans l'idée que chacun des membres du réseau adoptera un comportement coopératif (Poppo et Zenger, 2002). Même si chaque acteur pose l'hypothèse que les autres membres ont la volonté réelle de coopérer et que les comportements opportunistes seront ainsi quasi-absents, il est impératif de disposer d'une règle de réciprocité qui assure l'équité des transactions (Josserand, 2001 : 19). Dans le cas du réseau d'innovation, cette règle est celle de l'exclusion des individus qui ne se révèlent pas dignes de confiance. L'existence de cette règle, qui agit comme un garde-fou, sécurise les acteurs, incite au comportement coopératif (par exemple la transmission d'information) lequel, en retour, alimente la confiance au sein du réseau. Ainsi, non seulement le contrôle et la confiance sont des modes de coordination complémentaires mais ils s'influencent mutuellement (Goold et Campbell, 1987).

Outre la confiance et le contrôle, deux autres modes de coordination peuvent être mobilisés dans le cadre de l'enchevêtrement : le contrat et le type de technologie à travers notamment le concept de modulari-

té. Contrat et confiance doivent être compris comme des modes de coordination complémentaires (Brousseau, 2001), la confiance permettant tout à la fois de gérer la relation au quotidien et de pallier l'incomplétude dudit contrat. Cette complémentarité est exacerbée dans le cadre des activités d'innovation où l'initiative, difficile à prendre en compte *a priori* dans un engagement contractuel, est un facteur clé de succès. La confiance permet ainsi de gérer efficacement et à moindre coût la contractualisation. En retour, les contrats peuvent être des générateurs de confiance grâce à leur contenu (création de garanties crédibles) mais aussi en tant que processus. Le processus de contractualisation s'avère en effet générateur de confiance grâce à sa valeur symbolique et aux signaux coopératifs générés qui initient et entretiennent les « fragiles conjectures de confiances » (Brousseau, 2001 : 77).

La technologie est, à travers ses caractéristiques, également assimilable à un mode de coordination. Mangematin (1996) montre notamment comment celle-ci peut exercer une influence sur le mode de division du travail. La notion de modularité est ici centrale. Une technologie est dite modulaire lorsqu'elle permet l'intégration de différents sous-systèmes (les modules) dans des combinaisons variées sans changer l'architecture globale du système (Langlois et Robertson, 1992). La modularité du système permet une autonomie réelle des équipes et une possibilité de tests rapides de solutions alternatives favorisant un apprentissage de type essai-erreur. En terme de coordination, le réseau d'équipes est faiblement couplé, chaque équipe pouvant travailler de manière indépendante (le besoin d'ajustement entre partenaires est limité) en concentrant son attention sur des composants ou modules particuliers. Par ailleurs, si les technologies développées sont complémentaires, la valeur de la technologie dans son ensemble est supérieure à la somme des valeurs de chaque module, cette particularité créant une incitation intrinsèque à l'innovation (Mangematin, 1996).

## LA PRODUCTION DE CONFIANCE DANS LES RÉSEAUX DISTANTS D'INNOVATION

Les actes de confiance au sein des réseaux prennent la forme d'engagements qui, dans les projets d'innovation, s'intègrent dans une dialectique de dons et contre dons. Elle introduit une réciprocité dans l'échange déjà étudiée dans les relations entre les acteurs de l'innovation en général (Bouty, 1999, Ferrary, 2002) et ceux des logiciels libres en particulier (Loilier, 2002). Le mécanisme du don analysé notamment par Mauss (1950) en anthropologie, puis par Perroux (1960) en économie, se décompose en trois séquences : donner, recevoir puis rendre. Il convient bien à l'acte innovateur puisqu'il est lui-même un pari : il ne suppose aucun retour certain. Celui qui reçoit le don peut choisir de l'accepter ou de le refuser. S'il l'accepte, il va à son tour donner pour rééquilibrer la relation : il rend. Après évaluation de ce contre don, un nouveau cycle peut alors être enclenché. On assiste alors à un processus d'engagement progressif qui construit la

confiance comme le montrent Gomez, Korine et Masclef (2003) dans le cas de l'alliance Renault-Nissan. La rationalité du don est ainsi ambivalente dans la mesure où :

— tout don suppose la confiance puisque celui qui donne se trouve dans l'impossibilité d'évaluer *a priori* la valeur de l'éventuel contre don. Faire un don est donc un acte incertain qui peut être éloigné de la rationalité économique stricte ;

— le don n'est pas désintéressé dans la mesure où il présuppose un contre don. Le don « fournit aux individus des motivations personnelles qui permettent la contribution de tous au bon déroulement des échanges au niveau collectif » (Douglas, 1989 : 111). Perroux (1960) a d'ailleurs montré que, sous certaines conditions, la logique du don peut tout à fait renforcer l'ordre marchand.

Le problème posé par la distance entre les acteurs de l'innovation réside dans la difficulté de rencontre en face-à-face ou, plus généralement, de connaissance personnelle de l'autre. Comment parier sur la valeur du contre don quand on ne connaît pas l'autre ? Même si les TIC s'avèrent des moyens de communication puissants, il est largement admis que la construction de la confiance interpersonnelle passe avant tout par les face-à-face et les échanges directs entre les acteurs. Aussi cette forme de confiance est peu développée dans les réseaux distants.

Ceux-ci mobilisent davantage la confiance système et tout particulièrement la confiance institutionnelle pour pallier l'anonymat des acteurs. Cette dernière permet en effet, comme l'a démontré Zucker (1986), de se détacher des protagonistes en garantissant soit l'identité et la qualité de l'intermédiaire soit le respect de la qualité *via* des normes. Il est important de noter ici que cette confiance de niveau supérieur ne remplace par la confiance interpersonnelle (puisque au final ce sont bien les individus qui échangent) mais permet de la générer en socialisant l'échange. Celui-ci ne s'effectue plus hors contexte mais devient encadré et c'est cet encastrement qui produit la confiance. Si les acteurs ne peuvent être proches par leur connaissance mutuelle, ils vont le devenir à travers la connaissance partagée d'un tiers ou d'une institution (règle, norme...) qui va redonner du lien social et donc de la proximité. Bien entendu, cette proximité est d'autant plus efficace que tous ont connaissance de l'institution et s'y conforment (Mangematin, 2003)<sup>6</sup>.

Comment parvient-on à créer un contexte favorable à ces actes de confiance ? En analysant les travaux qui ont cherché à caractériser les conditions nécessaires à la production de la confiance dans les organisations virtuelles (Handy, 1995), les coopérations technologiques (Rallet et Torre, 2001 ; Ingham et Mothe, 2003), les équipes virtuelles globales (Jarvenpaa et Leidner, 1999 ; Lurey et Raisinghani, 2001), les équipes de R&D virtuelles (Duarte et Snyder, 1999 ; Maznevski et Chudoba, 2000 ; Nemiro, 2001 ; Letaief et Favier, 2003) ou, plus largement, dans les formes hybrides (Nohria et Eccles, 1992 ; Mangematin, 1999 ; Moon et Sproull, 2000), il est possible de mettre en avant neuf conditions de production de la confiance. Le **Tableau 2** précise ces conditions.

6. Cook et Luo (2003), à la suite notamment de Ho et Weigelt (2001), montrent comment les normes peuvent être utilisées comme des vecteurs de confiance dans le cas des achats par Internet. Dans ce cas, elles permettent des transferts de confiance d'un tiers (l'organisme qui se porte garant de la transaction) vers un inconnu (en l'occurrence le vendeur). Ce processus de transfert n'est possible que si l'acheteur est capable d'identifier les sources de preuve de la confiance et d'établir l'existence d'un lien de confiance entre la source de la preuve et le partenaire inconnu.

La suite de cet article vise l'étude du fonctionnement d'équipes-projets de la communauté des logiciels libres, en particulier celles dédiées au projet Linux. Notre objectif est de vérifier la présence ou l'absence des conditions recensées afin d'analyser les modes de production (**Tableau 1**) de la confiance entre des acteurs "étrangers".

## PRESENTATION DU CAS ET DE LA METHODE DE RECHERCHE

La deuxième partie est l'occasion de revenir sur l'histoire des logiciels libres et de présenter le dispositif de sélection et de catégorisation des données secondaires.

### PRESENTATION DU CAS : LE DEVELOPPEMENT DES LOGICIELS LIBRES ET LE PROJET LINUX

Apparus au milieu des années quatre-vingt, les logiciels libres mobilisent aujourd'hui plusieurs milliers de programmeurs indépendants. L'analyse des liens unissant ces informaticiens géographiquement disséminés a conduit à retenir le terme de communauté ou de réseau

**Tableau 2.** Les conditions de production de la confiance dans les réseaux distants

Conditions	Principaux arguments avancés
1. Constitution et identification claire du groupe de travail autour d'un objectif commun	Les membres de l'équipe ne peuvent avoir confiance que dans des individus qu'ils connaissent, qu'ils ont vu travailler. Cette condition tend à limiter la taille des équipes d'innovation (Handy, 1995)
2. Définition des objectifs et des délais	La confiance nécessite que le travail soit limité dans le temps et/ou par un objectif clair (Jarvenpaa et Leidner, 1999). La confiance « illimitée » est irréaliste (Handy, 1995). La clarification des objectifs augmente par ailleurs l'efficacité et la créativité des équipes virtuelles (Lurey et Raisinghani, 2001 ; Nemiro, 2001). Globalement, moins le projet est structuré, plus difficile est la coopération à distance (Rallet et Torre, 2001)
3. Mise en place de mécanismes d'apprentissage	Le sentiment de confiance est étroitement lié à la capacité d'apprendre des autres en échange du travail accompli à l'intérieur du projet collectif (Handy, 1995 ; Ingham et Mothe, 2003)
4. Mise en place de possibilités de contacts entre les membres	Tout au long du projet, il doit être possible aux membres de se rencontrer (Handy, 1995), d'échanger (Jarvenpaa et Leidner, 1999 ; Nohria et Eccles, 1992). Ces liens interpersonnels favorisent en outre la créativité (Nemiro, 2001 ; Letaief et Favier, 2003). La planification des occasions de rencontre augmente l'efficacité du travail accompli (Maznevski et Chudoba, 2000)
5. Relations antérieures fondées sur des liens professionnels	Les acteurs ont tendance à faire confiance à des partenaires avec lesquels ils ont l'habitude de travailler. La proximité « organisationnelle » (Rallet et Torre, 2001), c'est-à-dire les relations antérieures fondées sur des liens professionnels, génère de la confiance
6. Définition des engagements et obligations de chacun	Chaque membre du projet doit connaître les engagements et obligations des partenaires (Handy, 1995)
7. Mise en place de procédures de sanctions et d'éviction	Quand un acteur du réseau ne réalise pas ce pour quoi il a été intégré, par « ruse » ou par manque de compétences, il doit être possible de l'évincer (Handy, 1995)
8. Définition d'unités de pilotage du projet qui ont autorité sur le reste du réseau	Le réseau peut fonctionner avec un « leader » unique ou une collection de « leaders ». Dans ce cas, les zones de compétences doivent être clairement définies (Handy, 1995)
9. Création de routines communes aux membres de la communauté	Les routines rappellent ce qui est tenu comme acquis (« background expectations »), balisent le champ des possibles (vision commune) et les interprétations des actions de l'autre (Mangematin, 1999)

communautaire pour caractériser ce modèle innovateur qui intrigue les observateurs et les chercheurs. L'histoire du logiciel libre est en fait indissociable de celle d'Unix, système d'exploitation développé à partir de la fin des années soixante dans les laboratoires Bell, filiale d'AT&T (Logerot, 2003). Etant en situation de monopole dans le secteur des télécommunications, AT&T s'était engagée auprès du gouvernement fédéral américain à ne pas se développer dans l'informatique (machines et logiciels). C'est la raison pour laquelle, dès 1975, la société a offert aux universités et centres de recherche un accès gratuit au code source d'Unix. Une des conséquences majeures de cette mise à disposition est que le développement d'Unix va être largement le fait de ses premiers utilisateurs qui vont proposer un nombre important de versions<sup>7</sup>. Cependant, le démantèlement d'AT&T en huit sociétés distinctes lui permet de commercialiser, en 1984, Unix System V et de mettre ainsi fin au libre accès au code source.

La même année, Richard Stallman, un ancien chercheur du laboratoire d'intelligence artificielle du Massachusetts Institute of Technology, décide de développer un système d'exploitation totalement libre qui sera compatible avec Unix. Il baptise d'ailleurs son projet "GNU" pour « *Gnu is Not Unix* ». En 1985, Stallman crée la Free Software Foundation pour, d'une part, récolter des fonds pour le projet GNU et, d'autre part, commercialiser les copies des logiciels libres accompagnées d'un ensemble de prestations annexes telles que l'édition papier de manuels d'utilisation, l'assistance technique, la mise à disposition du CD d'installation...

L'ambition de Stallman est de proposer une alternative à la logique propriétaire qui est alors en train de s'imposer dans le monde de l'informatique. Il définit un nouveau type de licence, la *GNU General Public Licence* (GNU-GPL), dans laquelle le propriétaire du code source accorde aux utilisateurs le droit de le copier, le distribuer et le modifier. Pour garantir ces niveaux de liberté, ces logiciels sont protégés par le *copyleft* (l'opposé du *copyright*) qui interdit de dissimuler le code source du logiciel ainsi que celui de tous ses dérivés. Chaque utilisateur peut ainsi, dans une logique globale de création de valeur, améliorer le logiciel qu'il a reçu (gratuitement ou non) et ainsi rendre à la communauté ce qu'elle lui a donné<sup>8</sup>. Progressivement, un certain nombre de projets, qualifiés *d'open source*, et de sites dédiés à la communauté GNU vont se développer. Leur objectif est de concevoir des programmes complémentaires et de faciliter le transfert d'expérience de chaque utilisateur. De même, différentes licences *open source* vont être proposées (*Lesser General Public License—LGPL*, *Berkeley Software Design—BSD...*) même si la GNU-GPL est de loin la plus utilisée (Mangolte, 2003 : 2).

En 1990, bon nombre d'éléments constitutifs du système d'exploitation de Stallman sont réalisés mais des retards ont été pris dans la conception du noyau du système. Or, en 1991, Linus Torvalds, étudiant finlandais de l'université d'Helsinki, modifie le système Unix pour qu'il fonctionne sur des micro-ordinateurs. Une fois les lignes et les codes diffusés sur Internet, il reçoit rapidement d'internautes programmeurs des améliorations qu'il intègre à son système d'exploitation qu'il a baptisé

**7.** C'est la raison pour laquelle il existe de nombreuses versions d'Unix, pas toujours compatibles, et que l'histoire de ce système d'exploitation est assez complexe. Le lecteur intéressé pourra se référer à l'ouvrage de Logerot (2003).

**8.** Tout utilisateur d'un logiciel libre dispose de la possibilité de l'acquérir gratuitement (don) mais est aussi invité à fournir un contre don qui peut prendre au moins trois formes : contribuer à la pérennité du produit en augmentant mécaniquement le nombre de ses adeptes ; tester le produit et ainsi faire bénéficier l'ensemble de la communauté des utilisateurs de l'expérimentation ; concevoir des programmes informatiques libres, fondés sur le *copyleft*, qui viennent renforcer l'offre *open source* (Loilier, 2002).

9. Linux correspond ainsi à une version modifiée du système d'exploitation GNU par substitution d'un noyau par un autre. Il serait donc plus juste de parler de système d'exploitation GNU/Linux, ce que réclame Stallman.

Linux (le X est une référence à Unix). En particulier, en combinant Linux avec des éléments de GNU, des programmeurs vont parvenir à un système libre complet pouvant fonctionner sur n'importe quelle machine<sup>9</sup>. En mars 1994, Torvalds publie la version 1.0 du noyau Linux (*Linux Kernel*) avec une aide en ligne, bientôt suivie par d'autres versions de plus en plus abouties et comportant de plus en plus de fonctionnalités.

Linux est aujourd'hui un système d'exploitation constitué d'éléments d'origines diverses, notamment développés par la communauté *open source*. Le noyau du système est un fichier chargé en mémoire lors de l'initialisation de la machine qui contient l'ensemble des pilotes permettant d'assurer l'interface entre les applications utilisées et la machine. Le noyau est constamment disponible sous deux versions : une version dite de production (à utiliser) et une version dite de développement destinée à être améliorée. Linus Torvalds est la seule personne apte à approuver les modifications apportées au noyau. Le centre névralgique de la communauté Linux est le site <http://www.kernel.org> (Logerot, 2003 : 27-31). En juillet 2002, Linux Kernel (version 2.5.25) représentait plus de trois millions de lignes de codes et 2.263 contributeurs identifiés (Ghosh et David, 2003). En 2003, Linux est le système d'exploitation de 13,7 % des serveurs informatiques dans le monde (Ferrary et Vidal, 2004 : 2).

La deuxième vague de développement de l'environnement Linux a consisté à proposer un catalogue de produits liés de qualité. La gamme des logiciels libres est aujourd'hui assez étoffée : interface graphique, traitement de texte, tableur, traitement d'images, opérations graphiques, gestion de base de données... Mais il serait erroné de penser que la logique du projet GNU/Linux est strictement philanthropique et communautaire : elle s'apparente plutôt à une logique hybride, associant dons et services marchands. Autour de la communauté des logiciels libres gravitent diverses entreprises à vocation commerciale. Après l'arrivée de nombreux acteurs de l'informatique tels qu'IBM, Intel, Dell, HP et des distributeurs spécialisés, les sociétés de services et d'ingénierie informatique (SSII) ont elles aussi franchi le pas. En offrant des prestations sur mesure, elles ont fait exploser le marché : Linux, logiciel libre d'inspiration communautaire a commencé à devenir un véritable enjeu économique.

Il existe aujourd'hui plusieurs sous-communautés distinctes indissociables de projets très clairement identifiés, parfois concurrents : projet OpenOffice (développement d'une suite bureautique concurrente de Microsoft Office), projet Apache (serveur Web libre), Sendmail (agent de messagerie)... Plus de 72.000 projets de logiciels libres étaient référencés sur le site Sourceforge.net en décembre 2003. La taille et la durée de vie de ces équipes-projets, qui constituent la communauté des logiciels libres, sont très variables puisque la plupart d'entre elles disparaît lorsque les objectifs du projet sont atteints. À l'intérieur de cette communauté, les sources utilisées nous permettent d'analyser plus précisément le fonctionnement des équipes dédiées au système d'exploitation Linux, à son noyau (le Kernel) et au projet Freenet (logiciel réseau de type *peer-to-peer*).

## METHODE DE RECUEIL ET DE TRAITEMENT DES DONNEES

Notre observation s'appuie sur une ré-interprétation de travaux menés sur la communauté des logiciels libres. Il semble désormais largement admis de produire des connaissances en s'appuyant exclusivement sur des données secondaires. L'article de Weick (1993) sur l'incendie de Mann Gulch, qui s'appuie exclusivement sur l'ouvrage de Maclean, *Young Men and Fire*, en est une illustration flagrante. Le travail du chercheur devient alors un exercice de ré-interprétation des matériaux empiriques produits par d'autres. Les développements suivants ont pour objectif, d'une part, de justifier le recours à ce type de matériaux et de présenter la manière dont les données ont été sélectionnées, et d'autre part, de préciser les opérations de catégorisation et de codage qui nous permettent d'analyser les données et de tirer des enseignements du cas.

## CHOIX DE LA METHODE ET SELECTION DES DONNEES

Le choix de la ré-interprétation peut être justifié à deux niveaux : l'objet étudié et la variété des données disponibles. Le cas des logiciels libres (et en particulier celui du projet Linux) fait l'objet d'une attention soutenue de la part des chercheurs en organisation. Celle-ci se traduit ces dernières années par une quantité de données collectées<sup>10</sup> très importante matérialisée notamment par des numéros spéciaux de revues réputées (en particulier celui de *Research Policy* en 2003), des articles réguliers dans ces mêmes revues et des revues dédiées (en particulier *First Monday—Peer-reviewed Journal on the Internet*). La densité de ce matériau empirique incite à le réutiliser, à retravailler son interprétation. Ensuite, il est instructif de noter que le nombre élevé d'études empiriques s'accompagne d'une grande variété dans les objectifs des recherches, le mode de production des données et leur nature. Celle-ci garantit une vision globale assez fidèle de l'objet "réseau des logiciels libres" et permet ponctuellement d'utiliser les techniques de triangulation pour s'assurer de la robustesse des interprétations des enquêtes.

Concrètement, quatre sources de données principales ont été privilégiées : le projet FLOSS (2002), l'analyse de Ghosh et David (2003), l'étude empirique de Krishnamurthy (2002) et enfin l'analyse mono-cas de von Krogh, Spaeth et Lakhani (2003). Elles sont présentées en détail en **Annexe** et ont été ponctuellement renforcées par d'autres travaux académiques menés sur *l'open source*, systématiquement mentionnés dans le paragraphe consacré aux résultats de l'analyse. En outre, différents sites de la communauté des logiciels libres, des articles de presse et des ouvrages grand public ont été ponctuellement utilisés.

L'utilisation de données secondaires implique cependant une procédure stricte de sélection du matériau empirique. Il est en effet indubitable que la qualité des données utilisées s'avère le meilleur gage de la qualité même de la ré-interprétation proposée. Stewart (1984) pro-

**10.** Cette masse importante de données est sans doute aussi liée à la bonne volonté des acteurs eux-mêmes qui placent la transparence au cœur de leur activité et qui font preuve de beaucoup d'intérêt pour les analyses des chercheurs en sciences sociales. Il faut aussi y voir ici une illustration de la composition de cette communauté : elle est avant tout alimentée par les chercheurs des centres de recherches universitaires en informatique qui accueillent donc avec bienveillance les "intrusions scientifiques" de leurs collègues des sciences sociales.

pose un cadre systématique d'évaluation des sources de données secondaires en six points. Il s'agit alors pour le chercheur-utilisateur d'être capable de répondre sans difficulté majeure aux six questions détaillées dans le **Tableau 3** ; cette facilité de réponse étant un indicateur de la qualité des données considérées.

L'**Annexe** montre clairement que les quatre sources principales de données secondaires utilisées permettent des réponses claires et précises à au moins cinq des six questions posées par Stewart (1984) et s'avèrent donc des sources de données de haute qualité.

## CATEGORISATION ET CODAGE DES DONNEES

L'objectif de la recherche est de comprendre comment des équipes-projets composées d'individus distants géographiquement parviennent à créer et maintenir entre eux un sentiment de confiance. Les conditions recensées précédemment (**Tableau 2**) ont été utilisées comme des catégories *a priori* (Huberman et Miles, 1991) suffisamment précises pour ne pas nécessiter de redéfinitions ultérieures (Lincoln et Guba, 1985) et présentées sous forme de thèmes. Ce choix du thème comme étendue de la catégorie est lié à la méthode retenue pour l'étude de cas (comparaison de données secondaires) et aux matériaux utilisés (des articles de recherches aux objectifs variés) (Allard-Poesi, Drucker-Godard et Ehlinger, 2003 : 461).

Le contenu des différents textes utilisés a été ensuite découpé en unités d'analyses, elles-mêmes classées dans les catégories définies. Les unités d'analyse retenues sont des portions de phrases, des phrases entières, voire des groupes de phrases se rapportant à un même thème. Il s'agit ainsi d'unités de sens (Allard-Poesi, 2003) dont l'utilisation est cohérente avec l'objectif de la recherche qui est de décrire, comprendre et analyser le fonctionnement d'équipes-projets distantes. La taille des unités d'analyse a été validée par les deux critères de Lincoln et Guba (1985 : 345) : l'unité d'analyse sélectionnée doit aider à développer une compréhension, à faire sens au regard des questions de recherche posées ; elle doit être interprétable en l'absence d'informations additionnelles. L'inférence retenue pour lier les unités retenues aux catégories est l'inclusion (l'unité X est un type de catégorie Y). L'opération d'attribution d'une unité d'analyse à une caté-

### **Tableau 3.** Un cadre d'évaluation systématique des données secondaires\*

- 
1. Quel était le but de la collecte primaire ?
  2. Qui était responsable de la collecte ?
  3. Quelle information a été recueillie ?
  4. Quand l'information a-t-elle été recueillie ?
  5. Comment a-t-on obtenu l'information ?
  6. L'information est-elle corroborée par d'autres sources ?
- 

\*: Stewart (1984).

gorie se fait sans interprétation. Il s'agit donc d'un codage ouvert (Strauss et Corbin, 1990) ou descriptif (Huberman et Miles, 1991). Finalement, les documents sélectionnés ont fait l'objet « d'analyses thématiques qualitatives » (Bardin, 1993 : 148) destinées à vérifier la présence ou l'absence des catégories en tenant compte du contexte dans lequel le texte a été rédigé (notamment les objectifs de l'article et les contraintes éditoriales) (Allard-Poesi *et al.*, 2003 : 460-463). Le matériau empirique a été traité et synthétisé au moyen d'une métamatrice non ordonnée (Huberman et Miles, 1991). Ce protocole a été testé par une procédure de double codage (Weber, 1990). Il s'agissait de vérifier la définition des unités d'analyses et leur classification dans les catégories (fiabilité inter-codeurs) et de traiter les divergences. Le résultat de ce codage descriptif et son analyse sont présentés ci-après.

## **LA CONFIANCE DANS LA COMMUNAUTE DES LOGICIELS LIBRES : RESULTATS ET DISCUSSION**

Dans cette troisième partie, les résultats issus de la catégorisation des données sélectionnées sont tout d'abord présentés. Ils font ensuite l'objet d'une discussion qui est l'occasion de revenir sur la question initiale et de s'interroger sur la validité externe du cas.

### **LES CONDITIONS DE PRODUCTION DE LA CONFIANCE DANS LA COMMUNAUTE DES LOGICIELS LIBRES : RESULTATS**

Pour structurer notre analyse, nous avons distingué les conditions de l'émergence d'un comportement coopératif (conditions 1 à 5) des mécanismes de contrôle et de pilotage qui permettent les sanctions et facilitent le fonctionnement de la communauté (conditions 6 à 9).

#### **LE DEVELOPPEMENT DE LA CONFIANCE AU SEIN DE LA COMMUNAUTE**

Les deux premières conditions [1 ; 2]<sup>11</sup> mettent en exergue un besoin de structuration des projets d'innovation distants. Sur ce point, l'analyse du fonctionnement de la communauté Linux montre tout d'abord que la taille des équipes de développeurs est assez faible [1]. Dans la version 1.0 de Linux Kernel, 76,6 % des modules ont été développés par des équipes de moins de 10 personnes (Ghosh et David, 2003). La multitude des petites équipes est permise par la grande modularité du projet qui permet la spécialisation, le développement sans contrainte de coordination (Moon et Sproull, 2000) puisque c'est l'architecture générale du système qui assure la cohérence de l'ensemble. D'ailleurs, le nombre de développeurs travaillant sur plusieurs modules est très faible. Dans toutes les versions de Linux Kernel, plus de 70 % des auteurs travaillent sur un seul module (Ghosh et David, 2003). Plus généralement, seulement 5 % des développeurs de logiciels libres sont impliqués simultanément dans 6 projets ou plus, 56 % travaillant sur seulement un ou deux projets (Ghosh, Glott, Krieger et Robles, 2002). Si les listes de diffusion (*mailing-lists*) ont pour mission

**11.** Les numéros entre crochets renvoient au codage des catégories effectué dans le Tableau 2.

première d'informer les membres sur des questions techniques, elles jouent également un rôle prépondérant dans l'organisation des projets et la répartition des tâches [2]. La liste du noyau Kernel compte plus de 3.500 destinataires et des listes dédiées à des éléments particuliers du noyau (*subsystems*) se développent (Hertel, Nierder et Herrmann, 2003). Au delà, des sites centralisent les projets et les classent en fonction du stade de développement. Par exemple, Sourceforge.net est une plate-forme qui a été créée dans le but de référencer les projets *open source* et de faciliter la collaboration entre les développeurs en offrant notamment des outils de développement et en gérant des forums de discussion (Garcia et Steinmueller, 2003 : 10).

Les trois conditions suivantes [3 ; 4 ; 5] sont relatives à la nature des relations entre les membres. L'utilisation de standards de communication (*netiquette guidelines*), la publication systématique des travaux, l'archivage des questions posées (FAQ) et des pages de résolution de problèmes facilitent l'accès à l'information et constituent de véritables mécanismes d'apprentissage [3] très largement utilisés : chaque projet de logiciel libre a en moyenne deux forums de discussion propres et deux listes de diffusion (Krishnamurthy, 2002). Il faut cependant souligner que la modularité réduit considérablement les besoins de communication dans le travail de développement. Pour Ghosh et David (2003), il est faux de croire que dans le projet Linux (et plus largement les projets *open source*) les développements sont véritablement collaboratifs. La plupart des modules sont conçus par un ou deux développeurs. Pourtant, les auteurs multiplient les contacts électroniques avec les membres de la communauté [4]. Dans la version 2.0.30 de Linux Kernel, 75 % échangent des informations avec plus de 50 membres ; 30 % ayant même plus de 150 interlocuteurs (Ghosh et David, 2003). En d'autres termes, le travail technique n'impose pas des échanges fréquents et pourtant ces échanges ont lieu. Ce besoin de communiquer avec des membres de la communauté, non exigé par des contraintes techniques liées au développement, renforce la thèse selon laquelle les acteurs cherchent à socialiser les échanges.

La particularité du cas Linux vient plutôt de la manière dont ces échanges se réalisent [4]. Même si des manifestations sont organisées dans le monde entier pour permettre aux membres de la communauté de se rencontrer physiquement, les acteurs privilégient des moyens de communication électroniques. La dissémination des équipes étant totale, et la taille des équipes et le nombre de projets intégrés par les développeurs étant limités, les acteurs ne semblent pas exploiter des liens professionnels dans la constitution des équipes. Cela ne signifie pas pour autant l'absence d'évaluation des compétences des membres ni de hiérarchie. D'une part, tout auteur dont la contribution est intégrée au code source peut indiquer son nom, son adresse électronique et l'organisation qui le soutient. Ce système de citation permet une reconnaissance par les pairs d'une expertise technique, augmente la réputation de l'auteur et facilite son intégration dans d'autres projets [5] (Harhoff, Henkel et von Hippel, 2003). 94 % des développeurs de logiciels libres déclarent signer leurs contributions (Ghosh *et al.*, 2002). Dans certains projets (par exemple

Debian), celui qui se voit attribuer la responsabilité d'un module a le privilège d'obtenir une adresse électronique officielle (de type pseudonyme@debian.org) pour signaler son statut au reste de la communauté et augmenter sa réputation (Ferrary et Vidal, 2004 : 13). D'autre part, les observations qui ont été menées sur le fonctionnement des forums de discussion et des listes de diffusion (von Krogh *et al.*, 2003) montrent que les nouveaux entrants sont observés pendant quelques semaines, voire quelques mois, avant d'être véritablement intégrés aux discussions. Pendant cette phase d'évaluation, véritable processus d'adhésion (*joining script*) aux valeurs et habitudes de la communauté (Garcia et Steinmueller, 2003), les participants doivent démontrer leurs compétences en offrant une contribution jugée significative, c'est-à-dire donner à la communauté avant de pouvoir recevoir<sup>12</sup>. Dans le cas du projet Apache, si tous les développeurs peuvent librement contribuer au projet, seuls ceux qui ont démontré des compétences spécifiques se verront attribuer, par un vote des fondateurs du projet, des responsabilités particulières (Markus, Manville et Agres, 2000 : 21).

Finalement, les cinq conditions au développement de la confiance sont bien mises en évidence dans la communauté Linux même si les contacts entre les individus se font essentiellement par l'intermédiaire des TIC. Il est ensuite important de s'interroger sur les outils de gestion de la confiance pouvant agir comme des mécanismes de contrôle et de pilotage (**Tableau 2**, conditions 6 à 9).

#### LE CONTROLE ET LE PILOTAGE DANS LES PROJETS OPEN SOURCE

Une des particularités de la communauté des logiciels libres vient de la communication systématique sur Internet des obligations, règles et autres normes de conduite auxquelles les membres doivent se soumettre [6] (Hertel *et al.*, 2003). Il faut tout d'abord souligner que, dans les premières années de la communauté, des efforts ont été menés par les fondateurs pour préciser les droits et obligations des membres. Il s'agissait tout d'abord de définir les conditions à remplir pour qu'un logiciel puisse être qualifié d'*open source*. L'*Open Source Definition* précise ainsi les "9 commandements de l'*open source*" devant être respectés dans tous les projets. (Müller, Yamagata, Wall et Dougherty, 2001). Les possibilités offertes aux utilisateurs de ces logiciels libres ont également été précisées<sup>13</sup>. Enfin, la manière de se comporter dans les forums de discussion, et notamment de poser des questions, a été réglementée. Ces règles de conduite sont compilées dans le document "*Comment poser les questions de manière intelligente*" disponible sur différents sites (par exemple le site français de Linux). Certains forums ont des pages dédiées aux obligations de leurs membres (*guidelines*).

L'esprit communautaire qui règne au sein de ces réseaux ne suffit pas à éviter les comportements opportunistes et les désaccords quant aux axes de développement à privilégier. L'opportunisme est contrôlé grâce à l'infrastructure technologique qui permet de signaler rapidement aux membres le non respect des règles de la communauté par

**12.** Edwards (2001) distingue ainsi les *learners*, nouveaux entrants qui souhaitent intégrer la communauté et les *insiders*, membres qui participent activement aux projets et qui jouissent d'une réputation technique.

**13.** Il en découle quatre niveaux de liberté : 0/l'exécution du logiciel est libre, quel que soit son usage ; 1/l'étude du fonctionnement du programme est libre et donc adaptable aux besoins de chacun (accès au code source) ; 2/la distribution du logiciel est libre ; 3/l'amélioration, la transformation et la communication du logiciel sont libres.

certaines membres [7]. Le succès de ce réseau serait en partie lié à la manière dont les pratiques d'échanges d'informations ont été encouragées et la réputation sanctionnée (Moon et Sproull, 2000). Les sanctions sont centrées sur la réputation. Trois niveaux de sanctions existent : 1/ "*Read the F... Manual*" : la personne est invitée à relire les ressources documentaires ; 2/ "*Death penalty*" : la personne est considérée comme indésirable et ignorée, au moins pour un temps, dans les forums de discussion ; 3/ "*Reputation die and grow hard*" : la personne indésirable est référencée sur les sites web accessibles à partir des moteurs de recherche. Il lui sera très difficile d'intégrer de nouvelles équipes ou des forums de discussion (Tayon, 2002). On peut ajouter ici que la mission proposée par Stallman a probablement facilité l'intégration culturelle. La communauté des logiciels libres s'est en effet constituée autour de l'objectif de proposer un modèle alternatif à celui des éditeurs propriétaires (notamment Microsoft). Cette mission et la présence d'un adversaire clairement identifié tendent à renforcer la cohésion. Les études menées sur les motivations des membres des projets *open source* ont d'ailleurs souligné la volonté affirmée de participer à la lutte contre les logiciels propriétaires et la dimension politique de la communauté (Bezroukov, 1999 ; Ghosh *et al.*, 2002 ; Hertel *et al.*, 2003).

Les désaccords sont fréquents sur le réseau et portent notamment sur les options techniques à privilégier (Garcia et Steinmueller, 2003). Même si la communauté Linux a été qualifiée de modèle "bazar" (Raymond, 1998), l'analyse de son fonctionnement montre la présence d'unités de pilotage qui ont autorité sur le reste du réseau [8]. Tout d'abord, la communauté est hiérarchisée avec un noyau stratégique (Lorenzoni et Baden-Fuller, 1995), la *Free Software Foundation*, qui joue un rôle de coordination en étant à l'origine de plus de 17 % de l'ensemble des projets développés par la communauté (Ghosh et Prakash, 2000). Cette fondation a également un rôle de contrôle puisqu'elle s'assure du respect de la licence GNU (notamment du *copy-left*), encourage les membres à lui communiquer les cas de violation de cette licence et prend en charge les poursuites judiciaires (O'Mahony, 2003 : 1187-1188). Ensuite, les projets sont généralement pilotés par des administrateurs (en fait des programmeurs mainteneurs) en nombre limité qui contrôlent le développement et décident ou non d'intégrer les contributions nouvelles. Le cycle de développement d'un logiciel libre suit généralement les étapes suivantes (Edwards, 2001) : 1/un *leader* propose une version 0 d'un logiciel en précisant ses objectifs et en fournissant le code source ; 2/des contributeurs décident librement d'intégrer le projet (création d'une équipe virtuelle), téléchargent ces éléments et identifient les problèmes et les possibilités d'amélioration ; 3/les propositions sont adressées au *leader* et à l'équipe virtuelle via une liste de diffusion dédiée ; 4/les corrections sont discutées, parfois évaluées par des procédures de vote en ligne ; 5/le *leader* introduit les solutions validées et propose une version 1 en téléchargement, etc. L'étude de Krishnamurthy (2002) sur cent projets *open source* a montré qu'il y avait en moyenne deux administrateurs par projet (le mode étant égal à 1). Par exemple, Linus Torvalds, qui

se définit lui-même comme un “dictateur bénévole” (Hertel *et al.*, 2003 : 1161), est le seul qui puisse accepter l’intégration de nouveaux éléments (*patches*) dans le module Kernel. La communauté Linux est ainsi pilotée par diverses unités centrales qui sont à l’origine des projets et dont l’expertise technique est reconnue. Ces unités définissent les règles de base régissant les relations entre les membres, orientent les tâches réalisées à la périphérie et facilitent la mise en réseau en étant des points de rencontre entre les acteurs et des diffuseurs d’informations. Garcia et Steinmueller (2003) parlent à ce sujet d’autorité distribuée au sein de la communauté. Dans certains cas, le fondateur d’un projet délègue une partie de son autorité à différents administrateurs qui ont en charge le développement de modules liés. C’est ainsi que Linus Torvalds s’est entouré de “lieutenants de confiance” (*trusted lieutenants*) pour concevoir différents *patches* et assurer la sélection des propositions des membres de la communauté (Markus *et al.*, 2000 : 22). Il y a bien ici une collection de *leaders* avec des zones de compétences clairement définies [8]. L’autorité est cependant légitime et non statutaire puisqu’officiellement, tout individu peut librement intégrer ou quitter une équipe de développement<sup>14</sup>.

Finalement, la formalisation initiale des droits et obligations des membres, l’utilisation de standards de communication, la diffusion des pratiques répréhensibles et le contrôle réalisé par les unités de pilotage contribuent à créer des routines partagées [9] et, au-delà, permettent d’envisager le réseau Linux comme une communauté de pratique (Wenger, 1998 ; Loilier, 2002) structurée par un projet historique commun (le projet GNU), un engagement mutuel (le *copyleft* et les quatre niveaux de liberté de la communauté) et un répertoire partagé (ressources tangibles : prototypes, maquettes, ou intangibles : routines, procédures, symboles...).

Les neuf conditions recensées par la littérature (**Tableau 2**) ont donc été mises au jour dans le cas étudié. Elles font maintenant l’objet d’une discussion qui permet d’analyser la nature de la confiance finalement produite.

## NATURE ET ROLE DE LA CONFIANCE DANS LA COMMUNAUTE LINUX : DISCUSSION

### LES APPORTS DE LA RECHERCHE

Les éléments exposés ci-dessus offrent une perspective très analytique du processus de confiance au sein de la communauté des logiciels libres. Le premier apport de cette recherche passe par un retour explicite à la question fondatrice de notre réflexion : dans quelles conditions la confiance peut-elle être un mode de coordination quand il n’y a pas d’interaction directe sur le même lieu entre les acteurs ? Notre réponse, à la lumière de l’analyse du cas des logiciels libres, est la suivante : la logique du don/contre don qui gouverne les équipes d’innovation et la réputation dont peuvent bénéficier certains membres montrent qu’il peut exister une confiance relationnelle au sein d’une équipe. Toutefois, celle-ci s’avère insuffisante pour constituer un mode de coordination efficace. Celui-ci s’appuie sur une confiance institu-

**14.** Les projets *open source* respectent en effet la règle de la “bifurcation” (*forking*) : tout individu en désaccord avec les décisions prises par l’administrateur peut créer un nouveau projet pour exploiter d’autres choix techniques. Raymond (1999) note cependant que cette possibilité n’est quasiment jamais exploitée par les membres qui considèrent qu’elle serait de nature à affaiblir la communauté.

tionnelle grâce à des signaux émis par l'organisation (ici des règles et des codes éthiques principalement) qui sont complétés par des procédures de contrôle. Donc, il peut y avoir confiance sans se voir sous double condition de l'existence d'une confiance institutionnelle élevée et de la présence d'un dispositif de contrôle sanction. Les développements suivants reviennent sur cette double condition et sur la nature de la relation entre confiance relationnelle et confiance institutionnelle.

Il nous semble que la dissémination géographique des acteurs des projets *open source* limite la confiance relationnelle qui se développe avec la relation. Ce résultat va à l'encontre de Jarvenpaa et Leidner (1999) qui ont montré que la confiance personnelle peut être maintenue et développée dans les équipes virtuelles via les TIC. Cependant, les projets étudiés par ces auteurs diffèrent sensiblement de ceux analysés dans ce travail. D'une part, l'objectif à atteindre et les délais sont précisés d'emblée et ne nécessitent pas une co-création d'actifs spécifiques. D'autre part, les équipes constituées s'apparentent à des réseaux convergents, au sens de Callon, Laredo, Rabeharisoa, Gonard et Leray (1992). Dans ce type de situation, chaque participant a une connaissance précise de la composition de l'équipe et a la possibilité de contacter tous les membres. Ainsi, la densité du réseau relationnel est maximale puisque tous les éléments sont reliés entre eux par des liens directs (Aldrich et Zimmer, 1986). Or, dans la communauté des logiciels libres, la modularité des projets, les tâches assurées par les *leaders*, l'utilisation massive de moyens de communication asynchrone et la dispersion géographique ne permettent pas à chaque individu une connaissance précise de la constitution de l'équipe dédiée au projet et la multiplication des contacts avec chacun de ses membres. En d'autres termes, la densité du réseau relationnel est nettement plus faible. En revanche, les deux ressorts majeurs de la confiance (ressort moral ou technique) sont tout à fait présents au sein de la communauté. Ils prennent la forme d'une confiance institutionnelle (Zucker, 1986 ; Mangematin, 1999) dite aussi "confiance système" (Brousseau, Geoffron et Weinstein, 1997). Celle-ci peut être qualifiée de contextuelle : on ne fait plus confiance à l'autre (*trust*) mais à l'ensemble du contexte dans lequel s'insère la relation (*confidence*). La confiance repose sur la conviction que le partenaire respectera le *copyleft* (plus généralement les règles et normes sociales en vigueur) et non plus sur la personnalité des coopérants. La confiance attribuée à l'un des membres de la communauté n'est plus séparable de celle inspirée par le système.

Cette confiance système est-elle suffisante ? Autrement dit, les formes de confiance (relationnelle et institutionnelle) sont-elles substituables ? Zucker (1986) puis Greenwald et Stiglitz (1992) répondent plutôt par l'affirmative. Zucker (1986) montre notamment que la disparition de la confiance relationnelle est palliée par la production d'une confiance institutionnelle, la réciproque étant fautive. Toutefois, nos résultats semblent plutôt plaider en faveur d'une complémentarité des deux confiances dans la lignée de Mangematin (1996). Celui-ci montre clairement que les coopérations institutionnelles de recherche produisent de la confiance relationnelle, les chercheurs et ingénieurs développant

souvent des liens individuels privilégiés s'appuyant sur la logique de don/contre don. Nous nous inscrivons dans le même courant complémentariste : aucune des deux confiances n'est en soi suffisante, l'absence de l'une ou l'autre devant être palliée par des dispositifs explicites ou implicites.

Notre analyse offre un exemple éclairant de ce type de dispositif. L'absence (ou la quasi-absence) d'interactions directes et simultanées (en face-à-face ou en temps réel) limite considérablement la confiance relationnelle. Cette absence est certes compensée par une confiance institutionnelle élevée mais aussi (et surtout) par un dispositif formalisé de contrôle. C'est la combinaison de la confiance institutionnelle et du contrôle qui permet d'assurer un niveau de performance élevé au réseau distant. Aussi, nous nous éloignons des approches privilégiant la confiance comme une alternative au contrôle pour préférer un point de vue intégrateur. Dans des environnements de plus en plus turbulents, la confiance peut être complétée avec profit par des dispositifs formels (Goold et Campbell, 1987). Sitkin (1995) précise notamment l'importance des règles formelles et des procédures standardisées dans l'institutionnalisation de la confiance. Brulhart et Favoreu (2003) mettent à jour l'influence positive du contrôle sur la confiance dans 219 partenariats logistiques issus du secteur de l'agro-alimentaire. Précisément, le contrôle est ici composé de deux dimensions : le suivi et l'évaluation (formalisation de procédures) et la formalisation des accords (utilisation d'un contrat). Notre recherche permet donc d'étendre ce résultat à un cas exemplaire de développement d'une innovation de produit dans un secteur de haute technologie, l'informatique.

Elle permet aussi une autre extension des résultats issus du courant intégratif. Implicitement, ses auteurs se focalisent davantage sur le contrôle système producteur d'informations et de connaissances que sur le contrôle interprété comme un système de surveillance et de sanction. Ce contrôle dit rationnel est en effet négativement connoté. Les mécanismes formels de contrôle sont souvent perçus par ceux qui les subissent comme un manque de confiance du partenaire. En retour, les acteurs contrôlés réduisent eux-mêmes leur confiance vis-à-vis de leur partenaire (Argyris, 1952). Dans leur critique acerbe de l'économie des coûts de transaction, Goshal et Moran (1996) dressent un bilan sévère du contrôle rationnel. En s'appuyant sur une revue de la littérature, ils notent notamment :

- que le contrôle rationnel engendre un sentiment de défiance des contrôleurs vers les contrôlés ;
- qu'il menace la motivation et l'implication des contrôlés ;
- qu'il détériore l'image que les contrôlés se font d'eux-mêmes.

Or, nos résultats montrent que ce contrôle sanction peut être utilisé sans démotiver les intrapreneurs de la communauté Linux. Il existe clairement une procédure de sanction et d'éviction (*Reputation die and grow hard*) qui permet de limiter les risques qu'une personne indésirable parvienne à intégrer de nouvelles équipes et/ou des forums de discussion<sup>15</sup>. Ce contrôle *ex post* n'est toutefois efficace que parce qu'il vient compléter un système global de contrôle qui s'apparente à

**15.** Concrètement, cette personne est référencée sur les sites Web accessibles à partir des moteurs de recherche.

un contrôle social (Ouchi, 1979). Efficace dans les clans, cette forme de contrôle vise la création d'une intégration normative incitant les individus à internaliser les valeurs et les buts de l'organisation. Notre analyse semble montrer que, dans un réseau d'équipes virtuelles, le contrôle sanction peut être appliqué avec efficacité s'il est couplé à des dispositifs de contrôle social. La complémentarité confiance/contrôle peut donc être étendue au contrôle sanction. Il s'agit là d'un résultat novateur dans la mesure où, jusqu'à présent, cette forme de contrôle et la confiance étaient plutôt jugées antinomiques. Il nous semble que l'on peut émettre l'hypothèse que ce contrôle sanction est l'une des conditions de production de la confiance relationnelle au sein des équipes étudiées. Lorsqu'un individu opportuniste ou incompetent est détecté, il est évincé grâce au dispositif sanction. Il s'avère donc possible de faire confiance à un individu sur la base de sa réputation parce que celle-ci repose sur des mécanismes de sanction (Orléan, 1994 : 18).

#### LA VALIDITE EXTERNE DU CAS

Le second développement de cette discussion des résultats explore la validité externe de la recherche. Cette capacité à générer de la confiance en matière d'innovation sans se voir est-elle répliquable, réutilisable dans d'autres contextes ? Si oui, à quelles conditions ? Trois spécificités fortes du modèle de développement des logiciels libres doivent être soulignées ici : la faible proportion des connaissances tacites dans l'informatique, la systématisation de l'apprentissage expérimental et l'organisation particulière des équipes-projets.

Il est maintenant admis que la conception de l'innovation repose sur la mise en œuvre de savoirs à la fois tacites et formalisés. Le savoir formalisé (explicite, objectif) est une forme de connaissance non visqueuse<sup>16</sup> (von Hippel, 1994) puisqu'elle peut être transmise, codifiée sans perte d'intégrité. Le savoir tacite est par opposition une forme de connaissance impossible ou très difficile à communiquer par un discours écrit. En fait, le savoir formalisé est d'essence scientifique et échappe à son détenteur alors que la connaissance tacite est intimement liée à ce dernier (Reix, 1995). La proximité physique des acteurs est généralement considérée comme un moyen de diffusion de la partie non codifiable des connaissances et de limitation des coûts d'information (par exemple Belis-Bergouignan, 1997). Deux caractéristiques expliquent au moins pour partie le desserrement de la contrainte géographique. D'une part, l'informatique, et notamment le développement des logiciels, est un secteur dans lequel la part des connaissances tacites est considérée comme très faible. D'autre part, les acteurs ont pu exploiter trois éléments nécessaires à la codification (Cowan et Foray, 1997) : une grammaire communément acceptée, un vocabulaire partagé et un répertoire de phrases fabriquées à partir des deux éléments précédents. Ces trois éléments sont en effet explicitement présents au sein de la communauté des logiciels libres avec respectivement un langage de programmation commun (le langage C), des compétences en algorithmique partagées par l'ensemble des acteurs<sup>17</sup> et le noyau Kernel écrit par Linus Torvalds (Cohendet, Créplet et Dupouët, 2003). On comprend donc aisément que la confiance tech-

**16.** La "viscosité" (*stickiness*) d'une information tient compte de son coût d'acquisition, de transfert et d'utilisation par les autres acteurs du réseau. Cette viscosité est fonction de trois variables : la nature de l'information elle-même, la quantité d'information à échanger et les capacités intellectuelles des acteurs concernés.

**17.** L'étude menée dans le cadre du projet FLOSS a montré que 70 % des développeurs possèdent un diplôme universitaire et que 44,5 % sont des programmeurs et analystes programmeurs (Ghosh *et al.*, 2002).

nique n'ait pas besoin de face-à-face pour être présente au sein de la communauté dans la mesure où l'immense majorité des connaissances nécessaires à la bonne réalisation du projet est accessible à distance via Internet.

Ensuite, l'utilisation massive des TIC dans le processus innovateur a permis le développement de l'apprentissage entre les membres de la communauté avec en particulier la construction de sites de résolution de problèmes et d'archivage des questions posées (FAQ). Mais c'est sans doute l'apprentissage expérimental (Lundvall, 2000) qui est au cœur du processus de développement. Si cette capacité à expérimenter l'innovation délibérément auprès des utilisateurs pendant le processus de production n'est pas nouvelle<sup>18</sup>, elle a atteint avec les logiciels libres un niveau d'intensité très élevé. Celle-ci est liée au fait que, dès les débuts du projet, les développeurs étaient aussi les utilisateurs des applications produites et pouvaient donc en même temps identifier des problèmes et proposer les améliorations. Cette absence de séparation entre les innovateurs et les utilisateurs (Dang-Nguyen et Pénard, 1999 ; Loilier, 2002) a rendu ces apprentissages expérimentaux particulièrement efficaces grâce aux supports de communication fournis par les TIC, et ce malgré l'absence de face-à-face.

Enfin, l'organisation même des projets de la communauté *open source* est très particulière. Deux caractéristiques liées s'avèrent incontournables pour comprendre le fonctionnement des équipes virtuelles en général et la construction de la confiance en particulier : la taille des équipes et la modularité des projets. L'analyse très précise de Ghosh et David (2003) sur la communauté des développeurs du noyau Linux met en évidence la petite taille des équipes de développement au sein de ces projets. Pour une large part, elles ne dépassent pas cinq développeurs, un nombre significatif des modules étant même l'œuvre d'acteurs individuels<sup>19</sup>. Par ailleurs, plus de 70 % des développeurs ne participent qu'à un seul projet de développement au sein de la communauté. Ce processus d'innovation qui mobilise une multitude d'équipes de taille réduite n'est possible que parce que le projet Linux est modulaire. La modularité est directement issue des objectifs initiaux du projet de Stallman qui était de concevoir un système multi-tâches (autorisant le lancement simultané de plusieurs applications) (Mangolte, 2003 : 4). Cet objectif a impliqué une séparation stricte entre le noyau, qui gère les accès à la machine, et les applications utilisateurs qui n'accèdent pas directement aux ressources machines (Logerot, 2003 : 31). D'une part, les programmes peuvent ainsi être développés indépendamment les uns des autres. D'autre part, un effort a dû être effectué pour standardiser les liens (l'interface) entre les programmes et le noyau. C'est finalement cette interface standardisée qui assure la cohérence de l'ensemble, autorise l'indépendance des équipes et, plus largement, permet un développement du projet Linux sans véritable planification des tâches. On retrouve dans le cas des logiciels libres un phénomène déjà observé dans les autres branches de l'informatique, notamment le *hardware* (Baldwin et Clark, 1997 ; Langlois, 2002) : la modularité offre au concepteur de module une grande liberté dans la conception car il n'a pas à communiquer ses décisions aux concepteurs des autres modules

**18.** Voir notamment les travaux précurseurs de von Hippel (1986) à propos des *lead users*.

**19.** En 1997, les équipes composées de 2 à 5 développeurs ont mené 40 % de la totalité des projets de cette communauté, auxquels il faut ajouter 5 % de projets menés individuellement. En juillet 2002, ces projets menés par au plus cinq individus représentaient encore plus de 31 % du nombre total des projets développés. Par ailleurs, il est important de noter que cette utilisation massive d'équipes très réduites (au maximum composées de cinq acteurs) était plus importante aux débuts de la communauté : en mars 1994 (première version de Linux), ces équipes représentaient plus de 65 % des projets développés (Ghosh et David, 2003).

et à l'architecte du système. En revanche, la présence d'un *leader*, qui a le monopole sur les modifications du noyau, et qui ajoute les modules de manière incrémentale, permet d'inscrire l'intérêt de la coopération dans la technique. Le concepteur de modules doit en effet se conformer aux normes de la communauté et gagner sa confiance pour voir son développement intégré. Finalement, organisation des projets, taille limitée des équipes et modularité expliquent qu'à l'intérieur d'un projet le besoin de coordination est plutôt faible. Par conséquent, le besoin de confiance interpersonnelle (appréhendée comme un outil de coordination) est lui aussi plutôt faible. Autrement dit, si la communauté des logiciels libres s'est développée sans réel besoin de face-à-face, c'est donc, au moins pour partie, parce que son besoin de coordination à l'intérieur de chaque projet était lui-même de faible intensité.

## CONCLUSION

Au terme de cet article, il apparaît que la confiance à l'œuvre dans les équipes d'innovation distantes de la communauté des logiciels libres ne se fonde pas sur le développement des relations personnelles entre les individus mais relève d'un niveau supérieur : la confiance système ou institutionnelle. Cela ne signifie pas que la confiance interpersonnelle (notamment dans sa forme relationnelle) n'existe pas (la dialectique du don/contre don est bien au cœur des équipes étudiées) mais que son utilisation comme outil de coordination s'appuie sur le développement de la confiance institutionnelle. Toutefois, dans ce réseau distant, cette dernière ne s'avère pas suffisante pour construire une confiance interpersonnelle. Elle doit être complétée par l'utilisation de dispositifs de contrôle plus classiques alliant contrôle social et rationnel. Le modèle d'organisation des logiciels libres, souvent présenté comme révolutionnaire et anarchique, n'est donc pas si désordonné que cela. Même si l'on se trouve en présence d'une organisation en réseau, il demeure des liens hiérarchiques, c'est-à-dire la possibilité de donner des ordres et d'influer sur le comportement des autres membres (Josserand, 2001 : 18). La confiance à l'œuvre au sein de ces équipes n'est pas naïve. Les mécanismes de contrôle sont bien présents dans cette communauté à travers un contrôle social mais aussi, de manière plus surprenante, à travers un contrôle rationnel s'appuyant notamment sur des procédures de sanction très précises. Il s'avère donc possible de se faire confiance sans se voir pour innover en développant une confiance système à toute épreuve complétée par un système de contrôle rigoureux.

Ces résultats peuvent être rapprochés des travaux qui ont fait du territoire une solution de confiance qui facilite à la fois coordination, co-construction du résultat et socialisation des échanges (diminution de l'incertitude quant aux comportements des acteurs durant le processus d'innovation). Au sein d'une équipe-projet classique, la confrontation des points de vue, la divulgation d'informations stratégiques, plus largement les interactions fréquentes en face-à-face, facilitent la construction d'une confiance interpersonnelle et constituent sans

doute des moyens de contrôle informel des intentions véritables et des compétences des partenaires. En d'autres termes, confiance et contrôle informel se développent au sein de l'équipe. Les relations de face-à-face fournissent aux individus des occasions de montrer leurs compétences et leur capacité à donner autant d'informations qu'ils en reçoivent et d'approfondir leurs connaissances réciproques. Or, dans le cas d'un réseau distant, il n'est pas possible d'utiliser la proximité géographique pour générer de la confiance entre les partenaires. Ainsi, si les TIC permettent le développement de réseaux d'innovation distants, elle conduisent également à introduire des modalités d'organisation et de régulation totalement nouvelles, en particulier parce que les itérations, négociations et compromis entre les acteurs, indispensables à tout projet d'innovation, sont largement modifiés. Notamment, la confiance va naître en dehors de l'équipe, à un niveau supérieur, en référence à des pratiques, des normes partagées par une communauté qui incluent des procédures formelles de sanction.

Il serait toutefois très audacieux de tenter une généralisation de ce travail et ce pour au moins deux raisons. La première est liée aux spécificités fortes du cas des logiciels libres. D'abord, l'informatique est un secteur dans lequel la part des connaissances tacites est considérée comme faible (ce qui limite la nécessité du recours au face-à-face). Ensuite, les équipes virtuelles dédiées à l'*open source* peuvent utiliser de manière exacerbée l'apprentissage expérimental puisque les TIC constituent à la fois un moyen (actif disponible pour atteindre l'objectif du projet) et la production (actif co-construit au fil du temps) des utilisateurs-producteurs des projets. Enfin, au sein de ces équipes-projets, de taille réduite, le besoin de coordination est plutôt faible du fait de la grande modularité des projets.

La seconde raison est issue du choix méthodologique effectué dans cette recherche. En effet, la décision de porter essentiellement son attention sur le développement du système d'exploitation Linux, un indéniable succès, peut constituer un biais important. De très nombreux autres projets sont développés au sein de la communauté *open source* et tous n'aboutissent pas à une large diffusion. Il faut garder à l'esprit que cette communauté est aussi constituée d'équipes qui ont échoué, de logiciels qui ne seront jamais utilisés. En outre, de nombreux projets ne peuvent profiter d'une médiatisation aussi grande que Linux, en partie liée à l'histoire et à la personnalité de Linus Torvalds et d'un adversaire aussi clairement identifié. En effet, l'objectif de proposer un système d'exploitation alternatif à celui de Microsoft a sans aucun doute renforcé la cohésion de développeurs informatiques investis dès lors d'une véritable mission.

Ce travail nécessite donc d'être enrichi par des recherches futures sur un échantillon de projets plus représentatif de la communauté des logiciels libres. Il serait notamment intéressant d'utiliser des sites référençant les projets comme Sourceforge.net qui permettent d'exploiter des données primaires de type : date de démarrage, durée du projet, nombre, profils et coordonnées des développeurs... Il y a sans aucun doute matière à de nouvelles recherches fructueuses sur les liens qui unissent les membres de ce réseau si particulier.

**Note.** Les auteurs tiennent à remercier les trois évaluateurs de la revue pour la richesse de leurs observations et commentaires ainsi que Gérard Kœnig pour ses conseils, remarques et encouragements des plus stimulants.

**Thomas Loilier** est maître de conférences habilité à diriger les recherches à l'IAE de Caen Basse-Normandie. Au sein du CIME (Caen-Innovation-Marché-Entreprise), ses recherches portent sur le management stratégique. En particulier, elles se focalisent sur la gestion de l'innovation tant dans ses aspects organisationnels (management des équipes-projets) que stratégiques (politique coopérative de l'entreprise innovatrice).

**Albéric Tellier** est maître de conférences habilité à diriger les recherches à l'IAE de Caen Basse-Normandie. Les recherches qu'il mène au sein du centre de recherche de l'IAE (CIME) portent sur le management de l'innovation. Ses travaux se focalisent notamment sur la configuration et le fonctionnement des réseaux d'innovation, les situations de compétition technologique et les stratégies collectives d'innovation.

---

## REFERENCES

- Aldrich, H. E., and C. Zimmer 1986  
Entrepreneurship through Social Networks, in D. L. Sexton and R. W. Smiler (Eds.), *The Art of Science of Entrepreneurship*, New York: Ballinger, 3-23.
- Allard-Poesi, F. 2003  
Coder les données, in Y. Giordano (Ed.), *Conduire un projet de recherche : une perspective qualitative*, Colombelles : EMS, 245-290.
- Allard-Poesi, F., C. Drucker-Godard et S. Ehlinger 2003  
Analyses de représentations et de discours, in R. A. Thiétart (Ed.), *Méthodes de recherche en management*, 2ème édition, Paris : Dunod, 449-475.
- Argyris, C. 1952  
*The Impact of Budgets on People*, New-York: Controllershship Foundation.
- Baldwin, C. Y., and K. B. Clark 1997  
Managing in an Age of Modularity, *Harvard Business Review*, 75: 5, 84-93.
- Bardin, L. 1993  
*L'analyse de contenu*, Paris : PUF.
- Belis-Bergouignan, M. C. 1997  
Coopérations inter-firmes en R&D et contrainte de proximité : le cas de l'industrie pharmaceutique, *Revue d'Economie Industrielle*, 81: 3, 59-76.
- Bezroukov, N. 1999  
A Second Look at the Cathedral and the Bazaar, *First Monday*, 4: 12.
- Boissin, O. 1999  
La construction des actifs spécifiques : une analyse critique de la théorie des coûts de transaction, *Revue d'Economie Industrielle*, 90: 4, 7-24.
- Bos, N., D. Gergle, J. S. Olson and G. M. Olson 2001  
Being There versus Seeing There: Trust via Video, *Working Paper*, CREW, Ann Arbor: University of Michigan, School of Information.
- Bouty, I. 1999  
Décision individuelle d'échange au sein des réseaux informels : entreprise, chercheurs et communauté technologique, *Actes de la VIIIème Conférence de l'AIMS*, ECP, Chatenay-Malabry, 26-28 mai.
- Bradach, J., and R. Eccles 1989  
Markets versus Hierarchies: from Ideal Types to Plural Forms, *Annual Review of Sociology*, 15, 97-118.
- Brousseau, E. 2001  
Confiance ou contrat, confiance et contrat, in F. Aubert et J.-P. Sylvestre (Eds.), *Confiance et rationalité*, Versailles : INRA Editions, Les Colloques, 97, 65-80.
- Brousseau, E., P. Geoffron et O. Weinstein 1997  
Confiance, connaissances et relations inter-firmes, in B. Guillon, P. Huard, M. Orillard et J.-B. Zimmermann (Eds.), *Economie de la connaissance et organisations : Entreprises, territoires, réseaux*, Paris : L'Harmattan, 402-433.

- Brulhart, F., et C. Favoreu 2003  
Les modes de coordination et d'organisation des partenariats inter firmes: exploration du rôle et de l'impact respectifs du contrôle et de la confiance au travers du courant intégratif, *XIIème Conférence de l'AIMS*, Tunis, Les Côtes de Carthage, 4-6 juin.
- Callon, M., P. Laredo, V. Rabeharisoa, T. Gonard and T. Leray 1992  
The Management and Evaluation of Technological Programs and the Dynamics of Techno-economic Networks: the Case of the AFME, *Research Policy*, 21: 3, 215-236.
- Chanal, V. 2000  
La structuration d'un projet d'innovation par la communication électronique, *Actes de la IXème Conférence de l'AIMS*, Montpellier, 24-26 mai.
- Chiles, T. H., and J. F. McMackin 1996  
Integrating Variable Risk Preferences, Trust and Transaction Cost Economics, *Academy of Management Review*, 21: 1, 73-99.
- Cohendet, P., F. Créplet et O. Dupouët 2003  
Innovation organisationnelle, communautés de pratique et communautés épistémiques : le cas de Linux, *Revue Française de Gestion*, 29: 146, 99-121.
- Cook, P. C., and W. Luo 2003  
The Role of Third-Party Seals in Building Trust Online, *e-Service Journal*, 2: 3, 71-84.
- Cowan, R., and D. Foray 1997  
The Economics of Codification and the Diffusion of Knowledge, *Industrial and Corporate Change*, 6: 3, 594-622.
- Crisp, C. B., and S. L. Jarvenpaa 2000  
Trust over Time in Global Virtual Teams, *Annual Academy of Management Meeting*, Toronto, ON, August 4-9.
- Dang-Nguyen, G., et T. Pénard 1999  
Don et coopération dans Internet : une nouvelle organisation économique ?, *Terminal*, 80/81, 95-116.
- Douglas, M. 1989  
Il n'y a pas de don gratuit : Introduction à l'édition anglaise de *l'Essai sur le don* de M. Mauss, *Revue trimestrielle du Mauss*, 4: 2, 99-115.
- Duarte, D. L., and N. T. Snyder 1999  
*Mastering Virtual Teams: Strategies, Tools, and Techniques That Succeed*, San Francisco, CA: Jossey-Bass.
- Edwards, K. 2001  
Epistemic Communities, Situated Learning and Open Source Software Development, *Working Paper* for Epistemic Cultures and the Practice of Interdisciplinarity Workshop, Trondheim: Norges Teknisk-Naturvitenskapelige Universitet.
- Eisenhardt, K. 1989  
Building Theories from Case Study Research, *Academy of Management Review*, 14: 4, 532-550.
- Ferrary, M. 2002  
Pour une théorie de l'échange dans les réseaux sociaux, un essai sur le don dans les réseaux industriels de la Silicon Valley, in I. Huault (Ed.), *La construction sociale de l'entreprise : autour des travaux de Mark Granovetter*, Colombelles : EMS, 61-86.
- Ferrary, M., et P. Vidal 2004  
Les leçons de management de la communauté Linux, *XIIIème Conférence de l'AIMS*, Le Havre, 1-4 juin.
- Fukuyama, F. 1995  
*Trust: The social Virtues and the Creation of Prosperity*, New York: Free Press.
- Gallié, E. P. 2003  
Une grille d'analyse de l'usage des TIC dans les différentes étapes de la coopération technologique, *Sciences de la Société*, 59, 118-134.
- Garcia, J. M., and W. E. Steinmueller 2003  
The Open Source Way of Working: a New Paradigm for the Division of Labour in Software Development ?, *INK Research Working Paper*, SPRU, 1, Brighton: University of Sussex.
- Ghosh, R. A., R. Glott, B. Krieger and G. Robles 2002  
*Free/Libre and Open Source Software: Survey and Study – Part 4: Survey of Developers*, FLOSS Final Report, Maastricht : University of Maastricht, International Institute of Infonomics, <http://www.infonomics.nl/FLOSS/report/>
- Ghosh, R. A., and P. A. David 2003  
The Nature and Composition of the Linux Kernel Developer Community: a Dynamic Analysis, *Working Paper*, NSF Open Source Software Project, Stanford, CA: Stanford Institute for Economic Policy Research.
- Ghosh, R. A., and V. V. Prakash 2000  
Orbiten Free Software Survey, *First Monday*, 5: 7.
- Giddens, A. 1990  
*The Consequences of Modernity*, Cambridge, MA: Polity Press.
- Gomez, P.-Y., H. Korine et O. Masclef 2003  
Alliance stratégique et construction de la confiance : le cas Renault-Nissan, in V. Mangematin et C. Thuderoz (Eds.), *Des mondes de confiance*, Paris : CNRS Editions, 203-218.
- Goold, M., and A. Campbell 1987  
*Strategies and Styles: the Role of the Center in Managing Diversified Corporations*, New-York: Blackwell.
- Goshal, S., and P. Moran 1996  
Bad for Practice: a Critique of the Transaction Cost Theory, *Academy of Management Review*, 21: 1, 13-47.
- Greenwald, B., and J. E. Stiglitz 1992  
Information, Finance and Markets: the Architecture of Allocative Mechanisms, in V. Zamagni (Ed.), *Finance and the Enterprise*, London: Academic Press.
- Handy, C. 1995  
Trust and the Virtual Organization, *Harvard Business Review*, 73: 3, 40-50.

- Harhoff, D., J. Henkel and E. von Hippel 2003  
Profiting from Voluntary Information Spillovers: how Users Benefit by Freely Revealing their Innovations, *Research Policy*, 32: 10, 1753-1769.
- Hertel, G., S. Nierder and S. Herrmann 2003  
Motivation of Software Developers in Open Source Projects: an Internet-Based Survey of Contributors to the Linux Kernel, *Research Policy*, 32: 7, 1159-1177.
- Ho, T. H., and K. Weigelt 2001  
Trust Building Among Strangers, *Working Paper*, 99-008, Philadelphia, PA: University of Pennsylvania, Wharton Marketing Department.
- Howells, J. 1995  
Going Global: the Use of ICT Networks in Research and Development, *Research Policy*, 24: 2, 169-184.
- Huberman, A. M. et M. B. Miles 1991  
*Analyse des données qualitatives, recueil de nouvelles méthodes*, Bruxelles : De Boeck Université.
- Iacono, C. S., and S. Weisband 1997  
Developing Trust in Virtual Teams, in F. Nunamaker, Jr., and R. H. Sprague, Jr. (Eds.), *Proceeding of the 30th Hawaii International Conference on System Sciences*, Vol. 2, Piscataway, NJ: IEEE, 412-420.
- Ingham, M., et C. Mothe 2003  
Confiance et apprentissages au sein d'une alliance technologique, *Revue Française de Gestion*, 29: 143, 111-128.
- Jarvenpaa, S. L., and D. E. Leidner 1999  
Communication and Trust in Global Virtual Teams, *Organization Science*, 10: 6, 791-815.
- Josserand, E. 2001  
*L'entreprise en réseau*, Paris : Vuibert.
- Kanawattanachai, P., and Y. Yoo 2002  
Dynamic Nature of Trust in Virtual Teams, *Journal of Strategic Information Systems*, 11, 187-213.
- Krishnamurthy, S. 2002  
Cave or Community? An Empirical Examination of 100 Mature Open Source Projects, *First Monday*, 7: 6.
- Langlois, R. N. 2002  
Modularity in Technology and Organization, *Journal of Economic Behavior and Organization*, 49: 1, 19-37.
- Langlois, R. N., and P. L. Robertson 1992  
Networks and Innovation in a Modular System: Lessons from the Microcomputer and Stereo Component Industries, *Research Policy*, 21: 4, 297-313.
- Letaief, R., et M. Favier 2003  
Vers une meilleure compréhension de la créativité au sein des équipes virtuelles globales, GRH et TIC, *Journée d'étude et de recherche*, Laboratoire CREPA, Université de Paris Dauphine, mai.
- Lewicki, R. J., D. J. McAllister and R. J. Bies 1998  
Trust and Distrust: New Relationships and Realities, *Academy of Management Review*, 23: 4, 438-458.
- Lincoln, Y. S., and E. G. Guba 1985  
*Naturalistic Inquiry*, London: Sage.
- Lipnack, J., and J. Stamps 1997  
*Virtual Teams: Reaching across Space, Time and Organizations with Technology*, New York: Wiley.
- Logerot, P. 2003  
*Linux ou Windows ?*, Paris : Dunod.
- Loilier, T. 2002  
Gestion de l'innovation : quels enseignements tirer du cas des logiciels libres ?, *Finance-Contrôle-Stratégie*, 5: 3, 141-168.
- Loilier, T., et A. Tellier 2003  
Quelles configurations pour les réseaux innovateurs ?, in H. Laroche, P. Joffre et F. Fréry (Eds.), *Perspectives en management stratégique*, Tome 9, Colombelles : EMS, 159-183.
- Lorenzoni, G., and C. Baden-Fuller 1995  
Creating a Strategic Center to Manage a Web of Partners, *California Management Review*, 37: 3, 146-163.
- Luhmann, N. 1979  
*Trust and Power*, London: Wiley.
- Lundvall, B.-Å. 2000  
Understanding the Role of Education in the Learning Economy: The Contribution of Economics, in OECD-CERI, *Knowledge Management in the Learning Society*, Paris, OECD, 11-35.
- Lurey, J. S., and M. S. Raisinghani 2001  
An Empirical Study of Best Practices in Virtual Teams, *Information and Management*, 38: 8, 523-544.
- Maillat, D. 1998  
Organisations productives territorialisées et milieu innovateur, in G. Loinger et J.-C. Némery (Eds.), *Recomposition et développement des territoires*, Paris : L'Harmattan, 47-68.
- Maillat, D. 1996  
Systèmes territoriaux de production et milieux innovateurs, in OCDE *Réseaux d'entreprises et développement local*, Paris : Les Editions de l'OCDE, 75-90.
- Mangematin, V. 1996  
The Simultaneous Shaping of Organization and Technology within Co-operative Agreements, in R. Coombs, P. Saviotti, A. Richards et V. Walsh (Eds.), *Networks and Technology Collaboration*, London: Elgar, 119-141.
- Mangematin, V. 1999  
La confiance : un mode de coordination dont l'utilisation dépend de ses conditions de production, in C. Thuredoz, V. Mangematin et D. Harrisson (Eds.), *La confiance, approches économiques et sociologiques*, Paris : Gaëtan Morin, 31-56.
- Mangematin, V. 2003  
Cléopâtre et son goûteur, in V. Mangematin et C. Thuredoz (Eds.), *Des mondes de confiance*, Paris : CNRS Editions, 121-126.

- **Mangolte, P. A. 2003**  
Le chaudron magique et l'invention collective, *Cahier de recherche*, CEPN (IIDE), Villetaneuse : Université Paris-Nord.
- **Markus, M. L., Manville B. and C. E. Agres 2000**  
What Makes a Virtual Organization Work?, *Sloan Management Review*, 42: 1, 13-26.
- **Mauss, M. 1950**  
Essai sur le don : Forme et raison de l'échange dans les sociétés archaïques, in *Sociologie et anthropologie*, Paris : PUF, 143-279.
- **Mayer, R., J. H. Davis and F. D. Schoorman 1995**  
An Integrative Model of Organizational Trust, *Academy of Management Review*, 20: 3, 709-734.
- **Maznevski, M., and K. M. Chudoba 2000**  
Bridging Space over Time: Global Virtual Team Dynamics and Effectiveness, *Organization Science*, 11: 5, 473-492.
- **Minzberg, H., and J. A. Waters 1985**  
Of Strategies, Deliberate and Emergent, *Strategic Management Journal*, 6: 3, 257-272.
- **Moon, J. Y., and L. Sproull 2000**  
Essence of Distributed Work: the Case of the Linux Kernel, *First Monday*, 5: 11.
- **Müller, M., H. Yamagata, L. Wall and D. Dougherty 2001**  
*Le logiciel libre*, Paris : O'Reilly.
- **Nemiro, J.E. 2001**  
Assessing the Climate for Creativity in Virtual Teams, in M. Beyerlein, D. Johnson and S. Beyerlein (Eds.), *Virtual Teams, Advances in Interdisciplinary Studies of Work Teams*, Greenwich, CT: JAI Press, 59-84.
- **Nohria, N., and R. G. Eccles 1992**  
Face-to-face: Making Network Organizations Work, in N. Nohria and R. G. Eccles (Eds.), *Networks and Organizations: Structure, Form and Action*, Boston, MA: Harvard Business School Press, 288-308.
- **Nooteboom, B. 1996**  
Trust, Opportunism and Governance: a Process and Control Model, *Organization Studies*, 17: 6, 985-1010.
- **Nooteboom, B. 2002**  
*Trust: Forms, Foundations, Functions, Failures and Figures*, Cheltenham : Elgar.
- **O'Hara-Devereaux, M., and R. Johansen 1994**  
*Global Work: Bridging Distance, Culture and Time*, San Francisco, CA: Jossey-Bass.
- **O'Mahony, S. 2003**  
Guarding the Commons: How Community Managed Software Projects Protect their Work, *Research Policy*, 32: 7, 1179-1198.
- **Orléan, A. 1994**  
Sur le rôle respectif de la confiance et de l'intérêt dans la constitution de l'ordre marchand, *Cahier de recherche*, CREA n°9403A, Paris : Ecole Polytechnique.
- **Ouchi, W. 1979**  
Markets, Bureaucracies and Clans, *Administrative Science Quarterly*, 25: 3, 129-141.
- **Panteli, N. 2003**  
Situating Trust Within Virtual Teams, *Working Paper*, n°20, Bath: University of Bath, School of Management.
- **Perroux, F. 1960**  
*Economie et société : Contrainte, échange, don*, Paris : PUF.
- **Planque, B. 1991**  
Note sur la notion de réseau d'innovation : réseaux contractuels et réseaux conventionnels, *Revue d'Economie Régionale et Urbaine*, 3-4, 295-320.
- **Poppo, L., and T. Zenger 2002**  
Do Formal Contracts and Relational Governance Function as Substitutes or Complements?, *Strategic Management Journal*, 23: 8, 707-725.
- **Potter, R. E. 2000**  
Virtual Team Interaction: Assessment, Consequences and Management, *Team Performance Management*, 6: 7-8, 131-149.
- **Powell, W. W. 1987**  
Hybrid Organizational Arrangements: New Form or Transitional Development?, *California Management Review*, 30: 1, 67-87.
- **Rallet, A. 1997**  
Les technologies de l'information et de la communication et la coordination à distance dans les activités de recherche d'innovation, in A. Torre, A. Rallet et Y. Lung (Eds.), *Organisation spatiale et coordination des activités d'innovation des entreprises, Rapport pour le Commissariat Général au Plan*. Pessac : IERSO Université Bordeaux IV, 77-96.
- **Rallet, A., et A. Torre 2001**  
Proximité géographique ou proximité organisationnelle ? Une analyse spatiale des coopérations technologiques dans les réseaux localisés d'innovation, *Economie appliquée*, 54: 1, 147-171.
- **Raymond, E. S. 1998**  
The Cathedral and the Bazaar, *The First Monday Journal of the Internet*, 3: 3, 1-24.
- **Raymond, E. S. 1999**  
*The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*, Sebastopol: O'Reilly.
- **Reix, R. 1995**  
Savoir tacite et savoir formalisé dans l'entreprise, *Revue Française de Gestion*, 21: 105, 17-28.
- **Rooks, G., W. Raub, R. Selten and F. Tazelaar 2000**  
How Inter-Firm Co-operation Depends on Social Embeddedness: A Vignette Study, *Acta Sociologica*, 43: 2, 123-137.
- **Sako, M. 1991**  
The Role of Trust in Japanese Buyer-Supplier Relationships, *Ricerche Economiche*, 155: 2-3, 375-399.
- **Sitkin, S. B. 1995**  
On the Positive Effect of Legalization on Trust, *Research on Negotiation in Organizations*, Vol. 5, Greenwich, CT: JAI Press, 185-217.

- **Stewart, D. 1984**  
*Secondary Research: Information Sources and Methods*, Newbury Park, CA: Sage.
- **Strauss, A. L., and J. Corbin 1990**  
*Basics of Qualitative Research: Grounded Theory Procedures and Techniques*, Thousand Oaks, CA: Sage.
- **Tayon, J. 2002**  
Le projet Linux est-il un modèle possible d'entreprise innovante?, *Cahier de recherche*, v. 1.20, CNAM, Chaire de développement des systèmes d'organisation.
- **Teece, D. J. 1987**  
Profiting from Technological Innovation: Implications for Integration, Collaboration, Licensing and Public Policy, in D. J. Teece (Ed.), *The Competitive Challenge*, New York: Harper & Row, 185-219.
- **von Hippel, E. 1986**  
Lead User: a Source of Novel Products Concepts, *Management Science*, 32: 7, 791-805.
- **von Hippel, E. 1994**  
Sticky Information and the Locus of Problem Solving: Implications for Innovation, *Management Science*, 40: 4, 429-439.
- **von Krogh, G., S. Spaeth and K. R. Lakhani 2003**  
Community, Joining, and Specialization in Open Source Software Innovation: a Case Study, *Research Policy*, 32: 7, 1217-1241.
- **Wayner, P. 2000**  
*Free For All: How Linux and the Free Software Movement Undercuts the High-Tech Titans*, New York: Harper-Business.
- **Weber, R. P. 1990**  
*Basic Content Analysis*, Newbury Park, CA: Sage.
- **Weick, K. E. 1993**  
The Collapse of Sensemaking in Organizations: the Mann Gulch Disaster, *Administrative Science Quarterly*, 38: 4, 628-652.
- **Wenger, E. 1998**  
*Communities of Practice: Learning, Meaning and Identity*, Cambridge: Cambridge University Press.
- **Williamson, O. E. 1993**  
Calculativeness, Trust and Economic Organization, *Journal of Law and Economics*, 36: 1, 453-500.
- **Zucker, L. 1986**  
Production of Trust: Institutional Sources of Economic Structure: 1840-1920, in B. Staw and L. Cummings (Eds.) *Research in Organizational Behavior*, Vol. 8, Greenwich, CT: JAI Press, 53-111.

## ANNEXE : PRÉCISIONS MÉTHODOLOGIQUES

Cette annexe a deux objectifs : une présentation synthétique des quatre sources principales utilisées en insistant sur le mode de production des données, et une évaluation synthétique de chacune d'elles grâce à la grille de Stewart (1984).

### Le projet FLOSS (Ghosh *et al.*, 2002)

Mené par Ghosh et son équipe, ce projet de recherche est fondé sur une enquête en ligne portant sur 2.784 développeurs membres de la communauté des logiciels libres. Financé par l'*International Institute of Infonomics* de l'Université de Maastricht, cette enquête fait aujourd'hui référence tant pour ses résultats que pour leur mode de production. En effet, le recueil des données par questionnaire a été directement confié à la communauté elle-même, permettant ainsi un échantillon large et diversifié. Par ailleurs, un travail de validation des résultats a été effectué sur un sous-échantillon de 500 développeurs.

**Tableau A1.** Evaluation du projet FLOSS (Ghosh *et al.*, 2002) grâce à la grille de Stewart (1984)

Source	Ghosh, R. A., R. Glott, B. Krieger, and G. Robles (2002), <i>Free/Libre and Open Source Software: Survey and Study – Part 4: Survey of Developers</i> , FLOSS Final Report, Maastricht: University of Maastricht, International Institute of Infonomics, disponible sur <a href="http://www.infonomics.nl/FLOSS/report/">http://www.infonomics.nl/FLOSS/report/</a> .
Objectifs de la collecte	Découvrir les motivations des développeurs Découvrir les modes d'organisation de la communauté des logiciels libres Découvrir les caractéristiques individuelles (âge, expérience, rémunération motivations...) des acteurs de la communauté
Responsabilité de la collecte	Les chercheurs dans un premier temps puis, par effet boule de neige, les membres de la communauté eux-mêmes (auto-administration du questionnaire)
Nature des informations collectées	Sur les trois thèmes précités (motivations, modes d'organisation et caractéristiques individuelles), un mix de données qualitatives et quantitatives
Période de recueil	Février 2002 - Avril 2002
Mode de production	Questionnaire en ligne
Corroboration de l'information	Oui : the WIDI Survey et the Hacker Survey of BCG

Le **Tableau A1** montre que les critères de Stewart sont aisément remplis avec en particulier une volonté très nette de corroborer les résultats obtenus avec deux autres enquêtes majeures. Le lecteur intéressé trouvera des informations complémentaires sur ce projet sur le site <http://www.infonomics.nl/FLOSS/report/>.

### Le projet LICKS (Ghosh et David, 2003)

Mené en partenariat par l'*International Institute of Infonomics* de Maastricht et l'*Internet Institute* de l'Université de Stanford, ce projet, financé par la NSF, vise à compléter le projet FLOSS en étudiant l'évolution de la communauté et en se focalisant exclusivement sur une partie de cette communauté : celle centrée sur le développement du noyau Linux. Les auteurs ont choisi d'utiliser des techniques de statistique comparative en analysant les caractéristiques de développement de trois versions de ce noyau : Linux 1.0 (mars 1994), Linux 2.0.30 (avril 1997) et Linux 2.5.25 (juillet 2002). La technique de recueil des données et plus généralement la méthodologie employée sont décrites en détail dans Ghosh (2002, disponible sur le site <http://dxm.org/papers/toulouse2/>).

**Tableau A2.** Evaluation du projet LICKS (Ghosh et David, 2003) grâce à la grille de Stewart (1984)

Source	Ghosh, R. A., and P. A. David (2003), <i>The Nature and Composition of the Linux Kernel Developer Community: a Dynamic Analysis</i> , Working Paper, NSF Open Source Software Project, Stanford, CA: Stanford Institute for Economic Policy Research
Objectifs de la collecte	Analyser l'évolution de la communauté des développeurs du projet Linux kernel à travers l'évolution de la distribution de la production de lignes de codes signées par un auteur
Responsabilité de la collecte	R. A. Ghosh
Nature des informations collectées	Données quantitatives : tailles des équipes, nombre total de développeurs, nombre de sous-projets identifiés, nombre de collaborations inter-projets, part des lignes de codes anonymes
Période de recueil	2002
Mode de production	Extraction automatique de données par des outils informatiques développés pour l'occasion par Ghosh et Prakash et mis en œuvre par Prakash
Corroboration de l'information	Oui : Orbiten Survey 2000 (Ghosh et Prakash) et projet FLOSS

Comme dans le cas précédent, on peut noter que les six critères de Stewart s'avèrent remplis.

### La recherche de von Krogh, Spaeth et Lakhani (2003)

Cette recherche présente l'originalité de s'être focalisée sur un projet particulier (le projet Freenet) et non sur un échantillon important de projets comme les trois autres sources utilisées. Cette variété dans la méthode s'est avérée très fructueuse. On peut aussi noter la volonté de précision des auteurs qui ont mobilisé quatre modes de recueils de données complémentaires afin de pouvoir effectuer eux-mêmes des triangulations.

#### **Tableau A3.** Evaluation de la recherche de von Krogh *et al.* (2003) grâce à la grille de Stewart (1984)

---

Source	von Krogh, G., S. Spaeth, and K. R. Lakhani (2003), Community, Joining, and Specialization in Open Source Software Innovation: A Case Study, <i>Research Policy</i> , 32: 7, 1217-1241
Objectifs de la collecte	Comprendre comment un nouveau développeur intègre la communauté existante Comprendre comment il contribue à la production de nouveaux logiciels
Responsabilité de la collecte	G. von Krogh, S. Spaeth, K. R. Lakhani, auxquels s'est ajouté E. von Hippel
Nature des informations collectées	Données qualitatives : caractéristiques individuelles des développeurs et du projet étudié ainsi que les comportements et motivations des acteurs Données quantitatives : nombre de développeurs et nombre d'emails échangés durant la durée du projet
Période de recueil	Janvier 2000 – Mai 2001
Mode de production	Quatre modes distincts : 1/entretiens téléphoniques de type semi-directif ; 2/contenu des emails échangés au sein de la mailing list du projet ; 3/logiciel d'extraction des données permettant d'analyser la progression du projet (à travers les modifications du code source de Freenet) ; 4/documents complémentaires (pages Web, FAQ...)
Corroboration de l'information	Oui, en particulier avec les autres articles du numéro spécial de <i>Research Policy</i> et von Hippel et von Krogh (2003), Wayner (2000) et Raymond (1999)

---

Là encore, cette source de données remplit les six critères de Stewart.

### La recherche de Krishnamurthy (2002)

Cette dernière source est sans doute moins ambitieuse que les trois précédentes mais s'avère utile par l'originalité de ses objectifs et donc des données recueillies concernant notamment la taille des équipes de développeurs.

#### **Tableau A4.** Evaluation de la recherche de Krishnamurthy (2002) grâce à la grille de Stewart (1984)

---

Source	Krishnamurthy, S. (2002), Cave or Community? An Empirical Examination of 100 Mature Open Source Projects, <i>First Monday</i> , 7: 6
Objectifs de la collecte	Analyser le mode de production des logiciels libres Montrer que ce mode est davantage lié à des acteurs individuels solitaires ( <i>lone production model</i> ) qu'à une communauté travaillant collectivement sur des projets communs ( <i>community production model</i> )
Responsabilité de la collecte	S. Krishnamurthy
Nature des informations collectées	Données quantitatives : taille des équipes de développeurs Données qualitatives : nature des cent contributeurs les plus importants de la communauté (universités, individus, entreprises privées...)
Période de recueil	23 Avril 2003 – 1er Mai 2003
Mode de production	Compilation manuelle des données relatives aux cent projets étudiés disponibles sur le site Sourceforge.net
Corroboration de l'information	Partiellement avec le projet Licks (2003)

---

On peut noter que le dernier critère de Stewart ne s'avère pas complètement rempli. Toutefois, nous avons nous-même corroboré les données relatives à la taille des équipes avec les données issues du projet LICKS. En revanche, il n'a pas été possible de corroborer les données relatives à la nature des cent contributeurs les plus importants de la communauté.