

Le savoir en réseaux et l’empreinte inventive

Des droits coutumiers du gnu au dinosaure de Netscape

Il semble que le capitalisme se caractérise aujourd’hui de plus en plus par une dynamique de valorisation qui intervient dans le cadre du *réseau*, et non plus, ni de *l’atomisme* de l’agrégation d’actes marchands ponctuels, ni de *l’organicisme* propre à la planification industrielle et à la structuration hiérarchique. Ouchi, dans *Market, Bureaucraties and Clans* (Administration Science Quarterly, 1980) a par exemple mis en évidence que, entre le modèle du marché qui suppose une transaction purement ponctuelle et qui ignore le temps, et le modèle de la bureaucratie qui suppose des plans et des règlements, émerge un troisième modèle : celui du *clan*, notamment dans une situation de changement technologique rapide et où les coûts de transaction sont grands. Le réseau, lorsqu’il est défini de manière plate et non-politique, est un *lacs* (Saint-Simon articule une pensée du réseau à un projet d’enlacement du globe) souple, c’est-à-dire que ses éléments ne sont pas délimités par des frontières tracées *a priori*. Les frontières internes au réseau tout d’abord sont floues : la sociologie du travail a montré que, dans les organisations dites “distribuées”, les savoirs se constituent par des arrangements de proche en proche ; le management a récemment retourné ce projet critique, en opérant comme si les “humains” étaient devenus des automates aux nœuds d’un réseau connexionniste. Automates connexionnistes au sens où l’appréhension de la défaillance technique diffère du diagnostic formel, et se réalise par l’émergence d’une routine locale avec des moyens circonstanciels, mais aussi au sens où le savoir se caractérise par son état collectif et distribué.

Mais l’affaiblissement des frontières qui caractérise le flou concerne aussi les frontières *externes* de l’appareil productif : le réseau s’enfonce par capillarité jusque dans l’intime : goûts, affect, vie rêvée, amitié. Sous cet aspect, la mise en réseau correspond à une intensification de la mise au travail. C’est ce qu’a montré avec force l’approche opéraïste. Paolo Virno (1991) a ainsi montré comment peut être sollicitée l’expérience des individus coopérants dans ce qu’elle a de plus personnel, voire de plus *intime*. En effet, un intérêt majeur de l’invocation du réseau est de pouvoir faire utiliser par l’entreprise les savoir-faire des personnes précisément à partir du point de basculement où ils *deviennent* étrangers au domaine professionnel. L’informatique se caractérise par une prédisposition remarquable à se moduler selon cette articulation en réseau. Pour fixer les idées, on peut mentionner deux exemples récents. Tout d’abord, *l’utilisation de bénévoles*. America Online (AOL), le plus important fournisseur d’accès aux Etats-Unis, a été mis en cause en avril 1999 devant le ministère américain du Travail par sept anciens “bénévoles” pour une infraction au droit du travail¹. L’affaire a mis en lumière qu’AOL avait recours aux services de 10 000 bénévoles pour le soutien à la clientèle, l’animation de forums d’échange, le dépannage à distance, à raison chaque semaine de quatre heures de leur temps, plus des tâches administratives, en échange de quoi ils bénéficient d’un accès Internet gratuit chez AOL. “Les plaignants ont cité l’obligation de se conformer à des horaires établis par AOL, la fourniture par cette dernière d’outils de travail (logiciels, accès Internet), l’imposition de devoir signer des ententes de non divulgation, et l’exécution de tâches normalement dévolues à des salariés (...). Chez AOL, on parle de la “culture de partage” inhérente à Internet pour expliquer le recours à des bénévoles²”. Le problème est qu’AOL avait, sans préavis, mis fin à la convention de bénévolat avec certains des animateurs, leur retirant ainsi leur compte d’accès Internet, à la suite de critiques formulées publiquement contre eux sur les forums d’AOL. Cela souligne l’utilisation opportuniste, par certaines firmes de l’Internet, de l’énergie créatrice d’usagers et de programmeurs. Dans une autre direction, les firmes en réseaux s’orientent vers la *saisie de l’intimité de consommateurs*. Des fournisseurs d’accès gratuits demandent à leur client des informations très précises (connaissances en matière de presse, goûts musicaux), et vendent ensuite une base de données de profils d’utilisateurs. Ironie du sort, ce contrôle de l’intime a été soutenu par des revendications consuméristes (un boycott de l’Internet, organisé le 13 décembre 1998 à l’appel de certaines associations d’internautes) !

Promesse et amitié. Double violence ontologique du réseau

Cependant, notre tentative pour mettre en évidence la violence singulière et neuve d’un “capitalisme en réseaux” veut mettre l’accent sur un aspect qui outrepassa la conquête extensive du territoire de l’intime et les phénomènes d’implosion, de “lissage”, qu’implique la convocation intensive d’une *mobilisation* subjective. Nous voudrions insister sur les effets proprement ontologiques que produit la mise en réseaux, et qui sont de l’ordre d’une modification des rapports entre l’homme et son savoir. Puisque, dans le réseau, les entités ne sont pas prédéfinies, ce qui importe, c’est l’événement de la rencontre, et ce qu’il crée et produit : les

flux, les transferts, les échanges, les permutations, les précipitations, les déplacements. Aux ontologies essentialistes les réseaux substituent des mondes constructivistes, des mondes ouverts où les êtres sont constitués par leurs rapports et leurs compositions de rapports (malgré de grandes oppositions entre les approches de Deleuze, Guattari, Serres et Latour par exemple³). C'est exactement le pouvoir émancipateur des réseaux : il permet de se libérer des tutelles hiérarchiques ou des assignations identitaires. Cependant, si tout n'était qu'ouverture, le réseau n'aurait pas de consistance et se diluerait dans une liquidité sans texture : il ne connaîtrait que des flux sans que rien ne vienne jamais s'y former ou s'y produire : tout serait toujours emporté et dissout dans la soupe chaotique, dans le flot continu des ruptures. Il faut donc envisager le réseau comme une machine qui produit quelque chose d'intermédiaire entre une forme cristallisée et un informe chaotique. Si les êtres engagés dans un réseau travaillent à tout moment en plein brouillard, ils sont, comme les innovateurs sur les réseaux socio-techniques de la nouvelle sociologie des sciences, revêtus du masque de Janus (cf. Latour, *La vie de laboratoire*, 1989). Les dés qu'ils lancent une première fois sont l'affirmation du hasard, la combinaison qu'ils forment en tombant est celle d'une nécessité. Dans un passage très éclairant, Latour compare la pratique de l'innovateur au creusement progressif d'une vallée en V par le déplacement vers cette ligne de tous les ruisseaux. Le réseau est considéré comme ce qui réalise l'infléchissement des forces en les domptant pour qu'elles portent toutes vers la même ligne de plus grande pente. "L'asymétrie croît avec le flot de paroles" (*Microbes. Guerre et paix*, 1984, p.205). En transformant la légèreté d'un projet, souffle d'air sorti d'une bouche d'innovateur, dans la dureté pesante d'une chose, un objet, le réseau socio-technique produit une asymétrie, c'est-à-dire exactement une *information*, au sens d'une réduction du hasard. Dès lors, tout le problème, dans un réseau, consiste à passer du "kairos" de la rencontre à la "valeur" de l'information.

Les réseaux socio-techniques font ainsi reposer leur construction sur l'habileté, en leur sein, d'un stratège. Ce dernier en est le véritable moteur immobile, car c'est lui, en définitive, qui va transformer une succession de déplacements, qui sont aussi désignés comme des traductions, en des associations stabilisées. Chaque déplacement repose sur une opération qui peut être conceptualisée comme un *redressement* et dans laquelle il s'agit de faire en sorte que la nouvelle morphologie de chaque allié puisse se mouler dans le nouvel alignement ainsi formé. En effet, pour qu'une rencontre entre des perceptions puisse avoir une "valeur" sur un réseau, il faut que l'effet de cette rencontre soit repris, ou déplacé, par un *passeur*. Les passeurs dans un réseau ne sont pas uniquement des ponts ; ils sont aussi des créateurs d'informations (indépendamment du fait qu'ils peuvent ensuite la redistribuer pour les autres, ou la garder en secret). Dès lors, ils cumulent deux violences. Simondon a précisément insisté sur la violence des ponts, qui blessent en désancrant des territoires : "Jeter un pont sur un bras de mer, rattacher une île au continent, percer un isthme, c'est modifier la configuration de la terre, c'est attenter à son intégrité naturelle. Il y a un orgueil de domination dans cette violence, et l'homme se donne le titre de créateur ou du moins de contremaître de la création : il joue un rôle démiurgique" (*Du mode d'existence des objets techniques*, 1969, p.147). A cela s'ajoute la violence du stratège qui constitue le moteur immobile, sans cesse se déplaçant d'un front à un autre telle l'universelle araignée, et qui est une violence purement capitaliste : dans un tel capitalisme en réseaux, le principe de la valeur ajoutée est la capacité à tirer parti des connaissances, à combiner et à déplacer des informations, à les utiliser de manière opportuniste. Il s'agit dans un deuxième cas de *la violence opportuniste*. C'est exactement la thèse de Robert Reich dans *L'économie mondialisée* (1993). Dès lors, réseau et information constituent une même entité, l'une considérée sur le mode de la production et l'autre sur le mode de l'anti-production.

Ainsi, c'est autour de l'exercice de deux violences, l'une provoquée par la décontextualisation qu'opère l'insertion dans des réseaux de traçages hors des territoires, l'autre liée à l'opportunisme de l'usage, que se constitue la violence du réseau. Les deux violences sont conjointes, corrélées : l'une s'exerce contre la texture d'une consistance subjective, qui suppose une insertion dans la pâte colloïdale que constitue un tissu de sociabilité communalisé autour de connivences. La décontextualisation d'un actant dans un réseau de traçage renvoie ainsi à une violence d'arrachement et de déchirement ; d'une façon simplifiée, elle compromet comme on le verra la permanence d'une identité, qui s'exprime dans une "promesse". La destruction de la texture "colloïdale" du tissu de sociabilité et de connivence est manifeste dans l'aventure du logiciel libre, qui représente une tentative de transformation du savoir en actif "liquide". Sa liquidité est éventuellement encore assurée, dans certains mécanismes, par une monétarisation, dans le cadre de micro-paiements, des savoirs apportés par les membres. L'autre violence, qui provient de l'opportunisme de l'usage, s'exerce contre la dimension *exceptionnelle* ou "an-omale" d'une amitié : la formation d'un capitalisme en réseau est déclenchée par une asymétrie constitutive, qui consiste en un détachement, à l'intérieur du réseau, d'un "stratège" prêt à capitaliser pour son

propre compte les liens qui l'unissent aux autres membres. Il introduit un espace de calcul, dans lequel les relations sont comptabilisées et évaluées en fonction des trous structuraux qu'elles permettent de franchir. Mais la violence ainsi produite provient plus de la disparition de la mobilité latérale à l'intérieur du réseau pour tous les membres ; et, en cela, c'est la liquidation de la dimension anormale fondatrice de l'amitié. Ce qui fait la puissance de la "bande", c'est la possibilité pour chaque individu de développer beaucoup de positions possibles, ce qui fait de chacun, tantôt solitaire à côté de la bande, tantôt chef de brigade au sein de la bande, tantôt anonyme perdu dans les énoncés collectifs de la bande, un être exceptionnel. Bref, l'anormal, terme repris d'un substantif grec qui a perdu son adjectif, désigne le rugueux, ou l'aspérité, et plus précisément, dans le cadre du réseau, l'élément préférentiel de la bande, caractérisé par le fait que chacun y occupe une position symétrique de bordure. "L'Anormal, l'élément préférentiel de la meute" est un dispositif central pour comprendre la violence du réseau : l'anormal comme position dynamique, empressement à tracer la ligne et donc phénomène de *bordure* (Deleuze-Guattari, 1981, p.299-300).

Clarifier ces deux différentes violences susceptibles d'être exercées par un réseau nous semble ainsi essentiel, car le réseau pose deux problèmes absolument majeurs dans une perspective d'émancipation.

Le premier problème est que le réseau, contrairement par exemple au dispositif de surveillance, ne peut faire l'objet d'un diagramme le schématisant. Comme le remarquait Deleuze, dans les sociétés de contrôle, "on n'en finit jamais avec rien, l'entreprise, la formation, les services, étant les états métastables et coexistants d'une même modulation, comme d'un déformateur universel". Le mécanisme de contrôle "donne à chaque instant la position d'un élément en milieu ouvert, animal dans une réserve, homme dans une entreprise". Et, finalement, il "substitue au corps individuel et numérique le chiffre d'une matière dividuelle" (*Pourparlers*, 1991, p.242). Dès lors, le réseau ne peut être saisi par un diagramme, comme pouvait l'être par exemple le panoptisme. Le terme de "diagramme" est utilisé chez Foucault dans son chapitre de *Surveiller et punir* sur le "panoptisme", où il constitue un opérateur d'abstraction : défini énigmatiquement comme "fonctionnement abstrait de tout obstacle et frottement", il est le concept qui permet la généralisation du panoptique, comme système architectural, en panoptisme, comme mode d'exposition des rapports de force dans la société de surveillance (1974, 203-219). Le diagramme est *l'exposition des rapports de forces qui constituent le pouvoir*, en un sens il y a une troublante correspondance entre le caractère synoptique de sa vision surplombante et le caractère panoptique de la surveillance qu'elle surplombe. Le réseau, au contraire, est la dissémination de ces rapports de force.

Le second problème est *l'ambivalence de la notion de réseau*, qui se prête à des sens multiples et contradictoires. Cette ambivalence est un formidable moteur pour le capitalisme. En un sens, il a besoin d'elle. En effet, le capitalisme, comme forme d'accumulation illimitée, ne peut rien trouver en lui-même pour former des motifs d'engagement (autres que la dissuasion ou la contrainte, et pour une très infime minorité, l'enrichissement). Il est une des rares formes historiques ordonnatrices de pratiques collectives à être parfaitement détachée de la sphère morale au sens où il trouve sa justification en lui-même (l'accumulation du capital est un but en soi). De ce point de vue, pour maintenir sa légitimité, il doit aller puiser des ressources en dehors de lui-même, dans des idées qui lui sont au départ étrangères, et qu'il incorpore en en faisant son "esprit". Le capitalisme a tout particulièrement un besoin névralgique de susciter l'adhésion des nouvelles générations de cadres, sous la menace sinon d'être miné de l'intérieur. Ainsi, c'est par l'internalisation d'idées qui lui étaient au départ hostiles, que s'exerce fondamentalement sa capacité de mobilisation⁴. Or, l'une des principales idées incorporées, dans les vingt dernières années, a été la figure du réseau, héritée de 68, en tant qu'elle était valorisée par la critique sociale, notamment celle venue du tiers secteur (Revelli), et qui brandissait la figure du réseau comme le principe de "libération"⁵.

Ce long cadrage introductif nous a semblé indispensable car il vise à donner une indication du caractère *central* du capitalisme en réseaux. L'histoire du logiciel libre peut être examinée à la lumière de ce qui précède, comme la construction d'un réseau socio-technique par Netscape. Mais le phénomène majeur est que cette construction repose directement sur l'incorporation dans un réseau d'un mouvement radical. C'est même à plusieurs égards un exemple inouï d'accumulation capitaliste dans un secteur caractérisé par l'abolition de la propriété privée. Le caractère "inouï" s'explique par la nature des contenus qui se trouvent mis en réseaux : des formalismes intellectuels, qui en tant que tels *gardent une trace de la personnalité de leur inventeur*. Dès lors,

les problèmes de violence et d'opportunisme y sont exacerbés. Comment s'est établie une telle *accumulation* ? Quelles sont les violences exercées ? Et quelles sont les possibilités de libération qui nous sont laissées ? La réponse à ces questions oblige à considérer de manière plus précise ce que signifie précisément le fait de mettre le "savoir" en réseaux.

Un soulèvement au nom de "droits coutumiers" : les débuts du mouvement des logiciels libres :

Le mouvement du logiciel libre se constitue sur la volonté commune de mettre en réseaux des *formalismes intellectuels*, c'est-à-dire un ensemble de routines écrites dans un langage de programmation et correspondant à une fonction. Le principe consiste à distribuer des logiciels par forcément gratuits, mais dont les routines sont *disponibles* (elles sont diffusées en accompagnement du logiciel), et *librement modifiables*. Le premier avantage est l'évolutivité des outils intellectuels ainsi fabriqués, qui peuvent plus facilement s'ajuster à la singularité de chaque environnement de travail. C'est avec l'introduction d'une "licence" visant à protéger ce mode de distribution de toute absorption par une appropriation privée qu'on date le début du mouvement en 1984. Cette datation est erronée. La volonté de circulation libre des routines dans le milieu des informaticiens est issue d'une tradition qui s'exprimait auparavant non pas sous la forme juridique, mais sous la forme de droits coutumiers, bafoués par l'apparition des systèmes de propriété intellectuelle (brevets, droits d'auteur, secrets commerciaux) dans les années quatre-vingt.

En 1984, Richard Stallman, plus connu sous son label fantomatique, "RMS", qui lui servait d'identifiant sur les espaces électroniques, lança un gigantesque programme aux intentions politiques explicites. Celui qui était consacré comme l'un des plus grands programmeurs vivants dans les cercles du MIT bostonien ou de la Carnegie-Mellon University de Chicago, fonda le projet "GNU" de construction d'un environnement logiciel complet aux routines entièrement disponibles et librement modifiables, autour d'une version gratuite et ouverte du système d'exploitation Unix, lui-même portable mais "propriétaire". Au point de départ de sa colère, l'envahissement de son domaine technique par un système de "copyright", "d'accords de non-divulgaration", et de "contrats d'exclusivité", qui *confisquent leurs droits* aux utilisateurs, et *s'approprient les idées créatives* des travailleurs qu'ils emploient ou dont ils rachètent les créations⁶. Conséquence pour lui : la mort d'un système coopératif, qui avait pourtant prouvé sa performance depuis les années 1960 autour des entraides de programmeurs sous Unix. La dislocation d'un système de fraternité technique, organisé autour d'une culture du partage, de la proximité et de connivence. Ainsi, et c'est un point central, c'est autour de l'affirmation de la consistance collective d'un mode d'engagement dans la proximité que s'était instauré son mouvement, et notamment autour de la défense de "droits coutumiers".

Richard Stallman, lorsqu'il lance GNU, n'a pas d'autre optique que de donner une consistance pratique à l'éthique des "hackers", au moment même où elle est menacée. Le sens du terme de "hacker" renvoie chez Stallman à une double nostalgie : "hacker" désigne au départ non pas une catégorie professionnelle, mais une sociabilité *d'étudiants* dans un club, à travers le souvenir du sous-comité de signalisation et d'alimentation du club de modélisme ferroviaire du laboratoire d'intelligence artificielle du MIT à partir de 1958 (le fameux Tech Model Railroad Club). Le responsable de ce club était lui-même en charge du système téléphonique du campus. Mais le terme de "hackers" active une nostalgie pour une communauté d'artisans soudée par une communion dans le silence de la pratique, à l'abri du vacarme médiatique. "Hacker" désigne alors pour Stallman un ensemble d'utilisateurs qui se déterminent à partir de leur pratique, et composent un groupe qui ne peut pas être caractérisé par une similitude de statut. "L'utilisation du terme de "hacker" dans le sens de "qui viole des systèmes de sécurité" est un amalgame utilisé par les mass media. Nous autres hackers refusons de reconnaître ce sens, et continuons d'utiliser ce mot dans le sens "qui aime programmer et apprécie de le faire de manière astucieuse et intelligente" (Stallman, 1998).

Entrave technique et déchirement affectif : la dimension nostalgique d'un monde coutumier

Quelle fut l'éthique de cette communauté ? L'éthique des hackers s'est construite dans le silence de la pratique, et non comme un projet délibéré : implicite coalescent de la pratique collective, elle ne fut ni débattue ni explicitée dans des manifestes, comme le souligne Lévy (1984, p.39). Mais le point fondamental, dans l'élaboration d'un tel projet, est que les "hackers" ont défendu le logiciel libre *au nom de la défense de leurs droits coutumiers*. Pour justifier son basculement dans

l'engagement, Stallman fait ainsi référence à la nostalgie d'une "coutume", ou plus exactement à la *force d'une règle coutumière*

"Donc, la coopération passive était normale. On pouvait toujours supposer recevoir la coopération passive au moins; la coopération active pas toujours, les autres étant toujours très occupés. Mais, au moins, ils ne faisaient rien pour vous en empêcher. Par exemple, une fois le labo a eu besoin d'un assembleur pour le code de PDP 11, qui fonctionnait sur le PDP 10. Au lieu d'écrire un tel programme, je suis allé à Harvard où ils utilisaient le système d'exploitation de Digital. Ils avaient tel assembleur qui fonctionnait sur le système de Digital. Donc, j'ai fait une copie du code source qui n'avait pas de notice de droits d'auteurs. Je l'ai apporté au M.I.T. et j'ai fait des changements pour qu'il fonctionne sur notre système d'exploitation : le système de partage temps incompatible ou ITS. J'ai ajouté des fonctions de macros dans l'assembleur, comme ça nous l'avons utilisé pendant des années. Puis, quelqu'un qui travaillait à la société BBN voulait un tel assembleur pour leur système d'exploitation qui s'appelait Tenex. Donc, il a pris le code source de ma version, il a fait des changements pour le faire marcher sur Tenex et ajouter davantage de fonctions de macros. J'avais envie de ces autres fonctions, nous avons fait une version combinée qui fonctionnait sur les deux systèmes avec des conditionnelles, et qui avait toutes les fonctions de macros. Ensuite, nous avons fait la maintenance de cette version collectivement pour les deux systèmes, pendant plusieurs années. Et, le procédé de maintenance du logiciel fonctionnait comme ça, toutes les choses qui existaient étaient toujours disponibles dans notre communauté"(Stallman, 1998)

A ce point, il est délicat de parler d'éthique car celle-ci suppose une explicitation. L'explicitation est venue de l'entrave exercée par la brisure de la félicité des entraides. Une première entrave résulte de l'obstacle qu'exerce sur la pratique ordinaire la clôture des objets dans un régime de "propriété", et leur protection par des barrières telles que les mots de passe.

Richard Stallman est arrivé au MIT en 1971 (il avait 18 ans). "C'est une quête d'une atmosphère plus compatible avec le hack qui l'a amené du centre de calcul de Harvard, relativement autoritaire, vers le MIT : la chose qu'il aimait avant tout au MIT, *c'était qu'il n'y avait pas d'obstacles artificiels*" (Stallman, cité in Levy, p.416). Par obstacles artificiels, Stallman entend en fait des dispositifs industriels qui obligent à une planification (report dans le temps) ou à une préparation (protocoles d'accès) de l'action avec l'ordinateur. Il souligne la tension existant, à Harvard, entre un équipement industriel de la coordination et l'activité intime avec l'ordinateur : Stallman travaillait en "phase nuit" sur les consoles du laboratoire d'intelligence artificielle, et menait la journée, sans en avoir parlé aux autres membres du laboratoire, une recherche en physique fondamentale à Harvard. Il fit ses classes avec les programmeurs Richard Greenblatt et Bill Gosper.

Mais, au fur et à mesure que progressaient les années 1970, Richard Stallman commença à percevoir des changements dans son habitat naturel. C'est paradoxalement au moment de la liaison au réseau ARPAnet de la communauté du MIT, donc de la mise en réseaux, que la préoccupation sécuritaire s'est emparée du département : le Département de la Défense menaçait de retirer la machine du MIT du réseau au prétexte qu'elle ne se conformait pas aux chartes de protection d'accès édictées sur tout le réseau.

Il est remarquable que cette construction est fondée sur la confiance mutuelle, *c'est-à-dire la non-spécification délibérée des termes de la réciprocité*. Le respect du contrat repose non pas sur un contrat, susceptible de canaliser les indisciplines, mais sur l'implicite sur lequel reposent les réseaux d'amitié.

"Si vous possédez quelque chose qui peut aider la communauté, et que vous rencontrez quelqu'un qui en a besoin, vous le lui donnez. Quoi de plus naturel que cela. Mais il y en a qui vous appellent "pirates" si vous traitez vos amis comme des amis". "Le premier niveau est (...) la liberté de coopérer avec des amis"(Stallman, 1999)

Stallman compare les logiciels avec des "recettes de cuisine", qui constituent un savoir-faire qui appartient à l'univers domestique, et qui est marqué par du savoir tacite et une variabilité tâtonnante.

"Avant les ordinateurs, ce qui était le plus proche du logiciel c'était les recettes. Vous savez que les gens qui cuisinent partagent toujours des recettes et changent des recettes. Il est très important dans la vie des gens qui cuisinent, de partager, de changer des recettes. Donc, c'est la même chose pour le logiciel : la liberté de diffuser une version modifiée est très importante" (Stallman, 1998).

Mais c'est surtout une blessure provoquée par le délitement des réseaux d'entraides silencieuses caractéristiques de l'ancien monde qui est à l'origine de la volonté de Stallman de constituer un monde de logiciels libres. Parallèlement à la sécurisation, un autre changement affectait la composition de l'équipe du MIT : tous les anciens hackers, qui avaient formé la culture et constituaient l'épine dorsale du labo, le quittaient progressivement : beaucoup acceptaient les compromis qu'un travail dans le privé comprenait. C'est une ultime aventure qui le fait passer de l'indignation à l'activisme : les avatars de la commercialisation des machines LISP. Ce langage de programmation pour lequel des hackers du MIT avaient forgé un émulateur utilisable sur des machines PDP-10 était, du fait de ses propriétés d'extensibilité et de puissance, le fer de lance de la manipulation autonome des ordinateurs. Il délivrait des contraintes du temps partagé, sources de frustration par le manque d'autonomie qu'elles octroyaient à chaque utilisateur. Il permettait le contrôle de programmes sophistiqués dont l'exploration aurait satisfait la curiosité du hacker le plus assoiffé (Levy, p.420). Les machines LISP attirèrent les appétits commerciaux : en 1979, l'ancien *administrateur* du laboratoire d'IA du MIT, Russell Noftsker, manifesta son intention d'embaucher les concepteurs de machines LISP (et notamment le grand Richard Greenblatt, l'auteur du joueur d'échec et de MacLISP). Malheureusement, il y avait incompatibilité entre le style de vie de Greenblatt, et notamment sa volonté de ne pas faire la moindre entorse à l'éthique des hackers, notamment sur les questions d'accès libre à l'information, et les ambitions commerciales de Russell Noftsker. Son but, notamment, était non pas de réinvestir les gains à fins d'accumulation insatiable, mais de gagner suffisamment "pour contrôler sa propre destinée, sans avoir à cirer les pompes de quiconque" (id., p.422). Dès lors, Greenblatt fit sécession en fondant sa propre compagnie, LMI, mais quelques mois plus tard, débauchant d'autres programmeurs du laboratoire d'IA, Noftsker créa une compagnie concurrente, avec une meilleure capitalisation. Noftsker obtint, après une plainte au MIT qui continuait à rémunérer à temps partiel des hackers travaillant partiellement pour LMI, le départ de guerre lasse de la grande majorité des hackers restants du MIT. Comme l'écrit Levy, "la magie était désormais devenue secrets commerciaux", les membres de la communauté avaient anéanti la clé de voûte de leur éthique : le partage de l'information. Stallman en fut profondément affecté. "Cette période fut très triste pour moi (...) L'éthique se délitait. Les machines commençaient à dysfonctionner et il n'y avait plus personne pour les réparer. Les utilisateurs lambda réagirent à cela en se tournant vers des systèmes commerciaux et importèrent le fascisme (sic) et les contrats de licence commerciaux" (Stallman, 1998). "Je me souviens que j'arpentais alors la nuit des pièces vides, alors qu'elles étaient auparavant toujours pleines, et je pensais : "ô pauvre labo ! tu es en train de mourir et je ne peux te sauver". (Stallman, in Betz et Edwards, 1986). Stallman se désolait du fait qu'il ne lui était plus possible de téléphoner, à l'heure du dîner, au numéro de poste du laboratoire, qui était surnommé, parce qu'il se terminait par 6765, "le Fibonacci de 20", par les membres de la communauté. (Levy, 1984, p.425).

Auparavant, ce qui régulaient les conduites, c'était une émotion de fierté ou de honte, et non une dissuasion telle qu'elle peut être apportée par un contrat.

"Un jour, j'ai entendu dire qu'un chercheur de l'université de Carnegie Mellon avait une copie du code source de ce programme, plus tard, je suis allé à Pittsburg, à son bureau, lui demander : "voulez-vous me donner une copie de ce programme". Il m'a répondu : "non, j'ai promis de ne pas vous la donner". La parole m'a manqué, je suis sorti sans dire un mot. Parce que je ne savais pas répondre à une telle action honteuse (...) Il a trahi presque tout le monde" (Stallman, 1999).

la connivence avec le monde coutumier des hackers de l'Unix :

Les logiciels conçus par Stallman à partir de 1984, et le projet GNU lui-même, renferment des clin d'œil à la communauté des hackers. Cette façon de faire référence au groupe par un clin d'œil, qui forge une connivence, est typique de sa volonté de forger une tradition souterraine interne au monde des hackers. C'est l'épaisseur de cette tradition souterraine qui, selon lui, est susceptible de coaguler dans une force unitaire les tentatives disséminées de résistance.

"Ensuite, il fallait un nom, et comme j'étais hacker, j'avais besoin d'un nom rigolo et un peu méchant. Il y avait une tradition des hackers où, quand on écrit un programme qui est compatible avec un autre programme, on peut donner au nouveau programme un nom qui est un acronyme récursif qui dit "*ce programme n'est pas l'autre, il ressemble à l'autre mais il n'est pas l'autre*". Ainsi, dans les années 1960 et 1970, il y avait beaucoup d'éditeurs TECO et, pour la plupart, ils étaient appelés quelque chose-TECO, mais un hacker très doué a eu l'idée d'appeler son

programme TINT (Tint is not ECO). Et puis, par la suite j'ai écrit le premier éditeur Emacs, il y avait beaucoup d'éditeurs Emacs tous compatibles, la plupart appelés quelque-chose Emacs, il y en avait un appelé FINE (Fine is not Emacs). UN autre SINE (Sine is Not Emacs). Et puis EINE (Eine is Not Emacs). Plus tard, EINE fut presque réécrit entièrement, mais pas tout à fait, et la version nouvelle était appelée ZWEI (Zwei Was Eine Initially). Donc, je cherchais un nom de ce type, mais j'ai bientôt vu qu'il n'y a pas d'acronyme de 4 lettres qui fait un mot. Donc, j'ai cherché encore et j'ai trouvé l'acronyme de 3 lettres GNU (GNU's Not Unix) ”

L'orientation des conduites vers la coopération ne repose pas sur la canalisation des conduites dans un contrat “Comme j'avais mis la chose en partage, il était *de leur devoir* de partager à leur tour ; de travailler les uns avec les autres plutôt que les uns contre les autres ” (id.). De même, un défenseur de la Free Software Foundation, l'association à but non lucratif fondée par Stallman en 1985 pour promouvoir le logiciel libre, Faré Rideau, critique l'Open Software Initiative par le fait qu'elle donne trop d'importance au niveau juridique : “Les licences sont des documents légaux (...) ils décrivent seulement ce que les gens sont libres de faire ou non : ils ne décrivent pas ce que les gens font effectivement, ni pourquoi ils veulent faire ce qu'ils font ” (Rideau, 1998). Inversement, la licence GPL, qui est l'implémentation pour les logiciels du concept de distribution “copyleft ” proposé par Stallman, sera dénoncée ensuite par Eric Raymond, promoteur du rapprochement avec Netscape, pour sa trop faible spécification, qui laisse ouvertes des utilisations imprévues, voire nuisibles.

Rumeurs, historiettes édifiantes, socialisation par l'exemple :

La difficulté d'une explicitation sous la forme de justifications publiques des motivations de l'action est symptomatique du caractère peu codifié, peu explicite, peu formalisé de la régulation morale. Sur ce point, Eric Raymond est sans ambiguïté : “J'ai observé ces coutumes en action depuis 20 ans, depuis l'avant FSF, l'antiquité des logiciels au source ouvert. Elles ont un certain nombre de caractéristiques très intéressantes. L'une des plus intéressantes est que la plupart des hackers les ont respectées sans en être pleinement conscient. En effet, ce qui est écrit ici semble être la première mise au point consciente et raisonnablement complète jamais faite ” (ACN, §4). Raymond, plus loin dans son texte, détaille plus concrètement le format spécifique de ces règles de l'entraide, en prenant l'exemple de la façon dont sont “inscrites ” les règles fondamentales, dans la communauté, qui régissent par exemple la reprise d'un projet abandonné par son premier détenteur (respect d'un délai, publicité de l'appel à candidature). D'une part, il est clair que l'application de la règle suppose d'importantes compétences tacites, d'autant plus nombreuses que la règle contient des mots ambigus, sujets à litiges interprétatifs. Comment calculer précisément le délai “raisonnable ” à attendre? Où publier les appels à candidature pour une reprise de projet? L'interprétation de la règle engendre les apories déjà signalées par Wittgenstein, qui tiennent au caractère apparemment infini de la boucle récursive engendrée par le processus de son interprétation. Cependant, on peut spécifier plus avant l'aporie wittgensteinienne dans le monde du logiciel libre. Les travaux théoriques ayant mis en évidence cette aporie sont marqués par une opposition fondamentale entre la résolution qu'a tenté d'apporter Bloor et la résolution qu'a tenté d'apporter Lynch. Bloor mentionne la possibilité d'un arrêt de l'aporie en invoquant l'existence d'un savoir tacite communautaire, dont il dote la communauté des membres, et qui leur permet de saisir intuitivement la “bonne ” interprétation, en situation, du corpus réglementaire à appliquer. Lynch, à l'inverse, refuse ce bouclage communautaire au prétexte qu'il constitue une solution théorique trop exigeante quant à la façon dont elle attribue des propriétés aux agents humains. Pour Lynch, la solution du savoir tacite est irréaliste. Selon lui, les acteurs se tournent vers des interprétations qui conviennent en fonction de dispositifs représentationnels qui leur permettent d'appuyer leurs jugements. Leur format est bien sûr distinct de celui des règles, ainsi que du savoir tacite. La solution lynchéenne à l'aporie wittgensteinienne semble la théorisation la plus adéquate pour appréhender le fonctionnement de la régulation dans le monde du logiciel libre.

Le monde du logiciel libre est en effet peuplé de dispositifs servant à appuyer les membres dans leur travail interprétatif. Le dispositif majeur est ici la “rumeur ”. Celle-ci exerce une pression parce qu'elle construit durablement les réputations dans un univers marqué par la faiblesse des relations de face-à-face, donc l'aspect relativement superficiel des relations. Les jugements sur les uns et les autres se constituent dès les premières impressions comme des qualificatifs dotés de pouvoir convainquant. Les rumeurs sont rarement mises à l'épreuve par une vérification dans le face-à-face. Mais la rumeur a aussi une véritable efficacité pragmatique, car elle repose sur la circulation d'historiettes édifiantes, qui assurent la mémorisation des règles.

De plus, on ne trouve nul espace, avant les travaux de Raymond, où soit explicitement formulée la règle. Les personnes induisaient alors la règle à partir de leur expérience personnelle de socialisation.

“Bon nombre de règles ont été enseignées par l'exemple. Pour donner un cas simple, il existe une norme qui dit que toutes les distributions de logiciels doivent proposer un fichier LISEZMOI ou LISEZ.MOI qui contient les instructions nécessaires au parcours de la distribution. Cette convention a été établie depuis au moins le début des années 1980, mais jusqu'à présent elle n'a jamais été écrite noir sur blanc. Chacun l'apprend en regardant un grand nombre de distributions ” (ACN, §17). Le processus d'assimilation de la règle est l'induction à partir d'exemples fondateurs. Ces exemples ont la particularité de ne pas être des exemples à circulation générale, comme les récits édifiants qui sont colportés par la rumeur, mais ils ont la particularité d'être des exemples personnels, que chacun tire de son insertion dans un contexte social ”

“Chacun s'implique dans la culture à travers un attachement personnel à des projets spécifiques. Chaque projet possède un contexte social propre parmi les hackers que les prétendants contributeurs doivent démontrer et comprendre socialement aussi bien que techniquement pour s'y intégrer ” (ACN, §18).

Leur connaissance suppose un rapport de familiarité. Raymond signale notamment que c'est par l'insertion dans un contexte de familiarité, par un étirement le long de liens de proche en proche, que des aspects particulièrement cruciaux de la culture hacker sont appris puis intériorisés par les membres. Il note que cette importance de l'intériorisation des règles qui fondent le collectif dans le silence des apprentissages est propre à de nombreux collectifs techniques, fondés sur le secret des savoir-faire techniques. L'apprentissage s'identifie à “une initiation à certains mystères techniques ” (ACN, §18).

Le format “coutumier”, marqué par l'apprentissage par la rumeur colportant des scènettes édifiantes, et par l'apprentissage par l'exemple assimilé dans le cadre d'une socialisation contextuelle de proche en proche, se trouve durablement transformé par l'apparition de la codification juridique. Comment situer l'efficacité propre de la codification juridique? Que fait-elle et que ne fait-elle pas? Raymond avance des arguments entièrement tournés vers une valorisation des effets positifs de la codification juridique. Il se compare, toutes proportions gardées, à Locke, ou plus précisément situe son rôle comme analogue exactement à celui du sage dans la tradition islandaise. Selon lui, la codification juridique permet une “pacification” des relations, et il reprend cette valorisation positive à Miller (1990), un historien ayant étudié le rôle des “sagas” et de la codification opérée par les “sages” dans le passage de la coutume “du stade de l'accord tacite à celui de loi coutumière, et enfin à celle de loi écrite” (Miller, 1990, cité et commenté in ACN, §20).

“Nous avons trouvé une bonne analogie des coutumes des hackers dans la théorie de la propriété foncière selon la tradition législative anglo-américaine. Historiquement, les cultures tribales européennes qui ont inventé cette tradition ont amélioré leur système des conflits en passant d'un système de coutumes désarticulées et semi-conscientes à un ensemble de lois coutumières mémorisées par les sages de la tribu - et finalement couchées sur le papier. Peut-être qu'avec l'augmentation de notre population, l'acculturation de tous les nouveaux membres devenant plus difficile, il est temps pour la culture hacker de faire quelque chose d'analogue - c'est-à-dire d'inscrire un code de bonne conduite afin de résoudre les diverses sortes de conflits qui pourraient survenir dans le cadre de projets de logiciels au code source ouvert, et de créer une tradition d'arbitrage dans laquelle les aînés de la communauté pourraient être amenés à intervenir en tant que médiateurs dans les différends ” (ACN, §18).

du pingouin au dinosaure (Mozilla) : le redéploiement du capitalisme autour de la figure du réseau

Le 23 janvier 1998, la firme Netscape rendit publique une décision fracassante, en proposant une articulation juridique nouvelle pour son produit, dont un des éléments est la “licence dinosaure” (Mozilla) : dans un geste sans précédent, confirmant des rumeurs qui couraient depuis plusieurs semaines, elle décide de faire basculer dans le domaine libre le code source de son puissant navigateur, produit vedette de la compagnie⁷. L'arrivée de Netscape bouleversa pas mal de certitudes, qu'elles soient économiques ou politiques. D'un côté, un géant du software réputé pour l'agressivité de sa stratégie commerciale ; de l'autre, un mouvement de travailleurs et d'utilisateurs dont le soulèvement est profondément enraciné dans le lent mûrissement d'une exigence pratique. Jamais auparavant aucun éditeur important de logiciel n'avait franchi le pas de l'ouverture de son

code propriétaire La firme stupéfia le monde de l'industrie pour son culot. Et le culot est souvent l'arme des opportunistes.

A bien des égards, la décision de Netscape était un ingénieux stratagème opportuniste, une manœuvre stratégique. Netscape, en reprenant le concept du logiciel libre, a opéré un "déplacement". Le "déplacement" est la manœuvre qui constitue l'innovateur d'un réseau socio-technique, qui repose sur des opérations successives de "traduction" (Callon, 1990). Il est aussi ce qui permet de *trahir* un projet (tout en annexant de nouveaux acteurs, ici le monde de l'entreprise). Le nouveau mouvement du logiciel libre, constitué par Netscape et par les promoteurs de l'*Open Source Initiative*, a réussi à "rallier" dans un même réseau socio-technique des "actants" humains et non-humains : des collectifs déterritorialisés de programmeurs en forum électronique et appartenant à des milieux culturels distincts, des utilisateurs indociles qui rapportent avec beaucoup de zèle les erreurs, une armada de juristes confectionnant un "contrat de licence" lui permettant d'optimiser son profit en milieu ouvert. Netscape a su enrôler des spécialistes de publicité ayant eu la ruse d'accumuler un immense crédit d'opinion auprès d'une clientèle porteuse en dramatisant sa décision d'ouverture des sources, et en faisant de Netscape⁸ une "icône du Web". Elle a enrôlé des spécialistes de marketing lui faisant voir les avantages qu'elle avait à distribuer librement ses sources, pour familiariser de nombreux auteurs de sites avec ses routines, et renverser la tendance qui voyait le produit de son concurrent, Microsoft Explorer, s'établir comme standard *de facto*.

Mais, en associant à son projet Eric Raymond, celui qui revendiquait l'étiquette de "porte-parole des hackers", sur la base d'une épreuve d'opinion⁹, la firme Netscape consolidait encore son projet en enrôlant un actant collectif qui jouera dans cette aventure un rôle majeur : le groupe des "hackers". "Je n'avais pas encore la moindre idée que mon anthropologie amateur *se révélerait être un catalyseur significatif* qui provoquerait ces changements. Je fus plus surpris que quiconque d'observer cela. Mais les conséquences de cette surprise résonnent encore de nos jours sur la culture des *hackers* et les mondes des affaires et des nouvelles techniques", écrit Raymond (1998). Netscape réalisait la quadrature du cercle : trahir tout en conservant.

Comment en est-on arrivé à la "licence dinosaure", depuis le "système gnu"? Une double opposition les caractérise. Tout d'abord, une opposition des modèles de distribution. A bien des égards, le concept de logiciel libre désigne avant tout un *mode de distribution* et non un modèle économique. En cela, "libre", malgré l'ambiguïté entretenue par le mot anglais "free", n'a rien à voir avec gratuit. Le mouvement du logiciel libre est polarisé entre la FSF de Stallman, et son concept de "licence *copyleft*" dont une implémentation pour les logiciels est GPL (General Public Licence), et l'*Open Software Initiative* de Raymond, rallié à Netscape, et son concept de "certification *Open Source*". L'objet juridique de la certification *Open Source* est toujours de garantir à l'utilisateur un accès libre permanent au code source, et de ce fait de lutter contre toute appropriation privée du logiciel (autrement dit, le logiciel est *ouvert*, son code est disponible à tous) : en ce sens, la certification est accordée à un grand nombre de licences, du moment qu'elles respectent cette condition. C'est principalement l'article 10 de la licence GPL qui est remis en cause par la certification *Open Source*. Cet article 10 empêchait (sauf à avoir l'accord express du donateur) le licencié d'incorporer des parties du programme libre sous GPL dans d'autres programmes dont les conditions de distribution sont différentes (qu'ils soient "fermés" ou "libres" mais relevant d'autres licences, telles BSD ou Perl). La licence GPL était ainsi qualifiée de "virale", au sens où elle contaminait toutes les œuvres de collaboration dont certains morceaux étaient issus d'une distribution GPL. A l'inverse, et c'est là un déplacement de portée considérable par rapport à la licence GPL, l'auteur d'une modification peut, dans la certification *Open Source*, distribuer une version *dérivée* d'un programme *Open Source* sous des clauses plus restrictives, et notamment sous une licence fermée (et commerciale). Ainsi, la certification *Open Source* introduit la possibilité d'une asymétrie. Elle introduit une violence en ce sens qu'elle supprime la possibilité de l'auteur d'un formalisme intellectuel de maîtriser la destinée de son formalisme. Ainsi, des versions modifiées de celui-ci peuvent se retrouver privatisées et commercialisées dans des systèmes propriétaires.

*L'homme "svelte" connexionniste*¹⁰ :

Avant d'aborder ce point, nous voudrions traiter de l'opposition de mode d'organisation qui distingue le projet GNU initial et les systèmes *Open Source*, dont Linux constitue le modèle de référence : l'*Open Software* est fondé sur un modèle du "bazar", utilisant la forme réseau, tandis que la *Free Software* est fondé sur le modèle de la "cathédrale". L'opposition entre ces deux modèles correspond au triomphe de l'homme léger propre à l'entreprise connexionniste (Boltanski, 1999, 154-238). Tout d'abord, Netscape correspond à une entreprise "svelte".

L'entreprise svelte est ainsi le nouveau modèle productif ; c'est l'entreprise "amaigrie" de tout ce qui ne lui est plus essentiel. Considérons tout d'abord les erreurs :

"Étant donné un ensemble de bêta-testeurs et de co-développeurs suffisamment grand, chaque problème sera rapidement isolé, et sa solution semblera évidente à quelqu'un.

Ou, moins formellement, "Étant donné suffisamment d'observateurs, tous les bogues sautent aux yeux." C'est ce que j'appelle: "La Loi de Linus" (CB, §4)

Comment ne pas voir un air de parenté entre Netscape et sa sœur jumelle, l'entreprise svelte caractérisée par le mode de gestion par le Kan Ban, et théorisée par Ohno, "le système de production est un système permettant de faire remonter les sureffectifs à la surface" (!) (Coriat, 1991, p.33). Ainsi, pour Ohno, le stock est certes une technique de protection contre les aléas, mais c'est une technique frustrante et coûteuse. Elle dissimule à l'observateur et au gestionnaire tous les points où précisément les améliorations doivent s'appliquer. La gestion par constitution de stocks atteint toutes ses limites dans le marché à contraction de commandes, aux mouvements erratiques et imprévisibles. Aux contraintes de l'usine minimum s'ajoute la "chasse aux superflus par les yeux" : la règle pour Ohno est de "rendre visibles toutes les sortes de gras", de "les faire remonter à la surface", par le contrôle le plus rapide et le plus court : quoi de plus rapide et de plus droit qu'un regard! Ainsi, les yeux des bêta-testeurs qui voient du premier coup d'œil les rapports de bugs s'apparentent aux panneaux indicateurs à lumière orangée qui signalent les premiers bourrelets sur la chaîne productive.

La ressemblance va plus loin. Ohno, dans le système Kan Ban, veut établir l'affectation de tâches modulables et variables, où les frontières entre les postes sont virtuelles. Il présente son argument sous des dehors émancipateurs : il dit vouloir ainsi proscrire le lay-out en cages d'oiseaux, propre au taylorisme, où le travailleur est "prisonnier" de sa machine, "enfermé" dans son poste. Il a un sarcasme contre le vieux mot d'ordre de Ford : "nos travailleurs ne sont pas payés pour faire de la marche à pied". Il souhaite ainsi réaliser l'affectation modulable et variable, la virtualisation des frontières entre les postes. Et c'est alors qu'il a recouru au "sport" pour légitimer sa mise en réseau : "Le sport peut nous suggérer beaucoup de choses. Si l'on traçait des lignes sur un terrain de football délimitant rigoureusement la zone de nos activités, cela enlèverait beaucoup d'intérêt au jeu. La remarque s'applique aussi à l'usine". L'art de passer le témoin, le refus des territoires fixes, caractérise aussi la méthode "bazar", quand elle a un sarcasme contre la célèbre loi "fordienne" de Brooks :

"Le classique de Gerald Weinberg *"La psychologie de la programmation sur ordinateur"* apporta ce qu'on pourrait considérer après coup comme une correction vitale à Brooks. Dans sa discussion sur la "programmation non égoïste", Weinberg observa que dans les boîtes où les programmeurs ne marquent pas le territoire de leur code, et encouragent les autres à y chercher les bogues et les améliorations potentielles, les améliorations sont drastiquement plus rapides qu'ailleurs. Les termes choisis par Weinberg l'ont peut-être empêché de recevoir toute la considération qu'il méritait - et l'idée que les bidouilleurs sur Internet puissent être décrits comme "non égoïstes" fait sourire. Mais je pense que cet argument semble aujourd'hui plus vrai que jamais" (Raymond, 1998)

Enfin, dans ce modèle, il n'y a plus de chef, mais un entraîneur, qui sait intéresser les membres du réseau qu'il coordonne, et qui se caractérise par le fait qu'il accompagne le développement de leurs idées. De ce point de vue, on ne peut pas vraiment parler d'exploitation, au sens où le dévouement, en un certain sens, est intentionnellement recherché parce qu'il se révèle "payant". Le coordinateur (ici, l'exemple de réussite est celui de Linus Torvalds), le "chic type" doit être un "donneur de souffle". Le terme de donneur de souffle est l'expression-clé d'un auteur de management, Hervé Sérieyx¹¹. Il est plus besoin pour lui d'être à l'écoute de ses collaborateurs que d'avoir soi-même de bonnes idées. Une trop bonne qualité technique initiale du produit qu'il met en réseau est même, fondamentalement, un handicap.

Il vaut mieux avoir des structures de données intelligentes et un code stupide que le contraire.

Mais le problème avec l'intelligence et l'originalité dans la conception de logiciels, c'est que ça devient une habitude -- presque par réflexe, vous commencez à faire dans l'esthétique et le compliqué alors qu'il faudrait rester simple et robuste. Certains de mes projets n'ont jamais abouti à cause de cette erreur, mais ce ne fut pas le cas pour *fetchmail*.

Aussi suis-je convaincu que le projet *fetchmail* a réussi en partie parce que j'ai refoulé ma tendance à donner dans la subtilité; cela contredit (au moins) l'idée que l'originalité dans la conception est essentielle dans des projets réussis menés dans le style bazar. Et pensez à Linux. Imaginez que

Linus ait tenté de mettre en pratique des innovations fondamentales dans la conception de systèmes d'exploitation au cours du développement; est-il probable que le noyau qui en résulterait soit aussi stable et populaire que celui que nous connaissons ? (CB,)

En revanche, ce qui est décisif est de savoir écouter ses utilisateurs, et de faire preuve de qualités d'humanité voire d'humilité.

Comment ai-je procédé, vous direz-vous ? Comme suit: (...) J'ai agrandi ma liste de bêta-testeurs en y ajoutant tout ceux qui me contactaient et me parlaient de fetchmail. À chaque mise à jour, j'envoyais un petit palabre sur la liste bêta, encourageant les gens à participer.

Et j'ai écouté mes bêta-testeurs, en les sondant sur les choix de conception et en les caressant dans le sens du poil à chaque fois qu'ils m'envoyaient leur avis ou des corrections.

Le résultat de ces mesures élémentaires ne se fit pas attendre. La plupart des développeurs tueraient père et mère pour avoir des notifications de bogues de la qualité de celles que je reçus dès le début du projet, et la plupart du temps, elles étaient accompagnées de bonnes solutions. J'ai reçu des critiques réfléchies, du courrier d'admirateurs, des suggestions intelligentes proposant de nouvelles fonctionnalités. Ce qui nous donne:

Si vous traitez vos bêta-testeurs comme ce que vous avez de plus cher au monde, ils réagiront en devenant effectivement ce que vous avez de plus cher au monde.

Face à un tel modèle, des questions tout à fait décisives semblent émerger. Elles me conduisent à remettre en cause la vision courante qui opère une dichotomie entre une justification économiste de l'engagement dans le secteur du libre et une justification au contraire par le don. Une mauvaise lecture des deux moments du logiciel libre pourrait faire croire que celui-ci est passé d'un moment de solidarité et de partage à un moment d'utilitarisme et de rationalité. Par exemple, on expliquerait dans le premier cas l'engagement dans le libre des développeurs par leur volonté de faire défection, de défendre une communauté dissidente. Dans le deuxième cas, participer à un projet en libre devient simplement le moyen de se construire un carnet d'adresse et d'acquérir une réputation. Cette séparation n'est pas la bonne, l'intrication entre la sincérité du dévouement et ses bénéfices économiques est plus complexes. Le modèle du libre est en ceci caractéristique d'un nouveau capitalisme en réseaux, dans la mesure où il met en évidence que les engagements de proximité entre les personnes sont au principe même de l'efficacité stratégique. Entre la proximité du don de soi (don de temps ou savoir-écouter par exemple) et le désir machiavélien et stratégique d'exploiter "opportunément" les passions exploratrices des jeunes programmeurs, la frontière tend à se dissoudre. Les capitalistes en réseau sont ainsi caractérisés non par leur exploitation effrénée des nouvelles formes de coopération militante, mais par un *freinage* de ces récupérations par des limites éthiques. Il s'agit d'abord d'une sorte de générosité en retour qui fait que des sociétés comme *Red Hat*, *O'Reilly & Associates*, sont en train de bâtir ce qui constitue des moyens de recherche semi-indépendants grâce à des politiques de constitution et de conservation de véritables "universités sauvages" de la programmation libre (Raymond, 1999). De même, ces sociétés s'inscrivent dans une dynamique de production où elles sont non plus seulement indifférentes aux pratiques de piratage ou de sabotage (je pense à la confection de virus), mais où elles peuvent les susciter. C'est un intérêt bien compris tout d'abord : le piratage permet l'exploitation ultérieure de rendements d'adoption ; le sabotage, quant à lui, permet la constitution d'un nouveau marché, celui des logiciels de produits de sécurité. Cette situation n'est pas nouvelle : le piratage était, dans la période précédente, l'allié objectif des fournisseurs de matériel et de périphériques, comme notamment Philips en Europe récemment, du fait qu'il avait vendu son département multimédia, et faisait ses profits sur la vente de matériel. Mais ce qui est nouveau dans le capitalisme en réseaux, c'est que la promotion de ces pratiques fait l'objet d'une défense militante. Ainsi, Red Hat soutient ouvertement la réalisation de versions gratuites de sa distribution de Linux, "tolère sans ombrage" (l'expression est de Raymond, 1999) qu'un éditeur de livres et plusieurs autres distributeurs de CD-ROM fassent de la publicité pour des images CD-ROM de celle-ci, parfois avant même la date de sa sortie officielle. Cela leur est interdit par les normes morales propres à la communauté Linux.

Les transformations en cours dans le secteur du logiciel libre mettent en évidence une violence caractéristique des réseaux socio-techniques. La coopération décentralisée y est finalement articulée autour d'un modèle de l'entreprise svelte, où l'écoute, la bienveillance, le charisme, sont sans cesse utilisés de manière opportuniste pour étendre encore les méandres du réseau, et énerver plus profondément encore sa réticularité productive. Ils se disposent à prendre pour terreau les engagements les plus proches qui relient les personnes, jusqu'à l'attachement d'une amitié ou au serment d'une promesse.

Le don de savoir et la mise en réseaux :

Raymond va comparer dans des pages très lumineuses et parmi les plus belles de sa prose le “projet” coopératif avec une “terre” inconnue à conquérir. Cette comparaison lui permet d’asseoir la thèse que l’appropriabilité privée des “projets” se justifie par le caractère limité des ressources que déploie un projet. L’optimalité de la solution de la propriété privée peut ainsi être démontrée, dans la tradition lockéenne, sur la base du fait qu’elle constitue la meilleure allocation possible des ressources dans un contexte où l’autorité centrale est faible ou inexistante.

“A la limite du monde connu, là où les terres n’ont encore jamais appartenu à personne, on les conquiert en apportant son travail à la terre en friche, en la clôturant et en défendant son titre de propriété etc...” (ACN §5). Raymond tente de répondre à la seule question essentielle: qu’est-ce qui justifie l’apparition de ces droits de propriété? C’est la question lockéenne inaugurale, à l’origine de la théorie lockéenne de la propriété qui fait figure de mythe instaurateur du dogme libéral. C’est pour les avantages qu’elle procure (excès des bénéfices sur les coûts) que la solution de la propriété doit être adoptée pour les projets. Là encore, comparaison est faite avec un territoire, sur la base de l’opposition entre le *territoire de chasse*, qui reste dans le *domaine public*, et les *saillances en source d’eau*, qui font l’objet d’une *propriété privée* : “Dans la tradition des bushmen Kung San du désert de Kalahari, le territoire de chasse n’appartient à personne. Mais *il y a* une propriété des points d’eau et des sources selon un modèle qui ressemble à la théorie de Locke. L’exemple des Kung San est instructif, parce qu’il montre que les coutumes décrites par Locke ne surviennent que là où le bénéfice attendu d’une ressource dépasse le coût de sa défense. Le territoire de chasse n’est pas une propriété, car le profit de la chasse est hautement aléatoire (...) Les points d’eau, au contraire, sont vitaux et suffisamment localisés pour en défendre l’accès” (ACN, §5).

L’originalité de la conception propre au mouvement du logiciel libre est que l’appropriation n’y porte pas sur des idées, ni même sur des formes. La tentative la plus classique pour identifier ce qui, contrairement aux idées, est appropriable, a été celle du droit de la propriété intellectuelle: alors qu’on ne peut pas breveter une idée, on y brevète des formes, au sens d’implémentations concrètes d’idées, dans une oeuvre écrite, ou dans un brevet. Mais la justification apportée par le mouvement du logiciel libre n’emprunte pas ce cheminement classique. Les formes, telles que les logiciels, les collections de routines (programmes) ne sont pas l’objet d’une appropriation privée. La première formulation claire identifiant ce qui a été approprié par le mouvement a été effectuée par Raymond : pour lui, ce qui est appropriable, et approprié, ce sont des *projets*. “Les idées (ou éléments de la noosphère) n’appartiennent à personne, *alors que leurs instances sous forme de projets ont des propriétaires*” (ACN, §5). La raison en est que, contrairement aux idées, ce qui est appropriable doit constituer des “ressources limitées”. Or, les projets constituent de telles ressources limitées: dressant la liste non-exhaustive des ressources limitées déployées dans un projet, Faré, reprenant la conception de Raymond (pour ensuite la critiquer) énumère ainsi le temps passé à chercher les nouvelles idées, la revue des erreurs, la mise à jour, le travail de documentation, le temps passé à rechercher les projets qui seront le plus utiles, les tutoriaux pour délivrer les informations qui seront utiles pour aider à la diffusion des documents du projet, etc... (RF-HE, §4). La réalité des projets est très finement distinguée de ce avec quoi on oppose habituellement les idées, c’est-à-dire leur implémentation concrète dans des formes. Raymond s’oppose notamment, dans une argumentation techniquement complexe, à F.R. Faré sur la qualification précise susceptible d’être apportée à ce “dual” du monde des idées dont l’appropriation est effective dans le monde des logiciels libres. Ils s’accordent tous deux à appeler ce “dual” du monde des idées “l’ergosphère” (l’intuition est de Faré, cf RF-HE §1), mais lui donnent chacun un contenu différent. Pour Raymond, l’ergosphère est la somme des projets de conquête de l’espace vierge des idées encore non découvertes : son déploiement suppose une somme d’efforts coopératifs (ACN, §5). Pour Faré, l’ergosphère est le monde des services. Mais l’appropriabilité privée des projets ne tient pas seulement à la *rareté* de la ressource qu’ils exigent, le temps de programmeurs habiles: il tient aussi à l’importance des *économies* que permet l’appropriation privée (par un “meneur” ou par un “collectif de leaders historiques”) de chaque projet. C’est Raymond le plus explicite à décrire les avantages rendus possibles par la solution de l’appropriation du projet. Il énumère les coûts du projet: le coût de la monopolisation des efforts dans un projet est facile à situer. Il est le résultat des pertes de temps induites par ce modèle. Du temps substantiel, tout d’abord, est perdu pour “le travail de maintenance de l’historique des versions successives, le temps passé à faire des notifications publiques” (ACN, §5). Mais du temps est aussi virtuellement perdu, constituant une sorte de coût d’opportunité temporel de l’appropriation privée des projets: “le temps passé à ronger son frein avant de prendre possession d’un projet orphelin” (id.). Le gain apporté par la monopolisation par un individu d’un projet réside dans la meilleure coordination des efforts, dans l’incitation plus forte

des individus. Mais Raymond insiste sur un gain souvent inaperçu : le gain qui résulte de l'évitement, grâce à la centralisation des efforts permis par le projet, des redondances.

Ainsi, Raymond fonde plus largement une *justification par l'entraide* du logiciel libre. Ainsi, selon Raymond, le logiciel libre est justifié économiquement par le fait que ce qui constitue la valeur d'usage du logiciel libre, et est donc à la racine de la captation de la valeur d'échange, est la qualité du retour informationnel effectué par le produit sur le client: ce retour informationnel peut être effectué par une aide en ligne, mais à ce niveau il reste très insuffisant. C'est autour d'une théorie sur l'optimisation de l'entraide que l'on est conduit à engager une réflexion sur la valeur d'usage du logiciel. Le fait de rendre collectif un projet est justifié économiquement, par la différence entre le gain et le coût d'une telle mesure, mesurable en temps et en énergie évaporés. Le fait de rendre publique une routine permet de *pérenniser*, au-delà de la survie économique de la firme qui l'a fabriquée, le service après-vente sur un logiciel, ce qui permet d'augmenter considérablement et durablement sa valeur d'usage.

A l'inverse, la fondation sur les droits coutumiers du militantisme du libre se fonde plus précisément sur une *politique de la connaissance*. Une telle confrontation entre une économie des entraides et une politique de la connaissance est très visible aujourd'hui, autour de la combativité occasionnée par la résistance à l'assimilation à l'Open Source du secteur entier du logiciel libre. La résistance va au-delà de la simple dénonciation de la validité de la justification par l'économie ; au contraire, du côté de la GPL, il y a également une mise en évidence de la performance économique du système GPL. Par exemple, F.R. Faré met en avant le fait que la distribution historique GPL est source d'une valorisation lorsqu'elle exploite une dynamique de la qualité, de la "marque". Rob Young, fondateur de Red Hat, une distribution entièrement libre de Linux, compare son rôle à celui d'un vendeur d'eau minérale, dont les consommateurs achètent le produit en raison de la certification, dans une "marque" de fabrique, d'une qualité. La résistance n'incarne pas les forces anti-économiques contre les forces économiques, mais une perspective différente quant à l'appréhension des forces qui sont en jeu dans la circulation libre des formalismes intellectuels. Raymond appréhende les formalismes intellectuels comme des "entraides", et les restitue dans un tissu complexe de dons avec espoir de restitution ultérieure de contreparties.

une appréhension pratique du savoir : les droits coutumiers et la propriété indécise

A l'inverse, la référence des "hackers" du GNU à la coutume, pour défendre leurs droits, se double en fait, comme on le voit, d'une référence permanente à un "droit coutumier". La coutume est un mode de saisie de l'usage qui l'oriente vers des habitudes collectives. Le droit coutumier s'en distingue sur un point central : . Marx, déjà, opposait le "lex" et le "jus" : il opposait les droits idéels, qu'il conviendrait de distinguer cependant des simples "droits naturels", et les droits sédimentant la trace d'une tradition historique. Il attachait une importance aux "droits coutumiers". Alors qu'il était tout jeune gazetier pour la Rheinische Zeitung, Marx étudia ainsi le débat sur le "vol de bois". Dans le cadre de ce débat, il montre l'importance des "droits coutumiers" des paysans, qui reposent "sur le fait qu'un type de propriété avait un caractère *indécis qui ne déterminait pas, en dernière instance, si cette propriété était privée ou commune*"¹². A l'inverse, l'idéologie bourgeoise, selon Marx, ramène tout bien comme toute personne à un statut abstrait, unitaire et indivisible. Le droit coutumier, incarné par la licence *indécise* constituée par le GNU, correspond à un refus du détachement de l'objet usé dans une marchandise séparée de son producteur. Ce refus est particulièrement problématisé chez les "hackers" du GNU : il s'incarne dans l'énoncé explicite d'un droit moral de refuser qui utilise le produit par exemple.

"Ahem. Tout d'abord, prenez garde au mot "produit". Il implique qu'on fabrique quelque chose afin de le vendre. Je vous accorde que les sociétés de développement élaborent des logiciels afin d'en vendre des copies. Il s'agit donc bien d'un produit. Le but du projet GNU est, au contraire, de créer une meilleure communauté. Nous vendons des copies pour développer des logiciels. Il y a donc une différence fondamentale" (Stallman, 1997).

"Libre" n'a ainsi rien à voir avec "cession des droits". Lorsqu'un auteur diffuse son logiciel en libre, il conserve les droits sur celui-ci : ainsi, le concept de "libre" se distingue très clairement de celui de "domaine public", car dans le cas du domaine public le produit *ne conserve plus la trace des intentions de son auteur* :

“L'exemple de ce problème est le système X Window. Développé au MIT, et distribué sous forme de logiciel libre sous une licence permissive, il a rapidement été adopté par divers constructeurs. Ils ont ajouté X à leurs systèmes Unix propriétaires, sous forme binaire uniquement, en le frappant du même accord de non divulgation. Ces copies d'X n'étaient en rien du logiciel plus libre que le reste de l'Unix ”

Plus largement, la tradition qui dérive de Marx, invite à théoriser le caractère indivisible et le caractère intriqué des biens de connaissance, que ne fait que *sanctionner* son caractère coutumier (la coutume n'est qu'un *indicateur*)... Karl Marx, entre autres, a été frappé par le fait qu'il est possible de s'approprier la connaissance scientifique en vue de processus productifs, tout comme les forces physiques présentes dans la nature. L'indivisibilité de la connaissance trouve une extension exceptionnelle, sur une dimension temporelle et sur une dimension sociale. Temporellement, la connaissance présente une durabilité exceptionnelle. Socialement, elle rend impossible tout régime attributif. Le cas de la propriété collective pose des problèmes particuliers à l'analyse. D'une part, entre les co-participants, il y a une difficulté pour attribuer la somme due, à cause du problème des micro-paiements (CM §5). D'autre part, il y a une difficulté plus grande encore de détermination de la valeur de chaque contribution (CM §5).

Ainsi, c'est vers la mise en évidence d'une autre propriété que fait basculer la référence à la connaissance comme espèce exemplaire de bien public : sa nature cumulative, intriquée, et interactive. La nature cumulative, intriquée, et interactive de la connaissance est un élément tout à fait décisif de sa définition : il a souvent été mis en évidence dans la littérature, mais rarement été élevé au rang de propriété cruciale à partir de laquelle s'élaborent les autres propriétés, ramenées dès lors au rang de caractéristiques dérivées. La base de connaissance scientifique et technologique s'accroît par incrémentation, chaque nouvel apport s'appuyant sur des découvertes précédentes et en modifiant l'importance, et ce, de manière complexe et souvent imprévisible. Un tel constat est présent chez le sociologue des sciences britannique Michel Polanyi, dans le cadre d'une réflexion de l'auteur sur la notion de savoir “tacite”. Polanyi a affirmé que la loi sur les brevets était essentiellement déficiente parce qu'elle cherchait à “réduire en parcelles une chaîne de pensée créatrice” et la constituer en une série de revendications distinctes dont chacune constitue la base d'un monopole détenu de manière séparé, alors que “le progrès incrémentiel interagit à chaque stade avec tout le réseau de la connaissance humaine et fait appel à chaque instant aux stimuli les plus variés et les plus épars” (Polanyi, 1944, p.70-1). De même, la connaissance est susceptible d'une croissance et d'une réélaboration interne, par croisement, comme l'a montré Gabriel Tarde dans une sociologie moléculaire des courants d'imitation et d'invention : à l'inverse d'autres types de signaux comme les balises lumineuses, la connaissance technique et scientifique est susceptible de démultiplier sa valeur par hybridation avec d'autres types de connaissances. Le droit, qui découpe le tissu de connaissances en “parcelles” appropriables, est ainsi inadapté, car il disloque un objet qui est intriqué, impliqué.

Il nous semble que le point fondamental de cette politique de la connaissance, telle que l'institue la version radicale du mouvement du logiciel libre, tient dans cette mention d'une “propriété indécise”, propre aux droits coutumiers. Se contenter de revendiquer une communalité du savoir, du fait de sa nature cumulative, interactive et intriquée, ne résume qu'une partie de l'enjeu. Par “propriété indécise”, les droits coutumiers mentionnent la nécessité de revendiquer la légitimité d'un lien maintenu au savoir diffusé, d'une *attache* à la personne. Il ressort du domaine du logiciel libre une évidence plus troublante encore : ce qui constitue la force singulière de la licence GPL n'est pas tant cette affirmation d'une intrication collective du savoir, mais la mise en évidence d'une *trace personnelle* dans tout savoir, qui se fait l'écho, le reflet secret, des déambulations de celui qui lui a donné empreinte.

Textes cités en référence et mentionnés par une abréviation :

(CB) Raymond, E., 1997, “The Cathedral and the Bazaar”, intervention publique au Linux Kongress le 21 mai 1997, imprimé sous le même titre dans Raymond, E., 1999, *The Cathedral and the Bazaar. Musings on Linux and Open Source by an Accidental Revolutionnary*, O'Reilly and Associates. Traduction française par Sébastien Blondeel, http://blondeel/traduc/Cathedral-bazaar/Main_file.html.

(ACN) Raymond, E., 1998, “Homesteading the Noosphere”, diffusion publique sur le site Open Source Initiative le 1 avril 1998, imprimé sous le même titre dans Raymond, E., 1999, *The Cathedral and the Bazaar. Musings on Linux and Open Source by an Accidental Revolutionnary*, O'Reilly and Associates.

(CM) Raymond, E., 1999, *The Magic Chaudron*. Traduction française par Sébastien Blondeel, Emmanuel Fleury et Denis Vauldenaire.
(FR-HE) Rideau, F., 1998, "About Eric Raymond's Articles. Partie 1 : Homesteading the Ergosphere ", première diffusion publique le 23 mai 1998 en réponse à "A la conquête de la noosphère " de Eric Raymond.
(FR-SF) Rideau, F., 1999, "About Eric Raymond's Articles. Partie 2 : The Scientific Flask ", première diffusion publique le 7 août 1999 en réponse à "Le chaudron magique " de Eric Raymond.

Bibliographie :

Callon, M., 1990, "Éléments pour une sociologie de la traduction. La domestication des coquilles Saint-Jacques et des marins-pêcheurs dans la baie de Saint-Brieuc", *Année Sociologique*, n°36, 1986, pp. 169-208.
Levy, Steven, 1984, *Hackers, Heroes of the Computer Revolution*, Dell Publishing, New York.
Miller, W.I., 1990, *Bloodtaking and Peacemaking: Feud, Law, and Society in Saga Iceland*, University of Chicago Press.
Polanyi, K., 1944, *The Personal Knowledge*, Routledge, London.
Stallman, R., 1986, conducted by Betz, D., and Edwards, J., "GNU's not Unix. Richard Stallman discusses his public-domain UNIX-compatible software system ", Bytes, juillet 1986.
Stallman, R., 1998, "Conférence de Richard Stallman le mardi 10 novembre 1998 ", Université de Paris VIII, à l'invitation de l'association APRIL (la conférence a été donnée en français).
Stallman, R., 1999, "Le système d'exploitation du projet GNU et le mouvement du logiciel libre ", in Raymond, E., éd., *Open Sources : Voices from the Open Source Revolution*, O'Reilly, NY

¹ <http://www4.law.cornell.edu/uscode/29/ch8.html>

² Cloutier, *Chroniques de Cybérie*, 1999.

³ On connaît la position de Deleuze et Guattari, qui voient dans la communication la confusion de l'événement (comme non-rencontre) et de l'exposition de points de vue. Pour eux, le développement de la philosophie de la communication correspond "au mouvement général qui a remplacé la Critique par la promotion commerciale ". En prenant le point de vue du pervers contre le manager, il est clair que Deleuze et Guattari produisent le même déplacement révolutionnaire que lorsque Ivan Illich prenait l'exemple du paysan mexicain pour évaluer les effets du réseau autoroutier, et calculait les effets sur sa vitesse de déplacement lorsqu'il allait vendre le cochon au marché (sur cette remarque sur Illich, cf Boltanski, p.685). Dans les deux cas, l'épreuve de la rencontre ne conduit pas à une capitalisation de ses effets par des déplacements opportuns. A bien des égards, la révolution repose sur cette conversion de regard.

⁴ Cf Boltanski, *Le nouvel esprit du capitalisme*, 1999, p.42.

⁵ Là encore, Paolo Virno a montré que c'est parfois par un appel aux formes d'altérité les plus fortes, comme celles de la *contestation*, que s'exerce cette "mise au travail " des expériences personnelles : P. Virno a montré comment le capitalisme italien a réintégré et mis au travail les compétences acquises par les jeunes contestataires des années 1970 dans des activités militantes ou ludiques, en saisissant cyniquement la portée marchande du fait qu'elles relevaient de "l'invention de nouveaux modes de vie ".

⁶ Voici les jalons juridiques d'un tel *raidissement* : les logiciels informatiques sont protégés par le "copyright " aux USA. Après un long débat jurisprudentiel, la protection par les droits d'auteur des logiciels a été introduite en France par la loi du 3 juillet 1985, bien que le régime dérogatoire du logiciel ressemblât à une protection spécifique. Parallèlement, dès les années 1970, aux débuts de la micro-informatique, les contrats d'exclusivité se multiplient entre concepteurs solitaires et éditeurs. Puis, au début des années 1980, l'embauche de nouveaux salariés dans le secteur informatique s'accompagne de la signature "d'accords de non-divulgaration " entre patrons et salariés.

⁷ Hamerlin, J., Paquin, T., Walton, S., "Freeing The Source. The Story of Mozilla ", in *Open Sources : Voices from the Open Source Revolution*, 1999, Raymond, E., éd., O'Reilly, NY

⁸ C'est pourquoi nous avons limité notre présentation au rôle de Netscape. A vrai dire, beaucoup d'autres firmes se sont implantées dans le secteur du libre, et composent un subtil dégradé dans les formes d'opportunisme : Cygnus, la plus proche du modèle initial ; Red Hat de Bob Young ; puis, récemment, saisissant par les cheveux la fortune, Netscape ou Collab-Net.

⁹ La prétention de Eric Raymond à “représenter” le groupe diffus des “hackers” n’était pas une pure pétition de principe. Elle était éprouvée dans une épreuve d’opinion, comme le révèle cette confession publique de Eric Raymond: “J’avais été fasciné par la culture des *hackers* depuis de longues années, bien avant que j’aie édité la première édition de *The New Hacker's Dictionary*, en 1990. À la fin de l’année 1993, nombreux étaient ceux (et j’en faisais partie) qui en étaient venus à me considérer comme l’historien de la tribu des *hackers*, et leur ethnographe dépêché sur place. Ce rôle me plaisait bien” (Raymond, 1998c). Cependant, la validation dans une simple épreuve d’opinion de l’enrôlement des “hackers” dans le projet n’a pas d’épaisseur. Elle s’oppose à la proclamation d’un autre “porte-parole”, Richard Stallman, qui valide sa capacité à représenter ce groupe flou sur une épreuve de *confiance*, en mettant en avant que, contrairement à Eric Raymond, il a été socialisé durant de longues années dans le lieu d’émergence du “noyau dur” du groupe des hackers, le club de modélisme ferroviaire du laboratoire d’intelligence artificielle du MIT entre 1972 et 1983. D’ailleurs, Stallman est une figure centrale de l’histoire des hackers narrée par le journaliste Steven Levy (1984), tandis que Raymond ne figure même pas dans l’index de cet ouvrage...

¹⁰ L’expression est de Luc Boltanski, *Le nouvel esprit du capitalisme*, op. cit.

¹¹ Séryex, 1993, *Le big-bang des organisations*, Calmann-Lévy, Paris.

¹² p.141-2 in Lascombes P., Zander, H., *Marx : Du vol de bois à la critique du droit. Edition critique de “Débats sur la loi relative au vol de bois”*, Paris, PUF, 1984, 281 p.