Le logiciel libre dans le milieu du travail le cas de la plate-forme Moodle

Dans cette section, nous proposons une approche du phénomène des « logiciels libres » comme témoin du développement d'une forme « d'intelligence collective » ce qui nous permettra de préciser notre objet de recherche. Dans un premier temps, nous porterons succinctement notre regard sur le passé en revenant sur la « petite histoire » qui est à l'origine du « mouvement » du libre et nous expliquerons ce qu'il en est. Ensuite nous verrons que même si ce mouvement prône la « liberté » il est tout de même encadré par une variété de régimes juridiques. Pour continuer, nous aborderons le mode de développement et de production des ces applications et nous verrons qu'ils sont les produits d'œuvres collectives rassemblant une multitude d'acteurs. Nous poursuivrons en montrant l'intérêt que portent les chercheurs aux logiciels libres et à l'usage qui en est fait dans un contexte organisationnel et/ou institutionnel ; ce qui nous conduira regarder quelles sont les attentes des organisations et/ou des institutions en termes de compétences *Open Source* et de quelles manières elles sont construites.

Pour terminer, nous nous intéresserons aux plates-formes d'enseignement en ligne et plus particulièrement à la plate-forme « *Moodle* ». Nous en préciserons la genèse et l'évolution. Ensuite, nous ferons état de quelques statistiques qui montrent comment cette application distribuée est déployée au niveau international tout en s'inscrivant dans une perspective économique.

2.1 Les logiciels libres[59]

Les logiciels libres sont des logiciels dont le code source[60] peut être exécuté, copié, distribué, modifié et amélioré. À l'origine de ce terme, nous trouvons l'influence de Richard Stallman[61] qui, depuis plus de vingt ans, s'évertue à promouvoir le « mouvement du logiciel libre » par le biais de la *Free Software Foundation*[62] (*FSF*).

[59] Nous entendons également parler de *free software* (dans ce cas précis le terme *free* signifie libre et non gratuit), d'applications *Open source* ou de logiciels à code ouvert. Tout au long de ce travail, nous utiliserons les termes « *Open source* » et « logiciel libre ».

[60] Les lignes de programmation.

[61] Programmeur réputé de la communauté informatique et militant du logiciel libre, il est à l'origine système d'exploitation libre *GNU* (*GNU's NotUNIX - GNU* n'est pas *UNIX*) et de la licence publique générale

Stallman explique que le mouvement du « libre » trouve son origine dans les années 1970 alors qu'il travaillait au laboratoire d'intelligence artificielle (IA) du *Massachussets Institute of Technology* (*MIT*). Le point de départ du mouvement est, dit-il, l'histoire d'une imprimante bloquée. Le principal défaut de cette machine était une certaine propension au bourrage papier ce qui contraignait l'utilisateur à « rester planté devant la machine comme un valet au chevet de son maître ». Bien entendu, Stallman ne pouvait rien aux bourrages mécaniques. De la même façon, il lui était impossible d'accéder au logiciel qui contrôlait l'imprimante pour le modifier en fonction de ses besoins (Stallman, Williams, & Masutti, 2011, pp. 1-16).

« C'est secret, protégé par des droits et des brevets », lui rétorquait le constructeur. À cette opacité, une raison simple: le programme de l'imprimante était livré sans son code source, ses secrets de fabrication. Sans accès à ceux-ci, Richard Stallman ne pouvait adapter lui-même le logiciel rétif. Et, aurait-il pu y accéder, il n'aurait pas eu le droit d'intervenir sur leur contenu, ni de diffuser ses perfectionnements à d'autres utilisateurs de la même imprimante. Quelques années plus tard, en 1985, Stallman fondait la *Free Software Foundation* avec comme objectif de créer des logiciels diffusés avec leur code source. Des logiciels que tout un chacun pouvait copier à l'infini, modifier au gré de ses besoins, et redistribuer à sa convenance. Des logiciels que personne ne pouvait s'approprier. Des logiciels libres (Latrive, 2000, p. 12).

Selon la *FSF* un logiciel est qualifié de libre s'il présente pour ses utilisateurs les quatre libertés suivantes :

- la liberté d'exécuter le programme, pour tous les usages ;
- la liberté d'étudier le fonctionnement du programme, et de le modifier pour l'adapter à ses besoins (l'accès au code source est une condition nécessaire) ;
- la liberté de redistribuer des copies ;
- la liberté d'améliorer le programme et d'en publier les modifications afin qu'elles soient profitables à toute la communauté (GNU, 2013).

---

connue sous l'acronyme *GPL* (*GNU General Public Licence*). Il consacre la majeure partie de son temps à la promotion du logiciel libre auprès de divers publics un peu partout dans le monde. Depuis quelques années, il fait campagne contre les brevets logiciels et la gestion des droits numériques (*DRM - Digital Rights Management*)

[62] La *Free Software Foundation* est une organisation caritative dont les revenus sont utilisés pour employer un grand nombre de développeurs.

En définitive, le logiciel libre a « en quelque sorte restauré la logique première de l'industrie informatique où les logiciels étaient accompagnés de leur code source » (Benkeltoum, 2009, p. 38). Si ces libertés reflètent la philosophie (en termes d'exploitation et de développement) du mouvement *Open Source* il est nécessaire de noter que celui-ci est encadré par une variété de régimes juridiques.

2.1.1 Un mouvement encadré par des obligations contractuelles

Dans sa thèse, Bonneau (2012), souligne que le courant s'appuyant « sur les principes énoncés par la *FSF* représente la vision plus radicale et libertaire issue des idées de Stallman. Son refus de toute forme de droit d'auteur constitue pour plusieurs acteurs du libre un obstacle au déploiement des logiciels libres dans l'entreprise » (p. 22). C'est, explique la chercheuse le cas de Raymond et de Perens (2008) qui recommandent une approche « plus flexible étant davantage compatible aux hybridations entre le libre et le propriétaire, impliquant à la fois les particuliers et les entreprises » (*ibid*., p. 23). En effet, comme l'explicite Cardon (2005), il est parfois nécessaire pour les porteurs de projet de (notamment lorsque s'étend la nébuleuse des contributeurs) de se tourner vers des sponsors institutionnels ou privés ou de se prêter à des articulations avec l'univers marchand comme en témoigne par exemple, le partenariat de *Google* avec *Wikipedia*.

Ainsi, lorsque les pro-Stallmaniens militent pour ce que Bonneau appelle la « branche dure du mouvement du logiciel libre » (2012, p. 23), certains comme Perens et Raymond (2008) prônent une approche plus modérée en proposant l'expression « *Open Source Software* ». Ainsi, comme le souligne Bonneau même si de nombreux logiciels libres sont publiés sous la licence *GPL* (*General Public Licence),* le « mouvement de l'*Open Source* recouvre désormais plusieurs formes disparates puisqu'en plus de la licence *GPL*, il existe d'autres types de licences énonçant chacune des droits différents » (*op. cit.).* Mais, au-delà de ces questions de licence, il est à noter que les logiciels libres « sont le fruit d'un travail coopératif entre de nombreux développeurs qui ne se connaissent parfois que de manière virtuelle » (Dang-Nguyen & Pénard, 1999, p. 106) et qui sont au fondement d'un modèle de production de style « bazar ».

2.1.2 Une œuvre collective développée à distance : le style « bazar » comme mode de développement et de production

Dans l'article « La cathédrale et le Bazar » Raymond (1998), oppose deux modes de développement de logiciels. Le style « cathédrale » pour les logiciels propriétaires et le style « bazar » pour les logiciels libres.

Le style « cathédrale » s'inscrit dans la « logique traditionnelle de la division technique du travail, de sa planification et de son organisation rationnelle, qui privilégie l'approche centralisée et hiérarchisée » (Blondeau, 2000, p. 189). Selon ce point de vue, les logiciels doivent « être conçus comme des cathédrales, soigneusement élaborées par des sorciers isolés ou des petits groupes de mages travaillant à l'écart du monde » (Raymond, 1998). Il s'agit d'une production en série où « l'ingénieur élabore, le développeur développe et le consommateur consomme » (*op.cit*.).

Avec le style « Bazar », le cycle de production est parallélisé au moyen de ce que Raymond appelle la « Loi de Linus[63] » qui pose les assises d'une organisation déstructurée encourageant la créativité, l'initiative et renforçant l'efficacité des individus. L'idée est d'impliquer un nombre considérable d'utilisateurs dans l'amélioration du produit, de les stimuler par la « perspective auto-gratifiante » de prendre part au développement de celui-ci et de les récompenser par l'intégration de leurs idées dans le produit et par des mises à jour répétées. De fait, le développement de logiciel libre s'inscrit dans un mode de travail « communautaire » et devient ainsi un processus de coopération au sein duquel utilisation et production tendent à se confondre (les utilisateurs sont des développeurs tout comme les développeurs sont des utilisateurs). Si, de prime abord, ce mode de développement peut paraître anarchique, en y regardant de plus près on se rend compte qu'il est moins aléatoire qu'on ne l'imagine. En effet, tout projet de logiciel libre dispose d'un noyau de développeurs qui garantit sa cohérence. Ainsi, les axes futurs de développement sont discutés et fixés en commun ce qui permet de faire des choix raisonnables et de désigner des objectifs.

De plus, contrairement au « logiciel propriétaire », le processus de création des logiciels libres « échappe à toute approche marketing, se fondant plutôt sur la notion d'utilité sociale » (Moineau & Papatheodorou, 2000). Nous comprenons qu'outre leur mode de production, leur qualité technique et/ou leur rentabilité financière, les logiciels à code source ouvert peuvent être envisagés comme des vecteurs de participation sociale (Couture, Haralanova, Jochems, & Proulx, 2010, p. 45) fondés sur la « resocialisation de l'acte de créer » (Aigrain, 2005). Par conséquent, rendu possible par l'avènement d'Internet, le modèle de travail coopératif permet la participation des usagers au processus d'innovation du logiciel, ce qui les propulse de fait, au rang de co-développeurs.

[63] En référence à Linus Torvalds, qui dans les années quatre-vingt-dix, créa la surprise en proposant un nouveau système d'exploitation libre baptisé Linux.

2.1.3 Une communauté de « *Free-Rider* » et de co-développeurs

Comme le souligne Benkeltoum, « la nature du code informatique, la diffusion de langages de programmation, la démocratisation de la micro-informatique et d'Internet, ont rendu possible la conception distribuée de logiciels par des groupes plus ou moins importants d'utilisateurs experts » (2009, p. 38). L'auteur précise qu'il emploie « l'expression utilisateurs experts, car, originellement, le logiciel libre a émergé dans la sphère académique où les utilisateurs étaient aussi développeurs » (*ibid.*). Mais aujourd'hui ce n'est plus le cas, car un simple utilisateur peut participer à l'activité de conception et ainsi devenir partie prenante du processus de développement. C'est ce qu'essaie de démontrer Von Hippel (1998) lorsqu'il se penche sur le rôle joué par les utilisateurs dans les processus d'innovation. Cependant tous les utilisateurs ne participent pas au développement des logiciels libres ; ce qui soulève, comme le note Desbois, la question de leur « clivage (…) en deux catégories dont les motivations et l'implication diffèrent fondamentalement » (1999, p.4). C'est ce que nous allons voir dans les paragraphes qui suivent.

2.1.3.1 Les co-développeurs

Les utilisateurs développeurs constituent la force du monde du « libre ». Raymond conseille de les traiter comme des « co-développeurs ». C'est selon lui « le chemin le moins semé d'embûches vers une amélioration rapide » (1998) des logiciels libres. Si jusqu'à présent le développement de ces logiciels était une affaire d'informaticiens (programmeurs), aujourd'hui, il touche de simples utilisateurs souvent passionnés d'informatique et de TIC. La disponibilité du code source fait d'eux des « *hackers*[64] » efficaces. Ils diagnostiquent des problèmes, suggèrent des corrections ou des modifications du code (appelés *patch*) contribuant donc à son amélioration. Ainsi, comme le soulignent Perline et Noisette, en plus des programmeurs, les communautés du logiciel libre sont composées de toutes sortes d'individus : « les bêta-testeurs, les rédacteurs de modes d'emploi, les traducteurs, etc. » (2006, p.59). Nous comprenons donc que ce sont des organisations hétérogènes au sein desquelles des personnes apportent leur contribution en fonction des compétences et des motivations qui sont les leurs. C'est, selon ces auteurs, la raison pour laquelle la « notion de « développeur » est préférable à celle de « programmeur » lorsqu'on parle de la communauté

[64] Le terme de « *hacker* » fait ici référence à des individus qui ont une passion pour l'informatique et qui essaient de comprendre comment les choses fonctionnent, de savoir ce qui se cache dans les mécaniques diverses, et de les réparer ou de les améliorer. Ce terme ne doit pas être confondu avec celui de « pirate » informatique ou de « *cracker* ». Dans la version française de l'essai « La cathédrale et le bazar », Blondeel remplace le terme de « *hacker* » par celui de « bidouilleur »

du libre en général » (*ibid*.). En fin de compte, tant que les utilisateurs co-développeurs sont assez nombreux, le logiciel ne cesse de s'améliorer.

2.1.3.2 Les « Free-Rider »

Pour les simples utilisateurs (*free-rider*), ce sont : la gratuité du libre, le moyen d'échapper à l'enfermement des « logiciels propriétaires », la variété des produits proposés et la liberté de choix qui importent. Pour ces utilisateurs, le libre accès au code source ne revêt aucun intérêt particulier. Ils bénéficient tout simplement « des résultats des efforts collectifs de développement [des co-développeurs], sans qu'aucune contrepartie ne soit exigée de leur part ni sous forme pécuniaire ni sous forme de contribution à l'effort collectif » (Desbois, 1999, p. 5). On se trouve par conséquent « dans une situation de *free-riding* collectif » (*ibid*.) pouvant mettre en péril le modèle du « bazar ». Mais, comme le soulignent Von Hippel et Von Krogh (2003, p. 20) le *free-riding* n'est pas un problème, car s'ils ne contribuent pas au développement technique de l'outil, les *free-rider* interviennent tout de même dans le processus d'innovation en exprimant leurs besoins, en proposant de nouvelles idées et en identifiant des problèmes particuliers.

De ce fait, on peut dire qu'ils tiennent un rôle majeur dans le développement des applications. Benkeltoum, trouve qu'il s'agit d'une « vision simpliste de la dynamique concurrentielle dans l'open source ». Selon cet auteur, le « fait qu'un logiciel soit libre ne signifie pas que tout le monde peut l'améliorer et être concurrentiel en termes de services associés. Il s'agit, dit-il, d'un véritable mythe qu'il est nécessaire de briser » (2009, p. 196). Toutefois, comme le précise Bonneau, certaines études montrent « à quel point la participation des usagers au processus de développement d'une technologie est cruciale afin de bien comprendre leurs besoins et assurer le succès de l'implantation » (2012, p. 19). C'est le cas des travaux de Barki et Hartwick (1994) qui s'intéressent à la participation des utilisateurs dans le développement de système d'information. Il en est de même des recherches qui s'ancrent dans le courant du Design participatif (*Participatory Design*) des logiciels. Dans ce cadre, il est question d'intégrer les différents utilisateurs au processus de développement du logiciel en privilégiant le dialogue entre les utilisateurs et les développeurs.

De la sorte, la capacité des concepteurs et des développeurs de logiciels libres à être à l'écoute des utilisateurs qui proposent des améliorations est aussi un facteur déterminant pour ce mode de production. Mais, comme le souligne Bonneau la voix des utilisateurs « n'a pas

toujours un effet sur les propriétés matérielles de l'outil, puisque ce sont les développeurs qui ont le pouvoir de prendre les décisions relatives à ce qui sera implanté » (2010, p. 29)[65]. C'est ce qui ressort de l'étude menée par Livari (2010) qui s'intéresse à la participation des usagers non-développeurs dans le développement d'applications *Open Source.*

> *It seems that the non-developer users are not necessarily more empowered in the OSS development than in other development contexts. Even though the users are not totally silenced or ignored, a situation that could arise in commercial software development that can be carried out without any contact with actual users of the software, it might be that the voice of the OSS users is neither listened to even though the users have tried to contribute* (Livari, 2010, p. 126)

Dans le même ordre d'idée, Mateos-Garcia et Steinmueller (2003), indiquent que les utilisateurs qui sont techniquement moins informés, sont souvent considérés avec indifférence par les développeurs, ce qui peut expliquer les problèmes d'utilisabilité généralement associés aux projets *Open Source* :

> *In the pragmatic environment of 'code first and talk later' present in Open Source projects, users, who are less technically knowledgeable, are often regarded with indifference, derision or outright hostility. In the context of legitimate peripheral participation and distributed authority described above, the space for participation of individuals lacking a minimum of technical skills may be severely limited or even nonexistent. This ostracizing of users and the lack of attention to their feedback might be seen as one of the main reasons for the usability problems that are commonly associated with Open Source projects. Given the lack of alignment between the vision and intentions of developers regarding a particular program, and user's needs, and given that users lack the required skills to obtain the authority necessary to introduce their preferences in the program (by participating in its development), users' needs are neglected (or insufficiently addressed) (Mateos-Garcia & Steinmueller, 2003, p. 28).*

Mais, si la capacité d'agir des usagers-non développeurs peut être réduite, Bonneau note qu'ils affichent des capacités à « convaincre, argumenter et illustrer leurs idées par l'entremise

[65] Nous voyons ici, que le pouvoir est bien là. Il y a bien une séparation entre travail et pouvoir. Internet recrée des formes de hiérarchie.

des moyens de communication à leur disposition[66] » (2010, p. 29). Elle explique qu'il est essentiel de comprendre de quelle manière la « légitimité des usagers non-développeurs est construite et perçue par les développeurs à travers leurs interactions » (*ibid*., p. 30) notamment lorsque les développeurs et les utilisateurs se trouvent dans la même organisation.

Comme nous venons de l'observer, la littérature sur *l'Open Source* montre l'intérêt des chercheurs pour ce mode de développement. Cela se traduit par une multitude de recherches soucieuses d'expliquer et de décrire ce phénomène d'un point de vue scientifique.

2.1.4 Classification de quelques recherches sur *l'Open Source*

Selon le groupe de travail européen sur les logiciels libres[67] (2000), l'intérêt qui est porté aux logiciels libres est similaire à celui porté à la naissance du réseau Internet. Von Krogh et Von Hippel (2006, p. 977) proposent une caractérisation des recherches sur l'*Open Source* selon trois axes :

1) les motivations qui poussent les utilisateurs à contribuer au processus de développement (les motivations individuelles et celles des employés d'entreprises engagées dans le développement *Open Source* – la relation entre la motivation intrinsèque et extrinsèque dans le cas d'une contribution à un projet *Open Source* – les facteurs psychologiques et sociaux qui expliquent l'engagement dans des groupes d'utilisateurs, etc.) ;
2) la gourvernance, l'organisation et le processus d'innovation (la gouvernance de l'architecture du projet et du bien public – le fonctionnement et les types d'organisations dans des projets *Open Source* - le rôle pris par les contributeurs dans les projets – la coordination de l'innovation – les processus de développement et de maintenance – les facteurs expliquant l'évolution de l'architecture logicielle, etc.) ;
3) la dynamique concurrentielle (l'impact de l'*Open Source* sur la concurrence dans l'industrie du logiciel – les stratégies hybrides mêlant les solutions propriétaires et les les solutions *Open Source*, etc.).

À partir de cette caractérisation, Lindman, (2011, pp. 11-16) propose de classer les différentes recherches selon les thèmes et les concepts présentés dans le tableau suivant :

[66] Comme par exemple les forums de discussion
[67] The working group on Libre Software

Tableau 1 : Prominent OSS research themes (Lindman, 2011, p. 15)

Research theme	Developer motivation	Structure of community	Knowledge transfer model	Company - community relationship	OSS[68] licensing and business models
Topical research question	How to motivate OSS developers (often working without compensation)?	How are communities structured, in terms of contribution and communication?	How is knowledge transferred despite organizational borders?	How should companies build their relationship toward OSS communities and developers?	How should companies structure their revenue models and license their products?
Key concept	Developer	Community	Open innovation	Community management	Business model
Reference discipline	Psychology, Economics	Sociology (esp. network theory and SNA)	Economics, policy studies	Management and Information Systems Science	Management, Law
Example studies	Lerner & Tirole (2002); Hars and Ou (2002); Mustonen (2003); Hertel et al. (2003); Ke and Ping (2010)	Koch and Schneider (2002); Lin (2004); Shah (2006); Crowston and Howison, (2006); Bach and Carroll (2010)	Kogut and Metiu (2001); Von Hippel and Von Krogh (2003); Lanzara and Morner (2005)	West, (2003); Dahlander and Magnusson (2005, 2008); Shaikt and Cornford, (2010); Mehra et al. (2010)	Hecker (1999); Välimäki (2005); Osterwalder et al (2005); Fitzgerald (2006); Rajala et al. (2006); CasadesusMasanell and Ghemawat (2006); Bonaccorsi et al. (2006); Nagy et al. (2010)

[68] *Open Source Software.*

En définitive, ces recherches combinent différents niveaux d'analyse qui se concentrent sur l'individu (en termes de motivation) et sur l'organisation ou sur les communautés de développeurs et sur l'aspect économique de ce mode de production (Lindman, 2011, p. 16). Cependant, la littérature néglige l'aspect éducatif et, de ce fait, ne s'interroge pas sur les stratégies développées par les utilisateurs pour s'approprier les différentes applications libres implantées dans leur environnement professionnel. Afin de préciser nos objectifs de recherche, notons les spécificités se rapportant à l'usage des logiciels *Open Source* dans un contexte organisationnel.

2.1.5 L'usage des logiciciels *Open Source* dans un contexte organisationnel et/ou institutionnel

« L'informatique est en pleine mutation et le logiciel libre est l'un des catalyseurs de cette mutation ». C'est ce que révèle l'étude « Logiciel Libre France 2008 » menée par l'OPIIEC[69] au cours de l'année 2008. Selon cette étude, en France le logiciel libre représente 3,6% (1105 millions d'euros) de la demande en logiciels et services ». De plus, elle prévoit une forte croissance de ce marché[70] sur les quatre années suivant l'enquête (de 2009 à 2012), pour atteindre « près de 10% de la dépense en logiciels et services » (OPIIEC, 2008). Cette évolution est marquée par la rapide croissance des prestations relatives au logiciel libre (*cf.* figure suivante).

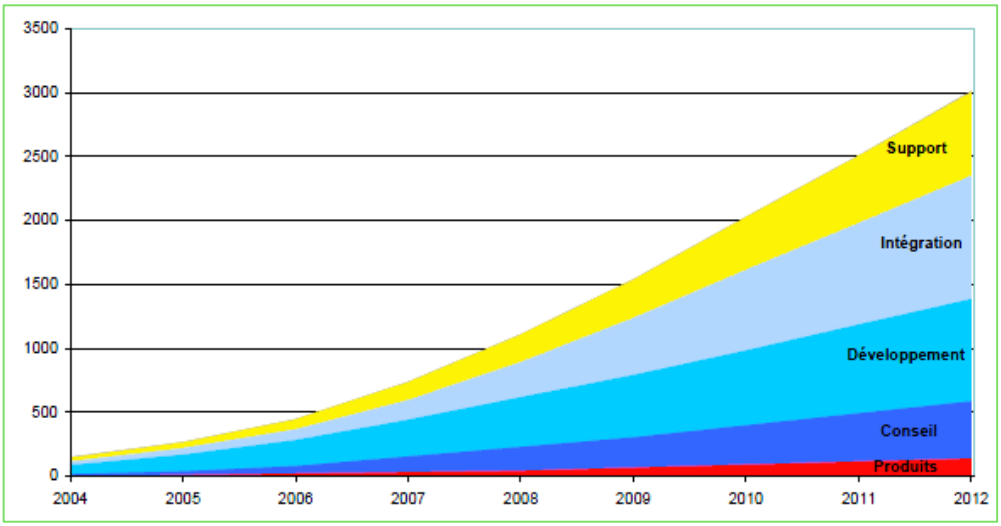

Figure 2 : Le marché du logiciel libre en France en millions

[69] Observatoire Paritaire des métiers de l'Informatique, de l'Ingénierie, des Etudes et du Conseil. Cette étude a été effectuée par le cabinet Pierre Audoin Consultants (PAC) auprès de 225 personnes (100 développeurs, 100 décideurs informatiques, et 25 dirigeants de Société Spécialisées en Logiciel Libre (SSLL)).

[70] 32,7% de croissance annuelle moyenne

Ce graphique dévoile « un marché qui mûrit rapidement, car souvent basé sur des technologies déjà banalisées ». Par ailleurs, notons que « les prestations d'intégration se développent fortement sous l'effet de la maturation de solutions packagées *Open Source* [et que] les phases de conseil et de support connaissent également une forte croissance » (*ibid*.).

Selon ce rapport, le secteur public était le plus ouvert au logiciel libre en 2007 (320 M€ soit 44% du marché total). Cela peut se justifier par des choix politiques et par l'activité d'équipes adeptes de ce type de développement. Les raisons évoquées sont : la réduction des coûts, la facilité de mutualisation entre les services de l'État, la réponse à des problématiques de standardisation et d'indépendance vis-à-vis des éditeurs de logiciels propriétaires, la réponse à des problématiques de maîtrise et de contrôle du code source, etc.

Cependant, si l'argument « logiciel libre » est devenu quasiment obligatoire sur une grande partie des appels d'offres (à fonctionnalités équivalentes/égales, le libre est privilégié), son adoption reste majoritairement limitée au poste de travail et plate-forme collaborative (OPIIEC, 2008, p. 23). Même s'il s'est déjà infiltré dans nos administrations, la position du gouvernement à l'égard du phénomène « *Open Source* » se trouve renforcée par la circulaire Ayrault (septembre 2012) qui préconise des « Orientations pour l'usage des logiciels libres dans l'administration ». Dans ce document sont soulignés les avantages intrinsèques à l'adoption de telles solutions « moindre coût, souplesse d'utilisation, levier de discussion avec les éditeurs ». Pour le Conseil National du Logiciel Libre[71] (CNLL) « cette circulaire constitue une avancée majeure pour le logiciel libre dans les systèmes d'information de l'État. (…) C'est un peu un *coming-out* du gouvernement sur le logiciel libre ! » (2012). Néanmoins, nous assistons à un revirement de situation puisqu'en juin dernier (2013) le gouvernement bascule et « le logiciel libre n'est plus une priorité pour le gouvernement dans son projet de loi de refonte de l'éducation [72]» (Chausson, 2013).

Selon l'INSEE (2011), le taux d'utilisation des systèmes d'exploitation libres augmente fortement avec la taille des entreprises. Ainsi, 21% des sociétés d'au moins dix personnes (de dix à deux cent quarante-neuf) recourent à ces systèmes alors que la proportion est de 50%

[71] Le CNLL regroupe dix associations dont deux pôles de compétitivité et représentent plus de trois cents PME acteurs de l'écosystème du logiciel libre en France, le CNLL est intervenu lors de la campagne présidentielle pour mettre en lumière les principaux défis politiques relatifs au logiciel libre.

[72] Il faut dire, souligne Chausson que « Syntec et l'Afdel, les deux instances représentatives des éditeurs et des SSII en France, dans une levée de bouclier magistrale, avaient décidé de monter au créneau. S'insurgeant contre cet amendement qu'ils jugeaient tour à tour discriminatoire, considérant comme une atteinte au principe de neutralité des achats des marchés publics, évoquant même «le risque de recours contentieux entre les opérateurs privés du secteur et les administrations» (2013).

pour les sociétés de deux cent cinquante personnes et plus. À l'inverse, l'utilisation des logiciels libres destinés à la bureautique est moins importante dans les entreprises de deux cent cinquante personnes et plus, puisque 33% d'entre elles les utilisent contre 43% pour celles de 10 à 249 personnes.

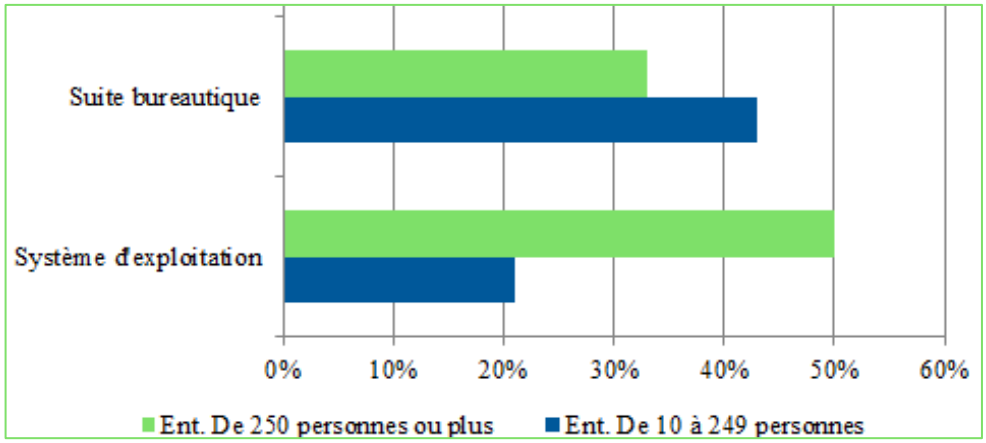

Figure 3 : Part des sociétés utilisant un système d'exploitation libre (INSEE, enquête TIC 2011, statistique publique[73])

Par ailleurs, une étude menée par Walli, Gynn, et Rotz, (2005, p. 4) auprès de 512 entreprises américaines a révélé que 87% d'entre elles utilisent des logiciels *Open Source.* Les auteurs de cette étude mentionnent que les principaux facteurs de motivation quant à l'utilisation de ce type d'applications sont ceux de la réduction des coûts informatiques ainsi que celle de leur dépendance à l'égard des sociétés des logiciels propriétaires. Bonneau (2010, p. 215) précise que les entreprises sont également attirées par leur évolution rapide grâce à la contribution et la mise en commun de nombreux individus et organisations. Nous le comprenons, le logiciel libre tient une place de plus en plus prépondérante au sein des organisations et/ou des institutions. Cela n'est pas sans conséquences sur l'évolution des métiers du numérique et plus particulièrement sur les besoins en compétences.

2.1.6 Attentes en termes de compétences Open Source

Une étude[74] menée par l'OPIIEC en juin 2013, montre que le niveau de compétences en termes d'*Open Source* varie en fonction des environnements technologiques concernés. Ainsi, en ce qui concerne « les logiciels d'infrastructures, les solutions bureautiques ou les systèmes d'exploitation, l'ensemble des sociétés intérrogées considèrent que les compétences sont

[73] Source : http://www.insee.fr/ - Consulté le 12/05/2012
[74] Méthodologie : 15 entretiens qualitatifs d'organisations utilisatrices et de fournisseurs de solutions informatiques - Enquête de terrain réalisée auprès de 250 organisations.

suffisantes ou très suffisantes ». En revanche, pour les « environnements Web ou les solutions de communication » les entreprises se répartissent de manière équivalente entre celles qui déclarent détenir un niveau suffisant de compétences et celles qui déclarent un niveau de compétences insuffisant. Enfin, concernant les progiciels de gestion et le décisionnel les compétences sont jugées majoritairement insuffisantes.

L'acquisition de ces compétences s'est majoritairement (84%) déroulée sur le tas. Un peu plus d'un tiers (32%) ont été acquises par le biais de formations professionnelles, 14% par des universités ou écoles d'ingénieurs généralistes et 7% par des universités ou écoles d'ingénieurs avec une spécialisation Open Source. Bien que certains environnements soient considérés comme suffisamment couverts en termes de compétences, cette étude fait ressortir des attentes de formation signe d'un besoin d'expertise (OPIIEC, 2013).

Tableau 2 : Besoins en formation selon l'OPIIEC (2013)

Utilisateurs[75]		**Fournisseurs**[76]	
Solutions bureautiques	10%	Linux	16%
Linux[77]	8,7%	Apache	12%
Java[78]	5%	Java	10%
Apache[79]	3%	NoSQL[80]	10%
HTML	3%		
Administration et supervision	3%		
Autres besoins			
PHP[81], MySQL[82], Tomcat, Javascript[83]. Cloud Computing[84] ou Hadoop[85]		Javascipt, Tomcat, MySQL - Cloud Computing ou Hadoop.	

[75] Le terme Utilisateurs se réfère à l'ensemble des organisations (entreprises et organisations publiques) utilisatrices de technologies informatiques (OPIIEC, 2013).

[76] Le terme Fournisseurs se réfère à l'ensemble des fournisseurs de solutions informatiques : services, logiciels, matériels (OPIIEC, 2013).

[77] Système d'exploitation à code source ouvert créé par Linus Torvald en 1991.

[78] Java : Langage de programmation conçu par Sun. Orienté objet, il permet de créer des applications qui viennent s'intégrer dans des pages HTML.

[79] Serveur le plus populaire du Web. Il est distribué selon les termes de la licence Apache.

[80] Not only SQL. Système de gestion de base de données (SGBD) qui n'est plus fondée sur l'architecture classique des bases relationnelles.

[81] Langage de programmation principalement utilisé pour produire des pages Web dynamiques

[82] Système de gestion de base de données (SGBD)

[83] Langage basé sur Java permettant au navigateur Web et à d'autres applications de faire tourner des programmes sur les machines des utilisateurs

[84] Ou Informatique en nuage. Désigne un mode de traitement de données dont l'exploitation s'effectue par Internet sous la forme de services fournis par un prestataire

[85] *Framework Open Source* conçu pour réaliser des traitements sur des volumes de données massifs. Il s'inscrit donc typiquement sur le terrain du *Big Data*. Géré sous l'égide de la fondation Apache, Hadoop est écrit en Java.

Au-delà de ces compétences technologiques, le rapport précise que d'autres compétences favorisant une meilleure maîtrise des environnements *Open Source* sont attendues par les entreprises. Par exemple : des compétences sur les projets d'intégration et les projets complexes – la gestion des licences – des capacités à travailler avec des communautés.

Ce manque de compétences est sûrement lié à la pauvreté de l'offre de formation en la matière. En effet, comme le montre le tableau suivant, en France, les établissements spécialisés autour du logiciel libre sont très peu nombreux.

Tableau 3 : Etablissements proposant une formation *Open Source* (OPIIEC, 2013, p. 26)

Site d'enseignement	Nom de la formation	Formation spécialisée logiciel libre	Nombre d'étudiants formés
Université du Littoral	Master Ingénierie du Logiciel libre	Oui	20
Université d'Angers	Licence Pro Open Source	Oui	20
	Licence Pro Logiciels libres et propriétaires pour les systèmes, réseaux et bases de données	Non	20
	Master Pro SILI	Oui	17
Université de Lorraine	Licence Professionnelle ASRALL		23
Université Evry Val d'Essonne	Master Miage	Non, mais partenariat avec Mozilla	20
Université Lyon 2 (ICOM)	Licence Pro CoLibre	Oui	20
Université Bordeaux 1	Licence Administrateur et développeur de systèmes informatiques sous licences libres et hybrides (ADSILLH)	Oui	20 Ouverture prévue en septembre 2014

Mais, si selon ce rapport, il n'y a que cinq diplômes « étiquetés » logiciel libre, il est à noter que nombreux sont les établissements qui en assurent la formation. Ainsi, la presque totalité des étudiants est « confrontée à l'*Open Source*, à minima dans un objectif pédagogique ». Par ailleurs, nous observons que certains projets *Open Source* sont initiés par des écoles ou universités (voir le tableau suivant).

Tableau 4 : Projets et initiatives Open Source dans les écoles et/ou universités (OPIIEC, 2013, p. 27)

Site d'enseignement	Nom du projet	Descriptif	Nb. d'étudiants impliqués
Centrale Paris	Projet VLC	Développement de Videolan (solution de lecture et de diffusion audio et vidéo)	35
Centrale Nantes UTBM EPITA /EPITECH	OpenOffice. org Education project	Contribution au développement d'OpenOffice par et pour les étudiants	NA
UPCM Université Paris VII	IRIL	Initiative pour la Recherche et pour l'Innovation sur le Logiciel Libre	NA
Université Paris VII	Centre de formation Logiciels Libres	Formation aux logiciels libres de tous les personnels des universités	NA

Pour terminer, cette étude montre que le logiciel libre « souffre encore auprès des étudiants d'une image d'environnement réservé aux *bidouilleurs* ». Ainsi, la crainte des étudiants d'être catalogués de la sorte « pousse les établissements à ne pas promouvoir les approches trop techniques » (OPIIEC, 2013, p. 28).

Au regard de ces études, nous comprenons qu'il y a un fort décalage entre la progression que connaît l'*Open Source* depuis quelques années et l'offre de formation en la matière. Ceci entraîne un manque d'expertise et conduit la plupart des individus à se former sur le tas. Nous pouvons donc poser l'hypothèse qu'il en est de même pour les personnes qui s'occupent de la mise en œuvre de la plate-forme d'enseignement en ligne à code source ouvert *Moodle*[86].

[86] Moodle est une marque déposée dans de nombreux pays.

2.2 Les plates-formes d'enseignement en ligne à code source ouvert : le cas de la plate-forme *Moodle*

À l'image des logiciels libres, les plates-formes d'enseignement en ligne *Open Source* entrent à grand train « dans le monde de la formation et de l'éducation amenant les acteurs à changer leurs pratiques, à proposer des ressources multimédias, à introduire de la distance dans les dispositifs, à mettre les réseaux sociaux au service d'une relation pédagogique renouvelée » (Loiget & Peuvrier, 2012). Outre leur gratuité, elles ont l'avantage d'être disponibles sur le réseau Internet où elles sont directement téléchargeables et il n'est pas besoin (en théorie) d'intermédiaires pour les installer. Donc, elles répondent particulièrement aux besoins des organisations et/ou des institutions qui souhaitent développer des dispositifs d'enseignement en ligne de manière indépendante.

Moodle est un système *Open Source* de gestion de cours *CMS* (*Content Management System)* également connu comme un *LMS*[87] (*Learning Management System*) ou encore comme un environnement d'apprentissage virtuel (VLE), permettant la mise en œuvre de sites d'enseignement en ligne.

Ses nombreuses fonctionnalités autorisent la création d'espaces de cours dynamiques[88] pouvant être agrémentés de nombreuses activités (tests, jeux de questions-réponses, sondages, etc.) et par des modules de communication et de collaboration tels qu'un forum, une messagerie interne et un Wiki. Notons ici que la plate-forme *Moodle* peut supporter des cours de type *MOOC*[89] (*massive open online course*). C'est le cas par exemple de l'université Australienne « UneOpen[90] » qui propose des cours gratuits en ligne avec une possibilité de valider des crédits pour l'obtention d'un diplôme universitaire[91].

[87] Terme utilisé pour désigner des systèmes qui organisent et donnent accès à des services d'apprentissage en ligne. Ces services comprennent généralement un contrôle d'accès, la fourniture de contenus d'apprentissage, des outils de communication, et d'organisation de groupes d'utilisateurs (Iskander, 2008, p. 59).

[88] Les espaces de cours de *Moodle* peuvent être enrichis grâce aux outils du Web 2.0 comme par exemple ceux destinés : à l'amélioration graphique - au stockage et au partage de ressources (flickr, dropbox, slideshare) - à la transformation , l'enrichissement ou la création de contenus médias – insérer des activités d'indexation et de veille, etc. Cela suppose l'acquisition de connaissances que nous dirons « collatérales » à l'utilisation de *Moodle.*

[89] Ce sont des cours en ligne ouvert à tous qui supportent un grand nombre de participants d'où le qualificatif « massif ». Pour en savoir plus voir l'article « Chronique des MOOC » rédigé par Cisel et Bruillard (2012).

[90] https://www.uneopen.com/

[91] Alors que l'inscription aux cours est gratuite, la validation des crédits ne l'est pas. La plupart des cours comportent un examen final qui coûte quelques centaines de dollars. Sont aussi proposées des sessions de cours non libres (1 heure payante) pour des groupes de huit personnes (maximum).

2.2.1 Genèse et évolution[92]

La conception de *Moodle* a été influencée par les travaux de Martin Dougiamas[93] dans le cadre de ses études doctorales sur les apports du constructivisme social dans la pédagogie en ligne.

Grâce au soutien d'une communauté (internationale) très active de bénévoles (utilisateurs ou développeurs) adeptes de l'initiative *Open Source* l'application *Moodle* est en perpétuelle évolution (de nouvelles fonctionnalités naissent quasi quotidiennement). Ainsi, à l'instar des organismes vivants, elle contribue à l'émergence de véritables écosystèmes[94] (communautés et réseaux d'utilisateurs) qui peuvent être vus comme des « systèmes ouverts auto-éco-organisés » fondés sur les connaissances, l'activité et les échanges entre utilisateurs. Cela se traduit par une évolution rapide et constante de *Moodle* puisque depuis l'année 2002, 142 versions[95] (ou mises à jour) ont été produites et mises à disposition des utilisateurs.

2.2.2 Distribution et déploiement

Distribuée sous la licence *GPL* telle que publiée par la *FSF*, l'application *Moodle* est disponible gratuitement. Tout le monde peut la télécharger et l'installer. De ce fait, le site communautaire *Moodle.org* a référencé 86 325[96] sites actifs dans 237 pays. En août 2013, le « Top 10 » diffusé par la communauté *Moodle* montre que ce sont les États-Unis qui occupent la première place du classement avec le plus grand nombre de sites enregistrés. Remarquons au passage que si la France affiche un nombre de sites relativement élevé (1 368), elle ne figure toujours pas dans le « Top 10 », malgré une nette progression par rapport à l'année 2007[97].

---

[92] Toutes les données (statistiques et autres) ont été recueillies sur le site Moodle.org (https://Moodle.org). La dernière mise à jour a été effectuée le 01/08/2013.

[93] Martin Dougiamas a été administrateur de la plate-forme WebCT (maintenant *Blackboard)* à *Curtin Technology University*, à Perth en Australie

[94] Selon Edgar Morin, l'écosystème est l'ensemble d'interactions au sein d'une unité géophysique contenant diverses populations vivantes constituant une unité complexe de caractère organisateur (1977, pp. 105-106)

[95] Source : http://docs.moodle.org/2x/fr/Historique_des_versions. Consulté le 02/08/2013. La dernière version 2.51 a été mise en ligne en juillet 2013 et la version 2.5.2 est en cours de développement ;

[96] Il est à noter que tous les sites ne figurent sûrement pas dans ce référencement puisque leur inscription ne revêt pas un caractère obligatoire.

[97] Le développement de *Moodle* est considérable car en 2007, dans sa thèse de doctorat Bos-Ciussi ne relevait que 19 sites enregistrés (p.97).

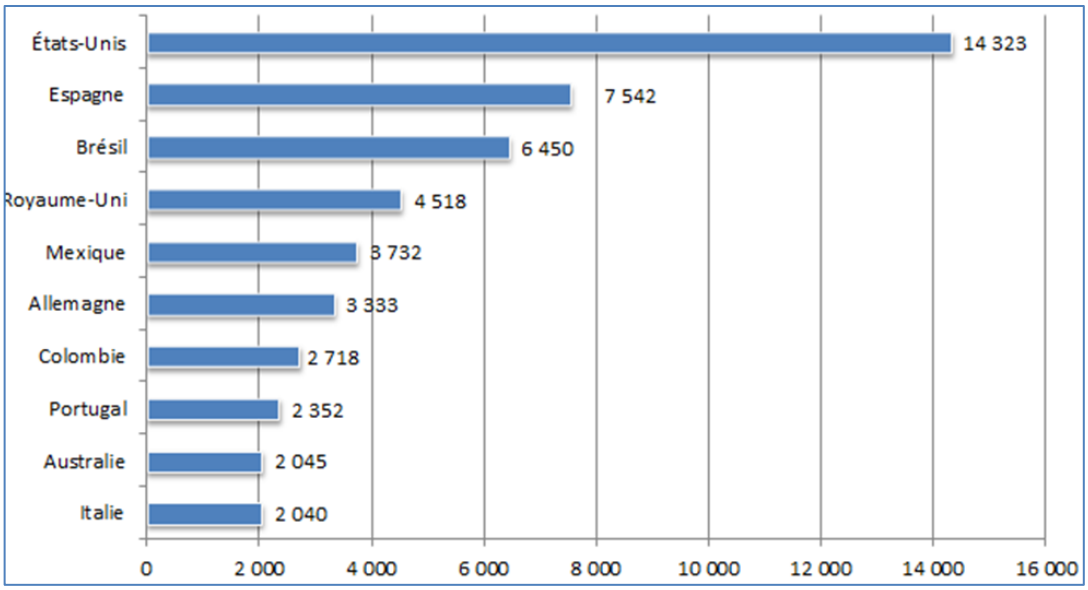

Figure 4 : Top 10 des sites enregistrés au niveau international

Au niveau mondial, le site Moodle.org est l'un des plus visités (7408[e] - *PageRank*[98] *Google 9/10*). En ligne depuis plus de onze ans[99], il compte 34 155 visiteurs uniques par jour pour 76 120 pages vues (WebStator, 2013). Par ailleurs, nous recensons un nombre important (17 657 577) de *backlinks*[100] *(ou inlinks)* pointant vers celui-ci. Ces données nous renseignent sur la réputation du site et sur l'intérêt des internautes pour l'application *Moodle*.

En effet, l'application *Moodle* rassemble des milliers de personnes communément appelées « *Moodleurs* ». À ce jour, la communauté *Moodle* enregistre près de 72 928 333 utilisateurs (dont 1 297 402 enseignants). Certains sont de simples utilisateurs (*free-rider*) et d'autres des co-développeurs.

[98] Le *PageRank* est un indicateur permettant de classer les pages du Web (dur une échelle allant de 0 à 10) dans les résultats de recherche de Google. Ce système a été inventé par Larry Page, cofondateur de Google.

[99] Créé en avril 2002

[100] Les *backlinks* sont des liens hypertexte pointant vers un site Web ou vers une page d'un site Web. Pour évaluer le nombre de *backlinks* pointant vers *Moodle* nous avons utilisé l'outil *Bulk Backlink* proposé par MAJESTIC SEO : https://fr.majesticseo.com/.

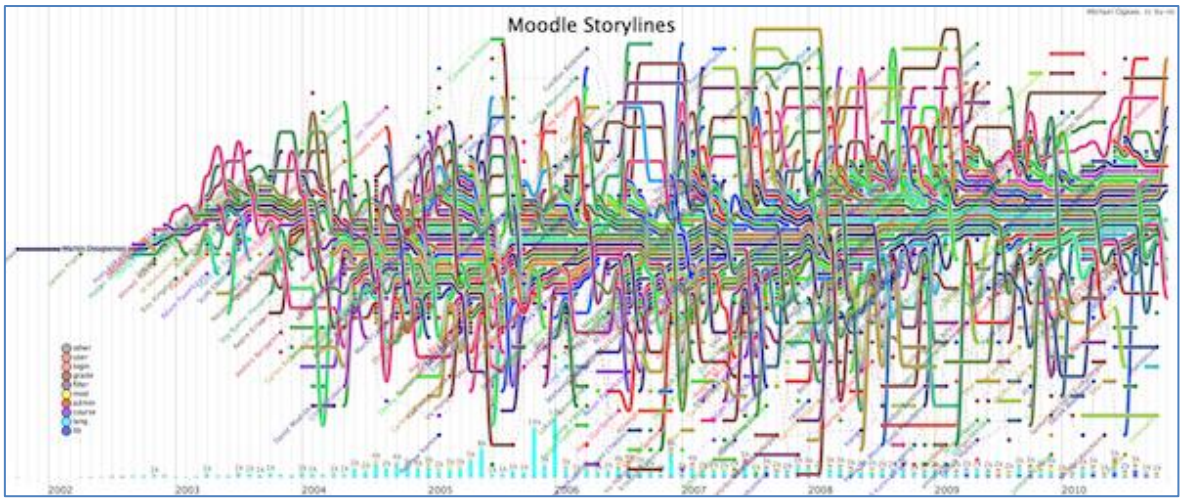

Figure 5 : Moodle story line (Martignoni, 2010)

La représentation graphique ci-dessus[101] rend compte de l'activité des « *Moodleurs* » en termes de dépôt de code source sur une période de huit années. L'axe horizontal représente le temps écoulé et les développeurs sont regroupés par mois en fonction des modifications apportées au code source. L'histogramme affiché au bas du graphique met en évidence le volume et le type de fichiers modifiés. Mais derrière le projet *Moodle*, se cache une entreprise.

2.2.3 *Moodle* un projet associatif :

Derrière le projet *Moodle*, se cache une association (*Moodle Pty Ltd* ou *Moodle HQ*) qui emploie une équipe[102] composée d'une trentaine de collaborateurs (y compris le fondateur et développeur principal Martin Dougiamas). Elle est responsable du développement et de la maintenance de l'application ainsi que de tous les sites communautaires de *Moodle.org*. Elle est également chargée de la certification des partenaires *Moodle* et de la protection de la marque.

D'un point de vue économique, le développement de *Moodle* est supporté par des donations faites par des clients qui ont besoin de développements spécifiques et qui sont prêts à payer pour cela, ainsi que par des *royalties* payées[103] par les partenaires certifiés *Moodle* (59 à ce jour). Ces entreprises fournissent des services tels que l'hébergement, la

[101] La version interactive du graphique peut être visualisée sur le blog de Martignoni : http://blog.martignoni.net/wp-content/uploads/2010/12/moodle.svg

[102] La plupart sont basées dans le bureau principal à Perth, en Australie et certains se trouvent en : Belgique, République tchèque, Nouvelle-Zélande et en Espagne

[103] 10% de leurs revenus.

personnalisation, le soutien, la formation et même la gestion complète d'un projet *Moodle*. La certification leur apporte quelques avantages par exemple :

- La possibilité d'utiliser les marques de *Moodle* dans leurs affaires
- Un accès direct au support et à la promotion via le *Moodle* HQ
- La priorité sur la résolution de *bugs*
- La priorité sur les notifications concernant les nouvelles versions et les problèmes de sécurité
- L'assurance de faire partie d'un réseau mondial de professionnels partageant les mêmes idées.

Outre ces considérations économiques, si elle peut paraître simple, la mise en œuvre de *Moodle* peut se révéler complexe.

2.2.4 Une mise en œuvre à la fois simple et complexe

Même si le concept général prône une facilité d'installation et d'administration, la mise en œuvre de cette application peut se révéler complexe, car en plus de l'insertion de nombreuses ressources et d'activités pédagogiques, elle s'appuie sur des technologies complexes par exemple le langage de scripts PHP permettant la production de pages Web dynamiques, le langage CSS (*Cascading Style Sheets* ou feuilles de style en cascade) facilitant l'application d'une mise en page (police, couleurs, etc.) à des pages Web ou sur un système de gestion de base de données relationnelles (SGBDR).

Au regard de ces informations, nous pouvons émettre l'hypothèse que l'inscription dans un projet d'enseignement en ligne *via* la plate-forme *Moodle* requiert de multiples compétences en termes de ressources humaines (des administrateurs, des enseignants, des webmasters, des développeurs, etc.). Or, comme nous le verrons par la suite, ce n'est pas toujours le cas. En effet, une observation de l'espace communautaire *Moodle* associée à une première lecture flottante des échanges se déroulant sur les forums de discussion en ligne de la communauté montre que dans un tel environnement, une même personne peut occuper plusieurs de ces fonctions et de ce fait peut être conduite à faire face à des problèmes opérationnels et à des imprévus, par exemple la gestion de l'application elle-même, le paramétrage des espaces de cours, la mise à jour de l'application, l'adaptation de scripts, etc.

Nous comprenons alors que les pratiques liées à l'activité sur *Moodle* nécessitent nécessairement la mobilisation et la construction de nouveaux savoirs et de nouvelles compétences. En conséquence, pour accompagner et soutenir ses membres dans leur activité,

la communauté *Moodle* met à leur disposition des artéfacts réifiés tels qu'un glossaire terminologique, une documentation, un guide, etc. ainsi que des espaces de communication (synchrones et asynchrones) favorisant les interactions entre pairs[104].

De ce fait, nous retrouvons sur le site *Moodle*.org de nombreux espaces de discussion en ligne (plus ou moins féconds en termes d'échanges) portant sur différentes thématiques[105] et rassemblant près de 75 287 280 participants à l'origine de 130 081 362 messages. La présence de ces traces écrites témoigne selon nos hypothèses de formes participatives contribuant au processus de développement professionnel des individus.

Enfin, si c'est le forum de discussion anglais « *Using Moodle* » qui est le plus actif, du côté francophone il en existe aussi deux principaux :

- le forum orienté vers l'aspect pédagogique de l'activité « Assistance pédagogique » et qui est destiné à fournir de l'aide dans le domaine pédagogique, lié à l'introduction de *Moodle* dans un enseignement ;
- le forum orienté vers l'aspect technique de l'activité « Assistance technique » qui attire la majorité des usagers francophones. Il est plus particulièrement destiné à apporter de l'aide sur les aspects techniques de l'application. Par exemple, l'installation, la migration, l'administration, le fonctionnement, les problèmes courants, etc.

De ces informations, il ressort que l'espace communautaire *Moodle* est un terrain privilégié pour l'étude du développement professionnel et que les outils de communication qu'il propose (notamment les forums de discussion en ligne) présentent un intérêt méthodologique que nous exposerons dans la troisième partie de ce travail (*cf.* p. 150).

[104] La communauté organise également des conférences annuelles « *Moodle*Moot ». Les éditions francophones 2005 et 2006 ont eu lieu à l'ENST (École Nationale Supérieure des Télécommunications) de Brest, celle de 2007 s'est déroulée à l'IUT (Institut universitaire de technologie) Paul Sabatier de Castres. En 2008 c'est l'École Nationale Vétérinaire de Nantes qui a reçu les conférenciers, en 2009 c'était au tour de l'INSA (Institut National des Sciences Appliquées) de Lyon. Pour sa sixième édition (2010), le *Moodle*Moot a eu lieu à l'UTT (Université de Technologie de Troyes), en 2011 c'est l'Université Virtuelle de Tunis qui a reçu les *Moodleurs*. En juin 2012 la conférence a été organisée par l'Université de Nîmes. Enfin, en 2013 (du 5 au 7 juin), c'est l'Université de Bordeaux qui a eu en charge l'organisation du colloque.

[105] Assistance pédagogique, traduction en français, développement, assistance technique, etc.

Synthèse du chapitre

Dans ce deuxième chapitre, nous nous sommes attachés à préciser l'objet de notre recherche. Pour ce faire, nous avons décidé de porter notre attention sur un cas spécifique, soit les logiciels à code source ouvert et plus précisément sur la plate-forme d'enseignement en ligne *Moodle*. Dans un souci de compréhension, proposons une approche de l'environnement du logiciel libre. Ce type d'application s'inscrit dans un mouvement (le mouvement du libre) basé sur quatre libertés qui permettent : d'exécuter le programme pour tous les usages, d'en étudier le fonctionnement et de le modifier pour l'adapter à ses besoins, de l'améliorer et d'en redistribuer des copies.

Nous observons que le mode de développement et de production de style « bazar » des logiciels libres est vecteur d'une participation sociale rendue possible par l'avènement d'Internet propulsant de fait les individus au rang de co-développeurs. Ceci soulève la question du rôle joué par les usagers dans le processus d'innovation. La revue de littérature qui s'intéresse à la participation des usagers aux processus de développement de logiciels libres, montre que ceux-ci peuvent faire l'objet d'une catégorisation : les « co-développeurs » et les « free-rider » dont les intérêts et les motivations diffèrent. De nombreuses études s'intéressent à l'*Open Source* elles s'orientent selon trois axes : la motivation qui pousse les utilisateurs à contribuer au processus de développement, la gouvernance de l'architecture du projet, la dynamique concurrentielle. De manière générale, ce corpus d'études combine différents niveaux d'analyse qui se concentrent sur l'individu, sur les communautés de développeurs, sur l'organisation, sur l'aspect économique ou sur le mode de développement et de production.

Dans le cadre de ce travail, ce sont les utilisateurs (*free-rider*) qui nous intéressent. Plus précisément, ce sont les stratégies qu'ils développent pour s'approprier les différentes applications à code source ouvert déployées dans leur environnement professionnel. Cependant, nous avons mis en évidence que la littérature sur l'*Open Source* néglige l'aspect éducatif et ne s'interroge pas sur ces processus d'appropriation. Or, nous relevons qu'ils se déroulent majoritairement sur le tas. Toutefois, nous n'avons pas de renseignements sur la manière dont ils se déroulent, ni sur les savoirs qui sont en jeu dans ce contexte précis. Cette question est cruciale, car, rappelons-le, dans la société cognitive qui est la nôtre, les organisations attendent de leurs acteurs qu'ils aient la capacité d'apprendre par la pratique, qu'ils se forment tout au long de leur vie et qu'ils adoptent la posture de « travailleurs du savoir » réactualisant sans cesse leurs connaissances. En bref, il est attendu qu'ils soient

acteurs de leur propre développement professionnel. Se pose donc de manière corollaire la question de la définition de ce développement professionnel.