

Produire du logiciel libre

Nicolas Jullien

ENST Bretagne / ICI

Nicolas.Jullien@enst-bretagne.fr

Impact du logiciel libre sur la production des technologies d'utilisation ?

Problème de la production d'un bien public hors de la sphère marchande. Trois questions :

- pourquoi produire un logiciel pour le mettre en libre ?
- pourquoi contribuer au développement d'un logiciel libre alors qu'on peut simplement l'utiliser ?
- est-ce que l'utilisation et les contributions éventuelles ne vont pas nuire à l'efficacité de la production de logiciel libre (encombrement) ?

⇒ problème classique du passager clandestin (Olson).

⇒ on retrouve les 3 étapes de la production d'un bien public (Heckarthorn[1996]).

« Except in the smallest or most highly centralized systems, collective action is organized incrementally. During **the initial phase**, either zealots (Coleman 1990, p 490) or a critical mass (Marwell and Oliver 1993) make the initial contributions. During **the intermediate phase**, the ranks of contributing actors continue to grow because of greater marginal returns to contributors, strategic interaction, or the operation of collective sanctions. Finally, during **the mature phase**, the limits of collective good production are approached because of physical constraints on further production of a dwindling pool of additional contributors. [...] The process of collective action also entails changes in the value of what may be termed the local collective good that each subgroup potentially produces. This is the portion of the global collective good that can be produced by the subgroup, so it reflects the marginal gains attainable through contributions by the local group's members. During the start-up phase, the value of the collective good is low because initially the slope of the production function is almost flat. During the intermediate phase, marginal gains increase, and so too does the value of the local collective good. Finally, during the mature phase, the slope decreases, reflecting diminishing marginal returns, so the value of the local collective good declines. Thus, organization of collective action is characterized by an increase in the value of the local collective good followed by a decline. [...] » (Heckathorn 1996, p 273-274)

la demande de technologies d'utilisation

On peut classer les demandes de technologies d'utilisation suivant trois axes :

- la spécificité de la demande (hétérogénéité) ;
- les externalités de réseau (directes ou indirectes) ;
- la dynamique d'évolution du logiciel/des besoins en terme de technologies d'utilisation).

Il faut mettre les caractéristiques de cette demande en perspective avec la production des technologies d'utilisation, composées de logiciel et de service.

quelles incitations à produire du logiciel libre ?

Deux types de récompenses :

- deux récompenses de court terme, l'acceptation d'une contribution à un logiciel libre et la critique d'une contribution ;
- une récompense de long terme, l'acquisition d'une certaine renommée parmi les producteurs de logiciels libres.

Il faut étudier comment ces récompenses sont valorisables et valorisées par les agents.

Il faut s'interroger sur les valorisations financières (ou non) et sur la l'horizon temporel de ce retour.

Différentes incitations pour différents types d'agents

- Il faut s'intéresser aux différents producteurs de logiciels/technologies d'utilisation :
 - deux types d'utilisateurs (marchands ou non-marchands) ;
 - deux types de producteurs (producteurs de logiciel et producteurs de services).
- les incitations joueront différemment suivant les utilisateurs et aussi suivant les caractéristiques des logiciels.

L'utilisateurs non-marchand

- Il est traditionnellement vu comme le producteur de logiciel libre.
- S'il dévoile le code, au pire, personne ne s'y intéressera et son coût sera identique au fait de le laisser fermé et secret.
- Il peut espérer, des récompenses de la divulgation, (signal envers des recruteurs, Lerner & Tirole[2000], transfert du coût de maintenance à la communauté et orientation du logiciel vers ses besoins)

=> contributeurs seulement chez les utilisateurs experts (designers/sophistiqués), plutôt sur les logiciels coûteux à maintenir (dynamique d'évolution forte, logiciels liés). Pas vraiment de différence entre production d'un logiciel libre ou contribution à un logiciel (si ce n'est le degré d'effort)

L'utilisateur marchand

- Cas de l'utilisateur-innovateur, développé par von Hippel.
- Coût de la divulgation plus élevé que l'utilisateur précédent, car elle peut profiter à ses concurrents (si ce n'est pas le cas, on est ramené au cas précédent).
- Vrai arbitrage diffusion/secret.

L'utilisateur marchand (2)

	Variation du coût de développement en libre par rapport au coût de développement propriétaire	Variation des recettes attendues du libre par rapport aux recettes attendues du secret
Impact positif	-- publicité sur la mise en libre du logiciel et introduction du logiciel dans une distribution officielle.	-rapidité avec laquelle ce développement sera réalisé par d'autres - orientation du logiciel vers ses besoins
Impact négatif	-importance des coûts de maintenance -facilité pour recruter une communauté de développement	- importance stratégique de la contribution pour l'entreprise

L'utilisateur marchand (3)

Les éléments ayant un impact sur la divulgation sont :

- la dynamique de production du logiciel (plus le logiciel est en évolution rapide, plus on a intérêt à divulguer) ;
- l'importance stratégique du logiciel pour l'entreprise (plus c'est important, moins on divulgue).

On retrouve des contributions d'entreprises utilisatrices dans des logiciels évoluant vite et assez génériques pour que leur contribution soit neutre du point de vue de l'avantage concurrentiel.

Le producteur : de nombreuses incitations à contribuer

	Type d'impact	Explication du mécanisme
Impact sur les coûts de production	- maintenance & amélioration de la contribution pris en charge par d'autres.	
	- possibilité d'améliorer la contribution en tenant compte commentaires des utilisateurs	
	- signal en direction de recrutables	fait connaître le niveau de l'entreprise, la qualité technique des groupes de programmeurs.
	- facilité pour recruter une communauté d'utilisateurs développeurs	conséquence de la renommée. N'a un impact que si l'entreprise cherche à développer d'autres projets de libre.
Impact sur les recettes	- signal pour les clients	baisse des asymétries d'information, construction d'une marque
	- extension de capacités du programme	permet d'élargir la taille potentielle du marché en intéressant de nouveaux utilisateurs au logiciel
	- orientation du logiciel vers ses besoins	permet de mieux répondre aux demandes sur ses produits propres
Impact sur les recettes et les coûts de production	- augmentation de l'expertise technique générale et de la connaissance du logiciel en particulier	effet de réputation (asymétrie d'information) et meilleure efficacité pour répondre à des problèmes posés par des clients ou pour proposer de nouvelles extensions.
	- création d'un bien public industriel	création d'un concurrent un logiciel établi ou création d'un standard pour la profession.

Le producteur de logiciel : renforcer son offre

Contributions de producteurs qui ne sont pas à l'origine du logiciel libre (Corel/Wine, SUN, IBM/Apache).

Trois types d'incitations expliquent la contribution :

- orienter le logiciel vers ses besoins ;
- en augmentant s'il le faut ses capacités ;
- envoyer un signal au client sur la compatibilité du logiciel avec celui que l'on vend.

=> subventions croisées entre logiciels (contribue aussi à renforcer le standard en augmentant l'offre de logiciels liés (effet classique en informatique))

Le producteur de logiciel : renforcer un standard dans une concurrence entre standard

- Les challengers peuvent avoir intérêt à subventionner un standard concurrent au standard dominant.
- Surtout, les entreprises qui dépendent d'un standard pour réaliser leur activité (comme SUN avec Apache) ont intérêt à ce que ce standard soit le plus ouvert possible, pour ne pas dépendre de la stratégie de l'entreprise qui fournit le standard.

Analyse

Comportement classique en informatique où, pour promouvoir son produit), il faut :

- « établir des alliances avec des co-producteurs » ;
- « sponsoriser les premiers utilisateurs et les premiers produits complémentaires » Genthon[2000].

En terme de type de technologies d'utilisation, les produits les plus susceptibles de recevoir ce type de contribution sont :

- les produits génériques, les plates-formes, avec un certain degré d'externalités de réseau (ce sont elles qu'il faut adapter à ses offres logicielles) ;
- des produits avec une dynamique de demande forte (obligation pour entreprises de contribuer au développement, pour transférer une partie de la maintenance de leur contributions et pour garantir que le logiciel reste orienté vers leurs besoins).

C'est la dynamique de l'innovation qui pousse à innover.

La mise en libre d'un logiciel propriétaire

- même stratégie, favoriser une offre propriétaire de logiciel complémentaire (côté gratuit du logiciel libre, course à l'audience, exemple de Zope)
- mais le fait de subventionner un logiciel pour pouvoir vendre un autre logiciel, complémentaire, n'est pas nouveau.

Si l'on comprend bien la gratuité, on peut s'interroger sur l'avantage concurrentiel que procure la mise en libre (cas Netscape vs Microsoft).

La mise en libre d'un logiciel propriétaire (2)

- externalisation d'une partie des coûts de production du logiciel ;
- augmentation de la réactivité vis-à-vis des besoins des utilisateurs => produit supérieur en terme de fonctionnalités, d'adéquation aux besoins.

Netscape :

- sans doute pas assez innovant, pas vraiment à un besoin non satisfait (navigateurs web de Gnome et KDE déjà créés, ``from scratch" , plus faciles)
- licence pas vraiment libre au départ (recrutement plus difficile) ;
- du côté de la demande, crainte de l'effet de lock-in dans un standard (effetUnix).

Principale raison de la création de la FSF, et de la licence GPL

La mise en libre d'un logiciel

propriétaire (3)

Deux autres incitations possibles :

- effet de signal en direction des recrutables (peut renforcer une stratégie de collaboration avec des communautés de producteurs de logiciels libres mais pas expliquer seule la mise en libre d'un logiciel) ;
- effet de signal en direction des clients capable d'évaluer la contribution de l'entreprise au logiciel libre. Efficace lorsqu'on s'adresse à des utilisateurs ``designers'', voir à des utilisateurs ``sophistiqués''(renforcent l'intérêt de produire du logiciel en libre lorsque la population d'utilisateurs-développeurs est importante). Nécessité de tiers de confiance pour évaluer la contribution d'une entreprises (utilisateurs naïfs)

=> problème de l'établissement des réseaux pour évaluer et garantir la qualités (voir services).

La mise en libre d'un logiciel propriétaire (4)

Stratégie des producteurs : développer des standards ouverts, (concurrer un standard déjà existant ou imposer un standard favorable à leur offre de logiciel).

Contributions d'autant plus importantes que le logiciel évolue vite(dynamique de la demande importante).

Dans l'autre sens, logiciel mis en libre d'autant plus "hacké" que la demande est dynamique, que les développeurs-utilisateurs sont nombreux et ont des besoins importants.

⇒ les entreprises devraient donc contribuer à des standards libres et proposer des logiciels libres qui évoluent très vite.

⇒ Problème : si la distribution du logiciel facilite le recrutement, elle facilite aussi la création de concurrents (voir entreprises de service).

Le service (1)

Trois types de finalités dans l'usage d'un service (trois grandes catégories de services) :

- services destinés à faciliter ou accompagner la mise en oeuvre de systèmes informatiques (conseil, assistance, développement de programmes, intégration de systèmes) : services ``supports" (Dréan) ou services de ``mise à disposition de capacités humaines" (de compétences, Gadray) ;
- services dits ``de substitution" (répondre à une demande d'alternative à l'acquisition de produits informatiques, services bureau des années 70 et 80, aux services de ``facility management", d'infogérance). Mise à disposition de capacités techniques entretenues (Gadray) ;
- services liés à l'informatique (services applicatifs), ou services informatisés (services qui préexistaient à l'arrivée de l'informatique, qui se sont informatisés). Pas abordés ici.

Le service (2)

- Problème de l'industrie des services : asymétries d'information (sélection adverse et/puis hasard moral).
- Services de support : services intellectuels, investissements faibles, coûts d'entrée bas (acquisition individuelle des compétences nécessaires) => secteur dispersé et marges faibles (Dréan[1996]) :
 - Assistance et conseil : engagement de moyens uniquement, facturation au temps et à la compétence (concurrence importante car peu de barrières à l'entrée et fort problème de sélection adverse) ;
 - développement de logiciel, ingénierie/intégration de systèmes : existence d'un produit médian (=> obligation de résultat spécifié par contrat), facturation +/- au résultat, mais, dans la réalité, codéfinition du système au cours du temps ; nouveaux entrants qui tirent les prix vers le bas (à nouveau sélection adverse puis hasard moral, voir De Bandt[1998])

Le service (3)

Le logiciel libre, un système pour signaler sa qualité (sélection adverse) ?

- importance des facteurs de renommé : construction d'une marque et possibilité **d'évaluation par les clients** ;
- ouverture du code limitation des problèmes externes (compatibilité) ;
- dans certains cas, maîtrise du logiciel, permet de garantir l'adaptation aux clients.

⇒ engagement répété et annoncé des entreprises du logiciel libre.

⇒ révision du code par des ``hommes de l'art" **Ⓢ** système de certification de qualité (voir Baudry[1995]).

Mais :

- Problème d'information en direction des utilisateurs naïfs et experts ;
- problème de l'évaluation de l'ensemble des logiciels. Nécessité de la construction de tiers de confiance (journaux, sites regroupant les contribution).

Le service (4)

Les logiciels libres, un système pour garantir son service (hasard moral)

- en utilisant des logiciels libres, on facilite la concurrence ;
- en utilisant des logiciels libres, on diminue aussi les coûts de sortie de la relation (logiciels connus) ;
- en contribuant au développement libre, on garanti au client la pérenité de la solution.

=> système d'engagement à la qualité de service (norme unilatérale, du type de celle des avocats).

Engagement d'autant plus crédible que le logiciel est un standard (logiciel connu) et la dynamique d'innovation est forte (il faut que la solution reste compatible avec le logiciel) ou que le logiciel est contrôlé par une entreprise (problème du ``lock-in'' sur un standard propriétaire).