

The art of (free software) war

Gabès Jean

1. Introduction sur ce texte

Ce texte fait suite à la lecture de trois ouvrages : « Richard Stallman et la révolution du logiciel libre », « L'art de la guerre » de Sun Tzu, et « Traité de l'efficacité » de François Jullien. Si le premier est une œuvre bien connue des acteurs du monde du libre, les deux dernières méritent également d'être lues. L'art de la guerre est le grand classique sur la vision des affrontements du point de vue chinois. Le second en est un approfondissement. Il présente les pensées occidentales et orientales de l'efficacité de la guerre, mais aussi plus généralement de toute activité humaine. Ce sont justement ces idées que je vais tenter d'appliquer au monde du libre. Je n'ai fait que tenter d'appliquer ce qu'il explique si bien à notre environnement.

2. En quoi le terme guerre est-il adapté à l'environnement des logiciels?

Le titre l'a déjà annoncé : nous allons adapter l'art de la guerre à l'environnement du Libre. Mais pourquoi *guerre*? Elle constitue le summum des affrontements antagonistes. L'environnement du libre, et plus généralement des logiciel, est particulier. Il représente à la fois ce qui est le plus binaire et platement théorique par son côté technique de codes et d'algorithmes, et sur ce point il est soumis aux règles strictes de la production. Il est également soumis à des conflits car hautement concurrentiel. Les projets s'affrontent sur des aspects techniques motivés par un environnement très lucratif.

Nous pouvons rajouter que l'accès au code source est également important car il rajoute une nouvelle dimension possible au conflit : en plus de la lutte entre les logiciels, nous pouvons avoir une lutte en leur sein. Là encore, les considérations techniques ne sont pas les seules à être importantes, loin de là. On pourrait presque dire qu'elles sont bien souvent de second ordre. Lorsque des conflits arrivent au sein des projets, ils sont bien souvent dû à une mauvaise entente entre les personnes, ou bien un problème d'orientation. Je ne traiterai pas ce genre de « guerre » qui est plus de l'ordre de la psychologie et des orgueils mal placés qu'autre chose.

3. Qui est l'*ennemi* de cette guerre ?

Il faut évoquer nos belligérants pour notre guerre des logiciels. D'un côté, nous auront les projets du Libre. De cet aspect *Libre* découle ce que nous pourrions appeler l'*ennemi* : les logiciels *privateurs*. Ce terme est plus adapté que *propriétaires* tel qu'il a été utilisé pendant des années comme nous le verrons par la suite.

Les intérêts financiers aidant, entre ces deux là, la guerre fait rage. L'auteur de ce document étant pro-libre, vous ne verrez pas ici un l'éloge des logiciels privateurs. C'est plutôt un texte sur comment le Libre peut (doit) prendre le dessus en étant le plus *efficace* dans cette guerre.

4. Différents modèles de théorie de la guerre dans le monde

Nous avons notre guerre de posée. Comme pour toute étude, partons de la théorie afin de mieux la comprendre avant d'aller dans les détails de son implémentation. Dans ce domaine, deux grandes

écoles s'affrontent. Nous avons d'un côté la pensée occidentale, dont le plus grand théoricien de la guerre est *Clausewitz*. De l'autre nous avons la Chine avec son célèbre *Art de la guerre* de *Sun Tzu*.

4.1 Modèle Occidental : modèle théorie/pratique, épopées et héroïsme

Il est intéressant de voir que les occidentaux ont du mal à définir une théorie de la guerre. Il y a bien des recettes pour la tactique d'attaque une fois sur le terrain, comme par exemple comment positionner ses troupes ou prévoir les ressources. Mais la stratégie, qui consiste à penser les engagements tactiques, n'a jamais été traitée efficacement. En fait pour Clausewitz elle est même in-modélisable. Pour lui il y a une trop grande part de hasard dans la guerre. Elle n'est pas prévisible, et c'est même sa principale caractéristique : « Ce qui fait la guerre c'est précisément cette distance inévitable qu'y prend le réel vis-à-vis de son modèle : penser la guerre, en somme, c'est penser comment elle est portée à trahir son concept ».

En fait, le problème de recherche de la théorie parfaite de Clausewitz est bien plus général que cette simple problématique de *guerre*. Il a tenté de la faire rentrer dans le schéma classique de la théorie-pratique, un héritage de la Grèce antique et de notre cher Aristote. Nous (occidentaux héritiers de ce mode de pensée) avons l'habitude de penser un modèle parfait, puis de le faire entrer dans les faits. Si cette méthode est idéale dans le mode *production*, elle n'est pas adaptée aux autres activités humaines. Le problème, et la guerre l'illustre bien, est que dans ce cas le réel est particulièrement mouvant. Le passage théorie-pratique n'est jamais parfait, voir même systématiquement éloigné de ce qui était prévu.

Ceci va amener notre civilisation à glorifier des héros spécialistes de l'action qui luttent contre cette mouvance, comme Achille dans l'Iliade. Ces héros ne peuvent réussir pleinement par eux même, du fait même de cette aspect aléatoire de la guerre. Dans notre cas, ce sont les dieux qui doivent pallier cela : leur omnipotence permet de ne plus *subir* l'aléatoire des affrontements, comme Ulysse qui ne peut rentrer chez lui sans le soutien d'Athéna.

Avec du recul, on voit que les écrits occidentaux classiques sur la guerre ne parlent pas réellement de sa *théorie*, mais simplement des *bonnes pratiques*. Ainsi, au lieu d'avoir penser la guerre, ils étudient et appliquent une sorte d'heuristique génétique en mélangeant des schémas qui ont déjà fonctionné afin d'affronter des situations totalement nouvelles. Si les engagements peuvent se ressembler, la stratégie à adopter est quant à elle tout le temps nouvelle.

4.2 Chine : utilisation de la propension de la situation

Les textes chinois anciens se piochent leurs idées dans une école de pensée bien différente de nos grecs : le *taoïsme*, l'art de rechercher le *tao*, la voie. Elle part du principe que toute action qui cherche à faire changer le réel de manière directe n'est pas idéale, car trop complexe et aléatoire. Il n'ont pas fait de distinctions entre un modèle parfait et le monde réel. Ils pensent qu'il faut faire avec ce que l'on a sous la main et ne pas chercher un idéal, mais plutôt améliorer ce qui est déjà réel. Sans cette distinction, qui est la base de la pensée occidentale, il n'y a pas besoin de revenir du modèle idéal pour l'appliquer au réel, avec toute la perte d'efficacité que ceci engendre.

Il est pour eux bien plus *efficace* d'étudier la situation et d'en voir les potentiels d'évolutions qui y sont présents afin d'influer sur eux, et non directement sur la situation en elle même. Ceci dans le but de faire évoluer cette dernière vers une direction qui est notre vrai objectif. La facilité de cette entreprise réside dans le fait que la transformation de ce qui est encore au stade de prémisses est facile. Cette *facilité* est à opposer à celle qui réside dans la *difficile modification directe* de la situation présente qui est déjà actualisée avec la seule utilisation d'actions.

Cette manière de procéder nécessite une étude approfondie de la situation actuelle et de savoir déceler au plus tôt sa *propension* afin de savoir vers où elle est sujette d'aller et d'en modifier le cours le plus tôt afin que ce soit le plus facile et le moins réversible possible.

Ce manque d'actions explique pourquoi les chinois n'ont pas ou très peu de grands *héros*. Ceci explique également l'absence d'*épopées* comme en occident.

Leur théorie se veut complète. Elle ne cherche pas à penser un modèle idéal pour ensuite le faire entrer à coup de marteaux dans le réel, mais bien d'étudier ce dernier afin de l'influencer. L'effet souhaité est induit par l'évolution de la situation, et nous verrons qu'il ne laisse pas de place au hasard.

5. Le libre : deux partis « ouverts », deux visions de l'affrontement différentes

Jusqu'à maintenant, nous avons parlé du Libre comme une seule entité. C'est inexact car en son sein, deux partis coexistent : l'*Open Source* d'un côté, le *Logiciel Libre* de l'autre. Regardons comment chacun s'affaire dans cette guerre, et si on peut le rapprocher d'une des deux *méthodes de guerre* que nous avons vues précédemment.

5.1 Open Source : des champions ouverts qui s'opposent, vision occidentale

L'*Open Source* consiste à fournir le code des programmes informatiques afin que tout-à-chacun puisse les étudier, les modifier et les redistribuer comme il le souhaite. Leur principe est que si un programme non open source existe et qu'il est meilleur que ses équivalents open sources, il n'y a pas de raison de ne pas l'utiliser. Par contre, ses membres vont travailler sur des équivalents libres jusqu'à ce qu'ils soient aussi performants, voir plus.

Cette vision est basée principalement sur la performance technique du modèle de développement plutôt que sur l'éthique. La liberté du code est utilisée comme un moyen d'obtenir de meilleurs outils ouverts, avec un mode de développement qui se veut le plus efficace possible.

Le mouvement de l'Open Source repose sur une concurrence sans merci entre les programmes ouverts. Cette concurrence renforce les programmes les plus en vus de cet écosystème, et en fait de véritables *champions de l'open source* comme par exemple Mozilla Firefox ou bien Ubuntu. Bien sûr des projets plus modestes existent afin de contenter certaines niches particulières non gérées par les champions. Ces projets peuvent être de fantastiques boîtes à idées pour ces derniers.

Cette vision de l'affrontement direct des projets techniques peut être mit en parallèle avec la vision occidentale de la guerre. Tel un Achille participant à la chute de Troie, Firefox participe activement à la chute d'Internet Explorer. Cette méthode demande beaucoup d'investissement en temps de développement afin d'obtenir des résultats satisfaisants face aux mastodontes de l'environnement privateur et leur position dominante.

Lorsqu'une victoire telle que celle de Firefox face à IE est obtenue, elle est annoncée en grande pompe afin de motiver le reste des projets.

5.2 Logiciels Libres : un travail de fond sur les principes, vision orientale

Fruit de la lutte de nombreux défenseurs des libertés, dont le plus connu est indiscutablement son précurseur *Richard Stallman*, le mouvement des *Logiciels Libres* met en avant l'éthique que représente les logiciels à code ouvert, comme un accès universel à la connaissance en matière informatique par exemple. L'ouverture du code et l'obtention de programmes disponibles pour tout le monde ne sont pas vus comme une finalité en soi, mais comme un simple moyen, une obligation en fait. Le but n'est pas d'avoir des logiciels performants de bonnes qualités, mais une informatique libre.

Ce mouvement présentant d'abord des idées, il n'est pas très médiatique, et ne repose pas sur des « grands champions » pour promouvoir ses idées. Il n'y a pas non plus de grandes dates ou

événements particuliers. Il cherche plutôt à faire changer les mentalités des « utilisateurs » d'informatique afin de leur faire prendre conscience qu'il est dans leur intérêt d'adhérer à ce mouvement qui souhaite tout simplement leur liberté et celle de leurs données.

Cette « transformation » des mentalités en utilisant ce qui est amené à se développer (comme par exemple la protection de la vie privée) est ce qui relie le Logiciel Libre et la vision chinoise de la guerre. Il ne joue pas sur des actions fortes et vues de tous afin d'arriver à ses fins, mais bien sur une transformation qui débouche sur une situation où toutes les libertés sont sauvées.

6. L'efficacité des deux modèles

Maintenant que nous avons survolé nos deux modèles, comparons leur efficacité.

6.1 Définition de l'efficacité

Tout d'abord, définissons rapidement l'efficacité :

efficacité = effets obtenus/efforts

Dans notre cas, les effets recherchés seront de proposer des logiciels ouverts afin que les utilisateurs puissent avoir un total contrôle sur leurs données et les traitements qu'elles subissent.

Nous allons voir maintenant l'efficacité des deux méthodes et comment ceci s'applique à notre exemple.

6.2 La part de chance

6.2.1 Au niveau des méthodes

Clausewitz a déjà admis que le modèle occidental n'était pas parfait, et qu'il faisait une part importante à la chance. L'application de l'heuristique et des meilleures méthodes est pratique dans un engagement tactique, mais plus on monte en complexité, moins elles sont efficaces. Arrivé au plan stratégique, elle n'évitent pas leurs parts de hasards et d'efforts nécessaires.

Ceci tranche avec la vision chinoise : ils assurent la victoire pour quelqu'un qui saurait parfaitement étudier la situation des forces en présence et savoir quelles sont les tendances qui y sont présentes.

6.2.2 Dans notre exemple

L'Open Source ne définit pas de *méthode* parfaite sur comment faire en sorte que le projet arrive à battre ses opposants privés. Ces derniers jouant dans la même cour, il leur *suffit* de les battre sur le plan marketing et/ou technique pour l'emporter.

A contrario, le Logiciel Libre a sa doctrine et s'y tient. Les logiciels privés ne peuvent rien contre cela, car ce qu'elle défend est avant tout le réel intérêt des utilisateurs : leur liberté.

6.3 Une reconnaissance inutile des efforts

6.3.1 Au niveau des méthodes

Les efforts sont glorifiés du côté occidental, avec la présence des « héros » qui agissent. Une entreprise n'est louée que si elle s'est faite avec difficulté face à une certaine adversité. Clausewitz affirme que l'efficacité a un coût important, qui est de l'ordre de la difficulté de ce que l'on entreprend.

Là encore, la différence est flagrante avec la Chine. Les maîtres de guerre pensent plutôt que

l'efficacité est inversement proportionnelle aux efforts consentis. Ou plutôt, que l'on a la meilleure efficacité lorsque l'on arrive à ses fins avec le moins d'efforts possibles. Ceci est plutôt en accord avec notre définition de l'efficacité. Ils disent même que le summum de l'efficacité est atteint lorsque l'on ne fait plus rien. En effet, à efforts qui tendent vers 0, l'efficacité tend logiquement vers l'infini (mais toujours avec les effets souhaités obtenus).

En fait le sage (un sage et un maître de guerre utilisent les mêmes méthodes du point de vue chinois) arrive à faire peu d'efforts en influant sur les tendances de la situation en amont de l'affrontement afin que, le jour où il se décide à attaquer à la vue de tous, la situation soit totalement opposée à l'adversaire. Cette transformation faite au plus tôt ne coûte presque rien, et une fois que la situation tend vers ce qui nous arrange, il n'y a plus besoin d'agir du tout. Au lieu d'agir il suffit « d'assister ce qui vient naturellement ».

6.3.2 Dans notre exemple

La lutte au niveau de l'Open Source implique d'être toujours plus performant et ergonomique que les logiciels privés. Même avec ses avantages, la guerre marketing tourne rarement dans le sens du libre. On glorifie des projets comme Firefox ou le noyau Linux, mais combien de lignes de codes et d'efforts marketing ont été nécessaires pour arriver à ce niveau de reconnaissance?

Mais que se passe-t-il lorsqu'un logiciel privé reprend la main sur le plan technique? La plupart des utilisateurs pensent Firefox comme un navigateur *gratuit* avant tout. Peu savent que c'est un outil dans lequel ils peuvent avoir confiance sur le sort de leurs données et qu'il ne leur cache rien. Lorsque Chrome aura autant de plugins que Firefox, ne verra-t-on pas une fuite des utilisateurs de ce dernier vers le navigateur de Google malgré les risques sur les données de leur vie privée?

Les mouvements des Logiciels Libres eux ne demandent pas autant d'effort pour soutenir leurs idées. Il n'y a pas besoin d'avoir un million de lignes de code pour prouver que l'on a raison. Il faut juste trouver le bon exemple pour convaincre l'utilisateur qu'avoir un plein accès à ses données est dans son intérêt.

Une fois la sensibilisation effectuée, l'utilisateur se posera la question sur qui possède réellement ses données. Même si techniquement supérieur, un outil privé ne sera pas vu comme une alternative acceptable. Et là, les efforts sont limités, voir nuls une fois la personne convaincue. Les choses suivent leur cours, l'efficacité est à son maximum.

6.4 Une action localisée face à une transformation globale

6.4.1 Au niveau des méthodes

Un autre problème vient de l'utilisation de l'action : si forte soit-elle, est nécessairement limitée dans l'espace et le temps. A contrario, une transformation s'étend à tout l'espace et dans la durée. C'est même de cette continuité d'où découlent les effets les plus importants car ils paraissent naturels. Là où le temps est un ennemi de l'action, il est le soutien de la transformation. Il en est de même pour l'espace : une transformation s'applique en tout point.

6.4.2 Dans notre exemple

Un logiciel comme Firefox impose le respect comme navigateur et influe sur les orientations du web dans le monde (ce qui n'est pas rien), mais il n'a aucun impact, par exemple, sur les logiciels de sauvegardes ou les tableurs. Un utilisateur peut très bien utiliser Firefox et Excel en même temps sans y voir le moindre inconvénient. Si l'on suit la logique de l'Open Source, tant que c'est techniquement supérieur, il n'y a pas de raison de ne pas l'utiliser. Firefox est supérieur aux autres navigateurs (tout du moins pour cet utilisateur), et Excel l'est également pour lui dans un autre domaine. L'un n'influe absolument pas sur l'autre.

Concernant le Logiciel Libre, une fois que l'utilisateur a compris que Firefox, en tant que logiciel libre, lui garantissait un total respect de ses données et des transformations qui y sont effectuées, il peut se poser la question pour tous ses logiciels, voir même plus globalement sur l'informatique dans son ensemble. Au lieu d'agir localement et ponctuellement dans le temps, cette prise de conscience va amener l'utilisateur à faire le même raisonnement pour tout ce qui le touche. Ce n'est pas une action, mais une transformation qui s'est effectuée en l'utilisateur. Elle perdure dans le temps, et n'a pas de limite dans son espace d'application.

6.5 Une recherche de gloire non efficace

6.5.1 Dans les méthodes

Ce qui se cache derrière la recherche d'une modification du réel par l'action est un besoin de reconnaissance, cette gloire immortelle apportée par nos épopées. Achille lui même a choisi une vie courte mais glorieuse à une longue vie tranquille mais sans éclats. C'est un certain plaisir de l'aventure qui est recherché, car dans la guerre par action il y a cette dimension de chance si jouissive. Mais c'est un plaisir qui est recherché en tout premier lieu, pas l'efficacité.

Les chinois se méfient tout particulièrement de cette recherche d'aventure au sein de la guerre. Ils recherchent l'efficacité brute, la victoire, pas la gloire. Pour eux, un grand général ne sera jamais connu, car il gagnera toujours des victoires faciles. Dans les faits, il a rendu ces victoires faciles même si ceci ne se voit pas de l'extérieur. Là où tout le monde ne regarde que l'engagement final, la grande bataille, les chinois prennent en compte tout ce qui s'est passé avant et qui a impliqué la situation du début du combat. Un bon général n'attaque que quand il sait la victoire acquise. Acquise car la situation où il se trouve à cet instant ne laisse aucun doute sur la défaite de l'adversaire.

Mais c'est lui qui a modifié bien en amont des tendances dans les situations de l'époque afin d'arriver à la situation actuelle où son armée équivaut à une tonne là ou celle de l'autre ne vaut pas plus d'un gramme. Il n'y a plus à louer un général pour le combat, car c'est *visiblement* une victoire facile. La différence de force impliquée par la situation empêche tout hasard quant à l'issue du combat.

6.5.2 Dans notre exemple

Combien de temps et d'efforts a-t-il fallu à Firefox pour arriver là où il est aujourd'hui? Son succès en fait un exemple qu'il faut montrer tant la lutte face à Internet Explorer a été (et est toujours) rude.

De l'autre côté, le logiciel libre ne fait pas de grandes annonces, n'a pas d'évènements marquants. Lorsqu'un utilisateur est convaincu ceci ne se voit pas, ou très peu. Mais la transformation est pérenne comme nous l'avons vu. Ce n'est pas spectaculaire, mais la transformation amorcée est irréversible. Lorsqu'un logiciel privateur sera proposé à un utilisateur au fait des problèmes qu'il rencontrera par son utilisation, la victoire sera assurée, sans grandes annonces. La guerre visible sera lors de ce choix. Elle est gagnée d'avance, même si tout s'est joué bien avant. Il n'y aura alors personne à louer, mais l'efficacité est assurée, car le gain est maximal et la peine pour y arriver minime.

6.6 Action/réaction, mais rien ne s'oppose à une transformation

6.6.1 Au niveau des méthodes

Un autre problème de l'action repose sur la réaction de l'adversaire, voir de l'environnement tout entier. Toute action implique une réaction. C'est particulièrement inefficace car le peu d'effet qu'avait apporté l'action est largement amenuisé par la réaction qui lui est opposée, généralement du

même ordre de grandeur. Ceci explique en grande partie la difficulté du modèle par action.

La transformation utilise la propension incluse dans la situation. Elle *pousse* ce qui tend déjà à augmenter. Il n'y a donc pas d'action contre-nature qui s'impose au réel. Celui-ci n'a donc aucune raison de lui opposer une « réaction ».

En fait, il peut y avoir des frictions dans l'application de cette transformation si elle n'est pas *pure* : si elle est teintée du sceau de son initiateur par exemple, elle risque de ne pas s'insérer parfaitement dans la situation. Si quelque chose dépasse, il va y avoir une réaction immédiate et la transformation ne sera pas pleinement efficace. C'est aussi pour cela qu'il n'y a pas à louer les grands généraux. Ils ne font rien pour être reconnu car ceci ne serait tout simplement pas efficace pour arriver à leurs fins.

Cette toute puissance de la situation, les chinois ne l'utilisent pas seulement contre l'adversaire, mais également pour leur propre troupe. Pour eux, leurs soldats ne sont jamais courageux par eux même, mais à cause de la situation. C'est pour cela qu'il se mettent dans une situation périlleuse (« zone de mort » de Sun Tzu) afin qu'elle implique du courage chez les soldats : ils n'ont pas le choix, donc ils combattent comme jamais, les plus poltrons deviennent courageux. C'est aussi pour cela que Jules César n'acculait jamais un ennemi, mais lui laissait toujours une voie de sortie : ce n'est pas par pitié (ce n'est pas le genre), mais pour éviter un regain de courage chez ses adversaires.

6.6.2 Dans notre exemple

Lorsqu'un logiciel ouvert amène une nouvelle fonction innovante, les logiciels privateurs vont vite le copier. A l'action tentée, une réaction de l'adversaire est vite proposée. La nouvelle situation va demander encore beaucoup d'effort pour arriver à vaincre un tel adversaire.

En ce qui concerne les logiciels libres, la transformation se fait au sein des utilisateurs. Respectant pleinement la morale, ses idées ne peuvent être combattues. D'ailleurs, les logiciels privateurs n'essaient même pas. L'effet est obtenu avec un effort minime car rien ne s'y oppose. Là encore, l'efficacité est maximale.

De plus, là où l'Open Source est dans l'affrontement direct et déjà présent, le logiciel libre peut s'appliquer bien avant qu'un conflit avec un logiciel privateur arrive. Une fois que le conflit explosera, vu que l'utilisateur sera déjà convaincu, le logiciel privateur pourra utiliser tous les effets 3D de la planète, il sera déjà défait avant même de pouvoir combattre.

6.7 Deux visions de la victoire parfaite

6.7.1 Au niveau des méthodes

Pour Clausewitz et les occidentaux il n'y a pas de doute : le but ultime de la guerre est de détruire l'ennemi. Sans cela, l'ennemi est toujours en mesure de contre attaquer.

Pour les chinois, la destruction de l'ennemi n'est qu'une demi victoire. Il est bien plus intéressant de prendre l'ennemi entier. L'idéal consiste à arriver à cette fin sans même avoir à combattre : nos pertes sont minimales, nos gains sont maximums. On a même bien plus de ressources après la guerre qu'au début. « L'art ultime est donc de vaincre avant d'avoir combattu ». Là dessus, le penseur chinois Mencius affirme qu'il est possible de prendre tous les peuples avec la morale, qu'ils vont aller d'eux mêmes vers le *prince* le plus juste.

Cependant, une telle chose ne peut s'acquérir avec la simple force : un peuple soumis par la force n'hésitera pas une seconde à se retourner au moindre signe de faiblesse de la part du prince. N'ayant plus rien à perdre, leur fureur sera terrible. Il faut les persuader de nous suivre afin que ce soit la situation qui tende à les garder sous notre bannière.

Si l'on devait faire un parallèle avec la physique, le cas de soumission sera notre état d'équilibre.

Cependant, ce dernier peut être de deux types : stable et non stable. On a $df/ds^2 < 0$ dans le premier cas, > 0 dans le second. Si c'est stable, comme dans le cas d'une transformation, chaque changement pourra être facilement infléchi avec juste une réaction (bien moins coûteuse qu'une action) afin de retourner dans l'état de soumission, car les personnes souhaitent cet état. Dans le second, chaque modification va donner l'occasion de changer cet état et va tendre à s'en éloigner. Il sera donc tout particulièrement instable.

6.7.2 Dans notre exemple

Le mouvement Open Source est orienté vers l'obtention des meilleurs outils ouverts. Leur lutte est directement orientée face à un ou plusieurs logiciels privateurs. Leur lutte a un objectif : que leur outil soit le seul utilisé. Si c'est une victoire possible, nous avons vu que ce n'était pas la meilleure.

Le logiciel libre, quant à lui, est bien plus *efficace* lorsque l'on parle de fin de guerre : au lieu de détruire l'ennemi, il l'absorbe. En effet, une fois la majorité des utilisateurs convaincus de l'avantage qu'ils ont d'utiliser des logiciels libres, que restera-t-il aux logiciels privateurs? Devenir libre. Et ça de nombreux éditeurs sont en train de le faire, comme Nokia qui ouvre Symbian par exemple. Par pure bonté d'âme? Surement pas! Mais pour attirer *ses* utilisateurs vers son système car ces derniers ont un intérêt d'avoir accès au source : ce sont les développeurs d'applications pour portable, véritable nerf de la guerre du mobile.

Comme un algorithme glouton, le logiciel libre progresse parmi les logiciels privateurs et les *libéralise*. Le mouvement en sort plus fort, avec de nouveaux outils plus performants, et des adversaires en moins. L'efficacité est à son zénith.

6.8 L'alliance des partis est indispensable

Comme nous l'avons vu, le modèle chinois est bien plus orienté efficacité que le modèle occidental. De même, le travail de fond du mouvement logiciel libre est plus efficace dans une guerre totale. Il y a cependant une guerre où ce mouvement n'est pas adapté : la guerre de l'image. Comme nous l'avons vu, le modèle chinois n'est absolument pas spectaculaire. C'est même totalement l'inverse. La guerre qui nous intéresse consiste à persuader des utilisateurs. Or l'image est une donnée particulièrement importante pour les utilisateurs occidentaux. Les utilisateurs en ont pleins les yeux et les oreilles. Il n'y a pas une journée sans *key note* ou annonces du même genre. Il est particulièrement complexe de faire passer un message sur la morale dans un tel flot. Si l'accès aux utilisateurs est *bloqué* la stratégie du logiciel libre s'effondre totalement.

Si nous avons vu que l'Open Source en lui même n'est pas efficace, il peut être un fantastique vecteur d'accès aux utilisateurs. Ceux-ci viennent aux applications pour leur fonctionnalités. Une fois à la vue de l'utilisateur, il serait bon de lui faire passer le message pour les logiciels libres de manière général. Ce message ne doit pas être un obstacle à l'utilisation du logiciel sous peine d'être un frein plutôt qu'un accélérateur du libre.

Les contres effets seront bien sûrs toujours présents, mais uniquement au niveau de l'open source. De toute façon, avec ou sans le message, ils l'auraient subis. Autant faire en sorte d'être le plus utile possible.

Finalement, les premiers à convaincre ne sont pas les utilisateurs, mais les gestionnaires des champions open source qui eux sont spécialistes des grands effets d'annonces. Si ces dernières ne sont pas utiles en tant que tel pour les utilisateurs, elles permettent d'être le vecteur d'information qui manque au logiciel libre.

Les deux partis doivent affronter un ennemi particulièrement puissant qui sait très bien manœuvrer sur les plan de l'espace médiatique. Si aucun des deux n'est en mesure de l'emporter (l'Open Source

par un manque de constance dans le choix des utilisateurs, le Logiciel Libre dans l'accès à la parole auprès des utilisateurs), leur alliance peut régler le problème et offrir aux utilisateurs une informatique performante et libre.

7. Plus loin dans l'art de la guerre

Nous avons vu que la stratégie chinoise est plus *efficace* que l'occidentale sur bien des points. Continuons d'étudier l'art de la guerre de Sun Tzu et identifions ce qui peut être utile dans notre contexte, ce que le libre fait déjà et ce qu'il devrait faire.

7.1 L'importance de la guerre et ses domaines

7.1.1 La guerre est indispensable

Sun Tzu commence fort logiquement par justifier l'importance de la guerre :

La guerre est d'une importance vitale pour l'État. C'est le domaine de la vie et de la mort: la conservation ou la perte de l'empire en dépendent; il est impérieux de le bien régler.

Il met l'accent sur son importance pour la conservation de l'état qui sera pour nous l'existence même des logiciels libres. Cette guerre est celle du choix : nous devons toujours pouvoir proposer ce choix aux utilisateurs. Il ne faut pas penser que nous avons pour but de faire couler tous les outils privés : et si certains utilisateurs souhaitent les utiliser en toute connaissance de cause, de quel droit les empêcherions nous?

Par contre, de quel droit les sociétés derrière les logiciels privés nous empêcherait de fournir ce choix aux utilisateurs ou au moins de les avertir? La guerre actuelle concerne ce point précis. Tout ce qui est contraire à cet objectif doit être éliminé. Le plus grand problème concerne bien entendu les brevets logiciels.

Ces derniers sont d'un certain point de vue la réponse *méthode chinoise* des logiciels privés : en orientant la situation vers une guerre de brevet, le libre n'est plus en mesure de combattre, et encore moins de vaincre. Si la situation aboutit à des brevets logiciels partout, la guerre est perdue.

7.1.2 Les domaines de la guerre

Il ajoute également :

Si nous voulons que la gloire et les succès accompagnent nos armes, nous ne devons jamais perdre de vue: la doctrine, le temps, l'espace, le commandement, la discipline.

La doctrine fait naître l'unité de penser; elle nous inspire une même manière de vivre et de mourir, et nous rend intrépides et inébranlables dans les malheurs et dans la mort.

La doctrine de Sun Tzu sera pour nous les grands principes du logiciels libres. Ils permettent de se placer sous un étendard bien identifié et sûr de cette force. Ses partisans n'hésiteront pas à la défendre face à l'adversité.

Si nous connaissons bien le temps, nous n'ignorerons point ces deux grands principes Yin et Yang par lesquels toutes les choses naturelles sont formées et par lesquels les éléments reçoivent leurs différentes modifications; [...]

L'espace n'est pas moins digne de notre attention que le temps; étudions le bien, et nous aurons la connaissance du haut et du bas, du loin comme du près, du large et de l'étroit, de ce qui demeure et de ce qui ne fait que passer.

Ici nous devons passer du monde bien réel de Sun Tzu au monde virtuel des logiciels et des libertés. Il met l'accent sur la connaissance approfondie de l'environnement. Sans cela, il n'y a pas à penser à la victoire car c'est là dessus que doivent se baser le logiciel libre afin de voir quelles sont les propensions de la situation qui vont aller dans son sens.

Discipline : posséder l'art de ranger les troupes; n'ignorer aucune des lois de la subordination et les faire observer à la rigueur;

Le général sera dans le cas du libre l'ensemble des responsables des principaux projets open source. Les *troupes* seront les défenseurs des logiciels libres et les responsables des projets de plus faible notoriété. Les *généraux* sont responsables de l'orientation et de l'image qu'ils donnent des logiciels libres et open source. La guerre de trolls n'est pas acceptable si elle ne fait pas progresser la situation dans un sens qui nous avantage, et doit être éradiquée sans la moindre once de pitié.

7.1.3 Une victoire assurée

A contrario de la pensée occidentale, Sun Tzu en digne héritier de la pensée chinoise affirme pouvoir déterminer à l'avance qui sera victorieux.

[..] de sorte que vous pourrez conjecturer assez sûrement quel est celui des deux antagonistes qui doit l'emporter; et si vous devez entrer vous-même en lice, vous pourrez raisonnablement vous flatter de devenir victorieux.

Comme nous l'avons vu précédemment à l'avance sera en fait avant l'engagement. Car au moment de l'engagement, tout est déjà joué si l'un des belligérants utilise l'art de la guerre chinois.

7.2 Prendre les peuples sans combattre

Sun Tzu fait écho à Mencius et son éloge de la morale sur le fait qu'il est possible, voir préférable, de prendre les peuples sans combattre :

Admirateurs de vos vertus et de vos capacités, les officiers généraux placés sous votre autorité vous serviront autant par plaisir que par devoir. Ils entreront dans toutes vos vues, et leur exemple entraînera infailliblement celui des subalternes, et les simples soldats concourront eux-mêmes de toutes leurs forces à vous assurer les plus glorieux succès.

Estimé, respecté, chéri des vôtres, les peuples voisins viendront avec joie se ranger sous les étendards du prince que vous servez, ou pour vivre sous ses lois, ou pour obtenir simplement sa protection.

Il affirme qu'un grand général inspire ses troupes et attire même les peuples voisins. Ici, un grand défenseur des logiciels libres va pouvoir donner du courage et de la détermination à tous les développeurs du monde libre. Il peut même attirer des peuples voisins qui seront soit des logiciels libres avec des principes un peu différents (comme BSD par exemple), voir même opposés comme des supporters des logiciels privés qui y verraient un avantage de ne plus subir de limitations artificielles.

7.3 L'art de la guerrilla

7.3.1 Maximiser les gains, limitez les pertes

Sun Tzu ne traite également de l'aspect guérilla de la guerre. Il la préconise même :

Également instruit de ce que vous pourrez et de ce que vous ne pourrez pas, vous ne formerez aucune entreprise qui ne puisse être menée à bonne fin.

Ne rien entreprendre si ce n'est pas sûr d'avoir un gain assuré. Ce principe est un peu contraire au fonctionnement actuel du monde libre, où tout le monde fait un peu ce qu'il veut, même si ça peut ne pas fonctionner. Sun Tzu y voit une perte de ressources, mais là où nous avons un avantage réside dans ses ressources justement : elles sont importantes et très variées. Au lieu de se disperser, on peut voir ce chaos comme une formidable manière d'explorer toutes les voies. Un petit peu comme pour les algorithmes imitant les fourmis et leur phéromones : une fois qu'un projet a trouvé une piste intéressante, il la marque et la partage avec le reste de la communauté. D'autres projets plus structurés peuvent alors prendre le relai et renforcer les champions avec une perte d'énergie pour ces derniers somme toute limitée.

7.3.2 Attirer des partisans ennemis

Sun Tzu pousse plus loin le principe de la guérilla : les ennemis peuvent, et même doivent, devenir une ressource. De cette manière, le gain est double : on se renforce d'autant que l'ennemi s'affaiblit.

Conserver les possessions des ennemis est ce que vous devez faire en premier lieu, comme ce qu'il y a de plus parfait; les détruire doit être l'effet de la nécessité. Si un général agit ainsi, sa conduite ne différera pas de celle des plus vertueux personnages; elle s'accordera avec le Ciel et la Terre, dont les opérations tendent à la production et à la conservation des choses plutôt qu'à leur destruction.

Dans notre exemple, il ne faut pas chercher à détruire la réputation d'un logiciel privateur, mais si possible l'amener dans nos rangs, donc de l'ouvrir. Ce fut le cas de Blender par exemple ou Symbian plus récemment. Ils peuvent devenir de futurs grands noms du libre. Leur légitimité sera peu être même plus importante au vue du grand public qu'un projet démarré libre dès le départ.

Vous profiterez de la dissension qui surgit chez vos ennemis pour attirer les mécontents dans votre parti en ne leur ménageant ni les promesses, ni les dons, ni les récompenses.

Comme pour la guérilla, attirer ses *ennemis* permet de grossir ses rangs. Ce coup double et doit être recherché à tout prix. Dans notre cas, les efforts pour convaincre un utilisateur du problème que pose les logiciels privateurs peut être double : un support de moins pour eux, et une personne qui peut expliquer les principes libres à d'autres.

Si vos ennemis sont plus puissants et plus forts que vous, vous ne les attaquerez point, vous éviterez avec un grand soin ce qui peut conduire à un engagement général; vous cacherez toujours avec une extrême attention l'état où vous vous trouverez.

Il est contre une attaque systématique et irréfléchie de l'adversaire. Si l'on est inférieur, on se place en simple défense. Ce point rejoint la pensée taoïste : si rien n'est propice dans la situation actuelle, le sage se retire et attends. La roue tourne, de nouvelles opportunités vont survenir, et il sera alors temps de sortir de sa torpeur qui n'est qu'apparente en fait.

La seconde partie est plus problématique avec les principes du libre : cacher son état est complexe pour un écosystème qui se veut le plus ouvert possible. Tous les échanges sont ouverts. Ceci donne un gage de confiance aux utilisateurs, mais fournit également des informations aux adversaires.

Lorsque vos gens auront pris sur l'ennemi au-delà de dix chars, commencez par récompenser libéralement tant ceux qui auront conduit l'entreprise que ceux qui l'auront exécutée. Employez ces chars aux mêmes usages que vous employez les vôtres, mais auparavant ôtez-en les marques distinctives qui pourront s'y trouver.

Les victoires et ceux qui l'ont rendu possible doivent être publique et faire du bruit. Ceci va inciter les autres membres de la communauté à faire de même.

Traitez bien les prisonniers, nourrissez-les comme vos propres soldats; faites en sorte, s'il se peut, qu'ils se trouvent mieux chez vous qu'ils ne le seraient dans leur propre camp, ou dans le sein même de leur patrie. Ne les laissez jamais oisifs, tirez parti de leurs services avec les défiances convenables, et, pour le dire en deux mots, conduisez-vous à leur égard comme s'ils étaient des troupes qui se fussent enrôlées librement sous vos étendards. Voilà ce que j'appelle gagner une bataille et devenir plus fort.

Sun Tzu insiste sur le fait de bien traiter les prisonniers afin d'en faire des ressources. Dans notre cas, les développeurs issus du monde privateur doivent se sentir plus à l'aise dans le monde libre que de là où ils viennent. De même, les projets issus du monde privateur, mais libérés comme Blender, ne doivent pas subir de moqueries dues à leurs origines. Ils doivent être pleinement intégrés, voir même écouté car ils apportent une vue nouvelle, donc potentiellement intéressante.

N'oubliez rien pour lui débaucher ce qu'il y aura de mieux dans son parti: offres, présents, caresses, que rien ne soit omis. Trompez même s'il le faut: engagez les gens d'honneur qui sont chez lui à des actions honteuses et indignes de leur réputation, à des actions dont ils aient lieu de rougir quand elles seront sues, et ne manquez pas de les faire divulguer.

Ici il faut attirer les développeurs du côté ouvert. Mais plus encore, il faut attirer *ceux qui savent* du côté ouvert. Ces personnes qui font référence en matière d'informatique dans le cocon familial ou entre amis. Ce sont eux qui conseillent et orientent les nouveaux utilisateurs. Il est capital de les convaincre que mettre du libre peut les arranger, ne serait-ce qu'en terme de licence, ou au moins avoir moins de cracks à chercher sur internet. Ils sont la première force de frappe marketing des logiciels privateurs. Cette force ne doit pas être sous estimée.

7.3.3 Attaquer où c'est faible

Sun Tzu reviens sur l'apparente tautologie du modèle chinois : pour gagner il faut attaquer là où l'ennemi ne se défends pas.

Lorsque l'ennemi est uni, divisez-le; et attaquez là où il n'est point préparé, en surgissant lorsqu'il ne vous attend point. Telles sont les clefs stratégiques de la victoire, mais prenez garde de ne point les engager par avance.

Un fantastique exemple est en train de prendre essor avec les bases *NoSQL* : les principales références dans ce domaines sont toutes libres! Là où les éditeurs privateurs s'affairent encore à avoir une offre sql classique performante, le libre a déjà occupé le nouveau terrain de jeu. L'occupation est d'une importance stratégique importante. La renommée va être simple, et les nouveaux arrivants vont avoir du mal à s'implanter, surtout s'il arrivent avec des licences onéreuses là où l'on parle de services web qui sont amenés à grandir fortement.

diminuez ses forces le plus que vous pourrez, en lui faisant faire des diversions, en lui tuant de temps en temps quelque parti, en lui enlevant de ses convois, de ses équipages, et d'autres choses

qui pourront vous être de quelque utilité.

Les ressources des adversaires des logiciels libres sont les utilisateurs et les développeurs de logiciels privés. Il faut faire en sorte de leur supprimer leur légitimité aux yeux de ces deux populations. Le reste se fera tout seul.

Concernant la diversion, le modèle chaotique du libre force ses adversaires à avoir une recherche et développement importante pour arriver à suivre. Microsoft commence par exemple à être de plus en plus diversifiés, mais sans être un acteur principal sur ses nouveaux marchés. La diversification à outrance sans vision globale n'est pas une bonne chose du point de vue stratégique. Une telle organisation n'a pas la souplesse qu'a un modèle organisé sur un chaos.

7.4 Ne pas faire durer la guerre

Sun Tzu revient un peu sur le *temps de guerre* qui ne doit pas être trop important.

Ils n'ignorent point, et vous devez le savoir aussi, que rien n'épuise tant un royaume que les dépenses de cette nature; car que l'armée soit aux frontières, ou qu'elle soit dans les pays éloignés, le peuple en souffre toujours;

Les projets dans la tourmente d'un conflit avancent beaucoup moins bien que d'habitude. Il faut donc que ces périodes soient les plus brèves possibles. Que dire du cas FreeBSD au début de sa vie? GNU-Linux ne serait peut-être pas passé devant sans le problème du droit d'auteur d'Unix.

Empêcher que les hameaux et les chaumières des paysans ne souffrent le plus petit dommage, c'est ce qui mérite également votre attention; porter le ravage et dévaster les installations agricoles de vos ennemis, c'est ce qu'une disette de tout doit seule vous faire entreprendre.

Sun Tzu ré-insiste sur le but premier de la guerre : conserver ses acquis. Dans notre analyse, l'exemple le plus frappant sera la lutte contre les brevets logiciels qui menace le monde du libre dans son entier. Prendre d'assaut les logiciels privés ne doit être qu'un moyen de se défendre contre cela.

7.5 Éviter les attaques de front, voir les affrontements eux mêmes

Pour Sun Tzu, l'attaque de front n'est pas une option intelligente. En cela, il suit bien la pensée chinoise classique.

Sun Tzu dit: Il est d'une importance suprême dans la guerre d'attaquer la stratégie de l'ennemi.

Une attaque de front n'est pas l'idéale. Un moyen de prendre les applications consiste à saborder leur support et à leur offrir une meilleure option dans le modèle libre. Un bon exemple ici est Nokia avec son système Symbian. Les applications sur portables sont le nerf de la guerre des téléphones. Avec de moins en moins de développeurs sur leur système, Nokia a choisi d'ouvrir leur système. D'autant plus, qu'avec QT, ils peuvent attirer facilement bon nombres de développeurs déjà friands des solutions libres ou des portages simples d'applications déjà existantes.

Celui qui arrache le trophée avant que les craintes de son ennemi ne prennent forme excelle dans la conquête.

En une phrase, Sun Tzu résume la philosophie chinoise que nous avons vu précédemment.

[..] que vous le pourrez d'en venir aux mains avec lui; la prudence et la fermeté d'un petit nombre de gens peuvent venir à bout de lasser et de dompter même une nombreuse armée. Ainsi vous êtes à la fois capable de vous protéger et de remporter une victoire complète.

Avec l'application des techniques chinoise, il n'y a pas besoin d'être beaucoup pour l'emporter. Cependant, vu que le libre rassemble beaucoup de participants, il serait dommage de ne pas en profiter.

7.6 Connaissance de l'ennemi et de soit même

Sun Tzu fait un résumé du calcul des chances de réussites assez facile à retenir :

Connais ton ennemi et connais-toi toi-même; eussiez-vous cent guerres à soutenir, cent fois vous serez victorieux.

Si tu ignores ton ennemi et que tu te connais toi-même, tes chances de perdre et de gagner seront égales.

Si tu ignores à la fois ton ennemi et toi-même, tu ne compteras tes combats que par tes défaites.

Il ne suffit pas de savoir à quel point les logiciels libres sont efficaces, il faut également démontrer que les opposants sont inefficaces et dangereux pour ses utilisateurs. C'est bien l'alliance de ces deux méthodes qui permettent d'obtenir la victoire. La connaissance des ennemis peut passer par exemple par une analyse fine et compréhensible des licences des principaux logiciels privés sur le marché. La plupart des clauses sont faites pour limiter les possibilités des utilisateurs. Ces derniers doivent être au courant de ce qu'ils signent.

7.7 Les lieux d'affrontements

Les lieux sont une données très importante pour Sun Tzu.

Sun Tzu dit: Une des choses les plus essentielles que vous ayez à faire avant le combat, c'est de bien choisir le lieu de votre campement. Pour cela il faut user de diligence, il ne faut pas se laisser prévenir par l'ennemi, [...] En général, il n'y a que du désavantage à camper après les autres.

Camper sera dans notre cas mettre un pied dans un nouveau domaine. Par exemple, les logiciels libres ont débutés originellement sur les outils de programmations et tout ce qui touche au système. Désormais, il faut se rapprocher de plus en plus vers les utilisateurs standards.

Un autre domaine qui émerge beaucoup des temps-ci pour les développeurs sont les bases NoSQL. Le libre est arrivé en force sur ce marché promis à un bel avenir.

Un autre point important réside dans l'art du général à faire siéger son adversaire là où il souhaite :

S'il est véritablement habile, il pourra disposer à son gré du campement même et de toutes les marches de son ennemi. Un grand général n'attend pas qu'on le fasse aller, il sait faire venir. Si vous faites en sorte que l'ennemi cherche à se rendre de son plein gré dans les lieux où vous souhaitez précisément qu'il aille, faites en sorte aussi de lui aplanir toutes les difficultés et de lever tous les obstacles qu'il pourrait rencontrer;

Si dans le cas NoSQL cette technique intéressante pour le libre, ce dernier s'est fait pendre à ce piège concernant les brevets : le projet Mono et l'aide de Microsoft pour son développement ne sont rien d'autre qu'une aide pour le libre afin que celui-ci mette le pieds dans la fosse aux brevets logiciels. Heureusement, peu de projets sont tombés dans le panneau, mais le risque est toujours présent. Il ne faut pas oublier que l'adversaire est loin d'être stupide. Il ne faut pas nécessairement

chercher à le blesser le plus possible, mais limiter les attaques qu'il peut nous faire. Le temps joue en faveur de ceux qui utilisent l'art de la guerre. Tachons juste de l'utiliser mieux qu'eux.

7.8 La maîtrise de l'adversaire

L'effet de surprise est un paramètre important pour Sun Tzu.

Rendez-vous dans des lieux où l'ennemi ne puisse pas soupçonner que vous ayez dessein d'aller. Sortez tout à coup d'où il ne vous attend pas, et tombez sur lui lorsqu'il y pensera le moins.

Là est une grande force du libre : son modèle distribué lui permet de rechercher dans toutes les directions. Il n'est pas possible de prévoir de la part de ses opposants un cheminement précis où ils peuvent déjà bloquer l'accès. Ils sont obligés d'aller partout à la fois, donc nulle part de manière efficace du fait de leurs ressources limitées.

La suprême tactique consiste à disposer ses troupes sans forme apparente; alors les espions les plus pénétrants ne peuvent fureter et les sages ne peuvent établir des plans contre vous.

Ce point est assez important et illustre bien l'efficacité du modèle *bazar*. Les adversaires n'ont pas à faire à une structure bien identifiée, mais à une myriade de personnes liés par une éthique. Elles s'affairent sur des projets séparés mais qui travaillent bien ensemble. Même depuis l'intérieur du *bazar* il est complexe de voir où il se dirige. Alors vu de l'extérieur...

7.9 L'adaptabilité

Point important soulevé par Sun Tzu : l'adaptabilité.

L'eau, dans son cours, suit la situation du terrain dans lequel elle coule; de même, votre armée doit s'adapter au terrain sur lequel elle se meut.

Ce point est peut-être le plus important de tout le traité : il faut s'adapter le plus possible. L'image de l'eau n'est pas anodine : elle représente chez les chinois ce qui a le moins de matière, de prise dans le réel. Pour les chinois, celui qui sera le plus à même de l'emporter sera celui qui est le moins pris forme dans le réel, celui qui peut encore changer la propension de la situation le plus en amont possible. Si l'un des acteurs est déjà en train d'affronter la réalité et n'est plus en mesure de s'adapter à la situation, alors il a de fortes chances que son adversaire utilise cette faiblesse contre lui en orientant la situation là où ça ne l'arrange pas. Ne pouvant s'adapter, la loi de Darwin fera son œuvre.

L'adaptabilité est donc d'une importance capitale pour Sun Tzu : il ne faut pas oublier que pour lui l'affrontement n'est que conclusif, la guerre est également, et surtout, constitué de tout ce qui a précédé et qui a figé le réel pour l'affrontement final. Arrivé au combat, c'est déjà fini. L'adaptabilité est la valeur clé dans cette phase capitale.

Ça tombe bien, c'est une qualité indéniable du modèle libre. Son modèle chaos est encore ici d'un avantage certain. Il n'y a pas de comité central qui dicte sa voix à tous les participants du monde libre. Si un membre souhaite explorer une nouvelle voie, il peut le faire sans soucis, le temps qu'elle est libre (dans le cas contraire tout ce qu'il risque c'est un kick sur un forum IRC...). Comme nous l'avons vu, si cette voie est prometteuse, l'ensemble de la communauté va pouvoir s'y engouffrer. L'effort a été limité pour elle, les gains sont énormes. L'efficacité l'est également.

L'adaptabilité vaut également pour les messages que l'on doit adresser aux utilisateurs : on ne convient pas un professionnel de l'informatique comme on convient un utilisateur final. Les premiers sont à la recherche d'une adaptabilité des sources et d'une certaine indépendance face aux éditeurs.

Les seconds ne sont pas intéressés par l'adaptabilité du code et pour cause : la plupart ne savent pas comment fonctionne un programme et s'en moque bien. Par contre, le message doit être orienté vers les données des utilisateurs. C'est sur elles que doivent porter nos efforts afin de leur faire comprendre qu'avec un modèle libre elles ne risquent rien, alors qu'avec un modèle privé, l'éditeur peu à tout moment décider de ne plus supporter leurs données. Avec un logiciel privé le vrai possesseur de ces données est l'éditeur, là où avec une solution libre, c'est bien l'utilisateur.

8. Au final

Ce texte souhaitait faire la lumière sur l'utilisation de la philosophie chinoise, et en particulier sa vision de la guerre, appliquée au monde du libre. Nous avons vu que le libre utilise déjà beaucoup de principes de la philosophie chinoise, sans en avoir forcément pris conscience. Les principales avancées pérennes engendrées par le libre sont par utilisation de cette méthode.

Si nous souhaitons avoir une informatique libre, il n'est d'autre choix que de continuer à adopter ce modèle. Les adversaires sont spécialistes de l'action, des grandes annonces et occupent déjà une bonne partie du terrain. Il faut opter pour une transformation plutôt que de se lancer bêtement dans un affrontement au corps à corps. Cette transformation passe par des messages efficaces et ciblés auprès des différents utilisateurs et décideurs de l'informatique. Ce fonctionnement est celui du Logiciel Libre. Si l'on rajoute la guerre médiatique d'aujourd'hui, un vecteur vecteur d'information est nécessaire. C'est indiscutablement les logiciels Open Source déjà présents chez les utilisateurs qui doivent être le faire de lance de ces messages.

Nous pouvons avoir confiance, Sun Tzu nous annonce une victoire si nous appliquons correctement ces principes. Ils ont fonctionné parfaitement depuis plus de 4000 ans, il n'y a pas de raison qu'ils ne soient pas efficaces à notre époque également.

Références :

- *L'Art de la guerre*, Sun Tzu, traduction du père Amiot, édition Mille et Une Nuits
- *Richard Stallman et la révolution du logiciel libre. Une biographie autorisée*, Sam Williams (Auteur), Richard Stallman (Auteur), Christophe Masutti, édition Eyrolles
- *Traité de l'efficacité*, François Jullien, édition Le Livre de Poche

Licence : ce texte est publié sous licence CC-BY-ND. Pour savoir ce que couvre cette licence, se référer à <http://creativecommons.org/licenses/by-nd/3.0/>.

Auteur : Jean Gabès, naparuba@gmail.com

Versions : 1.0 rc, 10 avril 2010 : version initiale

1.0, 18 avril : intégration remarques rms