

TECHNIQUE & PRATIQUE

Collection dirigée par Emmanuel Cornet et Alexandre Hérault

**Logiciels libres**

**Open source**

*qu'est-ce que c'est?*

BENOÎT CHEVALIER

Consultant

Ancien élève de l'École Normale Supérieure



## AVANT-PROPOS

On pourrait dire qu'il y a deux points communs majeurs entre l'amour et les logiciels libres. D'abord, des quantités de gens en ont peut-être sous les yeux tous les jours sans s'en rendre compte et sans les « voir » ; ensuite, les informations objectives et sans parti pris disponibles sur les deux sujets sont rares.

Les logiciels libres, *free software* ou *open source* en anglais, constituent un ensemble de programmes informatiques de grande qualité créés par des bénévoles diffusant largement leurs « secrets de fabrication » sur Internet.

Tout cela mérite une explication : pourquoi des experts en informatique et même des entreprises fourniraient-ils gratuitement le fruit de leur travail ? Est-ce qu'il s'agit d'une « nouvelle économie » – d'ailleurs, en quoi est-elle nouvelle ? Et pourquoi des gens continuent-ils alors à travailler avec des logiciels très chers s'il en existe de beaucoup moins chers de qualité aussi bonne voire meilleure, qui ont été choisis par des pays comme la Chine, le Brésil et le Pérou ?

Cette explication, vous l'avez à présent entre les mains.

Nous allons dans les pages qui suivent retracer l'histoire de cette idée puis nous décrirons les atouts (nombreux), les faiblesses (il y en a) et l'économie des logiciels libres, pour pouvoir répondre non seulement à la question « qu'est-ce que c'est ? » mais aussi, finalement, à « ai-je, en pratique, intérêt à utiliser ces logiciels ? » pour chaque catégorie d'utilisateur.

Tous les termes utilisés ici, indiqués à leur première occurrence dans le texte en PETITES CAPITALES, sont repris dans

un glossaire à partir de la page 115. Nous vous invitons à vous y référer à volonté, dès maintenant si nécessaire.

Délibérément, nous n'avons voulu faire ni un manifeste pour les LOGICIELS LIBRES, ni un pamphlet contre leur DÉVELOPPEMENT. De la sorte, ce livre est très différent de la quasi-totalité des sites internet et des quelques ouvrages déjà disponibles sur le sujet. En résumé : nous vous fournissons les informations, vous vous faites votre idée.

Ce livre n'est pas non plus un livre sur LINUX. Linux est un logiciel libre très important, et l'un des plus connus, mais c'est aussi l'arbre qui cache la forêt : le monde des logiciels libres ne se limite pas du tout à Linux.

Ce livre s'adresse aussi bien aux curieux désireux de mieux comprendre le phénomène qu'aux informaticiens qui veulent en savoir plus, et également aux décideurs qui cherchent à augmenter l'efficacité de leurs équipes ; enfin, à tous ceux qui ont entendu parler des logiciels libres et qui se demandent si c'est ou non adapté à leur usage ou leurs besoins.

Nous espérons que vous aurez autant de plaisir à lire cet ouvrage que nous en avons eu à l'écrire ; nous savons qu'il vous sera utile en pratique. Vos critiques comme vos éloges nous aideront à l'améliorer encore. N'hésitez pas à en faire part à l'éditeur, à l'adresse

`contact@H-K.fr`

Si vous rencontrez ce que vous estimez être une erreur ou une imprécision gênante dans l'ouvrage, nous vous serions reconnaissant de nous en faire part également.

Bonne lecture !

Benoît Chevalier

---

# Table des matières

---

<b>Avant-propos</b>	<b>3</b>
<b>1 Qu'a-t-on le droit de faire avec un logiciel ?</b>	<b>9</b>
1.1 Qu'est-ce qu'un logiciel ?	10
a. Le code source et le binaire	10
b. Ce que l'on achète quand on achète un logiciel	12
1.2 Qu'est-ce qu'un logiciel libre ?	13
a. Définition d'une licence libre	15
b. Ce qui n'est pas inclus dans la définition	16
c. Licence « libre » ou « open source »	18
<b>2 Bref historique des logiciels libres</b>	<b>19</b>
2.1 Une pratique issue du monde universitaire	19
2.2 Apparition des licences libres	20
a. L'imprimante de Richard Stallman	21
b. La <i>Free Software Foundation</i>	21
c. La licence GPL	22
d. Les autres licences	24
e. La naissance de Linux	25
f. 1998 : le début de la célébrité ?	27
2.3 Qui les utilise aujourd'hui ?	29
a. Et la réponse est...	29
b. Une présence majeure sur les serveurs	30
c. Des architectures mixtes libre/propriétaire	30
2.4 Quelques pistes pour le futur	33
a. De belles perspectives de croissance	33
b. À quand la GPL 3 ?	33

<b>3</b>	<b>Atouts et faiblesses des logiciels libres</b>	<b>35</b>
3.1	Comment sont-ils développés ?	35
	a. Les bâtisseurs de cathédrales	35
	b. Le développement libre en trois étapes	37
	c. Comment est-ce piloté ?	38
	d. Qui sont les contributeurs ?	40
	e. Une armée de bêta-testeurs	43
3.2	Les atouts des logiciels libres	44
	a. Pérennité	45
	b. Qualité du code	47
	c. Coûts	50
	d. Sécurité	52
3.3	Les faiblesses des logiciels libres	58
	a. Formation et support technique	59
	b. Absence de garantie	61
	c. Faible vitesse de développement	62
	d. Performance pas toujours optimale	63
	e. Absence de valeur ajoutée industrielle	64
3.4	Aspects communs	65
	a. Vulnérabilité face aux brevets logiciels	65
	b. Ergonomie	69
	c. Coût de sortie d'un logiciel propriétaire	69
<b>4</b>	<b>La montée en puissance</b>	<b>71</b>
4.1	Le contexte informatique	71
	a. Internet	71
	b. La standardisation des systèmes et des données	72
	c. La croissance des services	72
4.2	L'économie des logiciels libres	74
	a. Principes d'économie des logiciels	74
	b. Trois modèles de développement	75
	c. Spécificités des logiciels libres	80
	d. Comment gagner de l'argent avec les logiciels libres ?	82

4.3	L'émergence des schémas collaboratifs . . . . .	85
	a. Le modèle libre n'est pas réservé aux logiciels . . . . .	85
	b. Fédérer les idées innovantes . . . . .	87
<b>5</b>	<b>Les logiciels libres sont-ils faits pour vous ?</b>	<b>89</b>
5.1	Les particuliers . . . . .	89
	a. Les critères du choix . . . . .	89
	b. Le profil type . . . . .	90
	c. Les étapes de la migration . . . . .	91
5.2	Les entreprises et les administrations . . . . .	92
	a. Les critères du choix . . . . .	93
	b. Le profil type . . . . .	96
	c. Les étapes de la migration . . . . .	96
5.3	Les développeurs et les éditeurs . . . . .	98
	a. Les critères du choix . . . . .	98
	b. Le profil type . . . . .	101
	c. Les étapes de la migration . . . . .	101

## Annexes

<b>A</b>	<b>Les principales licences libres</b>	<b>103</b>
A.1	La GPL . . . . .	103
A.2	La LGPL . . . . .	104
A.3	La licence BSD . . . . .	105
<b>B</b>	<b>Le droit d'auteur des logiciels</b>	<b>107</b>
B.1	Introduction au droit d'auteur . . . . .	107
B.2	Les droits des auteurs de logiciels . . . . .	109
<b>C</b>	<b>Bibliographie</b>	<b>114</b>
	<b>Glossaire</b>	<b>115</b>

# Qu'a-t-on le droit de faire avec un logiciel ?

---

Ce qui était encore de la science-fiction pour nos grands-parents est devenu une banalité : l'informatique est partout. Du suivi des bouteilles de lait d'un supermarché à l'injection de l'essence dans le moteur d'une voiture, les machines interviennent dès qu'il faut traiter une information.

On devine sans peine que les LOGICIELS qui pilotent ces machines ont une grande valeur commerciale. Les entreprises qui se spécialisent dans la réalisation de logiciels<sup>1</sup> considèrent ces derniers comme leur propriété et leur secret le plus précieux.

Pourtant, l'entreprise qui écrit un logiciel n'est pas toujours la seule à l'employer : Microsoft a créé les SYSTÈMES D'EXPLOITATION de la famille WINDOWS mais tout le monde a le droit de les utiliser (moyennant finance). Comment protéger un secret dans ces conditions ?

À l'inverse, il existe des logiciels dont le fonctionnement interne est bien connu et ne cache aucun secret industriel. Vous en connaissez probablement certains, dont les noms sont de plus en plus fréquemment cités dans la presse : Linux, FIREFOX, THUNDERBIRD, APACHE ou OPENOFFICE en sont des exemples. On les appelle les logiciels libres.

Avant d'être en mesure de vous exposer ce qui fait leur spécificité, il nous faut revenir un peu en arrière : comment font les entreprises pour diffuser à tout le monde leurs logiciels sans révéler en même temps leur recette de fabrication ?

---

1. Par exemple Microsoft, Adobe, Electronic Arts, etc.

## 1.1 Qu’est-ce qu’un logiciel ?

Un logiciel (on dit parfois aussi une application ou un PROGRAMME) est un ensemble de codes et d’instructions destinés à être lus et utilisés par un ordinateur, qu’il s’agisse d’un ordinateur avec écran, clavier et souris, ou de l’ordinateur de bord d’une voiture.

### a. Le code source et le binaire

Les analogies pour décrire le fonctionnement d’un ordinateur sont nombreuses, nous retiendrons ici celle du pont<sup>2</sup>.

Pour construire un pont, il y a d’abord une phase de conception détaillée : définition du projet, premiers plans, validation par le calcul, maquette, etc. Ce processus coûte cher en temps et en salaire des ingénieurs, mais cependant dix fois moins que la construction du pont lui-même. La phase de construction fait appel à des entreprises spécialisées qui vont appliquer les instructions figurant dans les plans.

De même, pour « fabriquer » un logiciel, il y a une phase de conception, de rédaction d’instructions, et enfin la construction d’un résultat vraiment utilisable, c’est-à-dire utilisable par l’ordinateur. Cette dernière phase est très rapide (entre quelques secondes et quelques minutes) et s’appelle la COMPILATION. Elle est réalisée au moyen d’un logiciel particulier qui s’appelle le COMPILATEUR.

Mais elle doit être faite à partir d’instructions extrêmement détaillées, d’autant plus détaillées en fait que le processeur d’un ordinateur n’a pas d’opinion sur les ordres qu’on lui donne et ne sera généralement pas capable, au contraire d’une entreprise de travaux publics, de pointer une erreur dans les documents fournis.

Conclusion : c’est la construction qui coûte le plus cher et demande le plus de travail pour le pont, alors que pour le logiciel, la quasi-totalité du coût de la création est dans la partie conception.

---

2. Proposée par Jack Reeves dans son article *Code as design*.

L’ensemble des instructions servant à fabriquer un logiciel s’appelle le CODE SOURCE, tandis que le logiciel « fabriqué » et utilisable par l’ordinateur se nomme le code binaire ou simplement le BINAIRE.

Concrètement, le code source d’un logiciel est le texte qu’un programmeur tape sur son clavier. Il comporte souvent un grand nombre de COMMENTAIRES écrits par l’auteur et destinés à en faciliter la lecture. Il est dans ce cas relativement facile à un programmeur professionnel de retrouver une information ou la partie à modifier dans un code source, même s’il n’en est pas l’auteur.

Le code source est rédigé dans un langage de programmation informatique. Il existe un grand nombre de langages possibles mais dans la plupart des cas il s’agit d’une suite d’algorithmes mathématiques et de mots de liaison en anglais.

Voici un exemple de code source, écrit dans le langage de programmation C. Le programme ci-dessous calcule  $2^{10}$ , c’est-à-dire 2 à la puissance 10 :

```
int main() {
    int i;
    int res = 1;
    for (i = 1; i <= 10; i = i+1) {
        res = 2*res;
    }
    return res;
}
```

La compilation consiste à traduire ce code en une suite de commandes très simples exécutables avec une grande rapidité par le processeur mais presque illisibles par l’homme, même par un programmeur professionnel<sup>3</sup>.

---

3. Nous ne vous en donnons pas d’exemple ici mais il est possible d’« entendre » du binaire en écoutant les bruits d’un modem analogique en train de se connecter à Internet par une prise téléphonique. C’est tout à fait aussi incompréhensible que ça en a l’air.

## b. Ce que l’on achète quand on achète un logiciel

La plupart des logiciels que vous pouvez acheter dans un magasin ou sur Internet sont distribués sans leur code source. C’est en général uniquement le binaire que l’on trouve sur le CD-ROM d’installation. Le code source reste alors la propriété et le secret de l’éditeur du logiciel. Le jargon informaticien appelle cela un logiciel PROPRIÉTAIRE.

Il est en général illégal de copier un tel logiciel. La loi française autorise la création d’une unique copie, et seulement à des fins de sauvegarde ; le logiciel et sa copie ne peuvent être utilisés simultanément. Et il n’est évidemment pas envisageable de tenter de l’améliorer sans avoir accès au code source<sup>4</sup>.

La plupart des clients, bien entendu, n’ont pas besoin du code source et n’ont de toute façon aucune intention d’étudier le fonctionnement interne du logiciel, de même que la plupart des amateurs de musique ne demandent pas à avoir la partition livrée avec leur disque.

Cependant, même sans être informaticien, les utilisateurs apprécieraient de pouvoir dupliquer les logiciels qu’ils ont achetés et aimeraient aussi que les éventuelles erreurs soient corrigées rapidement : dans le cas des logiciels propriétaires, tout cela est interdit.

En effet, aussi étrange que cela puisse paraître au premier abord, le droit qui s’applique aux logiciels est le droit d’auteur, et plus particulièrement, en France, le très restrictif droit de la PROPRIÉTÉ INTELLECTUELLE, qui donne à l’auteur d’un logiciel les mêmes prérogatives que celles que l’auteur d’un livre ou d’une chanson a sur le fruit de son travail<sup>5</sup>.

---

4. En fait, il est possible de retrouver une certaine forme de code source à partir du binaire, opération qui s’appelle le *DÉSASSEMBLAGE* ou *reverse engineering* ; mais cette opération ne peut être faite que par un informaticien très spécialisé, demande des semaines ou des mois de travail même pour un logiciel relativement simple, et dans le cas d’un logiciel propriétaire elle peut être illégale.

5. Un projet de droit spécifique aux logiciels a été abandonné en 1983.

Afin que vous puissiez utiliser un logiciel, l’auteur doit vous transmettre certains de ses droits dans le cadre d’un contrat : la LICENCE. C’est le texte qui apparaît au moment où l’on installe le logiciel et que la plupart des utilisateurs ne lisent jamais, se contentant de cliquer sur les mots « J’accepte » pour pouvoir passer à l’étape suivante. On ne peut pas utiliser un logiciel si l’on n’accepte pas sa licence.

Si aucune licence n’accompagne le logiciel, celui-ci n’est donc pas utilisable, à moins que l’auteur n’ait pas non plus mis de *copyright* afin que son logiciel tombe dans le DOMAINE PUBLIC et soit utilisable par tous sans limitation<sup>6</sup>.

Les licences de la plupart des logiciels vendus dans le commerce restreignent très fortement l’usage qui peut être fait de ceux-ci, par exemple en limitant l’utilisation aux étudiants, alors que la version utilisable par tous – et techniquement identique – coûte sensiblement plus cher.

Elles interdisent le plus souvent d’étudier le fonctionnement du logiciel et limitent drastiquement la responsabilité de l’éditeur en cas de dommage résultant de l’utilisation de son produit.

En résumé, dans le cas d’un logiciel propriétaire, vous n’achetez pas vraiment le logiciel lui-même ; tout ce que vous pouvez acheter, c’est le droit de l’utiliser.

## 1.2 Qu’est-ce qu’un logiciel libre ?

Le système des logiciels propriétaires ne convenait pas à certains auteurs, qui ont voulu donner aux utilisateurs de leurs programmes des droits beaucoup plus étendus.

C’est précisément pour cela que la notion de logiciel libre a été créée dans les années 1980.

---

6. La même chose est arrivée à la suite d’une erreur au film *La Nuit des morts-vivants* de G. Romero : certaines copies envoyées aux salles de projection ne mentionnaient aucun *copyright*, elles sont donc automatiquement tombées dans le domaine public et ont été diffusées sans que l’auteur puisse revendiquer le paiement de ses droits.

Les deux idées fondamentales du logiciel libre sont :

- de rendre accessible<sup>7</sup> le code source ;
- de soumettre le logiciel à une licence permettant de le modifier et de redistribuer des modifications.

Ces deux conditions ne sont pas toujours indissociables :

	Licence libre	Licence non libre
Code source disponible	Logiciels libres : Apache, PERL...	Logiciels non libres : Java, PGP...
Code source confidentiel	impossible	Logiciels propriétaires : Photoshop, Powerpoint...

La conjonction de la disponibilité du code source et de l’autorisation de le modifier et de le redistribuer vous permet de faire et de diffuser – en toute légalité – toutes les vérifications, corrections et améliorations dont vous pouvez avoir besoin du moment que vous en avez la compétence.

Cela ne concerne pas uniquement les informaticiens, car si eux seuls sont capables de faire des modifications dans le code source du logiciel, tous les utilisateurs peuvent par la suite télécharger gratuitement sur Internet la version corrigée ou améliorée. Vous pourrez donc, quelle que soit votre spécialité, bénéficier du système. Et si vous êtes un décideur au sein d’une entreprise ou d’une administration, vous pouvez également demander à l’un de vos salariés ou à un consultant d’ajouter une fonction à un logiciel libre.

---

7. Il peut être soit fourni avec le binaire soit aisément téléchargeable.

### a. Définition d'une licence libre

Il y a plusieurs définitions précises possibles pour une LICENCE LIBRE<sup>8</sup>, mais toutes s'accordent sur les trois grands critères suivants :

#### ***Une licence libre permet à tous d'utiliser le logiciel***

Une licence libre ne peut donc pas interdire certains usages du logiciel, ni réserver celui-ci à une certaine catégorie d'utilisateurs comme les étudiants, ou encore à un usage non commercial.

#### ***Elle permet d'étudier et de modifier le code source***

Le code source est soit fourni avec le logiciel, soit aisément téléchargeable sur Internet.

Si vous avez acheté ou téléchargé un logiciel libre et son code source, et si vous en avez la compétence, vous pourrez à volonté étudier le fonctionnement interne du logiciel, tenter de corriger d'éventuelles erreurs, d'ajouter les fonctions dont vous avez besoin, etc. Vous n'aurez alors plus qu'à compiler à nouveau le code source pour obtenir un binaire tout neuf et meilleur.

Ce droit permet également d'insérer des parties du code source d'un logiciel libre dans un autre logiciel.

#### ***Elle donne le droit de diffuser le logiciel et ses modifications***

Cette condition signifie non seulement que vous pouvez copier librement un logiciel libre, mais aussi que vous pouvez le modifier et diffuser des versions modifiées.

C'est sur la base de ce droit que se sont constituées des équipes de programmeurs (parfois plusieurs centaines de personnes issues des cinq continents) autour de certains logiciels libres. Dans ce cadre, chacun apporte sa contribution à une partie du logiciel et la communique à tous par Internet, tout en bénéficiant des améliorations apportées par les autres.

---

8. C'est-à-dire une licence qui fait du logiciel qui lui est soumis un logiciel libre.

## b. Ce qui n’est pas inclus dans la définition

### **La gratuité**

Nous en savons assez à présent pour liquider le mythe de la gratuité. Si beaucoup de logiciels libres sont téléchargeables gratuitement sur Internet, d’autres sont payants, et il existe également des logiciels gratuits dont le code source n’est pas accessible : les FREEWARES<sup>9</sup>, qui sont toujours gratuits, et les SHAREWARES, qui sont gratuits pendant une « période d’essai » après laquelle l’utilisateur est invité à payer s’il continue à se servir du logiciel.

	Logiciels payants	Logiciels gratuits
Logiciels propriétaires	Logiciels commerciaux : MACOS, Word, Windows...	<i>Freewares</i> : Acrobat Reader, Kazaa...
Logiciels libres	Distributions : MANDRIVA, REDHAT, GNU Ada...	Logiciels libres téléchargés : OpenOffice, Firefox...

Pour citer l’analyste financier George Weiss : « [...] si les investisseurs se sont impliqués dans le développement de Linux, c’est pour réaliser des profits. Ils y réfléchiront à deux fois avant de parler de gratuité à leurs actionnaires. »

Le même logiciel, d’ailleurs, peut être gratuit ou payant selon le cas : ainsi, même si vous pouvez télécharger le logiciel libre Linux gratuitement, cela peut valoir la peine de dépenser quelques euros ou dizaines d’euros pour acheter en magasin un disque comprenant Linux et d’autres applications formant un ensemble, fournis avec une documentation et un

---

9. Ou graticiel, si l’on tient à utiliser un terme français très peu connu. Par exemple, le logiciel de téléchargement sur Internet KAZAA est gratuit, mais son code source n’est pas public. S’il l’était, des programmeurs auraient eu tôt fait de l’« améliorer » en supprimant le sous-programme GATOR, qui affiche de la publicité à toute occasion.

contrat d’assistance technique. Des éditeurs comme Redhat, Mandriva, ou encore Novell avec la distribution SUSE se sont spécialisés dans ce type de produits.

### ***Le copyleft***

Certaines licences libres comportent une caractéristique vraiment particulière, le *COPYLEFT*<sup>10</sup>. Si du code source issu d’un logiciel soumis à une telle licence est réutilisé dans un autre programme, l’ensemble du résultat devient automatiquement libre et soumis à la même licence.

Les licences appliquant le principe du *copyleft* sont souvent qualifiées par leurs détracteurs de licences « virales », dans la mesure où le caractère libre d’un segment de code régi par ce type de licence va d’après eux « contaminer » et rendre automatiquement libre tout logiciel dans lequel il est intégré.

Soyons clairs : un logiciel développé dans un environnement complètement libre et compilé avec un compilateur sous licence GPL n’a pas besoin d’être à son tour soumis à la licence GPL.

En revanche, le logiciel passe automatiquement sous licence libre dès lors qu’une ligne de code source sous licence de type *copyleft* est intégrée dans le programme. Cette intégration ne se produit pas à l’insu du programmeur (ce que le terme de licence « virale » pourrait suggérer), bien que ce dernier puisse parfois espérer que cela « ne se verra pas ».

### ***Le domaine public***

Certains auteurs de logiciels ont fait le choix de n’associer aucune licence ni aucun *copyright* à leur travail, qui appartient alors au domaine public, libre de droit, et utilisable par tous sans aucune condition. C’est une forme extrême de logiciel libre, pas du tout le cas le plus fréquent.

---

10. Il s’agit d’un jeu de mots : puisque le *copyright* est le droit d’auteur, le *copyleft* est donc le « gauche d’auteur ».

### c. Licence « libre » ou « open source »

Les trois principes que nous avons vus précédemment pour définir une licence libre ont été exprimés canoniquement par deux associations basées aux États-Unis qui font autorité dans le domaine des logiciels libres.

La première de ces associations, la *Free Software Foundation* ou FSF<sup>11</sup> parle des logiciels soumis à des licences libres sous la terminologie FREE SOFTWARE. Ceci peut cependant prêter à confusion en anglais pour les non-spécialistes car le mot *free* signifie « libre » mais également « gratuit ».

La seconde, l’*Open Source Initiative*, ou OSI<sup>12</sup> préfère la terminologie OPEN SOURCE pour éviter toute ambiguïté et dans le but de rassurer les acteurs économiques sur la valeur commerciale des logiciels libres.

« [...] Trop de gens voient les choses comme une guerre des logiciels libres contre les “forces du mal propriétaire”. [Dépasser, ou non, cette vision] est d’ailleurs, pour autant que je sache, la vraie distinction entre les mouvements “open source” et “free software”. »

Linus Torvalds

Ces termes sont donc historiquement liés aux associations correspondantes<sup>13</sup>.

Cependant, pour tous les usages pratiques, ils sont synonymes. Nous utiliserons donc désormais les expressions « licence libre » et « logiciel libre » mais en recommandant « open source » pour un usage en anglais afin d’éviter la confusion avec les *freewares*.

Maintenant que la définition est posée, nous avons fait le premier pas. Les suivants vont nous permettre de voir d’où vient la notion de logiciel libre, puis tout ce qu’elle signifie en pratique.

11. Fondation du logiciel libre.

12. Littéralement, « initiative du code source ouvert ».

13. Quelques militants de la FSF particulièrement radicaux considèrent même que la notion de *copyleft* est incluse dans le terme *free software*.

# Bref historique des logiciels libres

---

## 2.1 Une pratique issue du monde universitaire

Le premier ordinateur, le Colossus, a été conçu dans le plus grand secret en Angleterre en 1936. Utilisé par les militaires, il servait à percer à jour les codes secrets de leurs ennemis.

Cette technologie, stimulée par les besoins de la Seconde Guerre mondiale, est devenue accessible à la recherche civile à partir de 1945. Dès 1950, plusieurs universités américaines disposaient de gros ordinateurs (ils occupaient souvent des pièces entières) utilisés par les chercheurs dans leurs travaux scientifiques.

Jusque dans les années 1960, les logiciels étaient spécifiques à chaque modèle d'ordinateurs. Les gens qui les utilisaient disposaient le plus souvent du code source, soit qu'il ait été fourni avec le matériel informatique, soit qu'ils l'aient simplement écrit eux-mêmes. Dans tous les cas, le statut juridique du logiciel et de son code source restait assez flou et la circulation d'informations et de morceaux de programmes entre les différents laboratoires était une pratique courante.

Même quand les ordinateurs ont commencé à devenir de plus en plus standardisés, les codes sources ont continué à être librement échangés entre les équipes, comme le sont traditionnellement les résultats des recherches dans le monde universitaire.

À l'origine de l'informatique, les logiciels étaient donc plutôt « libres », mais personne ne s'en préoccupait vraiment.

# Atouts et faiblesses des logiciels libres

---

Tous les logiciels libres ne sont pas obligatoirement diffusés sur Internet ni utilisés par des milliers de personnes.

Nous nous concentrons cependant, dans cet ouvrage, sur les logiciels libres les plus importants, largement diffusés et ayant rencontré un accueil positif de la part des utilisateurs.

Nous savons que le premier fondement du logiciel libre est la disponibilité du code source. Or, assez peu de gens ont la compétence technique nécessaire pour lire ce code source. Pourtant, les différences entre les logiciels libres et non libres ne sont pas seulement compréhensibles par les experts en programmation. Elles seront manifestes pour vous quelle que soit votre spécialité parce qu'en raison de l'ouverture du code, ces programmes bénéficient d'un mode de développement unique en son genre qui entraîne des qualités mais aussi des défauts.

## 3.1 Comment les logiciels libres sont-ils développés ?

### a. Les bâtisseurs de cathédrales

Eric Raymond utilise l'expression « la CATHÉDRALE et le BAZAR » dans le livre du même nom pour décrire les deux organisations possibles d'une équipe de conception informatique. La cathédrale est la programmation d'un logiciel au sein d'une entreprise organisée et hiérarchisée. Ce modèle est plus efficace mais moins créatif que le bazar, modèle de développement collectif des logiciels libres, où tout le monde peut

# Pourquoi les logiciels libres montent en puissance

---

Le succès des logiciels libres, qui sont de plus en plus utilisés, ne repose pas seulement sur leurs qualités intrinsèques mais aussi sur un contexte informatique et économique favorable.

## 4.1 Le contexte informatique

Le succès des logiciels libres est encore modeste, mais il se traduit déjà par une croissance de plus de 35 % par an. Cette réussite est bien entendu liée aux aspects techniques décrits dans le chapitre précédent, mais également à la conjonction de trois grandes tendances : le développement d'Internet, les progrès de la standardisation, et la croissance de l'économie des services<sup>1</sup>.

### a. Internet

Dans le monde universitaire, les échanges d'informations sont la base du travail des chercheurs. Cela se fait depuis longtemps sous la forme de publications dans des journaux techniques spécialisés. Les chercheurs en informatique avaient besoin d'un outil plus réactif : ils utilisèrent le réseau américain d'origine militaire Arpanet, créé en 1969 et qui devint l'Internet que nous connaissons aujourd'hui<sup>2</sup>.

---

1. Pour plus de détails sur cette approche, voir Tim O'Reilly, *The Open Source Paradigm Shift*.

2. Le nom d'Internet est issu du protocole de communication IP (Internet Protocol) créé en 1974 pour Arpanet par Vinton Cerf et Robert

# Les logiciels libres sont-ils faits pour vous ?

---

Selon les cas, les avantages et les inconvénients des logiciels libres prennent plus ou moins d'acuité. Ce chapitre est donc divisé en trois parties, suivant que vous êtes un particulier, un décideur au sein d'une entreprise ou d'une administration utilisatrice de l'informatique, ou encore un programmeur ou un chef d'entreprise éditrice de logiciels.

## 5.1 Les particuliers

### a. Les critères du choix

#### *Les atouts des logiciels libres*

Les logiciels libres ont de multiples avantages pour un particulier : ils sont quasiment gratuits, ils ne « plantent » que très rarement, ils sont moins facilement inondés de publicité et de logiciels espions, et on n'est d'ailleurs pas obligé de passer tout son système sous logiciel libre en une seule fois : le navigateur Firefox, par exemple, ou le logiciel de retouche d'image Gimp, existent en versions Windows et MacOS en plus de Linux.

#### *La question du support technique*

Certains logiciels libres, et particulièrement les systèmes d'exploitation libres, peuvent être compliqués à utiliser<sup>1</sup>. Il est

---

1. Si vous utilisez un système d'exploitation comme Linux, vous serez souvent obligé de faire l'administration de votre ordinateur vous-même (installation des programmes, configuration de la connexion à Internet, mise en place des pilotes des périphériques, etc.).

# Les principales licences libres

---

Cette annexe apporte des informations techniques sur les trois licences libres les plus utilisées et vous indique où vous pouvez trouver leur texte intégral et officiel.

## A.1 La GPL

La *General Public License* de la FSF, créée en 1989 et modifiée en 1991, est la plus fréquemment utilisée de toutes les licences libres : 70% des logiciels libres s'appuient sur elle, y compris Linux, Gimp et le projet GNU.

Vous trouverez le texte officiel (en anglais) de la GPL à l'adresse suivante : [www.gnu.org/copyleft/gpl.html](http://www.gnu.org/copyleft/gpl.html).

Une version française sans valeur juridique<sup>1</sup> est disponible sur : [www.linux-france.org/article/these/gpl.html](http://www.linux-france.org/article/these/gpl.html).

La validité de la GPL a été récemment entérinée par la justice aux États-Unis et en Allemagne. L'affaire américaine concernait un logiciel d'ordinateur de bord de voiture<sup>2</sup>. Les deux jugements allemands ont été prononcés par la même cour au bénéfice du programmeur Harald Welte.

Le cas le plus récent portait sur un logiciel de sécurité informatique<sup>3</sup>.

Ci-après une traduction de la célèbre clause de *copyleft*.

---

1. Rendu nécessaire par la loi Toubon de 1994 relative à l'usage de la langue française, un projet d'écriture d'une version française officielle et juridiquement valable de la GPL est en cours mais n'a pas encore abouti.

2. Drew Technologies, Inc. contre Society of Automotive Engineers, Inc., et al., action civile No. 03-CV-74535-NGE-PJK (district Est du Michigan, 14 janvier 2005).

3. Harald Welte vs Fortinet Ltd., tribunal de première instance de Munich, 12 avril 2005.

# Le droit d'auteur français appliqué aux logiciels

---

On a noté dans le premier chapitre que le droit qui s'appliquait aux logiciels était le droit d'auteur. Nous allons voir dans les pages suivantes comment tout cela fonctionne en pratique.

## B.1 Introduction au droit d'auteur

Le droit d'auteur est défini en France par une législation spécifique : le code de la propriété intellectuelle<sup>1</sup>.

Ce texte reconnaît des droits à n'importe quel auteur d'une œuvre originale sans que celui-ci ait à accomplir une quelconque formalité.

*L'auteur d'une œuvre de l'esprit jouit sur cette œuvre, du seul fait de sa création, d'un droit de propriété incorporelle, exclusif et opposable à tous.*

*Code de la propriété intellectuelle*

Article L. 111-1

On parle bien de création, c'est-à-dire que l'œuvre, pour rentrer dans le système de la propriété intellectuelle, doit exister (les simples idées non encore mises en forme ne sont pas protégées par la loi), et doit bien être originale.

La Cour de cassation a précisé ce point en 1986, en demandant que la création porte « la marque d'un apport in-

---

1. Une version complète et annotée de ce code est disponible en ligne sur le site [www.celog.fr/cpi](http://www.celog.fr/cpi).

---

# Glossaire

---

Pour vous faciliter la lecture de cet ouvrage, vous trouverez ici dans l'ordre alphabétique toutes les définitions des termes techniques, des acronymes et les noms des principaux logiciels qui sont cités dans le texte.

**Abandonware** : logiciel en fin de vie commerciale et dont l'éditeur ne s'oppose plus à la diffusion gratuite (sur Internet par exemple). Les abandonwares sont en général disponibles sans leur code source.

**ADULLACT** : Association des développeurs et des utilisateurs de logiciels libres dans l'administration et les collectivités territoriales. L'ADULLACT mutualise notamment les développements de logiciels libres effectués pour le compte de collectivités locales françaises. Site web : [www.adullact.org](http://www.adullact.org).

**AFUL** : Association francophone des utilisateurs de Linux et des logiciels libres, créée en 1998 comme l'OSI et avec une approche pragmatique similaire. Site web : [www.iful.org](http://www.iful.org).

**Apache** : nom d'un logiciel libre de serveur web qui occupe 70 % du marché. Sa licence lui est spécifique : c'est la licence libre Apache. Il est téléchargeable gratuitement sur Internet à l'adresse [www.apache.org/](http://www.apache.org/).

**APRIL** : Association pour la promotion et la recherche en informatique libre, créée en 1996 sur une ligne proche de celle de la FSF. Site web : [www.april.org](http://www.april.org).

**AT&T** : *American Telephone & Telegraph*. Entreprise américaine équivalente à France telecom en plus gros.

**Bazar** : terme utilisé par Eric Raymond dans son livre *La Cathédrale et le bazar* pour désigner l'organisation du déve-

loppement d'un logiciel libre. Dans ce modèle, de nombreux programmeurs ou petits groupes de programmeurs travaillent séparément sur le logiciel. Le « bazar » est présenté comme peu organisé mais souvent très créatif.

**Bell labs** : département de la recherche et du développement de AT&T, créateurs du langage C et du système d'exploitation Unix.

**Bêta-testeur** : personne réalisant des tests et des recherches d'erreurs dans la version bêta d'un logiciel, c'est-à-dire une version déjà opérationnelle mais encore imparfaite.

**Bibliothèque** : composant d'un logiciel contenant des fonctions ou des instructions qui peuvent être appelées par divers autres logiciels. Les fichiers « .dll » sous Windows et « .so » sous Linux sont des bibliothèques.

**Binaire** : suite d'instructions élémentaires d'un logiciel, illisibles par l'homme mais exécutables par le processeur d'un ordinateur. Elles sont produites à partir du code source par la compilation. Les fichiers « .exe » sous Windows et les applications sous MacOS sont des binaires.

**BIND** : *Berkeley Internet Name Domain*, nom du principal logiciel libre qui gère le système DNS, c'est-à-dire les adresses sur Internet.

**BitKeeper** : nom d'un logiciel propriétaire qui était utilisé pour suivre les évolutions du code source de Linux, jusqu'à ce qu'un conflit juridique entraîne la fin de la collaboration entre les équipes de développement de ce système d'exploitation et l'éditeur de BitKeeper.

**BSD** : *Berkeley Software Design*. Ensemble de projets informatiques de l'université de Berkeley en Californie, financés à l'origine sur des crédits de recherche du ministère de la Défense américain. Ils incluent une licence libre, la licence BSD, et un projet de développement d'un système d'exploitation qui est devenu un logiciel libre à partir de 1992 et qui

a par la suite abouti aux trois systèmes OpenBSD, FreeBSD et NetBSD.

**Bug** : erreur dans un programme. Ce terme est utilisé aux États-Unis dans le sens d'erreur depuis le XIX<sup>e</sup> siècle. On écrit parfois aussi « bogue » en français. *Bug* en anglais signifie également « punaise » (l'animal) : on attribue souvent à tort l'origine du mot, dans le sens d'« erreur informatique », à un insecte qui a fait griller un composant dans un des premiers ordinateurs, l'ENIAC, en 1947.

**Cathédrale** : terme utilisé par Eric Raymond dans son livre *La Cathédrale et le bazar* pour désigner le mode de fonctionnement interne d'une entreprise qui édite des logiciels. La cathédrale ne désigne donc pas le produit fini, le logiciel, mais le mode d'organisation de l'équipe qui le développe.

**Code source** : le code source est la suite des commandes d'un logiciel que le programmeur écrit sur son écran. Elles sont lisibles par un professionnel ou un amateur averti, mais pas directement exécutables par l'ordinateur. Le code source est traduit en code binaire par la compilation.

**Commentaires** : mentions écrites par le développeur dans le code source d'un logiciel et destinées à en faciliter la lecture. Les commentaires ne sont pas traités par la compilation et il n'en reste aucune trace dans le code binaire.

**Compilateur** : logiciel dont la fonction est d'effectuer la compilation.

**Compilation** : transformation du code source d'un logiciel en code binaire.

**Copyleft** : jeu de mot sur le terme *copyright* qui se traduit par droit d'auteur et *left*, « laissé » ; le *copyleft* est donc le « gauche d'auteur » mais peut également se traduire par « copie laissée ». Ce principe inclus dans certaines licences libres consiste à dire que si un morceau de code issu d'un logiciel sous *copyleft* est intégré à un autre projet, alors l'ensemble

du projet résultant passe automatiquement sous la même licence libre. Ce système est parfois qualifié par ses détracteurs de « VIRALITÉ » ou de « licence virale » car le code libre va « contaminer » le logiciel dans lequel il est recopié et le transformer en logiciel libre. Attention : cette idée n'a rien à voir avec la notion de virus informatique.

Le *copyleft* est parfois représenté avec un symbole *copyright* à l'envers : ©

**Coût de sortie** : ensemble des difficultés occasionnées par l'abandon d'un système ou d'un logiciel. Elles sont liées au fait de devoir se passer de fonctions ou de prestations offertes par le système en question. Par exemple, l'abandon d'un logiciel propriétaire utilisant des formats de données non standards pour passer à un logiciel libre peut nécessiter de transférer les anciennes données dans un nouveau format, ce qui peut être une opération longue, complexe et/ou chère.

**Cybercommunisme** : expression due à R. Barbrook et présentée dans son ouvrage *The Californian Ideology*. Cela se réfère à l'idée (fausse) suivant laquelle les acteurs du logiciel libre seraient uniquement motivés par des raisons purement idéologiques et opposés au fonctionnement d'une économie de marché.

**Débogueur** : logiciel destiné à décomposer le fonctionnement d'un autre logiciel afin de l'étudier. Un débogueur sert en premier lieu à trouver et corriger les erreurs (bugs ou bogues) dans un programme. C'est aussi un composant essentiel de l'opération de désassemblage.

**Debian** : nom d'une distribution de Linux et de l'association à but non lucratif qui la diffuse. Site web : [www.debian.org](http://www.debian.org).

**Désassemblage** : action de tenter de reconstituer le code source d'un programme à partir du binaire. Cette opération n'est réalisable que par des informaticiens chevronnés et très spécialisés. Elle peut prendre un temps considérable (des se-

maines ou des mois de travail). Le désassemblage d'un logiciel propriétaire n'est souvent pas autorisé.

**Développement** : programmation d'un logiciel. La personne qui écrit le code source est donc un « développeur ».

**DNS** : *Domain Name System*, « système de noms de domaine ». C'est le mode d'identification des ordinateurs reliés par Internet.

**Domaine public** : en droit de la propriété intellectuelle, ensemble des œuvres sur lesquelles il n'y a pas de droits d'auteur à payer (en droit européen, c'est à partir de soixante-dix ans après la mort de l'auteur, ou parce que l'auteur en a décidé ainsi).

**Emacs** : nom d'un logiciel d'édition de texte du projet GNU. L'auteur qui a lancé Emacs est Richard Stallman.

**Ergonomie** : facilité d'utilisation d'un système ou d'un logiciel.

**EULA** : *End User License Agreement*, « contrat de licence de l'utilisateur final ».

**Firefox** : nom du logiciel libre de navigation web du projet Mozilla, concurrent de Internet Explorer. Téléchargeable gratuitement sur <http://frenchmozilla.sourceforge.net>.

**Fork** : subdivision du développement d'un logiciel libre en deux logiciels différents suite à des désaccords parmi l'équipe de programmeurs sur les choix techniques à effectuer.

**Free software** : expression anglaise pour désigner les logiciels libres, historiquement très liée à l'approche militante de la *Free Software Foundation* (FSF). La notion de *free software* est comprise par une minorité d'informaticiens comme incluant la notion de *copyleft*.

**Freeware** : logiciel propriétaire gratuit (par exemple, disponible en téléchargement sur Internet). Le logiciel propriétaire

Kazaa, destiné à effectuer des téléchargements de fichiers sur Internet, est un *freeware*. Le terme français académique correspondant est « graticiel ».

**FSF** : *Free Software Foundation*, ou en français « Fondation du logiciel libre ». Site web : [www.fsf.org](http://www.fsf.org). Cette association, créée par Richard Stallman en 1985, a pour vocation d'assurer la promotion des logiciels libres, de la transparence des systèmes informatiques et de la libre circulation de l'information. La FSF est également l'organe de diffusion commerciale des logiciels du projet GNU.



Le logo d'origine de la FSF

**Gator** : sous-programme de plusieurs *freewares*, y compris Kazaa. La fonction de Gator est d'afficher toutes sortes de publicités.

**GCC** : nom du compilateur libre du langage de programmation C. Il est issu du projet GNU et son auteur principal est Richard Stallman.

**GDB** : nom du débogueur libre écrit à l'origine par Richard Stallman pour le projet GNU.

**GFDL** : *Gnu Free Documentation License*, « licence libre GNU de documentation », destinée à encadrer les modifications de la documentation du projet GNU. La GFDL fonctionne notamment avec un système de parties invariantes, que l'on ne peut pas modifier et que l'on ne peut réutiliser qu'en un seul morceau.

**Gimp** : *Gnu Image Manipulation Program*. Nom d'un logiciel libre d'édition et de retouche d'image, concurrent d'Adobe

Photoshop. Gimp est téléchargeable gratuitement sur le site [www.gimp-fr.org](http://www.gimp-fr.org).

**Gnome** : interface graphique pour Linux, qui permet de donner à ce système une ergonomie et un design comparables aux principaux systèmes d'exploitation commerciaux.

**GNU** : *Gnu is Not Unix*, « Gnu n'est pas Unix », projet de création d'un ensemble de logiciels libres lancé par Richard Stallman en 1983. Site web : [www.gnu.org](http://www.gnu.org).

**GPL** : *General Public License*, « licence publique générale ». Licence libre publiée par la *Free Software Foundation* (FSF) en 1989 (version 1) et modifiée en 1991 (version 2, la plus utilisée aujourd'hui). Une nouvelle version est actuellement envisagée par la FSF. 70% des logiciels libres sont sous licence GPL. Cette licence applique le principe du *copyleft*.

**IDEALX** : société française spécialisée dans les services informatiques sur logiciels libres. Site web : [www.idealx.com](http://www.idealx.com).

**IIS** : logiciel de serveur web de Microsoft.

**IP** : *internet protocol*, système de transmission des données inventé par Vinton Cerf et Robert Kahn en 1974 pour le réseau Arpanet, l'ancêtre d'Internet. IP est toujours utilisé sur Internet, à qui il a donné le début de son nom.

**Kazaa** : logiciel *freeware* de téléchargement de fichiers sur Internet, financé par l'affichage de publicités gérées par son sous-programme Gator.

**KDE** : interface graphique similaire à Gnome.

**LGPL** : au début des années 1990, *Library General Public License*, « licence publique générale de bibliothèques », puis devenue *Lesser General Public License*, « licence publique moins générale ». La LGPL est une licence libre dérivée de la GPL mais comportant une clause de *copyleft* moins contraignante permettant d'intégrer des bibliothèques libres dans un ensemble qui ne l'est pas. Un peu plus de 10 % des logiciels libres sont sous licence LGPL.

**Licence** : contrat entre l'auteur ou l'éditeur d'un logiciel et l'utilisateur, définissant les droits, les obligations et les limites de la responsabilité des parties. On écrit *License* en américain.

**Licence libre** : licence d'un logiciel permettant l'accès au code source et son utilisation, y compris sa modification et le droit de redistribuer des versions modifiées ou non du logiciel.

**Linux** : système d'exploitation entièrement libre inventé par Linus Torvalds en 1991 et publié officiellement sous ce nom pour la première fois en 1994. On parle parfois de GNU/Linux pour désigner le noyau du système d'exploitation Linux assorti d'un certain nombre de logiciels issus du projet GNU.

**Logiciel** : ensemble d'instructions permettant de commander un système informatique. Word et Outlook, par exemple, sont des logiciels.

**Logiciel libre** : logiciel dont les utilisateurs ont accès au code source et soumis à une licence libre.

**MacOS** : nom de la série des systèmes d'exploitation livrés par Apple sur ses ordinateurs des marques MacIntosh, Imac, Powerbook et Ibook.

**Mandriva** : nom d'une distribution de Linux et de la société qui la diffuse, fusion en février 2005 des sociétés Mandrake et Connectiva. Site web : [www.mandrivalinux.com/fr](http://www.mandrivalinux.com/fr).

**Minix** : système d'exploitation créé dans les années 1980 par A. Tannenbaum, célèbre professeur d'informatique de l'université d'Amsterdam. Minix est principalement destiné à être un outil pédagogique pour la formation des étudiants en informatique. Linus Torvalds s'en est inspiré pour son projet Linux. Minix, bien que disponible avec son code source, n'était pas un logiciel libre (il était interdit de redistribuer d'éventuelles modifications) jusqu'à son passage sous licence BSD en avril 2000.

**Mono** : nom d'une architecture libre destinée à rendre compatible le développement de logiciels libres avec la plate-forme .NET de Microsoft.

**Mozilla** : nom d'un projet libre issu de la diffusion libre du code source de Netscape en 1998, qui a donné notamment les logiciels Firefox pour surfer sur le web et Thunderbird pour lire le courrier électronique. Ils sont téléchargeables gratuitement sur <http://frenchmozilla.sourceforge.net>.

**MySQL** : nom d'un logiciel libre de gestion de bases de données informatiques.

**Navigateur** : logiciel destiné à lire les pages web, dit aussi « butineur ». Exemples : Internet Explorer, Firefox.

**.NET** : nom d'un ensemble de logiciels de développement de programmes commercialisé par la société Microsoft.

**Netscape** : société américaine notamment éditrice des navigateurs web Netscape Navigator et Netscape Communicator. Netscape a vu ses parts de marché fondre après la sortie de Windows 95 dans lequel le navigateur concurrent Internet Explorer était fourni en standard, et a décidé en 1998 de diffuser sous licence libre le code source de ses logiciels, ce qui a engendré le projet Mozilla.

**OpenOffice** : nom d'un logiciel libre de bureautique (traitement de textes, tableur, etc.) équivalent de Microsoft Office et disponible pour de nombreux systèmes d'exploitation différents. Le projet *Moxie study* a estimé, grâce à des téléchargements aléatoires de fichiers sur Internet en janvier 2003, qu'OpenOffice était compatible avec 97 % des fichiers issus de traitements de textes, 98 % des feuilles de calcul et 94 % des présentations. OpenOffice est téléchargeable gratuitement sur le site [www.openoffice.org](http://www.openoffice.org).

**Open source** : expression anglaise pour désigner les logiciels libres. Le terme *open source* est historiquement très lié à la

démarche de marketing des logiciels libres initiée par l'association OSI.

**OSI** : *Open Source Initiative*, ce que l'on pourrait traduire en français par « Initiative pour les logiciels en code source ouvert ». Association fondée en 1998 par Bruce Perens et Eric Raymond avec l'objectif de promouvoir les logiciels libres auprès des acteurs économiques traditionnels. De nombreuses informations sur les logiciels libres sont disponibles sur le site web de cette association : [www.opensource.org](http://www.opensource.org).

**Perl** : nom d'un langage de programmation très utilisé dans le monde du logiciel libre. Les outils permettant d'utiliser Perl sont libres, gratuits et téléchargeables sur [www.perl.com](http://www.perl.com).

**Pilote** : dit aussi *driver*, logiciel fourni avec un périphérique matériel et permettant au processeur de communiquer avec lui.

**Programme** : synonyme de logiciel, désignant souvent plus précisément le code source.

**Propriétaire** : adjectif désignant un logiciel diffusé uniquement sous forme binaire et dont le code source reste la propriété et le secret professionnel de l'auteur ou de l'éditeur. Propriétaire n'est pas exactement le contraire de « libre » puisqu'il existe des logiciels dont le code source est disponible mais que l'on ne peut pas modifier et redistribuer, comme Minix avant avril 2000, PGP ou Java<sup>1</sup>. C'est aussi le cas de certains logiciels Microsoft<sup>2</sup>.

**Propriété intellectuelle** : système juridique permettant de protéger les créations originales en attribuant à l'auteur, du

---

1. La *Sun Community Source License* (SCSL) du logiciel Java de Sun Microsystems permet de distribuer le code gratuitement mais impose de payer une commission à l'éditeur d'origine pour chaque copie vendue.

2. La MSSSI ou *Microsoft Shared Source Initiative* constitue un projet très innovant de Microsoft qui diffuse certaines parties du code source de ses logiciels à des utilisateurs triés sur le volet et qui s'engagent à ne pas le redistribuer.

seul fait de sa création, des droits permanents sur le fruit de son travail.

**Python** : langage de programmation libre créé par Guido Van Rossum.

**RedFlag** : nom de la version chinoise de la distribution de Linux RedHat, et de la société chinoise qui la diffuse.

**RedHat** : nom d'une distribution de Linux et de la société qui la diffuse. Site web : [www.fr.redhat.com](http://www.fr.redhat.com) .

**Samba** : nom d'un logiciel libre de partage des ressources en réseau, compatible avec Linux et Windows NT serveur. Samba est téléchargeable gratuitement sur [www.samba.org](http://www.samba.org) .

**Sendmail** : nom d'un logiciel libre de routage du courrier électronique.

**Shareware** : logiciel propriétaire que l'on peut obtenir gratuitement pour le tester mais dont les fonctions sont limitées ou qui devient payant si on en fait usage au-delà d'une certaine période d'essai. Le logiciel Winzip est un *shareware*. Le terme français académique pour désigner ce type de logiciel est « partagiciel ».

**Spam** : courriers électroniques publicitaires et indésirables.

**SpamAssassin** : nom d'un logiciel d'identification et de destruction de spam.

**SuSE** : nom d'une distribution de Linux éditée par la société Novell. Site web : [www.suse.de/fr](http://www.suse.de/fr) .

**Système d'exploitation** : logiciel à la base du fonctionnement de l'ordinateur, servant à afficher les fenêtres, gérer les fichiers, etc. Un ordinateur ne peut pas fonctionner sans système d'exploitation. Exemples : MacOS X, Windows XP, Linux, Unix, OpenBSD, etc.

**TCO** : *Total Cost of Ownership*, « coût total de possession d'un logiciel », incluant le coût d'achat, mais aussi les coûts

d'installation, de formation, de maintenance, et le coût entraîné par les pertes de temps ou de données dues aux pannes.

**T<sub>E</sub>X** : nom du logiciel libre d'édition et de mise en page de textes créé par Donald E. Knuth. De nombreuses fonctions complémentaires ont été écrites pour ce logiciel, l'ensemble constituant le projet L<sup>A</sup>T<sub>E</sub>X.

**Thunderbird** : nom du logiciel libre de courrier électronique du projet Mozilla, concurrent de Microsoft Outlook. Thunderbird peut être téléchargé gratuitement sur le site web : <http://frenchmozilla.sourceforge.net> .

**Viralité** : synonyme péjoratif de *copyleft*.

**Web** : *World Wide Web* (littéralement, en français, « toile mondiale »). Ensemble de toutes les pages accessibles sur Internet et affichables à l'aide d'un navigateur. Peut être utilisé comme un nom propre ou un adjectif.

**Windows** : nom de la série des systèmes d'exploitation développés par Microsoft et vendue en standard avec la plupart des ordinateurs de type PC.

**Winmodem** : abréviation de « Windows modulateur démodulateur ». Il s'agit d'un appareil qui permet à un ordinateur sous Windows de se connecter à Internet via une communication téléphonique. C'est donc quasiment la même chose qu'un modem analogique « normal » sauf qu'une partie des fonctions réalisées par l'électronique du modem analogique sont effectuées par le logiciel Windows dans le cas du winmodem. Ce dernier est donc légèrement moins cher, grâce aux économies réalisées sur les composants, mais ne fonctionne qu'avec un système d'exploitation Microsoft.