

## 1.2. Prise en compte de la dimension technique : rappels sur les livrables d'un projet informatique

Par le groupe de travail Syntec Informatique-FniLL

Créé le 10/04/2010 - 09:28

Soumis par [Benjamin Jean](#) [1] le lun, 10/04/2010 - 09:28

### 1.2.1. Description du livrable

Les livrables d'un projet s'entendent de ce qui est défini comme tel dans le contrat. Rien ne doit être implicite.

Le livrable peut très utilement être décrit dans un document technique dit de « fourniture logicielle ».

Que l'on soit dans un schéma de distribution de Logiciels Libres ou non, un certain nombre de bons comportements, de « Bonnes Pratiques » doit être retenu<sup>1</sup>. Il est ainsi notamment nécessaire de :

- **Définir** le contenu et le périmètre technique : ce qui correspond généralement à l'objet principal du contrat de fourniture de logiciel (nom et descriptif fonctionnel sommaire de la livraison, pré-requis de déploiement, liste détaillée de la fourniture logicielle et des documentations qui l'accompagnent, procédures de test et de validation, procédures de recette, etc.).
  - **Justifier** clairement des licences et des noms des titulaires de droits sur les composants qui le constituent ou qui ont participé à sa constitution ([Cf](#) [2]. supra concernant l'établissement de ladite liste).
  - **Définir** clairement d'éventuelles limites de responsabilité sur des composants externes non conçus par le prestataire et livrés par le prestataire et/ou requis pour le déploiement. Il faut notamment décrire le niveau de version et/ou l'état technique sur lequel le prestataire aura validé la compatibilité ou le bon fonctionnement de ces composants externes choisis ou imposés par le client/l'intégrateur.
  - **Servir**, le cas échéant, de document de référence en cas d'arbitrage ou de litige, que celui-ci porte sur des aspects de responsabilité civile ou contractuelle, de propriété, de contrôle par une autorité de tutelle, etc.<sup>2</sup>. Ce document pourra être une annexe du contrat (de licence ou de fourniture) ou du procès-verbal de recette définitive<sup>3</sup>. Enfin, ce document pourra aussi servir de description de référence dans le cadre d'une réponse à appel d'offres, affichant clairement la liste des composants externes et les conditions de leur réutilisation.
- 
- <sup>1</sup>. Voir en annexe 2, exemple d'« annexe contractuelle de description d'une fourniture logicielle ». La liste d'informations donnée n'est pas exhaustive du point de vue de la livraison d'un logiciel puisqu'elle ne prend pas en compte les contextes projets spécifiques (ex : liste des exigences couvertes, liste des anomalies corrigées...). Pour autant, les informations décrites permettent de délimiter précisément de quoi le logiciel est constitué et quels sont les droits qui lui sont attachés.
  - <sup>2</sup>. Il pourra en effet être utilisé par des arbitres, des experts ou toute personne ayant à en connaître ou compétente pour en traiter.
  - <sup>3</sup>. Pour la livraison initiale et pour toute livraison de version majeure.

[Back to Top](#)

### 1.2.2. Description du code source

Les avantages techniques et les aspects juridiques du Logiciel Libre doivent être appréhendés dès la phase de conception. Deux recommandations peuvent ainsi être formulées concernant la « rédaction » de code :

- **Veille quant à l'état de l'art.** [3] Avant de développer un [composant](#) [4], il est nécessaire d'établir un état de l'art et rechercher s'il n'existe pas un logiciel ou une suite de logiciels libres appropriés. Cette pratique permet un gain en temps et en effort de développement. Il est en général moins coûteux de chercher et d'adapter un [composant](#) [4] existant et éprouvé que de le développer en partant de rien. Les Logiciels Libres sont souvent les solutions les plus adaptées (absence de frais de licence, possibilité de les adapter, support des communautés, etc.), à condition de réaliser le travail de renseignement, de validation des licences, et de songer au caractère du projet (communautaire, attaché à un éditeur, etc.) pour prévoir les pistes d'évolutions possibles.
- **Détection de codes.** Parallèlement à un l'état de l'art, il faut que les chefs d'équipe s'assurent de l'absence de copie de code et donc de contrefaçon (codes sans indication des sources, absence de licences, etc.). Ainsi, les entreprises peuvent recourir à des outils de détection afin de faciliter, automatiser et fluidifier le travail d'audit de code. Il existe des outils qui permettent d'assister tout ou partie de ces tâches : par exemple des logiciels de détection de [code source](#) [5] ou de licences. Ces produits (et services) aident les entreprises à comprendre la composition de leurs logiciels et à connaître l'origine de chacun des composants. Ces logiciels peuvent rapidement avoir une place prédominante dans la mise en place d'une [gouvernance](#) [6]1, et ils permettent notamment de gérer la réutilisation des composants tout au long du cycle de développement logiciel en toute sûreté tout en respectant les obligations des licences et en minimisant les risques.

Les principaux logiciels permettant d'auditer un code sont :

- **BlackDuck** [7] : Solution logicielle de gestion de la conformité et de la sécurité permettant la maîtrise des risques généralement associés à l'Open Source en automatisant les processus liés à la gestion du [code source](#) [5] tout au long du cycle de vie du développement des applications - recherche, sélection, approbation, validation et surveillance.
- **FOSSology** [8] (FOSS pour Free and Open Source Software) : Outil d'analyse de code logiciel construit autour d'une architecture ouverte et modulaire. Les modules inclus permettent l'analyse des licences, l'extraction de méta données, l'identification du type MIME (Multipurpose Internet Mail Extensions). Il dispose d'une large communauté et provient d'un développement interne rendu libre sous licence GPLv2 par la société [HP](#) [9] et continue de bénéficier de son soutien pour le développement de futures [versions](#) [10].
- **Palamida** [11] : Solution industrielle de sécurité dédiée au logiciel sous licence Open Source, elle permet d'identifier rapidement et de suivre du code non documenté, les vulnérabilités associées, ainsi que les problèmes de propriété intellectuelle ou de conformité.
- **OSLC** [12] (Open Source License Checker) : outil Open Source ([GNU GPL](#) [13]) d'inspection et d'analyse des licences de packages Open Source. OSLC analyse les informations de licences par l'extraction des informations de licence de tous les codes sources, puis il effectue une comparaison avec le texte de référence archivé dans la base et finalement résume toutes les informations de licence du paquetage.
- **Protecode** [14] : Outil permettant de gérer le développement de logiciel en temps réel grâce à la mise en place d'une [gouvernance](#) [6] efficace et la rationalisation de l'actif de [code source](#) [5], afin d'augmenter la valeur de l'entreprise et réduire les coûts.
- 1. Ce n'est d'ailleurs pas une surprise si la société Hewlett-Packard a développé son propre outil de détection pour maîtriser le risque d'utilisation involontaire d'un [composant](#) [4] open source, allant même jusqu'à en faire un logiciel libre : Fossology.

## 1.2. Prise en compte de la dimension technique : rappels sur les livrables d'un projet info

Publié sur Guide Open Source (<http://guideopensource.info>)

---

**URL source:** <http://guideopensource.info/guide/prise-en-compte-dimension-technique>

### Liens:

- [1] <http://guideopensource.info/user/1>
- [2] <http://guideopensource.info/glossary/term/26>
- [3] <http://guideopensource.info/glossary/term/15>
- [4] <http://guideopensource.info/glossary/term/3>
- [5] <http://guideopensource.info/glossary/term/4>
- [6] <http://guideopensource.info/glossary/term/9>
- [7] <http://www.blackducksoftware.com/>
- [8] <http://fossology.org/>
- [9] <http://guideopensource.info/glossary/term/46>
- [10] <http://guideopensource.info/glossary/term/14>
- [11] <http://www.palamida.com/>
- [12] <http://forge.ow2.org/projects/oslc3/>
- [13] <http://guideopensource.info/glossary/term/33>
- [14] <http://www.protecode.com/>