

DOCUMENT D'ANALYSE  
DE LA COMMISSION D'ÉVALUATION DE L'INRIA

# Évaluation des logiciels et autres réalisations

Préparé par David Margery, Jean-Pierre Merlet,  
Cordelia Schmid, Agnès Sulem, Paul Zimmermann (rapporteur)

Validé par la Commission d'Évaluation du 12 septembre 2007

## Résumé

La Commission d'Évaluation (CE) de l'INRIA a décidé d'entamer à partir de septembre 2006 une réflexion sur les éléments intervenant dans l'évaluation de l'activité scientifique des chercheurs (concours, promotions). Parmi ces éléments figurent les logiciels et autres réalisations. Un groupe de travail – constitué de David Margery, Jean-Pierre Merlet, Cordelia Schmid, Agnès Sulem et Paul Zimmermann – a élaboré un premier document, qui a été finalisé après présentation à la CE le 23 février 2007. Ce document prend en compte les modifications proposées lors d'une seconde discussion en CE le 12 septembre 2007.

## TABLE DES MATIÈRES

1. Introduction	2
2. Qu'est-ce qu'une réalisation ?	2
3. Comment évaluer une réalisation ?	3
4. Conclusions et recommandations	4
Documents de référence	6

### 1. INTRODUCTION

Outre les publications scientifiques (articles de journaux et de conférences), la recherche en informatique — notamment à l'INRIA — fait une place importante au logiciel et autres réalisations. Ces « réalisations » viennent souvent valider ou nourrir une recherche plus fondamentale.

Or, si on a de nombreux outils pour évaluer les publications scientifiques (comité de lecture des revues et conférences, taux de sélection, facteur d'impact, *citeseer*, *google scholar*, *h-number*<sup>1</sup>), on se retrouve souvent démuné pour évaluer une réalisation (originalité, difficulté, impact) et la contribution d'un candidat à celle-ci. Ce document donne quelques pistes pour l'évaluation des logiciels et autres réalisations. Il est destiné (i) à usage interne à l'INRIA : commission d'évaluation (rapporteurs des dossiers), direction, plan stratégique; (ii) à usage externe à l'INRIA : aider les candidats à mieux décrire leurs réalisations.

### 2. QU'EST-CE QU'UNE RÉALISATION ?

Dans le contexte INRIA, on pense souvent à des « réalisations logicielles », or le terme « réalisation » couvre un spectre beaucoup plus large : une plateforme de réalité virtuelle, un CAVE, un workbench, un robot, un volant de retour d'effort, un système embarqué, un prototype destiné au domaine médical sont en effet des exemples de **réalisations non logicielles**. On peut également considérer une activité de standardisation, notamment si elle s'accompagne d'un prototype (un démonstrateur IETF par exemple); voir aussi l'OMG (*Object Management Group*), les normes ISO-OSI, ou encore RM-ODP (*Reference Model of Open Distributed Processing*, <http://www.rm-odp.net/>). D'autres exemples sont : l'extraction de données pertinentes à partir d'une base fournie par un partenaire industriel, une étude auprès du grand

---

<sup>1</sup>Voir le document *Que mesurent les indicateurs bibliométriques*, A.-M. Kermarrec, E. Faou, J.-P. Merlet, Ph. Robert, L. Segoufin, septembre 2007, 25 pages.

public et son dépouillement (par exemple dans les domaines de la réalité virtuelle, de l'ergonomie, des interfaces utilisateur). Une **réalisation informatique** n'est quant à elle pas forcément un programme : cela peut être une base de donnée (exemple le LEFFF ou lexique des formes fléchies du français, <http://www.lefff.net/>), ou encore la base d'images Caltech 101, [http://www.vision.caltech.edu/Image\\_Datasets/Caltech101/Caltech101.html](http://www.vision.caltech.edu/Image_Datasets/Caltech101/Caltech101.html)), un benchmark (par exemple la *Frisco test suite*, <http://www-sop.inria.fr/saga/POL/>), voire un *challenge* comme le challenge RSA pour la factorisation d'entier (<http://www.rsasecurity.com/rsalabs/node.asp?id=2093>), ou le *Visual Object Classes Challenge* du réseau d'excellence européen Pascal (<http://www.pascal-network.org/challenges/VOC/>).

Dans le contexte de l'évaluation d'un dossier de chercheur, on ne considère ici que des **réalisations innovantes**, qui s'inscrivent dans une activité de recherche et sont évaluées en tant que telles<sup>2</sup>.

La note de G. Giraudon de 2001 (voir les références en fin de document) distingue **objet de recherche** et **objet de transfert**. La CE considère que cette distinction n'est pas claire (dans quelle catégorie se trouvent des logiciels comme Caml ou Coq?). D'autre part une telle classification a peu d'intérêt dans le cadre — considéré ici — d'un dossier de candidature ou de promotion, où un transfert éventuel sera indiqué par ailleurs. À noter cependant la classification TRL (*Technology Readiness Level*, cf [http://en.wikipedia.org/wiki/Technology\\_Readiness\\_Level](http://en.wikipedia.org/wiki/Technology_Readiness_Level)), qui mesure le degré de maturité d'un logiciel ou plus généralement d'une technologie. Cette classification, utilisée en particulier par la NASA, va du niveau 1 (*Basic principles observed and reported*) au niveau 9 (*Actual system 'flight proven' through successful mission operations*).

### 3. COMMENT ÉVALUER UNE RÉALISATION ?

Trois points sont importants quand on évalue une réalisation : (i) que fait cette réalisation ? (ii) qui l'utilise ? (iii) quelle est sa diffusion ?

Il s'agit d'évaluer à la fois la **difficulté d'une réalisation** et sa **pertinence scientifique**. Les deux aspects ne vont pas forcément de pair : un logiciel écrit en une journée peut être utile à des milliers de chercheurs ; au contraire un logiciel ayant demandé plusieurs années et une haute technicité peut n'avoir qu'un faible intérêt scientifique.

---

<sup>2</sup>On pourrait citer ici de nombreux exemples de réalisations « non innovantes », mais qui s'inscrivent dans une démarche expérimentale, et qui ont conduit à de nouveaux résultats de recherche, soit parce qu'elles ont posé de nouvelles questions, soit parce que l'expérience a donné des résultats différents de ceux attendus.

On distingue l'évaluation **on-line** quand la réalisation peut être effectivement testée par l'évaluateur (cas d'un logiciel téléchargeable sous forme source ou binaire) de l'évaluation **off-line** quand au contraire la réalisation ne peut pas être essayée (exemple du robot humanoïde de Toulouse, <http://www2.cnrs.fr/presse/journal/2901.htm>).

Outre le « volume » de la réalisation (par exemple nombre de lignes de code pour un logiciel, nombre de composants pour une réalisation matérielle) il est important de pouvoir mesurer le caractère innovant et l'impact de la réalisation, son utilité pour la recherche personnelle ou les autres, le temps passé à cette réalisation (par exemple le temps d'intégration pour une réalisation matérielle). S'il s'agit d'un logiciel ayant subi plusieurs refontes, compter uniquement le nombre de lignes de code de la dernière version ne rend pas justice aux auteurs.

Il est parfois difficile de mesurer l'impact d'une réalisation, par exemple d'une bibliothèque logicielle, qui peut être utilisée dans plusieurs distributions Linux, sans forcément que les auteurs le sachent. Ceci peut arriver aussi pour une réalisation matérielle : exemple d'un robot qui est copié par des concurrents.

#### 4. CONCLUSIONS ET RECOMMANDATIONS

Il est important de bien distinguer la **contribution d'un candidat** à une réalisation de l'**évaluation globale** d'une réalisation ou d'un logiciel. Pour la contribution d'un candidat, qui est ce qui nous intéresse ici, nous proposons une fiche détaillée pour la section « réalisations » des dossiers de candidature.

Concrètement, la CE propose de modifier la section 4 (*Réalisation et diffusion de logiciels*) des dossiers de candidature, afin d'avoir un certain nombre de **fiches détaillées** (facultatives) pour les logiciels/réalisations les plus significatifs (avec un nombre maximum comme pour les publications les plus significatives, par exemple 2 pour les dossiers de CR, 4 pour les dossiers de DR). Voir la proposition de fiche en annexe<sup>3</sup>.

Concernant l'évaluation (globale) de réalisations ou logiciels (*software review* en anglais), il s'agit d'un sujet important en soi, sur lequel il reste beaucoup de chemin à faire pour atteindre un même niveau de qualité que l'évaluation des articles scientifiques. La principale difficulté consiste à mettre en œuvre une évaluation par les pairs, telle celle qui se fait de façon anonyme

---

<sup>3</sup>On pourrait d'ailleurs reprendre ce modèle dans les fiches de synthèse lors des évaluations d'équipe-projet.

dans les revues à comité de lecture. On pourrait alors parler de véritable **publication logicielle**. Des initiatives, pour la plupart isolées, existent déjà : citons par exemple sans recherche d'exhaustivité la revue *ACM Transactions on Mathematical Software*, la conférence *ACM Symposium on Discrete Algorithms*, les *tool presentations* de la conférence *Computer Aided Verification*, l'*editorial board* des logiciels CGAL ou GAP, le système de revue formelle des modules de la distribution Linux Fedora. On pourrait par exemple imaginer avoir pour chaque logiciel un **software impact factor**, similaire au *journal impact factor* des revues. On pourrait de même généraliser le fameux « *h-number* » aux logiciels : le plus grand entier  $h$  tel que  $h$  personnes ont téléchargé chacune (au moins)  $h$  versions différentes de ce logiciel.

Pour généraliser le *software review*, il faudrait pouvoir avoir l'avis d'experts de renommée internationale, et il semble par conséquent difficile de mettre en place cela à l'échelle restreinte d'un « petit organisme ». Rien n'empêche cependant d'en discuter au sein de l'INRIA.

## DOCUMENTS DE RÉFÉRENCE

- *Développement de logiciels à l'INRIA. Objectifs et organisation*, G. Giraudon, sept. 2001, <http://www.inria.fr/interne/dirdri/developpement/logiciel/developpement-10.pdf>, 7 pages. Cette note aborde les quatre points suivants :

- Quels sont les types de logiciels développés dans l'institut et pour quels usages ?
- Qui développe des logiciels et dans quel cadre ?
- Quels bilans peut-on en tirer ?
- Comment améliorer ce cadre, quelles retombées peut-on en attendre ?

Cette note s'intéresse donc plus à l'organisation du développement logiciel à l'INRIA qu'à l'évaluation proprement dite.

- *Contribution à la réflexion sur le développement logiciel à l'INRIA. Rapport et recommandations*, Éric Gautrin, Bernard Martin, décembre 2002, 36 pages, <http://www.inria.fr/interne/dirdri/developpement/contributionsGM/mission-gautrin-martin.pdf>. Ce rapport de mission très complet aborde les points suivants : le développement logiciel dans les projets de recherche, les métiers du développements logiciel, la diversité des besoins et attentes, et formules diverses recommandations. La section qui nous intéresse plus particulièrement ici est 2.1.6 *Motivations : Se faire évaluer à l'INRIA*, qui dit : « Ils [les chercheurs] considèrent être essentiellement évalués sur des critères de publications, sans réelle prise en compte des activités de développement. »

- *Évaluation des activités non académiques. Avis sur le texte « Recommandations aux évaluateurs » du 29/04/02*, E. Ledinot, février 2004, 3 pages. Le présent rapport s'inspire largement de la section 3 (*Activités produisant des résultats*) de ce document, et de sa recommandation finale.

# Projet de Fiche de Description de Logiciel ou Autre Réalisation<sup>4</sup>

Décrire ici vos logiciels ou réalisations (au plus 4), en détaillant les plus significatifs (au plus 2) au moyen de la fiche détaillée ci-dessous, dont vous ferez autant de copies que nécessaire<sup>5</sup>. Vous pouvez faire valider l'impact et la difficulté de ces réalisations via l'une des lettres de recommandation.

---

<sup>4</sup>Cette fiche vise à compléter la section 4 (*Réalisation et diffusion de logiciels*) des dossiers de candidature, que nous proposons de renommer en *Logiciels et autres réalisations*. Tout le texte (sauf les notes en bas de page) est à ajouter à ces dossiers. Cette fiche est destinée aux candidats considérant qu'ils ont une contribution logicielle (ou de réalisation) dont l'impact est significatif ou résulte d'efforts importants. Cette fiche vise à donner les moyens de faire une évaluation aussi approfondie du logiciel — ou de la réalisation — que celle d'un article scientifique par le comité de lecture d'une conférence ou revue sélective.

<sup>5</sup>Nous proposons une limite de 4 logiciels dont 2 fiches détaillées pour les concours CR et les promotions CR2-CR1 ; de 8 logiciels dont 4 fiches détaillées pour le concours DR et les promotions DR1 et DR0.

<b>Description courte du logiciel ou de la réalisation</b>
Que fait le logiciel ou la réalisation ? Est-ce un prototype ou un logiciel à vocation plus pérenne ? Est-ce un logiciel généraliste ou spécialisé ?
<b>Contribution du candidat</b>
Indiquer les différents auteurs, et votre contribution personnelle (temps et/ou nombre de lignes, etc) à la fois pour la conception, la réalisation, la documentation, la maintenance.
<b>Originalité et difficulté</b>
Quels sont les logiciels concurrents ? En quoi le logiciel est-il nouveau, original ? Quels ont été les connaissances nécessaires, les choix techniques, éventuellement l'encadrement ? Quels sont les principaux résultats obtenus ou à obtenir ?
<b>Validation et impact</b>
Qui utilise votre logiciel et pour quoi (membres de l'équipe, participants d'un projet européen, transfert industriel, ...) ? Quel est l'impact du logiciel ou de la réalisation sur vos travaux de recherche, sur ceux de vos collègues, sur la communauté scientifique (mailing lists, forum de discussion, ...) ? Indiquer le cas échéant le nom et les coordonnées d'une personnalité pouvant attester de cet impact.
<b>Diffusion</b>
Mentionner les publications liées à la réalisation, les brevets éventuels, et autres éléments de diffusion (dépôt APP, page web, documentation, licence, tutoriaux, ...) Éventuellement : historique, évolutions futures.