R++, the Next Step

C. Genolini[a], **J. Falcou**[b] et **R. Tournier**[c]

[a]UMR 1027 INSERM
37 allée jules Guesde
31000 Toulouse
<cgenolin@u-paris10.fr>

[b]LRI
Université Paris-Sud
91400 Orsay, France

[b]IRIT-SIG
Univ. Toulouse 1 Capitole
31000 Toulouse, France

Résumé : *R* est un langage de programmation dédié à l'analyse statistique. Intégré dans un environnement spécifique, il met à disposition des utilisateurs une grande variété de techniques statistiques et graphiques.
Logiciel gratuit et open source, le développement de *R* s'appuie sur une large communauté. Malheureusement, *R* souffre également d'un certain nombre de faiblesses qui nuisent à ses performances, comme la gestion du parallélisme ou des grandes dimensions.
R++, the Next Step est un projet de développement d'une nouvelle implémentation de *R*. Il a pour vocation d'être **compilable**, de permettre en natif **la gestion du parallélisme** et l'exploitation des **bases de données de grande dimension**. *R++* sera également intégré dans une interface homme machine dédiée, spécifiquement conçue pour les analyses statistiques.

Mots clefs : Big data, parallélisme, interface homme machine, logiciel métier

1 Forces et faiblesses de *R*

R est un langage riche présentant de nombreux avantages. Gratuit et open source, ce qui dans la période actuelle de difficultés financières pour nombre d'instituts académiques est non négligeable, il permet une rapide intégration de toutes les nouvelles méthodes statistiques modernes en permettant aux utilisateurs d'eux-mêmes enrichir le langage. Le nombre de packages disponibles ne cesse d'augmenter. Pratiquement toutes les méthodes statistiques de pointe sont disponibles sous *R*. La communauté ne cesse de grandir.
Malheureusement, les points que nous venons de présenter sont à mettre en regard d'un certain nombre d'inconvénients et de limites. Comme beaucoup de langages, *R* n'a pas été conçu pour gérer le parallélisme ou la concurrence. De même, la taille maximale des bases de données exploitables est de l'ordre de la dizaine de giga alors que les spécialistes des grandes dimensions travaillent d'ores et déjà avec des dizaines voire des centaines de terra. Enfin, *R* n'est pas compilable, ce qui entraine un manque à gagner important en termes de performance.
Ces points sont régulièrement pointés par la communauté [1]. Différents packages se proposent d'y remédier. Malheureusement, ils ne présentent pas à ce jour des performances très prometteuses. En tout état de cause, aux dires de certains experts en parallélisme, un langage non initialement conçu comme parallèle pourra difficilement être rendu vraiment efficace par des surcouches. De même, sous *R*, les appels de fonctions se font par valeur, ce qui signifie que les variables passées comme arguments sont dupliquées en mémoire. Ce mode de fonctionnement présente certains avantages et prévient bien des erreurs de programmation. Par contre, dans le cadre de manipulation de données de grandes dimensions, il n'est clairement pas optimum (dans *R*, la modification d'une valeur dans une base de données entraine la duplication de toute la base.)

2 Projet *R++, the Next Step*

R++, the Next Step est un projet de développement d'une nouvelle implémentation de *R*. L'objectif est de conserver les forces de *R* tout en améliorant ces faiblesses : *R++* a pour vocation d'être compilable, d'intégrer en natif la gestion du parallélisme et de permettre l'exploitation des bases de données de grande dimension. *R++* sera également intégré dans une interface homme machine dédiée, spécifiquement conçue pour les analyses statistiques.
R++ est clairement un projet ambitieux. Afin de le mener à bien, nous avons réuni des experts de différents domaines. Actuellement, quatre équipes de chercheurs participent au projet, chacune sur un axe spécifique.

- **Compilation haute performance et parallélisme** : mené par l'*UMR 8623 LRI - Équipe Parallel Systems, Orsay* (l'équipe qui a développé **NT2** [2]), cet axe vise le développement du cœur de *R++*. Trois sources de parallélisme sont envisagées : le multi-cœur, la carte graphique et le cloud.

- **Big data** : L'*UMR 5505 IRIT - Equipe Système d'Informations Généralisées, Toulouse* est plus spécifiquement en charge d'étudier l'exploitation et le stockages des grandes bases de données (plusieurs To, via la plateforme d'expérimentation OSIRIM).

- **Interface homme machine** : *InSitu*, équipe commune entre le LRI et INRIA, travaille sur l'interface homme machine. A travers des séances de prototypage vidéo, elle permet aux statisticiens de mettre en avant les faiblesses des logiciels actuels et propose des outils plus adaptés aux analyses modernes (comme une interaction bi-directionnelle continue entre les graphes et les données).

- **Logiciel métier** : Enfin, les *UMR 1027 INSERM, Toulouse* et *UMI 669 INSERM, Paris* sont des unités composées en parti de médecins et de bio-statisticiens. Elles sont en charge de la gestion générale du projet. Elles participent aux différents axes, s'assurent que *R++* reste le plus proche possible de *R* et soit effectivement un logiciel métier.

3 Avancement du projet

A ce jour, l'écriture d'un compilateur parallèle et haute performance se basant sur le module NT2 est en cours.
Un étudiant M2R recruté par l'équipe ParSys travaille actuellement à la parallélisation de méthodes statistiques emblématiques et modernes, comme les imputations multiples ou le bootstrap.
Deux étudiants M2R ont également rejoint l'équipe SIG. Ils explorent les méthodes de calcul de statistique partiel afin de pouvoir utiliser le swap.
Enfin, sous l'égide de l'équipe InSitu, un groupe mixte de statisticiens et d'informaticiens se réuni régulièrement pour faire du prototypage vidéo.

Références

[1] Ross Ihaka (2010) R: Lessons Learned, Directions for the Future. In *Joint Statistical Meetings*

[2] Falcou, J., Sérot, J., Pech, L., et Lapresté, J. T. (2008). Meta-programming applied to automatic smp parallelization of linear algebra code. In *Euro-Par 2008-Parallel Processing* (pp. 729-738). Springer Berlin Heidelberg.