

# Discussion autour de LEMA et d'autres sujets

Compte rendu des discussions du 19 octobre  
Deuxième réunion EVA-Flo : 18-19 octobre 2007

Secrétaire de séance : Nathalie Revol

## Présents

Sylvain Chevillard, Sylvain Collange, Marc Daumas, David Defour, Florent de Dinechin, Claude-Pierre Jeannerod, Nicolas Jourdan, Christoph Lauter, Vincent Lefèvre Philippe Langlois, Nicolas Louvet, Ivan Morel, Hong Diep Nguyen, Nathalie Revol, Guillaume Revy, Gilles Villard.

## Discussion autour de LEMA

David : quelles sont les notions utiles qui ont été extraites pour LEMA, depuis la dernière réunion ?

Vincent : des lambda-expressions avec des éléments de base : nombres divers et variés, intervalles. . . Il manque l'expression des nombres binaires à la C99.

David rappelle que l'objectif est de définir un langage qui permette à tous d'exprimer nos besoins : mémoire, ulp, erreur absolue, erreur relative, précision. . .

Vincent : en MathML il y a plus que ce dont nous avons besoin dans certains domaines, dans d'autres domaines il manque des notions, qui permettent d'exprimer que tel calcul est effectué dans telle arithmétique, avec telle précision. Le langage XML est fait pour permettre d'ajouter ces notions, il est extensible.

David : est-il pertinent de prendre en compte des contraintes architecturales ? Peut-on le faire ?

Vincent : on peut tout à fait étendre le langage pour pouvoir les exprimer, on peut aussi ignorer des informations quand il y en a trop (ou qu'on ne sait pas les traiter).

David : est-ce que toutes les personnes ici présentes seraient capables de définir précisément ce dont elles ont besoin ?

Christoph n'a pas besoin de notions particulières, mais surtout de pouvoir communiquer des briques de base : des expressions, des booléens, une erreur, une liste de points. . .

Florent : on peut échanger des expressions, on sait communiquer quelques précisions, à savoir double, double-double, triple-double, mais pas de précision complètement paramétrée, en revanche on ne sait pas faire grand-chose avec les informations architecturales.

David fait référence aux simulateurs d'architectures, qui sont paramétrés et très modulaires : un module par unité.

Florent : pour le moment, lorsque l'on communique un polynôme et une fonction, une information implicite est que le polynôme approche la fonction.

Sylvain Ch. confirme que tous les appels de fonctions de Sollya fonctionnent ainsi, avec une sémantique implicite.

Vincent : il faudrait éviter ce qui est implicite.

Marc : il faut séparer ce qui est une donnée (par exemple un polynôme) des outils qui sauront travailler si on leur fournit les bonnes données.

Christoph : mais que faire d'une soupe de données ? L'objectif d'EVA-Flo est algorithmique.

Marc : les outils peuvent prendre une donnée, peut-être la transformer et la remettre dans la soupe.

David : attention à ne faire pas faire comme SimpleScalar, qui est une grosse tambouille dans laquelle chacun a rajouté selon ses besoins et où on ne peut maintenant plus rien modifier par peur des effets de bord. Je suggère que l'on fonctionne par passage de messages, pour plus de sûreté.

Nicolas L. : dans ce cas on perd la trace des modifications effectuées. Comment compte-t-on fonctionner dans EVA-Flo : en conservant un fichier de log ?

Marc : oui.

Christoph : apparemment Marc a envie de conserver toutes les informations, mais cela va requérir trop de mémoire pour les fichiers de log.

Marc : veut-on définir des outils qui agissent sur des fichiers de données ou autre chose ?

Christoph : on devrait encapsuler la moulinette en une succession de petites procédures.

Nicolas J. : le problème est alors de définir le format d'échange entre les petits blocs.

Christoph : il y a deux sortes de problèmes, d'une part faire communiquer deux blocs de Sollya et c'est facile, d'autre part faire communiquer un bloc de Sollya ou un outil externe avec un (autre) outil externe.

## Discussion autour de la prise en compte de l'architecture

Sylvain Co. : il faut pouvoir exprimer ce que l'on veut, comme " évaluer  $e^{\cos x^2}$  en double précision avec arrondi fidèle " pour la partie mathématique.

David : pour la partie matérielle, les paramètres importants sont : la latence, le débit, la mémoire, le nombre de registre, le temps d'accès aux registres...Le problème est qu'il y a beaucoup d'informations, lesquelles sont pertinentes ?

Florent : attention à ne pas faire une procédure par architecture existante.

David : il faut envoyer toutes ces informations à un outil, qui les utilisera s'il sait le faire et les ignorera sinon.

Florent : pour ne pas perdre le caractère quasi-optimal d'un outil tel que celui de Christoph, on ne peut pas le généraliser à tout. Autrement dit on ne sait pas faire un outil qui sait utiliser toutes les informations et qui retourne un résultat de qualité.

L'idée est d'en faire assez à la main avant d'être capable de dégager des principes qui permettent d'automatiser.

Nathalie : ce n'est pas la question, de savoir si un outil saura traiter toutes les informations. La discussion diverge.

Florent, Sylvain Co. : non, on ne diverge pas, il faut avoir résolu assez de problèmes à la main pour savoir quelles sont les informations pertinentes à mettre dans le langage.

Christoph : il faut un langage très expressif, mais on ne saura effectivement pas construire d'outils qui sauront tout traiter. D'ailleurs, on n'est pas obligé de donner toutes les informations à tous les outils.

## Discussion autour de la preuve

Marc : on a besoin de pouvoir exprimer des affectations, des intervalles et des informations du type  $x$  *approche*  $y$ , de traduire en Gappa et d'un outil qui sache générer des hints pour Gappa si jamais Gappa n'arrive pas à conclure tout seul.

Nathalie : et comment traiter les connaissances que Nicolas L. a acquises autour des algorithmes compensés par exemple ?

Christoph : comment exprimer des informations supplémentaires et la connaissance en ingénierie que nous avons ?

Marc : il faut avoir à disposition un vaste langage sur lequel nous pourrions nous greffer

...

comment définir une erreur absolue entre  $x$  et  $y$  ? Nous, nous savons que  $x$  est plus précis que  $y$ , mais comment l'exprimer ?

Marc suggère d'utiliser un code exact (par exemple si on cherche l'erreur entre un produit de matrices exact et un produit de matrices calculé en double précision), ou alors d'exprimer qu'il s'agit d'une erreur entre un objet mathématique et un objet obtenu par un procédé d'approximation, si on cherche à évaluer  $e^x$  par exemple.

en utilisant l'exposé de Gilles, discussion sur la façon de définir l'erreur entre  $R$  et  $\tilde{R}$  : cette erreur ne provient pas de l'algorithme utilisé.

## Conclusion

Un groupe de réflexion autour de l'architecture doit se créer et avancer, avec David, Sylvain Co., Florent, Guillaume, Claude-Pierre.

La prochaine réunion EVA-Flo aura lieu en avril 2008. (Rappel : deux réunions par an.)