

## Title: Noyau réflexif et Isolation

Keywords: language design, isolation, proxy, compiler

Laboratoire, institution et université: INRIA Lille Nord Europe

Location: lille, France

Team: [Equipe RMoD – INRIA Lille Nord Europe](#)

Supervisor

Stéphane Ducasse [stephane.ducasse@inria.fr](mailto:stephane.ducasse@inria.fr)

<http://stephane.ducasse.free.fr>

## Contexte

De plus en plus d'applications nécessitent des comportements dynamiques évolués souvent basés sur la possibilité de l'application de modifier son propre comportement. Ces comportements sont souvent basés sur l'utilisation des aspects réflexifs des langages. Nous voyons aussi apparaître des langages dynamiques tels Lua qui sont utilisés et embarqués dans des applications desktop comme Adobe Photoshop ou des langages de scripts comme F-Script sur Mac OS-X qui peuvent être injectés dans n'importe quelle application Cocoa. De plus en plus d'applications se mettent à jour en chargeant dynamiquement du code.

Le chargement dynamique de code potentiellement hostile (untrusted) a été abordé pour les langages statiques par la vérification de bytecode, mais le typage dynamique et la réflexion limitent fortement ces approches.

Les approches de type bac à sable sont sécurisées par construction mais limitent les extensions et les interactions avec le reste du système. Dans un même temps, de nouveaux langages dynamiques tels que Ruby, Python ou Pharo deviennent de plus en plus utilisés pour la construction d'applications sensibles. Dans de tels langages, lors de l'exécution du programme, il est possible de pratiquement modifier quasiment tous les aspects de ces langages ce qui donne leur puissance mais est clairement un problème pour la sécurité. Suivant les circonstances, il est capital de pouvoir ou non accéder aux aspects dynamiques du langage : ainsi Seaside un framework de développement d'applications dynamiques est basé sur la possibilité de manipulation de la pile. De nouveaux langages comme Erights proposent de nouvelles abstractions de sécurité mais cela reste insuffisant dans un monde de plus en plus interconnecté.

**Objectifs** : Ce stage a pour objectif d'étudier comment il est possible de réconcilier des langages extrêmement ouverts et dynamiques avec une problématique de sécurité. L'idée n'est pas de valider du bytecode comme cela se fait déjà mais d'étudier comment par construction une application peut être à la fois extensible, dynamique mais sécurisée.

Le travail pourra suivre le plan suivant:

- Etude des abstractions ouvertes (ou protocol à meta objets): Dans les langages dynamiques existants nous allons classifier les différentes opérations et abstractions (telles que le contrôle

de l'envoi de message, la recompilation dynamique, l'extension de packages ou de classes) en fonction de leur pouvoir d'expression et donc dangerosité.

- Implementation du noyau d'un langage à objet et un meta-object protocol sous forme de couches.

Le stage s'effectuera en utilisant Pharo <http://pharo.org> un langage dynamique réflexif pur. L'étudiant pourra suivre l'excellent Mooc <http://mooc.pharo.org> pour découvrir le langage.

La lecture de <http://books.pharo.org/booklet-ReflectiveCore/> est un point conceptuel important.

## References

Gilad Bracha and David Ungar, "Mirrors: design principles for meta-level facilities of object-oriented programming languages," Proceedings of OOPSLA '04, ACM SIGPLAN Notices, ACM Press, New York, NY, USA, 2004, pp. 331-344.

Stéphane Ducasse, "Evaluating Message Passing Control Techniques in Smalltalk," Journal of Object-Oriented Programming (JOOP), vol. 12, no. 6, June 1999, pp. 39—44.

Gregor Kiczales, Jim des Rivières and Daniel G. Bobrow, The Art of the Metaobject Protocol, MIT Press, 1991.

Camille Teruel, Stéphane Ducasse, Damien Cassou, Marcus Denker, Access control to reflection with object ownership, DSL 2015.

G. Polito, S. Ducasse, L. Fabresse, N. Bouraqadi, B. Van Ryseghem: Bootstrapping reflective systems: The case of Pharo. In Science of Computer Programming 96: 141-155 (2014)

Compétences espérées

Curiosité

Envie d'apprendre

Bonne communication

Bon programmeur

Anglais