

Sujet de stage L3 Info

« Maintenance évolutive pour ProofGeneral »

Encadrant : Érik Martin-Dorel (erik.martin-dorel@irit.fr)

Février 2020

1 Informations pratiques

- Stage de L3 Info à temps plein
- Nature du travail : développement logiciel
- Proposition de dates : 6 avril → 30 mai 2020 (8 semaines)
- Établissement : [IRIT](#)

2 Description du contexte du sujet

ProofGeneral¹ (PG) est un plugin (le terme exact étant *major mode*) pour l'éditeur GNU Emacs, permettant d'interagir avec des assistants de preuve interactive. Initié en 1994 par David Aspinall à l'Université d'Edinburgh, son développement s'effectue maintenant sur la plateforme GitHub. Comme tout mode Emacs, ProofGeneral est écrit en Emacs-Lisp, langage fonctionnel de la famille des langages LISP.

Le mode est activement utilisé² et trois prouveurs sont actuellement supportés (Coq, PhoX et EasyCrypt).

Le support de ces prouveurs s'appuie actuellement sur une interaction de type REPL (Read-Eval-Print Loop), intrinsèquement séquentielle, tandis que Coq dispose depuis la version 8.5 d'un nouveau protocole basé sur des messages XML, permettant d'éditer un *document* Coq de manière plus efficace et asynchrone.

Une nouvelle version de ProofGeneral avec un support asynchrone des preuves Coq a été récemment développée, cependant l'implantation en question est actuellement au stade de *proof-of-concept* et celle-ci a été développée en remplaçant complètement le back-end REPL par le nouveau back-end, ce qui implique qu'il n'est pas possible d'utiliser une seule version de ProofGeneral pour pouvoir effectuer des développements Coq et (PhoX ou EasyCrypt).

3 Description du travail demandé

Le but de ce stage est de se familiariser avec un nouveau langage fonctionnel (Lisp) et de participer au développement du mode afin de :

1. modifier son architecture logicielle pour pouvoir bénéficier des deux back-ends (REPL et async.) en fonction du prouveur lancé ;
2. contribuer à la maintenance évolutive du back-end asynchrone.

1. <https://proofgeneral.github.io/>

2. 321 « ★ » sur [GitHub](#), 13000⁺ téléchargements depuis que PG figure dans [MELPA](#)

La mise en œuvre de l’item 2 pourra donc s’appuyer sur le code du proof-of-concept (ainsi que sur la spécification du protocole Coq/XML) et des tests spécifiques pourront être envisagés pour compléter la chaîne d’intégration continue actuellement mise en place.

4 Environnement technologique

Environnement de développement : [GNU Emacs](#), [Git](#), [Travis CI](#),

Langages : [Emacs Lisp](#), [Coq](#)

5 Prérequis

Le candidat devra avoir de bonnes aptitudes d’analyse, de rigueur et de communication (notamment en anglais écrit), une bonne connaissance du paradigme de programmation fonctionnelle, ainsi qu’une première expérience d’utilisation de l’assistant de preuve Coq (par exemple acquise dans l’UE TAPFA). Mais aucune connaissance préalable en Lisp n’est requise.

Les candidats intéressés doivent prendre contact dès que possible avec erik.martin-dorel@irit.fr, qui fera la liaison avec l’équipe de développement de Coq-ProofGeneral.