PSTL : Interface Web Autobill

Fazazi Zeid Luo Yukai Brahima Dibassi

23 mars, 2023

Contents

1 Contexte du projet

1.1 Que-est-ce qu'est Autobill ?

1.2 Comment on rentre dans ce projet ?

Le sujet de notre Projet STL va donc être de soutenir l'effort de développement en proposant une interface sur le Web permettant la libre manipulation de l'outil. En effet, il n'est pour l'heure uniquement utilisable via l'invite de commandes, en ayant au préalabe cloner le repertoire GitLab où il est hébergé, et suivi les étapes d'installation, nécessitant des utilitaires de paquetes comme opam et dune.

Notre approche vise donc à faciliter l'utilisation d'Autobill avec une interface Web qui prendrait la forme d'un "mini" environnement de développement, avec un éditeur de code et une sortie standard sur le côté. Aussi, pour le rendre le plus accesibble, en entrée, un langage avec un syntaxe similaire à OcamL sera disponible en entrée et pourra être utilisé pour écrire les programmes à tester. Plusieurs modes d'évaluation seront disponibles, comme la possibilité d'interpréter ou d'afficher l'occupation mémoire minimum du programme en entrée.

Néanmoins, le langage accepté d'Autobill étant Call-By-Push-Value, il est nécessaire de pouvoir traduire le langage camélien pour permettre l'évaluation des contraintes et de l'allocation mémoire du programme. Ainsi, un travail sur la compilation et la traduction d'un langage à un autre va avoir lieu, passant par les étapes de construction d'AST camélien et par la traduction de ce dernier en AST compréhensible par Autobill.

2 Avancement

2.1 MiniML

- Description du language
- Description traduction vers lcbpv
  - Expliquer lcbpv (Contexte,Fonctionnement,Difficultées)
- Choix de Conception

2.2 Archi Full Client

- Design du client (Images ? Inspirations ?)
  - Schema
- React/Js_Of_Ocaml
  - Interlanguage Pb Rencontrée (Thread Block , Exception …)
  - Pourquoi React

        * Tenter un lien PC3R ??
    - Description Js_Of_Ocaml
        * Neccesaire dans l'archi full client (Pourquoi)
- Solveur (Web Assembly)

2.3 Archi Serveur + Client

- Design du server (Images ? Inspirations ?)
    - Schema
        * Montrer les echanges de données
    - Pourquoi Node ? (Et pas ocaml par exemple)
    - Tenter un lien PC3R ??
- Solveur

2.4 Client VS Serveur

- Avantages Inconvénients

3 Projections (Rapport Suivant)

3.1 MiniML

- Description + Full Spec
- Lazyness
- Bibliothéque de tests (Structure de donéees complexes)

3.2 Serveur

- ?

3.3 Client

- ?

3.4 Tests

- Comparaison Client vs Servers
- Comparaison YAML vs Autobill

4 Bibliographie

- Reprendre le carnet de bord