

Compilateur GCC

Un compilateur est un programme informatique qui transforme un code source écrit dans un langage (langage source) en un autre langage (langage cible) [¹]

On s'intéresse à la suite GCC (GNU Compiler collection). Il s'agit d'une chaîne de compilation. L'utilisateur peut écrire son programme dans trois langages différents (C, C++ et Java), et produire du code exécutable pour des cibles différentes. Au plus haut niveau d'abstraction, la chaîne se présente sous la forme de trois parties, appelées respectivement *frontend*, *middle-end* et *backend*, qui sont exécutées l'une après l'autre.

La partie *frontend* permet de lire le code source, de le parser et de le convertir en un Arbre Syntaxique Abstrait (AST). Pour cela, chaque programme est d'abord analysé par un *préprocesseur*, chargé de détecter les fautes de syntaxe. Le résultat est transmis à un *parseur* qui produit un AST. Chaque langage a un préprocesseur et un parseur spécifique. A ce stade, la signification d'un AST est différente selon le langage source. C'est pourquoi une étape d'uniformisation est effectuée par un composant appelé *généraliser*. Le résultat est un AST dans le langage GENERIC.

La partie *middle-end* est commune à tous les langages. Elle permet d'effectuer plusieurs transformations successives à partir d'un AST. Sans rentrer dans les détails, on considère que ces transformations successives correspondent respectivement à de l'*analyse sémantique*, à de la *génération de code intermédiaire* et à de l'*optimisation*. A la fin de l'optimisation, on obtient une représentation du programme en RTL (*Register Transfer Language*).

La partie *backend* permet la génération de code finale. GCC permet en particulier de produire de code objet pour adaptée à la cible d'exécution. Ainsi, GCC permet de produire du code objet pour ARM, SPARC, VAX, etc. Il y a un composant de génération de code par cible d'exécution.

Exercice 1

On veut appliquer la démarche de représentation des architectures vues en cours pour GCC.

- Identifier les acteurs et proposer un diagramme de contexte
- Identifier les parties principales du système et décrire la vue logique de haut niveau. Les acteurs doivent figurer dans ce diagramme.
- Pour chaque partie, décrire une vue logique détaillée représentant les composants et la façon dont ils sont liés.
- Identifier deux comportements du système, et représenter les sous la forme de diagramme de séquence.
- Donner une vue physique pour ce système.

Exercice 2

Ce système fonctionne comme une suite de transformations successives. C'est caractéristique d'un style architectural classique.

- Comment s'appelle ce style ?
- Quels sont les points forts et les faiblesses de ce style ?
- Pouvez-vous citer d'autres exemples de systèmes basés sur ce style ?

¹ http://icps.u-strasbg.fr/~bastoul/teaching/compilation/bastoul_introduction_compilation.pdf