Traduction – Chapitre 1 : Introduction

Parser : Analuseur syntaxez

# Qu’est-ce que la compilation ?

* Un logiciel qui traduit un programme qui est écrit dans un langage de haut niveau (C, Java, …) en instruction exécutable.

La compilation : l’ensemble des algorithmes qui construisent ce programme

Un compilateur est différent d’un interpréteur.

Compilateur :

* Texte source
* Compilo
* (Edition de lien)
* Programme cible 🡨 Exécutable , avec des données🡪 Résultat

Compilateur : programme de traduction automatique, le fichier qui résulte de la compilation est exécutable une fois pour toute.

Interpréteur : C’est aussi un programme.

Résultat

Erreurs

Interpréteur

Programme source

Données

Il n’y a aucun fichier cible.

L’interpréteur analyse une instruction après l’autre et l’exécute immédiatement. Il s’agit d’une boucle infinie tant que le fichier source n’est pas finie.

Il existe des langages entre deux comme Java : On les appelle les langage P-Code. Le fichier source est compilé sous forme de binaire compact.

BYDECODE INTRPRETATEUR

Compilateur

Javac

Sortie

f-class

f-java

Java

Interpréteur est aussi une machine virtuelle.

Dans un compilateur, on a de la grammaire, une notion d’arbre, la structure des données, génération dans un fichier des instructions assembleurs.

Schéma général d’un compilateur

ANTLR (descendant) / LEX-YACC (montant)

PHRASE

Programme source

TOKEN

AST + TDS

Parseur

(Grammaire)

Analyse sémantique

Analyse

Syntaxique

Analyse

Lexicale

Lexeur

AST

Arbre abstrait

Peut aussi avoir une TDS

Optimisation

(Tous ne le font pas)

Génération de code assembleur

OPTIMISATIONS

Génération de code intermédiaire

Code à 3 adresses

Cible

Légende couleur :

* Orange : Partie front
* Jaune : Partie cœur (Pas dans le projet)
* Vert : Partie Back

Lexeur et parseur sont souvent dans la même boite

L’analyse lexicale est un automate fini : notion de langage relationnelle.

L’analyseur syntaxique est un automate à états fini / pile : langage algébrique

TOKEN : unitée lexicale.

Arbre abstrait : arbre syntaxique auquel on a enlevé tout ce qui ne sert à rien.

TDS : Table des symboles.

# Quelques dates

* 1954 : FORTRAN (1er compilateur ~ 1950)
* 1959 : COBOL (Langage orienté objet, gestion)
* 1966 : SIMULA (Notion de classe et objet, dédié à la simulation)
* 1971 : PASCAL ( Langage pédagogique)
* 1972 : Langage C, PROLOG
* 1976 : ADA (domaine de l’aviation, orienté objet)
* 1982 : C++ (couche objet de C, C & SIMULA)
* 1986 : EIFFEL (langage objet)
* 1991 : JAVA
* HTML XML
* BDD
* Tex, Latex
* …

La liste n’est pas exhaustive, de nombreux sont créés régulièrement.

GCC : Crée en C mais compilé via un compilateur précédent.