Langages Enchâssés

Enseignants : Robert Strandh & David Sherman

Planning : 7x3h de cours/TD/TP + un examen

Horaires : mercredi 14h-17h

Lieu : Salle TD13 ENSEIRB

Support de cours

Robert Strandh et Irène Durand :
Traité de programmation en Common Lisp

Paul Graham :
On Lisp : Advanced Techniques for Common Lisp

Transparents

Petits langages

Ce sont des langages spécifiques à un domaine :

- requêtes dans une base de données,
- description d'un circuit électrique,
- description d'une imprimante,
- dessin de courbes,
- etc.

Exemples de petits langages

- termcap (description de terminaux),
- X11 ressources (interfaces graphiques),
- TCL (interfaces graphiques),
- sendmail (courrier électronique).

Implémentation traditionnelle

L'implémentation d'un petit langage se fait traditionnellement d'une des manières suivantes :

- préprocesseur,
- interpréteur,
- compilateur.

Difficultés :

- préprocesseur : ordre et compatibilité entre plusieurs pré-processeurs,
- interpréteur : performance + réimplémentation de fonctions de base,
- compilateur : difficulté d'implémentation + réimplémentation de fonctions de base.

Définition de Langage Enchâssé

C'est une technique du Génie Logiciel pour implémenter des petits langages.

L'idée est d'augmenter un langage existant (le langage hôte) avec les fonctionnalités nécessaires pour le petit langage.

Le petit langage sera donc enchâssé dans le langage hôte.

Abstraction

Il y a trois mécanismes d'abstraction nécessaires :

- abstraction de données (structures, classes),
- abstraction de contrôle (procédures, méthodes),
- abstraction syntaxique (formes spéciales).

Les langages habituels ne sont pas capables d'exprimer l'abstraction syntaxique.

Pourtant elle est nécessaire pour la construction d'un langage enchâssé.

Objectif du module

Présentation des techniques d'implémentation des langages enchâssés grâce aux macros Common Lisp.

Avantage :

- imbrication de plusieurs langages possible,
- compilation si le langage hôte est compilé,
- importation en masse des fonctions de base du langage hôte.

Contenu Cours

- présentation rapide de Common Lisp (comparaison avec Scheme),
- les macros Common Lisp,
- présentation d'un langage enchâssé.

Contenu TD/TP

- Emacs pour la programmation en Common Lisp,
- écriture et mise au point de macros.

Examen

Je ne sais pas encore.