

Programmation concurrente

SmartFolder

David Gonzalez - Claudio Sousa

23 décembre 2016

1 Introduction

Ce TP de deuxième année en programmation système consiste à implémenter un programme similaire au SmartFolder sur MacOSX.

Le SmartFolder sur MacOSX recherche sur le disque des fichiers correspondant à un/des critères et pour chacun des fichiers trouvés, le programme crée un lien dans un dossier spécifié.

1.1 Spécification fonctionnelle

Ce programme possède deux modes de fonctionnement.

1.1.1 Mode recherche

Le premier est le mode *recherche*. C'est le mode par défaut qui simule le SmartFolder sur MacOSX. Par ailleurs, le programme tourne indéfiniment tant qu'aucun signal d'arrêt n'est reçu.

Ce mode prend 3 paramètres :

- *<dir_name>* : chemin où stocker les liens ;
- *<search_path>* : chemin de recherche ;
- *[expression]* : critères de sélection.

<dir_name> et *<search_path>* sont de simples chemin vers des dossiers. Concernant *[expression]*, ceci correspond à une liste de critères dont l'interface est identique à celle de *find*.

Expression

...

1.1.2 Mode stop

Le deuxième est le mode *stop*. Il permet d'arrêter une recherche en cours.

Ce mode prend 2 paramètres :

- *-d* : paramètre qui permet de lancer ce mode ;
- *<dir_name>* : chemin traité par le SmartFolder à terminé.

2 Development

2.1 Architecture

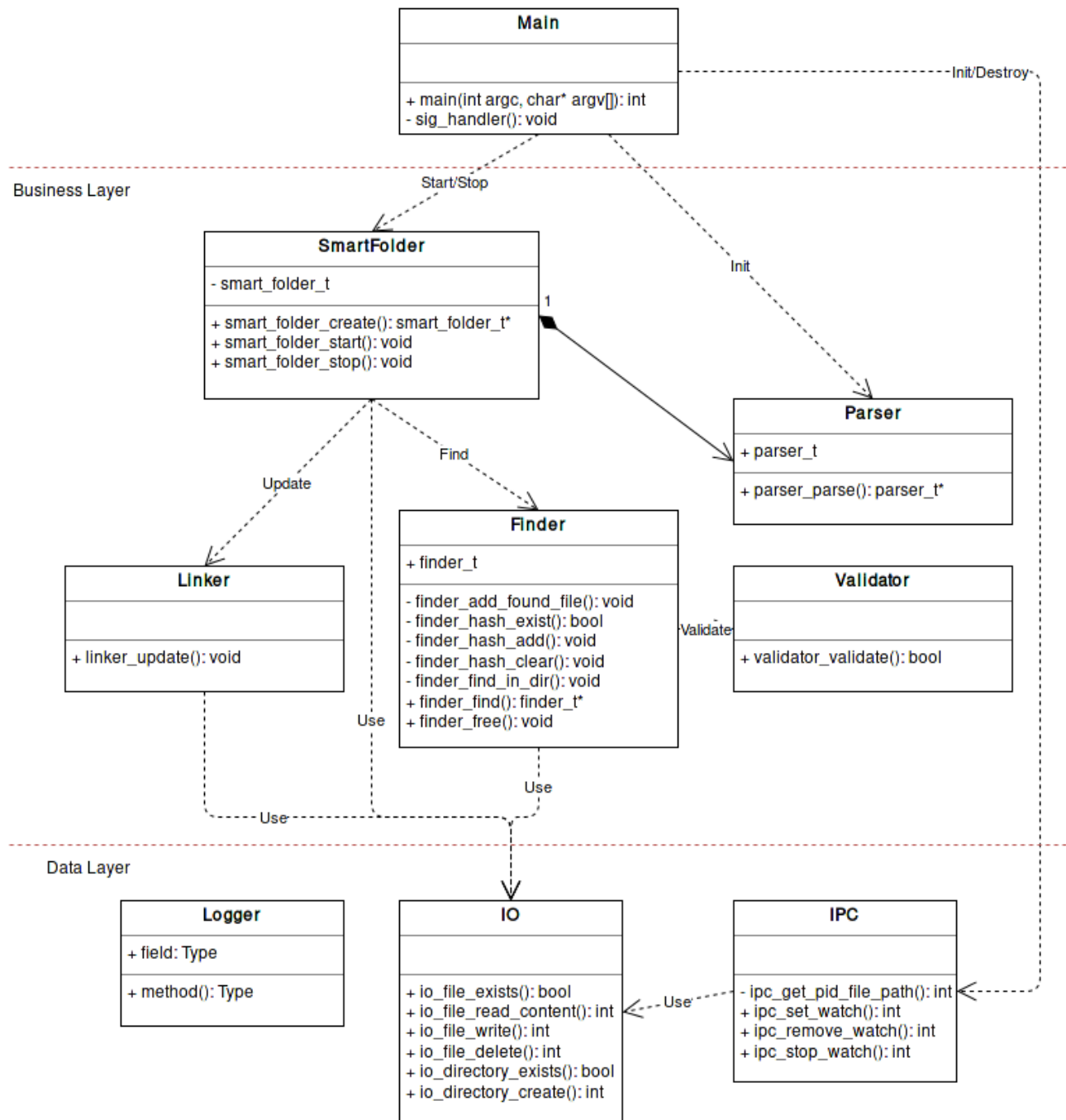


FIGURE 1 – Architecture du SmartFolder

2.1.1 Main

Le programme principal a pour rôle de vérifier les arguments et de sélectionner le bon mode de fonctionnement.

Dans le mode *recherche*, il a pour tâche de :

- mettre le processus en arrière-plan ;
- traiter l'expression ;
- poser le *pidfile* ;
- initialiser et lancer la recherche.

Lorsqu'il reçoit un signal d'arrêt, il libère la mémoire, efface le *pidfile* et ce termine.

Dans le mode *stop*, son seul rôle est de signaler l'arrêt (voir IPC).

2.1.2 SmartFolder

SmartFolder est le module principal qui va orchestrer la recherche.

Lorsque lancé, il va continuellement recherche les fichiers correspondant au critère, puis demander la mise à jour du répertoire de destination.

A noter qu'entre chaque recherche, il y a une pause de quelques secondes.

2.1.3 Parser

2.1.4 Validator

2.1.5 Finder

2.1.6 Linker

2.1.7 IPC

2.1.8 IO

Le but de ce module est d'offrir une interface simple aux appels systèmes.

2.1.9 Logger