

Design Patterns

OS Virtuel

Axel GENDILLARD Audoin DE CHANTÉRAC

Étudiant ingénieur
Groupe ESEO

14 décembre 2015

Sommaire

- 1 Incrément 1 : Singleton
- 2 Incrément 2 : Observer
- 3 Incrément 3 : Visiteur
- 4 Incrément 4 : Commande
- 5 Incrément 5 : Commande ++
- 6 Incrément 6 : Proxy

Incrément 1 : Singleton

Objectifs

- Créer un nouvel utilisateur à chaque validation ;
- Avoir un utilisateur unique par login ;
- Créer plusieurs terminaux par utilisateur ;
- Interdire la création d'utilisateur hors de la fenêtre.

Patron Singleton

- Garantir l'unicité de l'instance d'une classe ;
- Accéder globalement à l'instance.

Incrément 1 : Singleton - UML

fr.eseo.os

User

- instances : Map<String, User>

- User (login : String, password : String)

+ getInstance (login : String, password : String) : User

Incrément 2 : Observer

Objectifs

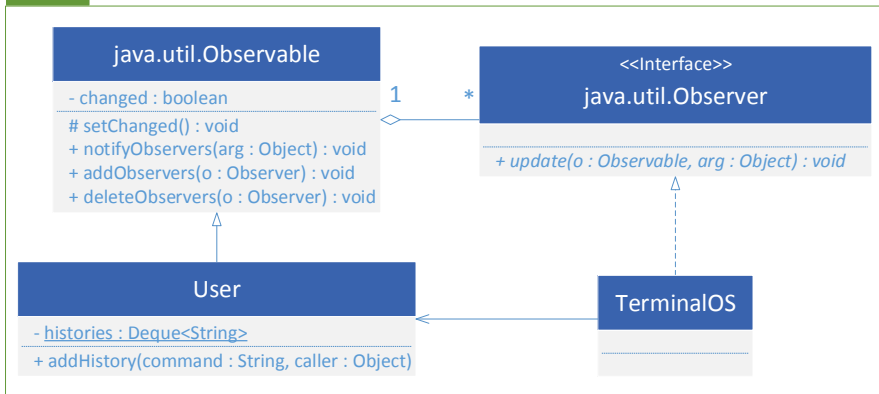
- Synchroniser les historiques des terminaux ;
- Mettre à jour tous les terminaux lors de l'entrée d'une commande.

Patron Observer

- Permet de mettre à jour les objets observés en fonction de l'objet observateur.

Incrément 2 : Observer - UML

fr.eseo.os



Incrément 3 : Visiteur

Objectifs

- Interpréter des commandes du terminal (ls et cat) pour chaque type de noeud ;
- Utiliser le patron Visitor.

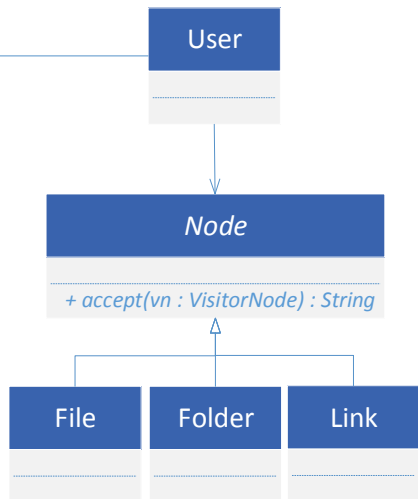
Patron Visiteur

- Sépare l'algorithme de la structure de données ;
- Chaque élément implémente une méthode d'acceptation de visiteur ;
- Chaque visiteur implémente une méthode *visit()* pour chaque élément (file, folder et link).

Incrément 3 : Visiteur - UML

fr.eseo.os

visitor



<<Interface>>
VisitorNode

+ visit(file : File) : String
+ visit(folder : Folder) : String
+ visit(link : Link) : String

VisitorLS

VisitorCAT

Incrément 4 : Commande

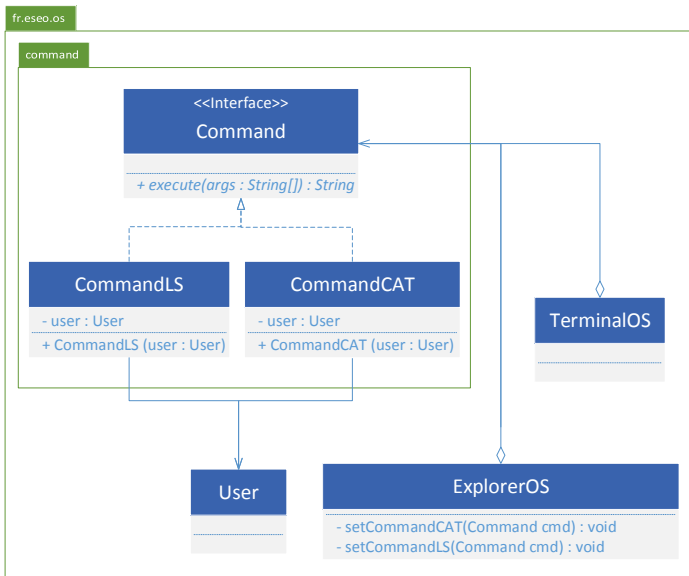
Objectifs

- Réutiliser les traitements interprétés par le terminal ;
- Exprimer de manière unique le traitement des commandes ls et cat.

Patron Commande

- Isole une requête ;
- Ces requêtes peuvent provenir de plusieurs émetteurs (*TerminalOS* et *ExplorerOS*) ;
- Ces émetteurs doivent produire la même requête ;
- Ces requêtes doivent pouvoir être annulées.

Incrément 4 : Commande - UML



Incrément 5 : Commande ++

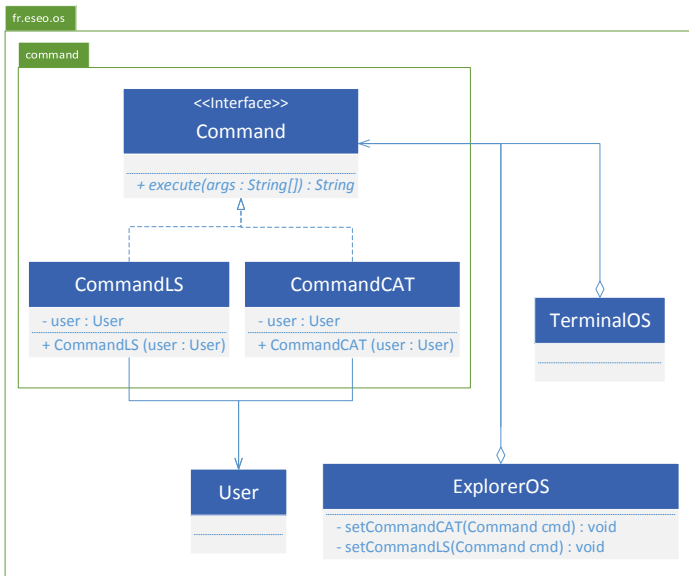
Objectifs

- RM : Supprimer un dossier, un fichier ou un lien ;
- MKDIR : Créer un dossier ;
- TOUCH : Créer un fichier ;
- LN : Créer un lien.

Patron Commande

- Isole une requête ;
- Ces requêtes peuvent provenir de plusieurs émetteurs (*TerminalOS* et *ExplorerOS*) ayant le même fonctionnement ;
- Ces requêtes doivent pouvoir être annulées.

Incrément 5 : Commande - UML



Incrément 6 : Proxy

Objectifs

- Lecture seule : Pas de modification des données des utilisateurs ls et cat sont autorisées ;
- Lecture/écriture : Les commandes mkdir, touch, ln et rm sont autorisées.

Patron Proxy

- Rôle d'aiguilleur en fonction des droits de l'utilisateur ;
- Couche d'abstraction entre le client et la commande.

Incrément 6 : Proxy - UML

