

## TP : Les lignes de produits en pratique

Dans ce TP vous allez utiliser un environnement (FeatureIDE) pour manipuler et développer des lignes de produits logiciels.

### Partie 1 : Installation et mise en route

Dans cette première partie vous allez utiliser

1. Télécharger l'outil FeatureIDE/Eclipse  
[http://www.iti.cs.uni-magdeburg.de/iti\\_db/research/featureide/#relateddownloads](http://www.iti.cs.uni-magdeburg.de/iti_db/research/featureide/#relateddownloads)

2. Lancer eclipse et changer la perspective vers FeatureIDE.

### Partie 1 : la ligne de produit « Notepad » en Java

Dans cette première partie de TP, vous allez manipuler une ligne de produits simple qui concerne l'éditeur classique Notepad. Cette ligne de produits est fournie comme un exemple dans l'outil FeatureIDE.

1. Importer le projet NotepadPL : menu *File* → *New* → *Example* → *FeatureIDE Examples*. Sélectionner le projet *Notepad-FH-Java*.
2. Visualiser et analyser le feature model proposé (le fichier : *model.xml*)
3. Créer des configurations de produits de votre choix: clique droit sur le répertoire *config* → *New* → *Configuration File*
4. Dériver vos produits en générant leur code Java : clique droit sur la configuration et choix de l'item **Set as a current configuration**. Le code généré est dans le répertoire *src*.
5. Analyser le code qui a été généré et comparer les différents produits et lancer la version Notepad générée (Il faut juste exécuter la class Java Notepad). N'oubliez pas de recopier aussi les images.
6. Tester la fonctionnalité de FeatureIDE permettant la génération de tous les produits possibles.

Clique droit sur le racine du projet *FeatureIDE* → *Build All Valid Configurations*

Nous souhaitons maintenant prendre en compte une nouvelle feature « *Tools* » pour *NotepadPL*

- a. Ajouter cette feature au feature modèle et fait en sorte pour qu'elle soit optionnelle.
- b. Ajouter aussi une contrainte de dépendance spécifiant que cette nouvelle feature exige la présence de feature *Cut*.
- c. Ajouter le code nécessaire pour implémenter cette feature. Pour simplifier l'implémentation, nous supposons que cette feature raffine la classe Notepad en ajoutant un menu « *Tools* » et un item « *Language* » dans ce nouveau menu.
- d. Régénérer du nouveau tous les produits possibles et vérifier que cette nouvelle feature est bien implémentée.

En plus de l'exemple de Notepad, FeatureID intègre plusieurs exemples de lignes de produits avec un support de plusieurs langages de programmation : C, C#, ... Vous pouvez jouer avec l'exemple Email-System-C qui implémente une famille de système de gestion de mail en C.

- Visualisez le feature modèle de cette LdP.
- générez le code C de quelques variantes et exécutez le code généré.

### **Partie 3 : Une ligne de produits pour la téléphonie Mobile**

Nous souhaitons implémenter une ligne de produits regroupant les logiciels d'une famille de téléphones mobiles. Pour simplifier l'exemple, nous considérons seulement la variabilité qui concerne les facteurs suivants :

#### **Les fonctionnalités supportées par le téléphone:**

- Passer des appels qui est une fonctionnalité obligatoire (heureusement☺). Un téléphone peut supporter les appels par voix ou par image ou les deux.
- Envoyer des messages : SMS ou MMS ou les deux.
- Les alarmes
- Les jeux. Cette fonctionnalité exige la présence de la plateforme Java.

#### **La configuration**

- OS : Android ou iOS
- La présence ou non de la plateforme Java

#### **MultiMedia**

- MP3
- MP4
- Camera

#### **Connectivité**

- WIFI
- Bluetooth
- USB.

Utilisez l'outil FeatureIDE et FeatureHouse pour implémenter cette ligne de produits.

(a) Créer un projet FeatureIDE /FeatureHouse

(a) Créer un feature model spécifiant la variabilité dans cette ligne de produits.

(b) Proposez une implémentation simple basée sur le raffinement de code proposé dans FeatureHouse. Vous pouvez juste proposer des classes **.java** simples avec des affichages sur la sortie standard. Pour les plus motivés, vous pouvez aller plus loin ☺

