

# Environnements de développement (intégrés)

## Développement de greffons

Patrick Labatut

labatut@di.ens.fr

<http://www.di.ens.fr/~labatut/>

Département d'informatique

École normale supérieure

Centre d'enseignement et de recherche en technologies de l'information et systèmes

École des ponts

# Plan

- 1 Développement de greffons
  - Plateforme Eclipse
  - Greffons
  - Exemples

# Plateforme Eclipse

Eclipse = plateforme + greffons

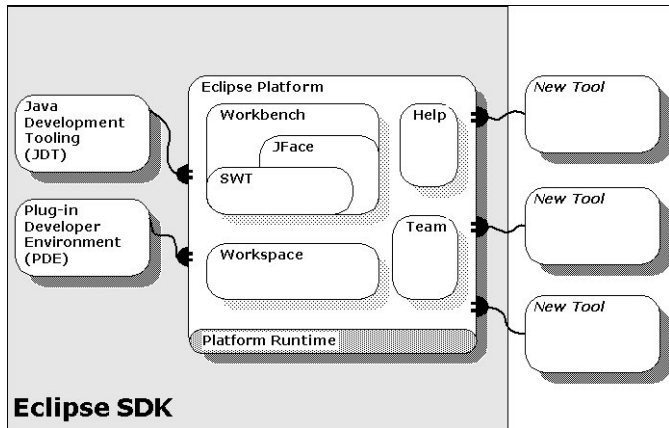
- plateforme :
  - un exécutif<sup>1</sup> indépendant du SE (JVM),
  - un ensemble basique de greffons<sup>2</sup> extensibles,
  - des mécanismes (API), règles et outils pour construire des greffons,
  - un moteur pour découvrir, charger et exécuter des greffons.
- greffon : la plus petite unité qui peut être développée et utilisée séparément ;
  - se connecte à des points précis de la plateforme,
  - remplit une tâche (pas forcément exécutable),
  - offre des points d'extension,
  - coexiste avec d'autres greffons,
  - instance : ensemble de greffons qui coopèrent pour offrir un EDI.

---

<sup>1</sup>runtime, en anglais

<sup>2</sup>plug-in, en anglais

# Architecture de la plateforme Eclipse



# Développement de greffons

Règles :

- 1 Tout est greffon,
- 2 Un greffon n'est chargé en mémoire qu'au moment où il devient nécessaire,
- 3 On ajoute des fonctionnalités mais on ne remplace pas l'existant,
- 4 Les greffons doivent respecter les interfaces,
- 5 Commencer toujours par récupérer la structure d'un greffon similaire,
- 6 Ne contribuer que lorsque vous avez à apporter quelque chose d'utile au niveau fonctionnel,
- 7 Autant que possible, laisser aux autres la possibilité de contribuer à votre greffon,
- 8 En tant que fournisseur d'un point d'extension, votre greffon doit se protéger des comportements erratiques des greffons.

# Constituants d'un greffon

- Interfaces configurables (*extension points*) : nom, type, paramètres [ *Search / Search... / Plug-in Search / Extension Points* ]
  - utilisables : *Declarations*,
  - implémentées : *References*.
- Sources internes : *src/*, *doc/*
  - code Java, archive(s) JAR,
  - ressources non-exécutables.
- Sources/exécutables externes : *JRE*, *Plug-in Dependencies*
  - bibliothèques externes,
  - plateforme d'exécution.
- Description : *plugin.xml*, *META-INF/MANIFEST.MF*
  - identification : nom, version, fournisseur,
  - interfaces (implémentées et introduites),
  - organisation interne.

# Environnement de développement de greffons

Le *Plug-in Development Environment* (PDE) permet de construire des greffons en utilisant les points d'extension disponibles.

- Choix du projet de greffon à développer :
  - Projet de greffon : standard,
  - Projet de « fragment » : ajout à un greffon,
  - Projet de *feature* : contient un ou plusieurs greffons,
  - Projet de mise à jour : mise en ligne des *features*.
- Assistant de greffons : squelettes de greffons classiques,
- Éditeurs et formulaires spécialisés pour l'écriture des fichiers nécessaires aux greffons (`plugin.xml`, `META-INF/MANIFEST.MF`, `build.properties`),
- Perspective spécialisée : *Plug-in Development*,
- Plateforme d'exécution : permet de lancer les greffons en environnement de test.

## En pratique. . .

L'expérience montre que :

- le développement d'après exemple est majoritaire,
- l'extension des niveaux bas de la plate-forme demandent plus d'attention,
- la majorité des extensions sont au niveau UI (`org.eclipse.ui`),
- le JDT/PDE sont indispensables,
- connaître le langage XML peut être utile. . .



## Greffon et points d'extension

- Le fichier XML `plugin.xml` décrit le greffon,
- Chaque point d'extension a son propre schéma (DTD), à consulter dans :  
*[ Extensions / Open extension point description ],*
- Quand on déclare un point d'extension, il faut indiquer un schéma (DTD) pour indiquer son utilisation.

Les CDATA correspondant aux noms de greffons ont une syntaxe « hiérarchique ».

# Exemples

Greffons d'UI d'Eclipse :

- Contribution d'une action : *Hello, World*
- Contribution d'un éditeur multi-pages : Éditeur de texte
- Contribution à un menu contextuel,
- Contribution d'une vue.