Intégration de l'interface graphique de Ptidej dans Eclipse

Driton Salihu (salihudr@iro.umontreal.ca)
Lulzim Laloshi (laloshil@iro.umontreal.ca)

Ptidej
Pattern Trace Identification, Detection, and Enhancement in Java

Département d'informatique et de recherche opérationnelle
Université de Montréal

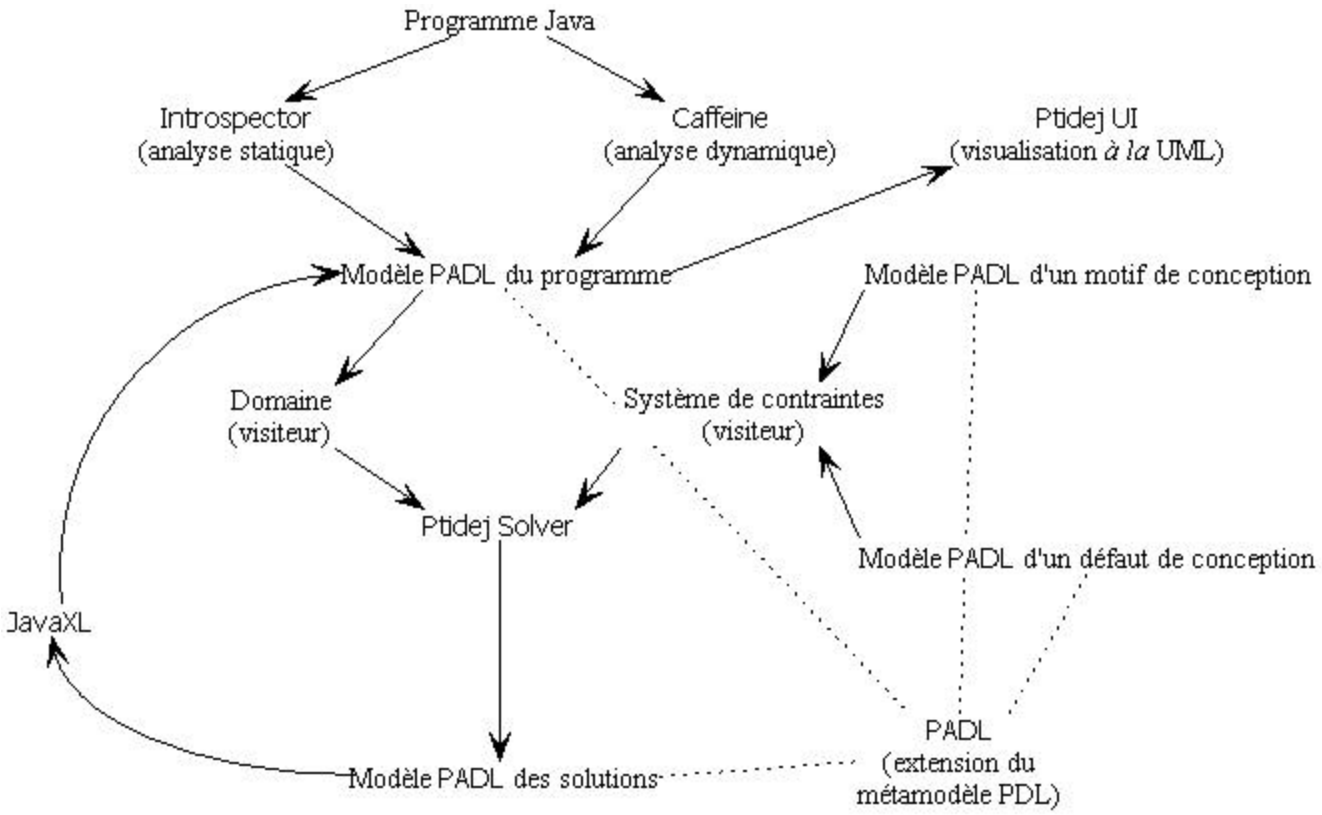
Introduction et description du problème

- Contexte
  - Les défauts des outils de rétro-conception industriels
  - L'implantation et le développement d'un ensemble de définitions dans l'outil Ptidej qui permet de
    - Analyser statiquement le code source d'un programme
    - Identifier des relations abstraites entre classes, telles que les relations d'association, d'agrégation et de composition
    - Appeler un solveur de contraintes pour identifier les micro-architectures similaires à la solution d'un patron de conception donné
  - L'intégration de cet outil avec l'environnement de développement Eclipse

Introduction et description du problème

- Travail demandé
  - Implanter l'interface graphique de Ptidej dans Eclipse
  - Développer Ptidej pour atteindre une intégration complète
  - Implanter toutes les fonctionnalités disponibles
  - Visualiser le modèle correspondant à l'aide d'un éditeur de diagrammes directement dans Eclipse
  - Afficher les résultats de l'appel d'un solveur de contraintes pour identifier les micro-architectures similaires à la solution d'un patron de conception donné
  - Faire de tests JUnit

Situation de Ptidej UI par rapport au projet Ptidej



Concepts théoriques

- Eclipse est un environnement de modélisation et de développement générique, ouvert et extensible
- Eclipse est générique puisqu'il permet le développement peu importe le langage utilisé sur de nombreux systèmes d'exploitation (Linux, Windows, Solaris)
- Eclipse est une plate-forme ouverte puisqu'elle est offerte sous licence *Common Public License*, c'est-à-dire que le code source est libre de redevance n'importe qui peut le redistribuer et les travaux dérivés sont autorisés
- Eclipse est également extensible puisque c'est un *framework* permettant de construire et d'intégrer des outils de développement de toute nature

Concepts théoriques

- Eclipse est composé de trois parties
  - Workbench
  - Plug-in Development Environment (PDE)
  - Standard Widget Toolkit (SWT)
- Eclipse offre aussi une interface pour travailler avec CVS

La conception de Ptidej et approche

- La structure des modules d’extension par défaut
  - Le dossier bin
  - Le dossier src
  - Les éléments du module d’extension à la racine
  - Le fichier .template
  - Le fichier build.properties
  - Le fichier plugin.xml (Manifeste)

La conception de Ptidej et approche

Voici un exemple de manifeste :

```
<?xml version="1.0" encoding="UTF-8"?>
<plugin
   id="idduplugin"
   name="nom du plugin"
   version="1.0.0"
   provider-name="Université de Montréal"
   class="test.testPlugin">
<runtime>
     <library name="boot.jar"/>
</runtime>
<requires>
<import plugin="org.eclipse.core.resources"/>
   <import plugin="org.eclipse.ui"/>
</requires>
```

La conception de Ptidej et approche

```
<extension point="org.eclipse.ui.editors">
<editor
      name="Exemple d'éditeur XML"
         icon="icons/sample.gif"
         extensions="e"
         ContributorClass="org.eclipse.ui.texteditor.
               BasicTextEditorActionContributor"
         class="test.editors.XMLEditor"
         id="test.editors.XMLEditor">
     </editor>
</extension>
</plugin>
```

La conception de Ptidej et approche

- Nous avons suivi une démarche particulière adaptée à la situation
- Nous n’avons pas fait de conception traditionnelle avec des diagrammes de classes UML
- Nous avons défini les classes à modifier et à implanter tout en expérimentant le développement des modules d’extensions
- Nous avons utilisé cette méthode déconseillée car le développement de modules d’extension nous étaient inconnus

Les fonctions implantées

1. Traitement des fichiers .class
   a. Les analyser dans le même éditeur de diagrammes
   b. Les analyser dans un nouvel éditeur de diagrammes
2. Traitement des fichiers .jar
3. Traitement des fichiers .java
4. Traitement des fichiers .ptidej
5. Traitement des répertoires qui contiennent des fichiers .class
6. Traitement des répertoires qui contiennent des fichiers .java
7. Traitement des paquetages Java

Les fonctions implantées

- Les fonctions implantées par Driton
  - Sous-fonctions *a* des fonctions 1 à 5
  - Sous fonctions *a* et *b* des fonctions 6 et 7
- Les fonctions implantées par Lulzim
  - Sous-fonctions *b* des fonctions 1 à 5

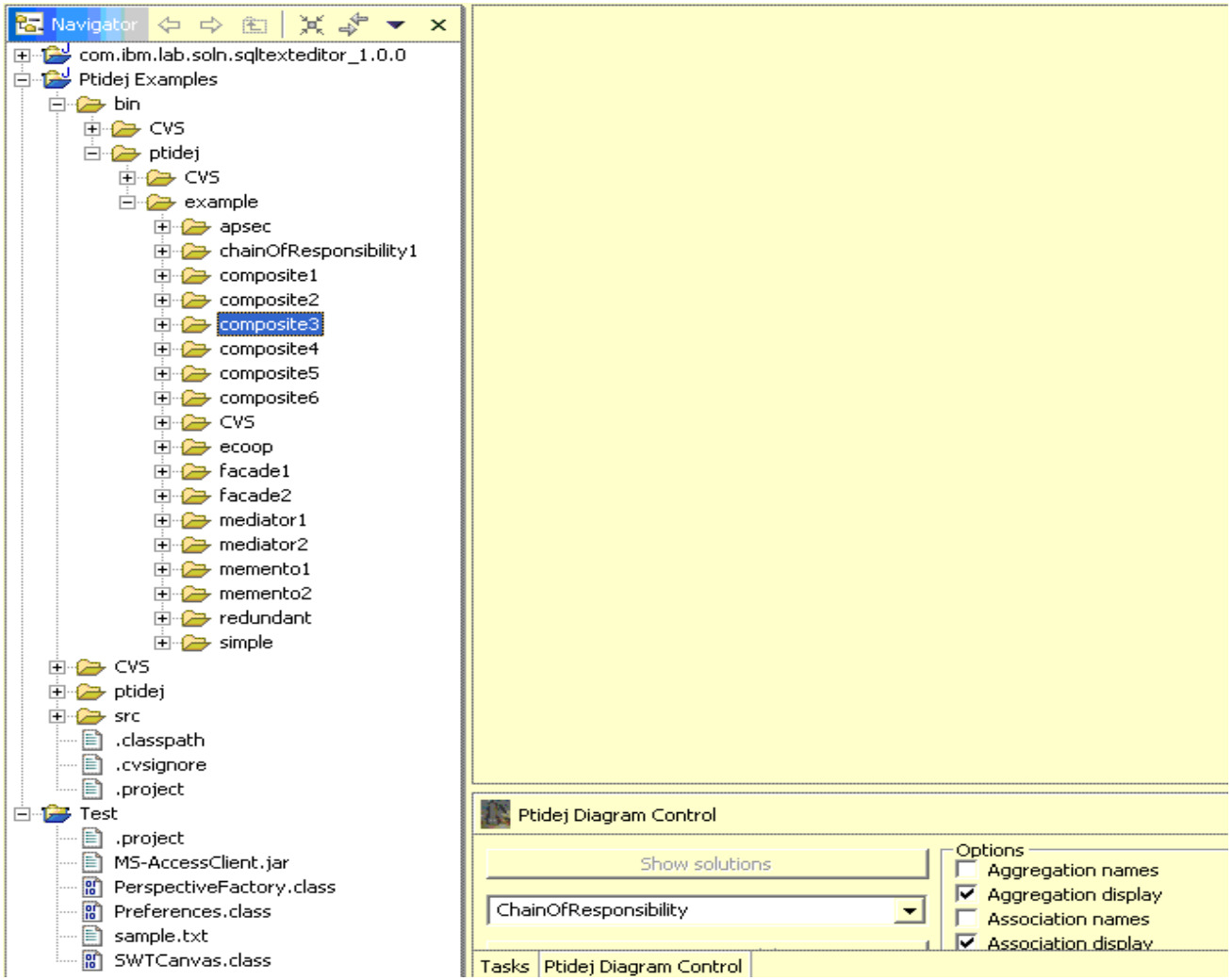
Les classes implantées (26)

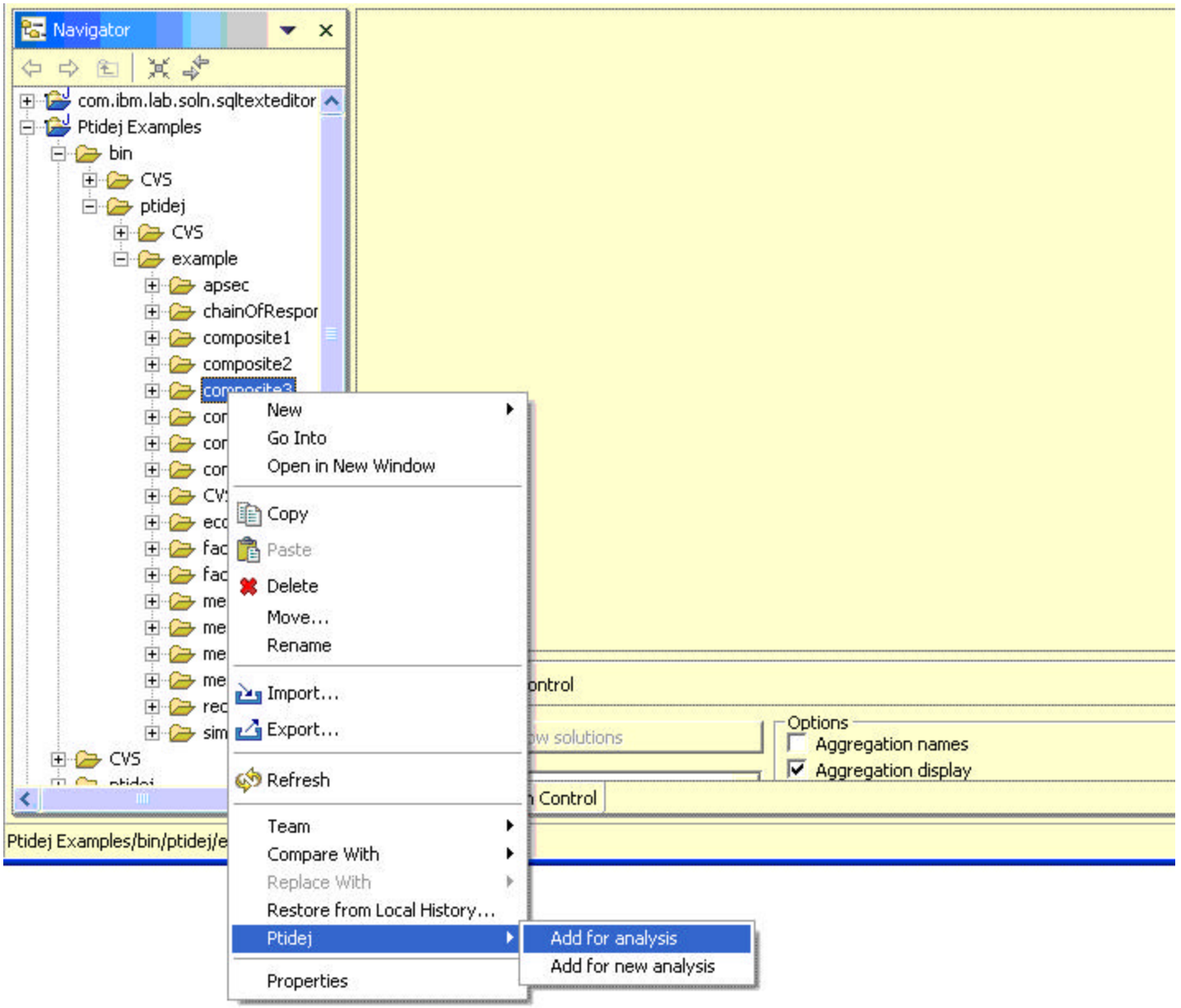
- AddFolderAction.java (D)
- AddJavaAction.java (D)
- AddJDTPackage.java (D)
- AddJDTPackageAction.java (D)
- AddNewFolderAction.java (L)
- AddNewJavaAction.java (L)
- AddNewJDTPackage.java (L)
- AddNewJDTPackageAction.java (L)
- PopActionClass.java (D)
- PopActionJava.java (D)
- PopActionJavaPackage.java (D)
- PopActionNewClass.java (L)
- PopActionNewJava.java (L)

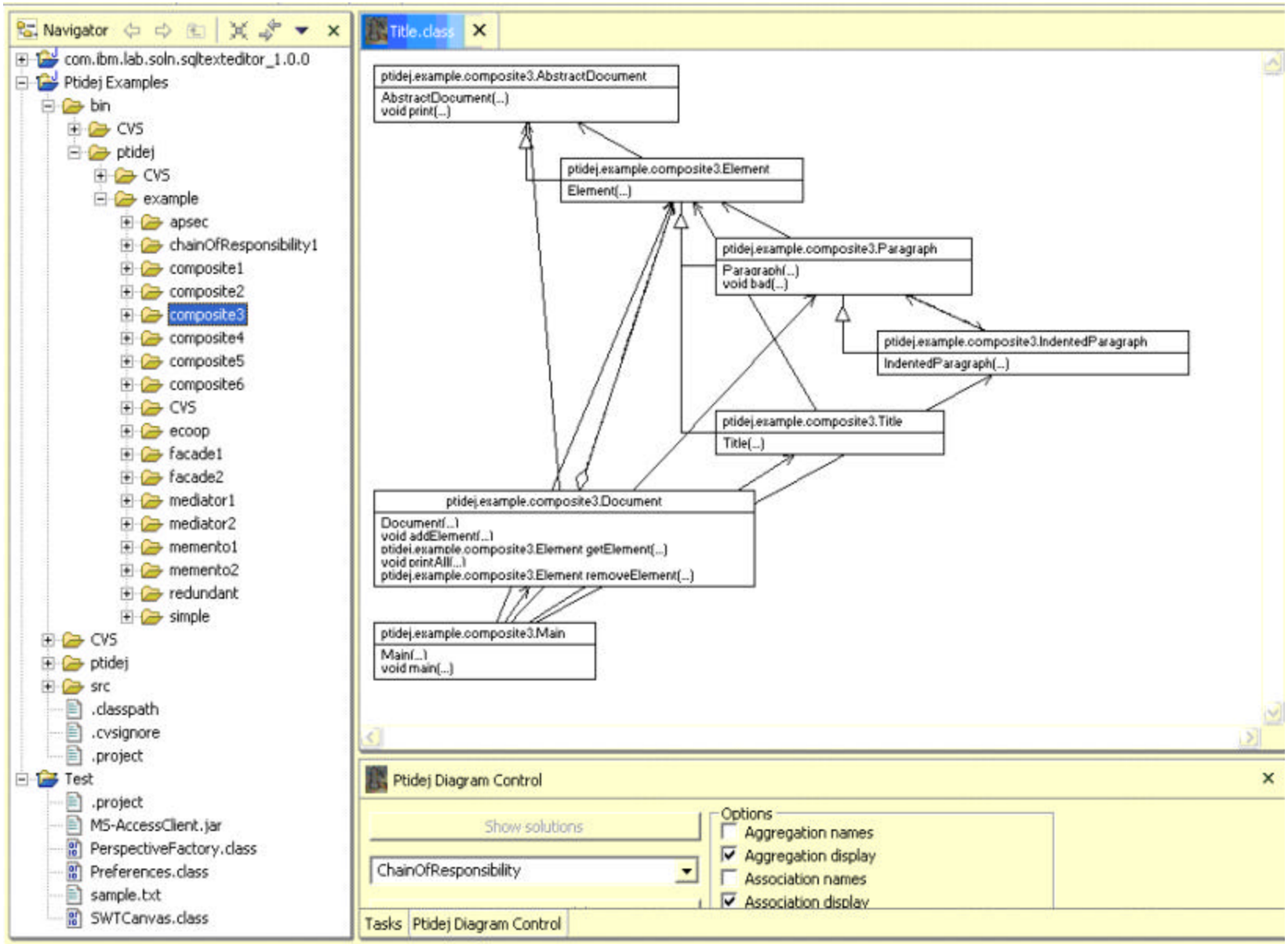
Les classes implantées

- PopActionNewJavaPackage.java (L)
- PopActionNewPackage.java (L)
- PopActionNewPtidej.java (L)
- PopActionPackage.java (D)
- PopActionPtidej.java (D)
- DiagramEditor.java (D)
- DiagramEditorContributor.java (D)
- PtidejDiagramEditor.java (L)
- PtidejDiagramEditorContributor.java (L)
- ClassFileFilter.java (D)
- ClassJarDirFilter.java (D)
- Explorer.java (D)
- Util.java (D)

Les résultats obtenus




Navigator
com.ibm.lab.soln.sqltexteditor
Ptidej Examples
bin
CVS
ptidej
CVS
example
apsec
chainOfRespor
composite1
composite2
New
Go Into
Open in New Window
Copy
Paste
Delete
Move...
Rename
Import...
Export...
Refresh
Team
Compare With
Replace With
Restore from Local History...
Ptidej
Properties
Add for analysis
Add for new analysis
Options
Aggregation names
Aggregation display
Control
Ptidej Examples/bin/ptidej/e


Navigator
com.ibm.lab.soln.sqltexteditor_1.0.0
Ptidej Examples
bin
CVS
ptidej
CVS
example
apsec
chainOfResponsibility1
composite1
composite2
composite3
composite4
composite5
composite6
CVS
ecoop
facade1
facade2
mediator1
mediator2
memento1
memento2
redundant
simple
CVS
ptidej
src
.classpath
.cvsignore
.project
Test
.project
MS-AccessClient.jar
PerspectiveFactory.class
Preferences.class
sample.txt
SWTCanvas.class
Title.class
ptidej.example.composite3.AbstractDocument
AbstractDocument(...)
void print(...)
ptidej.example.composite3.Element
Element(...)
ptidej.example.composite3.Paragraph
Paragraph(...)
void bad(...)
ptidej.example.composite3.IndentedParagraph
IndentedParagraph(...)
ptidej.example.composite3.Title
Title(...)
ptidej.example.composite3.Document
Document(...)
void addElement(...)
ptidej.example.composite3.Element getElement(...)
void printAll(...)
ptidej.example.composite3.Element removeElement(...)
ptidej.example.composite3.Main
Main(...)
void main(...)
Ptidej Diagram Control
Show solutions
ChainOfResponsibility
Options
Aggregation names
Aggregation display
Association names
Association display
Tasks
Ptidej Diagram Control

Le travail non-réalisé

- L’appel d’un solveur de contraintes pour identifier les micro-architectures similaires à la solution d’un patron de conception donné
- Le chargement dynamique des bibliothèques du plugin
- Les tests JUnit, mais on a effectué des tests d’intégration

Le travail bonus

- La compatibilité avec Eclipse 3.0.0
- Les métriques de qualité

Les obstacles et les difficultés rencontrées

- La difficulté de planifier les tâches à accomplir
- L’établissement d’un échéancier réaliste
- La recherche des documents pertinents sur les modules d’extensions
- Le chargement dynamique des bibliothèques .jar
- Le manque de temps

Les bénéfices apportés

- C’est la première fois qu’on participe dans un projet d’une telle envergure et qu’on ait accès au code source d’un outil aussi avancé et extensible que Eclipse
- Cela va nous aider dans les futurs développements envisagés
- Une analyse et une compréhension plus efficace du code source
- Nous avons acquis de nouvelles connaissances et de meilleures habitudes de programmation

Les bénéfices apportés

- Nous avons maintenant
  - Une meilleure approche de la résolution des problèmes divers concernant les applications aussi grandes
  - Une meilleure planification et organisation
  - Un meilleur contrôle du développement d'un projet
  - Une expérience et une préparation pour les travaux futurs dans le développement et la maintenance des logiciels

Les améliorations

- Intégrer plus efficacement l'interface graphique de Ptidej dans la version 3 de Eclipse
- Réfactoriser et simplifier le code source
- Améliorer certains algorithmes de recherche de fichiers .class
- Internationaliser Ptidej

Compatibilité

- Incompatibilité avec Eclipse 3 (niveau développement)
  - Le «binary runtime compatiblilty »
    - permet aux modules d'extension existants qui utilisent les méthodes supprimées *openEditor* ou *openSystemEditor* de continuer leur fonctionnement normal malgré les changements dans l'API.
  - La résolution de cette incompatibilité
    - les méthodes supprimées ont été redéfinies dans la classe IDE comme des méthodes statiques avec une petite différence que nous pouvons les utiliser directement pour ouvrir un éditeur

Compatibilité

Exemple d'utilisation

Eclipse 2.1.2

```
IEditorPart editorPart = activePage.openEditor(
                                fileToOpen,    // IFile
                                editorID,      // String
                                true);         // Boolean
```

Eclipse 3.0.0

```
IEditorPart editorPart = IDE.openEditor(
                                activePage,    // IWorkbenchPage
                                fileToOpen,    // IFile
                                editorID,      // String
                                true);         // Boolean
```

Remerciements

Nous souhaitons remercier Yann-Gaël Guéhéneuc qui nous a permis de travailler sur ce projet et nous a accordé une grande autonomie

Bibliographie et références

- Ouvrages
  - « The Java Developers Guide to Eclipse », un groupe d'auteurs, Addison-Wesley
  - Exemples sur le CD-ROM qui accompagne ce livre

- Documentation sur l'Internet
  - http://www.eclipse.org

    Site officiel d'Eclipse
  - http://www.beyondcode.org/articles/eclipse-plugins.html

    Documentation sur les modules d'extensions Eclipse
  - http://drdb.fsa.ulaval.ca/sujets/boisvertAlain/html/bk03.html

    Preséntation de Eclipse
  - http://www.eclipse-plugins.info

    Site officiel regroupant des modules d'extensions Eclipse

Des questions ?

- ?