

Tutorial 2

OpenOME distilled

On the Requirements, Design and
Implementation of the legacy tool

<http://www.cs.toronto.edu/~yijun/OpenOME.html>

<http://sourceforge.net/projects/openome>

Java - Eclipse Platform

t1 Protégé 2.1.2 (file: c:\workspaces\OpenOME\OpenOME\projects\t1.pprj, Standard Text Files)

Project Edit Window Help

Classes Slots Forms Instances Queries OME

Classes

src

- com.keypoint
- com.sixlegs.image.png
- edu.stanford.smi.protege.queries
- edu.stanford.smi.protege.queries
- edu.stanford.smi.protege.util
- edu.stanford.smi.protege.widget
- edu.stanford.smi.protege.widget
- edu.stanford.smi.protege.widget
- edu.stanford.smi.protege.widget
- edu.stanford.smi.protege.widget
- edu.stanford.smi.protege.widget
- edu.stanford.smi.protege.widget
- edu.stanford.smi.protege.widget
- edu.stanford.smi.protege.widget
- edu.toronto.cs.ome
- edu.toronto.cs.ome.OME
 - AbstractPluginMethod.java
 - BMPWriter.java
 - Canvas.java
 - CheckboxChooser.java
 - Choice.java
 - ContinueCancelDialog.java
 - CreateElementMethod.java
 - CreateLinkMethod.java
 - D.java
 - GraphicView.java
 - GraphicViewCanvas.java
 - GraphicViewElement.java
 - GraphicViewFrame.java
 - GraphicViewLink.java
 - GraphicViewObject.java
 - ImageTable.java
 - InputDialog.java
 - JMenuPlus.java
 - JPopupMenuPlus.java
 - JTelosUtil.java
 - KBManager.java
 - MenuMethod.java
 - ModelAttribute.java
 - ModelElement.java
 - ModelLink.java
 - ModelManager.java
 - ModelObject.java
 - ModelValueAttribute.java
 - NewModelInfo.java
 - ObjectMethod.java
 - OMEDefaultPlugin.java

OMEObject

OMELink

- NFRLink
 - NFRSoftgoalWeaving
 - NFRSoftgoalCorrelation
 - NFRSomeNegativeCorrelation
 - NFRSomePositiveCorrelation
 - NFRUnknownCorrelation
 - NFRHelpCorrelation (2)
 - NFRHurtCorrelation (1)
 - NFRMakeCorrelation
 - NFRBreakCorrelation
 - NFRSoftgoalContribution
 - SomeNegativeContribution
 - SomePositiveContribution
 - UnknownContribution
 - HelpContribution
 - HurtContribution
 - EqualContribution
 - AndContribution (3)
 - OrContribution
 - MakeContribution (1)
 - BreakContribution
- OMEElement
 - OMEGrowableElement
 - NFRElement
 - Task (3)
 - Goal
 - Aspect
 - NFROperationalization (1)
 - NFRClaim
 - NFRSoftgoal (2)
- NFRSoftgoalPriority
- NFRConflictSoftgoalLabel
- NFRSatisfiedSoftgoalLabel

c:\workspaces\OpenOME\OpenOME\projects\Telos\graph\streamline.tel

File Edit View Window Help NFR Options

Save NFRSoftgoal Operationalization Claim Goal Task Aspect Contribution Link Correlation Link Weaving Link

```

run protege Windows [Program] c:\workspaces\OpenOME\OpenOME\run_protege.bat

kr.re.etri.ezowl
org.algernon
uk.ac.ecs.iam.akt.tgviztab
uk.ac.man.cs.mig.coode.owlviz
uk.ac.man.cs.mig.coode.owlwizard

Telos Parser Version 0.1: Telos program parsed successfully from
c:\workspaces\OpenOME\OpenOME\projects\Telos\graph\streamline.tel.
Telos model loaded from kb.
Load time for file:c:\workspaces\OpenOME\OpenOME\projects\t1.pprj = 2 sec (project) + 5 sec (ui)

```

OpenOME

start Total Commander ... Microsoft PowerPoi... Java - Eclipse Platf... t1 Protégé 2.1.2 ... EN 10:27 AM

Contents

1. Historical retrospective
2. Requirements and features
3. Design and patterns
4. Implementation and issues
5. Relation to the course project

1. Historical retrospective

- OME stands for Organizational Modeling Environment. It was part of the *Tropos* project to support goal-oriented and agent-oriented requirements engineering methodologies (at least 5 years development involving 10 man-year efforts)
- OME has been widely used by more than 130 users (across the globe)
- Every OME user must sign an agreement with *Techne* because the Knowledge Base was a module protected by the license
- To enlarge the user-base, we decide to open-source it last year ... OpenOME

2. Requirements and features

- Is a Graph editor
 - A graph has elements and links in various form, basic operations include: Load, Save, Insert, Delete, Select, Cut, Paste, Hide, Highlight, Labelling, etc.
 - Multiple views (under development)
- Supports requirements engineering
 - Goal-oriented: goal reasoning through label propagation (NFR)
 - Agent-oriented: group goals into agents rationale (i*)
- Interchanges with other graph editors
 - Semantic Web queries: Protégé (OWL)
 - Layout algorithms: AT&T Graphviz (DOT)
 - Scalability: Microsoft Visio (XSLT)under development
 - Model-driven development: Rational Rose (EMF/XMI)
.....under planning

3. Design: MVC

- Model-View-Controller design pattern
 1. *Model*: The Telos Knowledge Base representation and OME models
 2. *View*: Graph presentation
 3. *Controller*: commands in menu, toolbar and various methods

3.1 Model

- *ModelManager*
- *Telos**: requirements as knowledge
- *Telos* as metamodeling language
 - Level: Token, SimpleClass, MetaClass, MetaMetaClass, Builtin classes ...
 - *.tel: L X IN {Y}* ISA {Z}* WITH {attribute,U:V}*
 - ER, NFR(vgraph), ISTAR, GRL
 - From *jtelos.dll* to *TelosParser*
 - Export *Telos* model to other models: *JTelosUtil.java*
OTelos (ConceptBase), *Protégé (KnowledgeBase)*
TODO: Eclipse Modeling Framework (XMI)

3.2 View

- *GraphicView* is a collection of *GVElement*, *GVLlinks*, maps the tokens in Telos model into geometric shapes in the presentation
GVE\$Record, *GVL\$Record* ...encodes the location of the shapes, states of the presentation, etc. They are saved as *SerializedViewObjects*
- *GVElement*, *GVLlink*
Visitor pattern and Decorator pattern
- They are extended by the OME plugins

3.3 Controller

- *OMETab*: run it as standalone Java application, or as a plugin for Protégé or Eclipse (under development)
- *GraphViewFrame* and *OMEDefaultPlugin*: control the menu, toolbar and methods
A method is interpreted as commands
 - No argument command: Layout
 - With one argument: Insert, ...
 - With two arguments: CreateLink, Move ...
 - With multiple arguments: Select, ...
- They are extensible using the OME plugins

4. Implementation issues

- OME: 90% Java + 10% C/C++
- Recently
 - OpenOME: 99% Java + 1% scripts
 - Use the Eclipse IDE
 - CVS, bug report: host at SourceForge
 - 3 research developers + some contribution from you J

5. Relation to your project

- It is the graph editor client of the choice for your OmniGraphEditor project. You may choose additional open-source graph editor as bonus point (such as Dia, Visio, Eclipse GEF etc.), but that is not recommended because of the large efforts
- Opportunities:
 - You may add junit test cases to the code base to reveal bugs (publish it to the bug tracking system) and fix them (+5%)
 - You may apply design patterns, refactoring techniques on this legacy code base, showing as an improved complexity metrics (+2.5%)
 - You may tune the performance of the system to speed up the display, load/save for scalable graphs (+2.5%)
- Don't forget your major project task (up to 100%!)
 - To study the editor methods in the OpenOME and adapt them to the OmniGraphEditor web service.