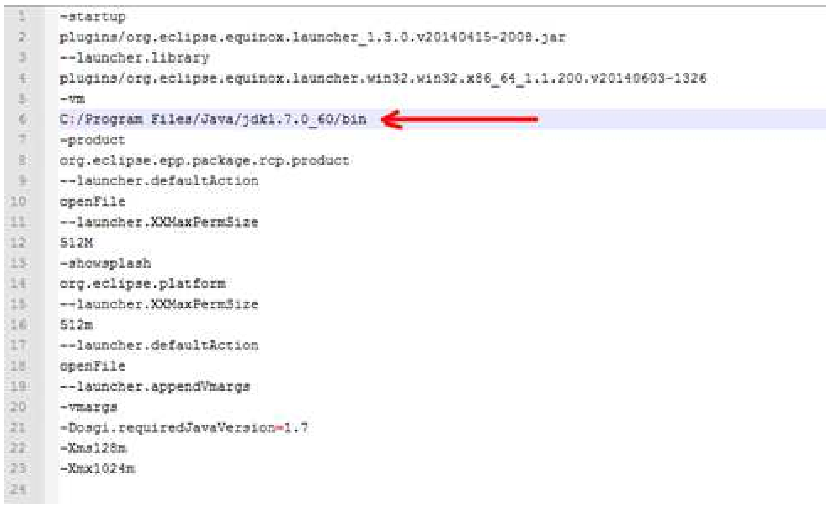
"*SoPeTraceAnalyzer*": a tool for Exploiting Traceability Uncertainty between Software Architectural Models and Performance Analysis Results

Installation of the tool

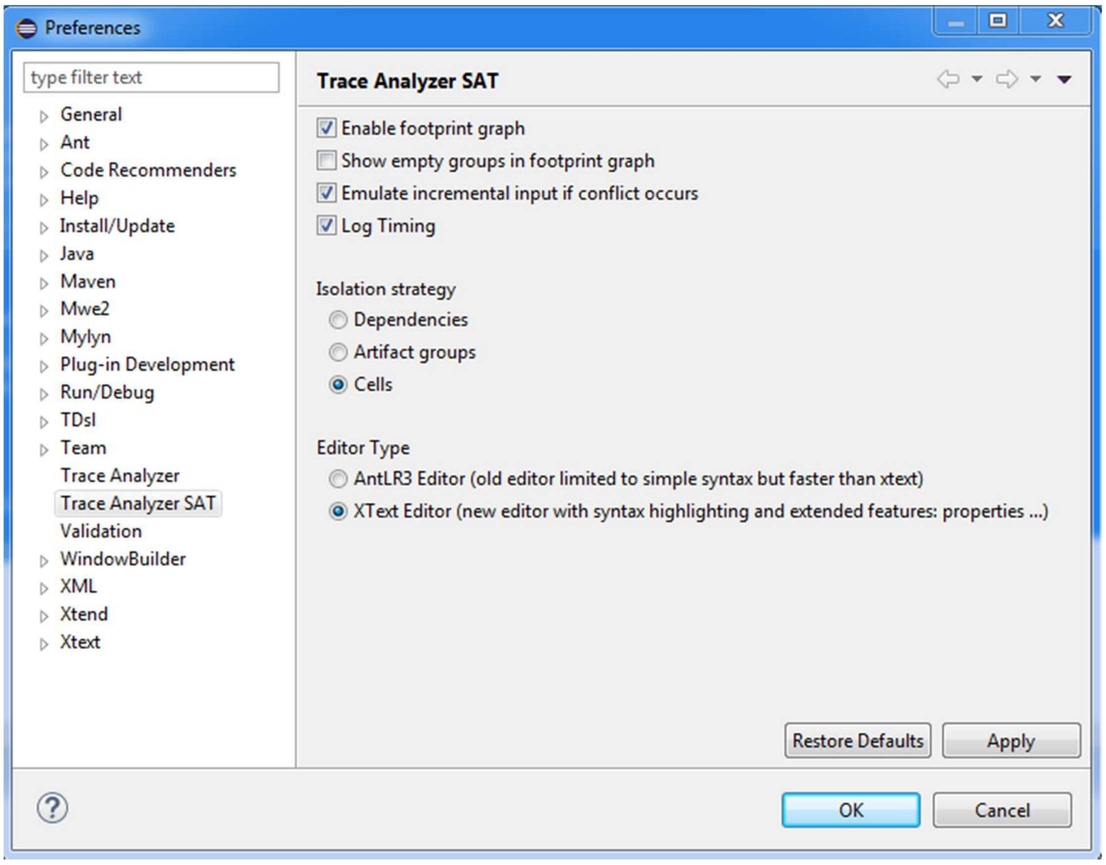
1- Extract the zip file.

2- Open the file 'eclipse.ini' and point the –vm path to your Java JDK:



```
-startup
plugins/org.eclipse.equinox.launcher_1.3.0.v20140415-2008.jar
--launcher.library
plugins/org.eclipse.equinox.launcher.win32.win32.x86_64_1.1.200.v20140603-1326
-vm
C:/Program Files/Java/jdk1.7.0_60/bin
-product
org.eclipse.epp.package.rcp.product
--launcher.defaultAction
openFile
--launcher.XXMaxPermSize
512M
-showsplash
org.eclipse.platform
--launcher.XXMaxPermSize
512m
--launcher.defaultAction
openFile
--launcher.appendVmargs
-vmargs
-Dosgi.requiredJavaVersion=1.7
-Xms128m
-Xmx1024m
```

3- Start Eclipse and make sure the configuration of "Trace Analyzer SAT" is as follow:



Trace Analyzer SAT

- [✓] Enable footprint graph
- [ ] Show empty groups in footprint graph
- [✓] Emulate incremental input if conflict occurs
- [✓] Log Timing

Isolation strategy

- ○ Dependencies
- ○ Artifact groups
- ● Cells

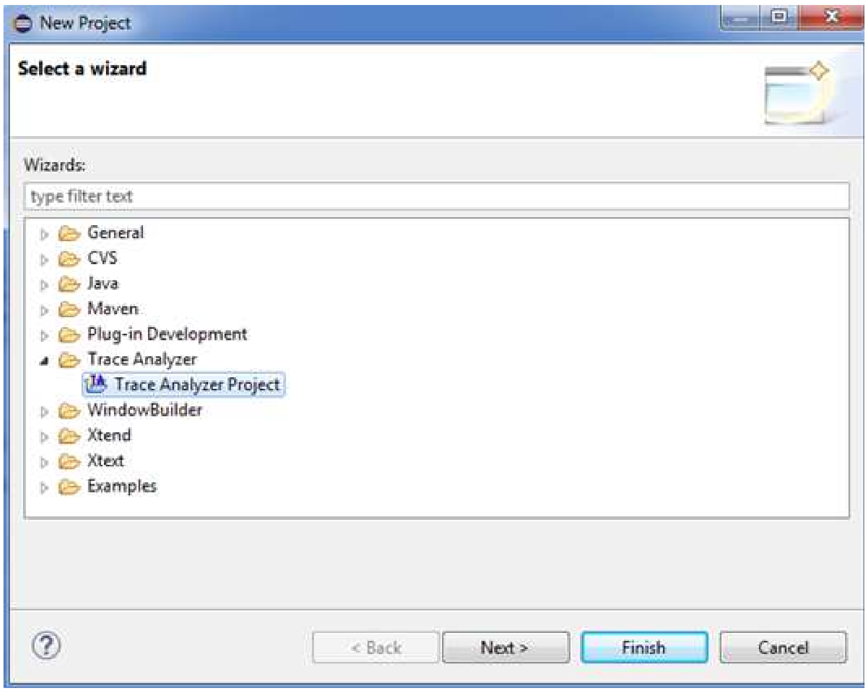
Editor Type

- ○ AntLR3 Editor (old editor limited to simple syntax but faster than xtext)
- ● XText Editor (new editor with syntax highlighting and extended features: properties ...)

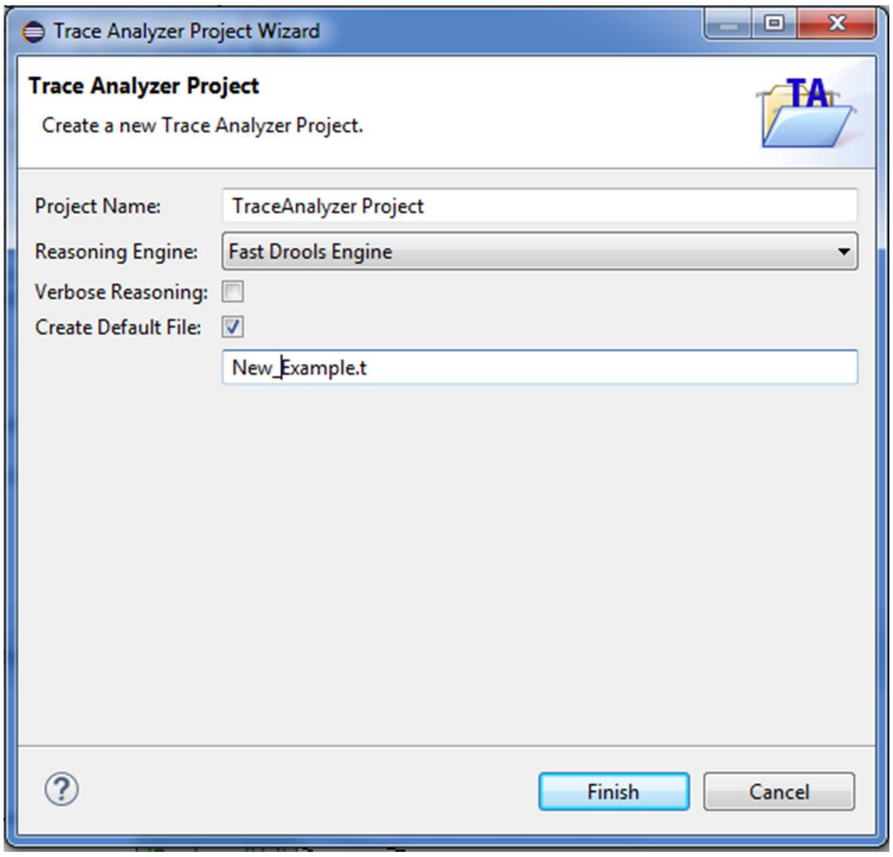
The workspace delivered with the zip file already contains a project with a case study example, i.e. the E-Health System (EHS). You can run it directly or create a new one.

Create New Example (Optional)

1. Create a new TraceAnalyzer Project



2. Define the Project Name and the Default File.


Trace Analyzer Project Wizard
Trace Analyzer Project
Create a new Trace Analyzer Project.
Finish
Cancel

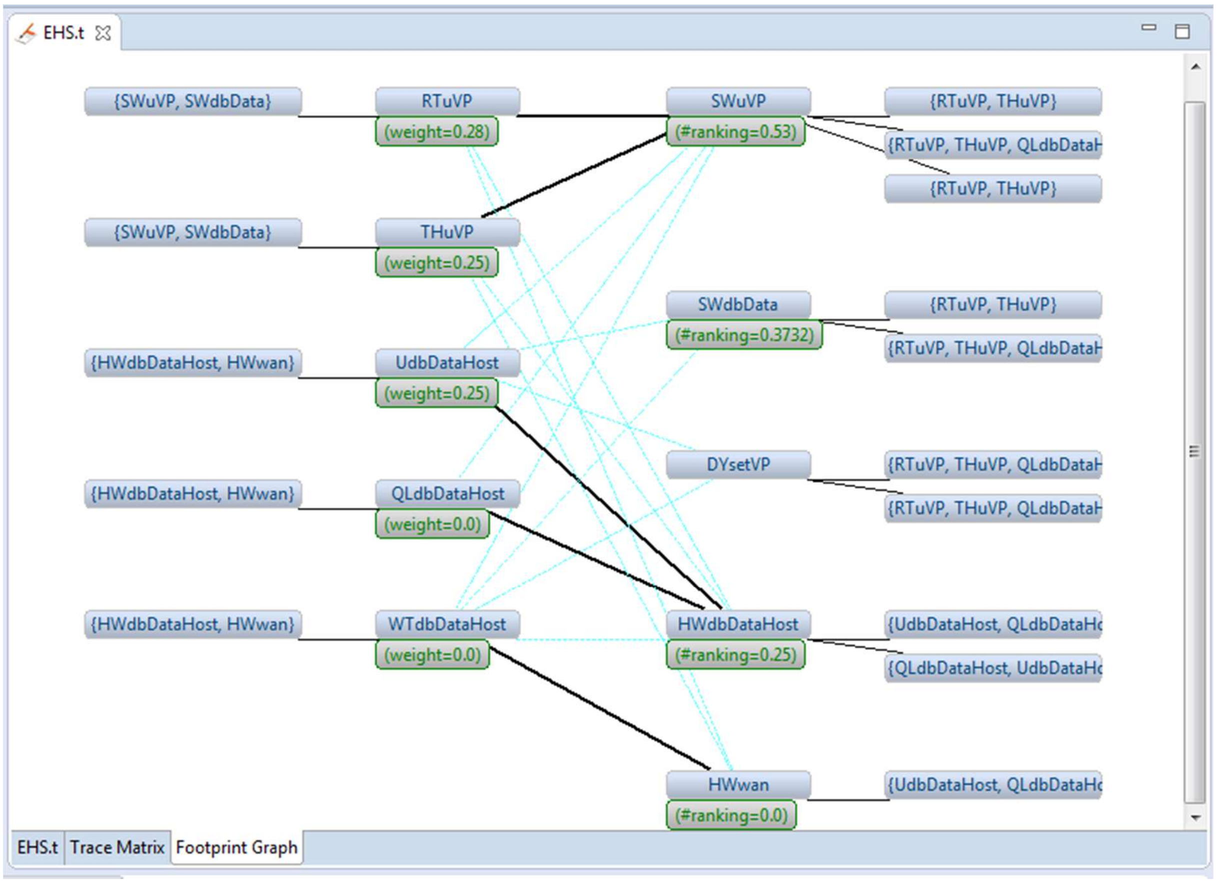
Project Name: TraceAnalyzer Project
Reasoning Engine: Fast Drools Engine
Verbose Reasoning: ☐
Create Default File: ☑
New_Example.t

Most important is that the default file should have the extension ".t"

3. The newly created file will contain the EHS example as before. You can change the data as you want:
    a. Add/Remove Result Elements (RE) or software Model Elements (ME).
    b. Add/Remove/Change the weights.
    c. Add/Remove/Change the guilt values.

Running the tool with the E-Health System (EHS) example

By clicking on the file EHS.t (or the default file that you created in the new Project wizard), a multi tabbed editor will be opened. In the "Footprint Graph" tab you will find the Footprint graph, as discussed in the paper.



What is new in *SoPeTraceAnalyzer* with respect to TraceAnalyzer[Ghabi-Egyed-Wicsa-2012]:

1. The possibility to **define properties** in the input:

```
define visible property weight;
define property guilt;
```

2. The possibility to **provide weight values** for each Result Element (RE):

```
set number weight : RTuVP = 0.28;
set number weight : THuVP = 0.25;
set number weight : UdbDataHost = 0.25;
set number weight : QLdbDataHost = 0.0;
set number weight : WTdbDataHost = 0.0;
```

3. The possibility to **provide guilt values** estimating how much a Model Element (ME) is “responsible” for the corresponding RE to which it is connected. Numbers reported in the following comes from the application of our guilt-based approach [Trubiani-etAl-JSS-2014].

```
set number guilt : RTuVP -> SWdbData = 0.44;
set number guilt : THuVP -> SWdbData = 1.0;
```

4. The possibility to **deduce weight values** for each ME node. Such values are computed by the tool and shown in the Footprint Graph along with the visible properties.

References

[Ghabi-Egyed-Wicsa-2012] Achraf Ghabi and Alexander Egyed, “Exploiting Traceability Uncertainty between Architectural Models and Code”, in the Proceedings of Joint Working Conference on Software Architecture and European Conference on Software Architecture (WICSA/ECSA), 2012.

[Trubiani-etAl-JSS-2014] Catia Trubiani and Anne Koziolek and Vittorio Cortellessa and Ralf Reussner, “Guilt-based handling of software performance antipatterns in palladio architectural models”, in the Journal of Systems and Software (JSS), volume 95, pp.141-165, 2014.