

Open Provenance Model Tutorial

Session 8: OPM Toolbox

Luc Moreau

L.Moreau@ecs.soton.ac.uk

University of Southampton

Session 8: Aims



In this session, you will learn about:

- How to install the OPM Toolbox
- How to write a Java program that generates an OPM graph
- How to serialize an OPM graph to XML or RDF
- How to convert and pretty print OPM graphs from the command line
- How to contribute to the OPM toolbox development

Session 8: Contents



- Installation
- Command line commands
- Java programming
- Open source development

OPM TOOLBOX

OPM Toolbox



- A set of utilities to manipulate OPM graphs in Java
- Binary distribution: toolbox-1.1.X-release.zip
- Available from
<http://openprovenance.org/java/maven-releases/org/openprovenance/toolbox/>
- Unzip it!

Binary Release Contents



```
-rw-r--r--@ 1 lavm staff 1083 Sep 15 08:26 README-licenses.txt
-rw-r--r--@ 1 lavm staff  944 Sep 15 08:26 README-opm2dot.txt
-rw-r--r--@ 1 lavm staff 1070 Sep 15 08:26 README-opmconvert.txt
-rw-r--r--@ 1 lavm staff  640 Sep 15 08:26 README-opmrdf2xml.txt
-rw-r--r--@ 1 lavm staff  453 Sep 15 08:26 README-opmxml-validate.txt
-rw-r--r--@ 1 lavm staff  632 Sep 15 08:26 README-opmxml2rdf.txt
drwxr-xr-x 16 lavm staff  544 Sep 15 08:26 bin
drwxr-xr-x  9 lavm staff  306 Sep 15 08:26 examples
-rw-r--r--@ 1 lavm staff 1095 Sep 15 08:17 license.txt
drwxr-xr-x 22 lavm staff  748 Sep 15 08:26 repo
```

License



Copyright (©) 2008, 2009, 2010 University of Southampton

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

COMMAND LINE EXECUTABLES

Converting to pdf



- Graphviz must be installed, and dot executable in PATH
- Usage:

```
opm2dot opmFile.xml out.dot out.pdf [configuration.xml]
```

- Example:

```
cd opmToolbox  
cat examples/bad-cake.xml  
bin/opm2dot examples/bad-cake.xml cake.dot cake.pdf
```

Converting to RDF



- Usage:

```
opmconvert -xml2rdf fileIn fileOut NS [yes]
```

```
opmconvert -xml2n3 fileIn fileOut NS [yes]
```

```
opmconvert -rdf2xml fileIn fileOut NS [gid]
```

- Example:

```
cd opmToolbox
```

```
cat examples/bad-cake.xml
```

```
bin/opmconvert -xml2n3 examples/bad-cake.xml cake.n3 http://ex.com/
```

WRITING A JAVA PROGRAM THAT CREATES AN OPM GRAPH

A simple maven project



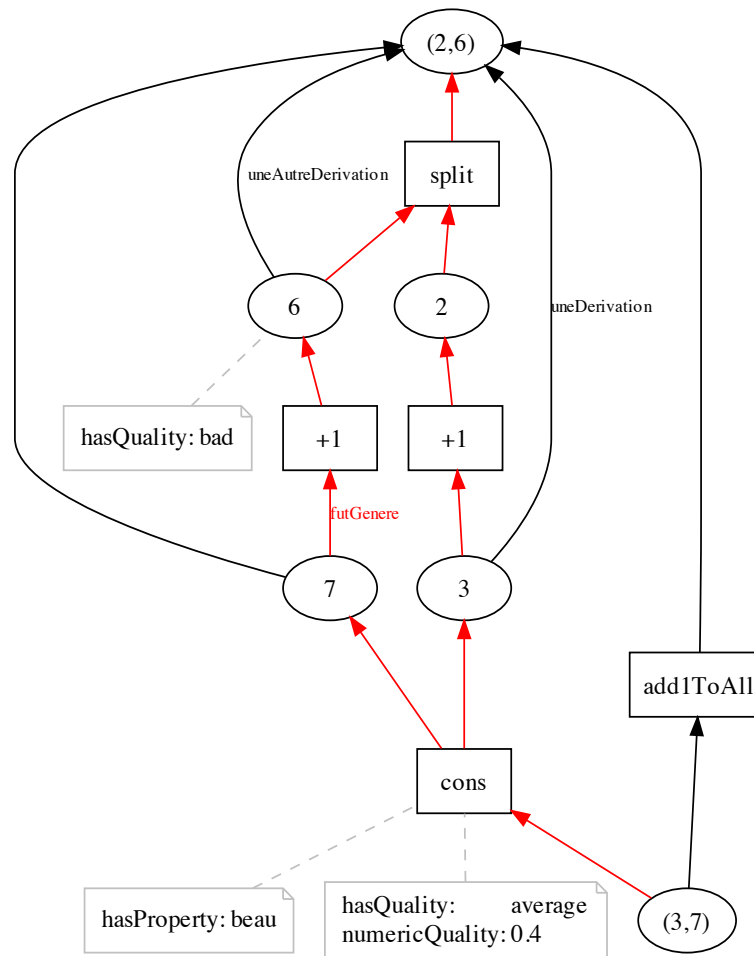
- opmexample2.tar.gz from the tutorial distribution

```
tar xvf opmexample2.tar.gz
```

```
cd opmexample2
```

```
mvn clean install
```

Resulting graph in target/



Maven Repository



```
<repositories>
```

```
  <repository>
```

```
    <id>opm</id>
```

```
    <name>OPM releases Repository</name>
```

```
    <url>http://openprovenance.org/java/maven-releases/</url>
```

```
  </repository>
```

```
  <repository>
```

```
    <id>opm-snapshots</id>
```

```
    <name>OPM snapshot Repository</name>
```

```
    <url>http://openprovenance.org/java/maven-snapshots/</url>
```

```
  </repository>
```

```
</repositories>
```

Programming with the toolbox



- Java model derived from XML Schema (JAXB)
- Key class (and subclasses): OPMFactory
- Contains factory methods for all OPM objects
- Graph generator parameterized by factory

```
public OPMGraph getGraph(OPMFactory oFactory)
{
    OPMGraph graph = ...;
    return graph;
}
```

Full example in `src/main/java/uk/ac/soton/ecs/opmdemo/MyGraph.java`

Artifact/Process creation



```
Artifact a4=oFactory.newArtifact( "a4",  
                                   orange,  
                                   "6" );
```

```
Process p5=oFactory.newProcess( "p5",  
                                orange,  
                                "http://process.org/cons" );
```

Arguments are: node id, list of accounts, label for pretty-printing

Edge creation



```
Used u1=oFactory.newUsed(p1,  
                           oFactory.newRole("in"),  
                           a1,  
                           green);  
WasGeneratedBy wgl=oFactory.newWasGeneratedBy(a2,  
                                                oFactory.newRole("out"),  
                                                p1,  
                                                green);  
WasDerivedFrom wdf1=oFactory.newWasDerivedFrom(a6,  
                                                  a1,  
                                                  green);
```

Arguments refer to previously created processes or artifacts

Graph creation



```
OPMGraph graph
=oFactory.newOPMGraph( green_orange,
                        new Overlaps[] {},
                        new Process[] {p1,p2,p3,p4,p5},
                        new Artifact[] {a1,a2,a3,a4,a5,a6},
                        new Agent[] {},
                        new Object[] {u1,u2,u3,u4,u5,u6,
                                     wg1,wg2,wg3,wg4,wg5,
                                     wdf1,wdf2, wdf3},
                        new Annotation[] {} );
```

Arguments refer to previously created nodes, edges, and accounts

Annotation creation



```
EmbeddedAnnotation ann
=oFactory.newEmbeddedAnnotation
    ("an15",
     "http://property.org/hasQuality",
     "average",
     green);

oFactory.addAnnotation(a1,ann15);
```

Serialization



- Get thread-safe serializer
- Invoke serialization method with file, graph, and flag to pretty print

```
OPMGraph graph=new MyGraph().getGraph(oFactory);
```

```
OPMSerialiser serial  
    =OPMSerialiser.getThreadOPMSerialiser();
```

```
serial.serialiseOPMGraph(new File("target/example.xml"),  
                           graph,  
                           true);
```

Full code in `src/test/java/uk/ac/soton/ecs/opmdemo/ExampleTest.java`

Deserialization



- Get thread-safe deserializer
- Invoke deserialization method with file

```
OPMDeserialiser deserial  
    =OPMDeserialiser.getThreadOPMDeserialiser();
```

```
OPMGraph graph  
    =deserial.deserialiseOPMGraph(new File("exmpl.xml"));
```

Conversion to dot/pdf



- Create converter
- Invoke conversion method (like command line)

```
OPMToDot toDot=new OPMToDot();
```

```
toDot.convert(graph,  
              "target/example.dot",  
              "target/example.pdf");
```

What about generation of RDF?



- The Java data model is directly derived from the XML schema
- JAXB converts from Java to XML and back
- RDFOPMFactory is a subclass of OPMFactory, which also constructs an RDF representation in a Sesame model
- Relies on Sesame's elmo generating Java interfaces/classes from an OWL ontology
- RDF representation compatible with OPM OWL ontology by construction!
- Serializing is by dumping the RDF model

Conversion to RDF



- Create the RDFOPMFactory:

```
oFactory=  
    new RdfOPMFactory(new RdfObjectFactory(manager,  
                                           TEST_NS));
```

- Dump RDF (relies on helper class)

```
File file = new File("target/example.n3");  
  
rHelper.dumpToRDF(file,  
                  (SesameManager)manager,  
                  RDFFormat.N3,  
                  prefixes);
```

Full example in `src/test/java/uk/ac/soton/ecs/opmdemo/RdfTest.java`

Other stuff



- IndexedOPMGraph: a class that allows incremental creation of OPM graph, indexing of its nodes, and easy traversal in a forward or backward manner
- Basic support for graph normalization and comparison
- Intended to support OPM graph algebra

CONTRIBUTING TO THE OPM TOOLBOX

Source code on github



github SOCIAL CODING

lucmoreau Dashboard Inbox 0 Account Settings Log Out

Explore GitHub Gist Blog Help Search GitHub...

lucmoreau / **OpenProvenanceModel** Admin Watch Pull Request Download Source 5 1

Source Commits Network (1) Fork Queue Issues (3) Downloads (5) Wiki (0) Graphs Branch: master

Switch Branches (4) Switch Tags (5) Branch List

The overarching project of Java code related to the Open Provenance specifications.
<http://openprovenance.org/>

SSH HTTP Git Read-Only `git@github.com:lucmoreau/OpenProvenanceModel.git` This URL has Read+Write access

license

lucmoreau (author)
about an hour ago

commit: `d1e7924ce22eb4c4a5a1`
tree: `eee616f68db9d8a89dbe`
parent: `50eb78fd3b8e71948b6f`

OpenProvenanceModel /

name	age	message	history
<code>.gitignore</code>	October 28, 2009	minor fixes [Luc Moreau]	
<code>RELEASES.txt</code>	about an hour ago	license [lucmoreau]	
<code>TODO.txt</code>	February 05, 2010	sig [lucmoreau]	
<code>build.xml</code>	October 24, 2008	*** empty log message *** [lucmoreau]	
<code>elmo/</code>	about 3 hours ago	[maven-release-plugin] prepare for next develop... [lucmoreau]	
<code>license.txt</code>	about 12 hours ago	readme [Luc Moreau]	
<code>opm/</code>	about 3 hours ago	[maven-release-plugin] prepare for next develop... [lucmoreau]	
<code>pom.xml</code>	about 3 hours ago	[maven-release-plugin] prepare for next develop... [lucmoreau]	
<code>src/</code>	August 25, 2009	git-svn-id: https://provenance.ecs.soton.ac.uk/ ... [lucmoreau]	
<code>toolbox/</code>	about an hour ago	license [lucmoreau]	
<code>tupelo/</code>	about 3 hours ago	[maven-release-plugin] prepare for next develop... [lucmoreau]	

We couldn't find a README for this repository, we strongly recommend adding one. For more details on what formats we support, visit [github/markup](https://github.com/markup)

Blog Support Training Job Board Shop Contact API Status

© 2010 GitHub Inc. All rights reserved. Terms of Service Privacy Security

Powered by the Dedicated Servers and Cloud Computing of Rackspace Hosting®

Get Involved



- Source code:
 - <http://github.com/lucmoreau/OpenProvenanceModel>
- Currently three branches: master, opmsig, opmowl
- Download, compile by a simple mvn clean install (note build relies on existence of unix executables such as sed in the path)
- Fork in github.com and modify

Conclusion



- OPM Toolbox intended to support useful and common functionality for manipulating OPM graphs
- Community involvement would be desirable
- Try and experiment with it!