Taverna as an Apache Incubator project

Stian Soiland-Reyes
myGrid team
University of Manchester

http://orcid.org/0000-0001-9842-9718

This work is licensed under a Creative Commons Attribution 4.0 International License.
A brief history of Taverna development

... or at least according to a quick look at sourceforge/cvs/svn/git history and occasional dips into Stian's memory
2001-2006: Prototyping

myGrid: 6 academic institutions and 8 industry partners

Challenge: Create a graphical workbench for bioinformaticians to combine data and web services

Taverna Workbench releases:

- 2003: 0.1 beta of Taverna Scufl workbench
- 2005: 1.0 release, using Freefluo engine

Open source:

Source+binary releases on SourceForge.net
Local CVS servers → SourceForge.net CVS
2006-2009: Productizing

Apache Maven-based build

executeworkflow command line

Version 1.5:
Raven plugin system

Version 1.7:
Taverna Remote Execution Service
Preview of t2core workflow engine
2007-2014: Taverna 2.x

Version 2.0 (2008)
SourceForge CVS → GoogleCode SVN
t2core workflow engine
Reimplemented workbench

Version 2.1.2 (2009)
Improved support for 3rd-party plugins

Version 2.2 (2010)
Taverna Server + ruby gem

Version 2.5 (2014)
Domain-specific editions
(astronomy, bioinformatics, biodiversity, digital preservation)

GoogleCode SVN → GitHub
Taverna 2 issues

Plugin system too strict on versioning of dependencies
Every new release of Taverna Workbench made existing plugins incompatible
Maintaining our own classloader implementation - hard!
   Building a release difficult -
   several weeks including testing
2012-: Taverna 3

Version 3.0 alpha (2013)

**OSGi**-based plugin system

**Engine** separated from workbench/commandline/server

Independent **APIs** for workflow and data format

Taverna **Platform** - execution API

Version 3.0 (2015)

**Apache** Taverna (incubating)
Apache Incubator

Taverna
Apache Foundation
Non-profit organization, forming a community of open-source software projects.

Strong emphasis on openness, collaboration and a consensus-based development process.

Examples: Apache HTTP server, Tomcat, Maven, Jena, CXF and OpenOffice.
Why Apache Taverna?

- Fully open development
- Encourage further developer involvement in core codebase
- Remove impression of Univ of Manchester as leader
  - Reduce *us vs them* conversations
- Independence from Univ of Manchester
  - Centrally managed infrastructure
  - Less political to include Apache Taverna in say a grant proposal
  - Longer term sustainability - self-managed community
Apache Incubator

Graduation path for becoming an Apache project
Taverna Incubator Proposal

**Champion:** Andy Seaborne (Apache Jena)

**Proposal** submitted 2014-09-23

Volunteering **mentors:**

- Chris Mattmann
- Suresh Srinivas
- Suresh Marru
- Marlon Pierce
- Michael Joyce
Questions raised

- **Third-party LGPL dependencies?** (incompatible with Apache releases)
- **Should future developer-contributed plugins be part of "Apache Taverna"?** Danger of **fragmenting existing community**
- **How to diversify core development outside University of Manchester?**
- **Adequate rights to change the license wholesale?**
- **Build infrastructure required by Apache?** Migration of 85 git repositories **a concern.**
Incubator voting

+13 votes: Accepted!
Next: house keeping

1. Migrate to new mailing lists, e.g. dev@taverna.incubator.apache.org
2. Apache Contributor License Agreement (CLA) needs to be signed for each Taverna committer
3. Apache account per committer - e.g. stain@apache.org
4. Paperwork handover from Univ. of Manchester
Initial goals

1. Fully investigate/resolve incompatibly licensed dependencies
2. Stage **git repositories** for move at https://github.com/taverna-incubator:
   - Restructure git repositories (to ~ 10 repos?)
   - Update headers/metadata to indicate Apache License 2.0
   - Rename Maven groupIds to **org.apache.taverna.**
   - Rename packages to **org.apache.taverna.**
   - Move staged Github repositories to Apache git
3. **Automated builds** in Apache's Jenkins
4. Update to **latest dependencies**
5. Avoid dependencies on myGrid Maven repository
6. Propose updated **release and testing procedure**
7. Move **Website** and documentation **wiki**
Git repositories

1. Reduce our current 85 github repositories to a more manageable number
2. Stage merged repositories at https://github.com/taverna-incubator
3. Request Apache INFRA team to import code at git.apache.org
4. Mirrored back to https://github.com/apache
   1. Committers: push to git.apache.org
   2. Others: Github pull requests
   3. Pull requests merged manually by a committer
## Proposed git structure

<table>
<thead>
<tr>
<th>GroupId / Java package name</th>
<th>GroupId / Java package name</th>
</tr>
</thead>
<tbody>
<tr>
<td>apache/taverna-maven-parent</td>
<td>org.apache.taverna.parent</td>
</tr>
<tr>
<td>apache/taverna-utils</td>
<td>org.apache.taverna  # Still needed?</td>
</tr>
<tr>
<td>apache/taverna-engine-api</td>
<td>org.apache.taverna.engine</td>
</tr>
<tr>
<td>taverna-core-api</td>
<td>org.apache.taverna.core</td>
</tr>
<tr>
<td>taverna-platform-api</td>
<td>org.apache.taverna.platform</td>
</tr>
<tr>
<td>apache/taverna-engine-impl</td>
<td>org.apache.taverna.engine.impl</td>
</tr>
<tr>
<td>taverna-core-impl</td>
<td>org.apache.taverna.core.impl</td>
</tr>
<tr>
<td>taverna-platform-impl</td>
<td>org.apache.taverna.platform.impl</td>
</tr>
<tr>
<td>taverna-rest-activity</td>
<td>org.apache.taverna.rest.activity</td>
</tr>
<tr>
<td>taverna-script-activity</td>
<td>org.apache.taverna.script.activity</td>
</tr>
<tr>
<td>taverna-interaction-activity</td>
<td>org.apache.taverna.interaction.activity</td>
</tr>
<tr>
<td>apache/taverna-scuf12</td>
<td>org.apache.taverna.scuf12</td>
</tr>
<tr>
<td>apache/taverna-databundle</td>
<td>org.apache.taverna.databundle</td>
</tr>
<tr>
<td>apache/taverna-workbench-api</td>
<td>org.apache.taverna.workbench.api</td>
</tr>
<tr>
<td></td>
<td>org.apache.taverna.workbench.api.activitypalette</td>
</tr>
<tr>
<td></td>
<td>org.apache.taverna.workbench.api.menu</td>
</tr>
<tr>
<td>apache/taverna-workbench-impl</td>
<td>org.apache.taverna.workbench.impl</td>
</tr>
<tr>
<td></td>
<td>org.apache.taverna.workbench.impl.activitypalette</td>
</tr>
<tr>
<td></td>
<td>org.apache.taverna.workbench.impl.menu</td>
</tr>
<tr>
<td></td>
<td>org.apache.taverna.workbench.impl.diagram</td>
</tr>
<tr>
<td></td>
<td>org.apache.taverna.rest.workbench</td>
</tr>
<tr>
<td></td>
<td>org.apache.taverna.script.workbench</td>
</tr>
<tr>
<td>apache/taverna-commandline-product</td>
<td></td>
</tr>
<tr>
<td>apache/taverna-workbench-product</td>
<td></td>
</tr>
<tr>
<td>apache/taverna-server</td>
<td>org.apache.taverna.server</td>
</tr>
<tr>
<td>apache/taverna-plugin-biomart</td>
<td>org.apache.taverna.biomart</td>
</tr>
<tr>
<td></td>
<td>org.apache.taverna.biomart.activity</td>
</tr>
<tr>
<td></td>
<td>org.apache.taverna.biomart.workbench</td>
</tr>
<tr>
<td>apache/taverna-plugin-soaplab</td>
<td>org.apache.taverna.soaplab</td>
</tr>
</tbody>
</table>
Suggested release plan

1. Support libraries (e.g. taverna-scuf12 and taverna-databundle)
2. Apache Taverna **Command Line** 3.0 beta 3
   (infrastructure test)
3. Apache Taverna **Command Line** 3.0
4. Apache Taverna **Server** 3.0
5. Apache Taverna **Workbench** 3.0 beta 3
6. Apache Taverna **Workbench** 3.0 (core edition)
7. (Third-party plugins/editions, e.g. AstroTaverna)
What about Taverna 2.x?

Not going into Apache Incubator

"The final release": Taverna 2.5.1 (really soon now)

Security fixes only until Apache Taverna 3.0.0 is released
Changes to work practices

Fully open development
All discussion and decisions to be made on dev@taverna.incubator.apache.org mailing list
Mailing list open to all - not just committers
Voting on releases
Developer infrastructure managed by Apache
Meritocracy
Earn it through doing well
Who is the Taverna developer community?

**Plugin** developers (e.g. Kevin, Dmitry)

**Integrators** (e.g. Vadim, Susheel, Yassene)

**Core** developers (e.g. Manchester, Dmitry)

Scientific Workflow **researchers** (e.g. Gabor, Pinar)
How do I become a committer?

Submitting patches/pull requests
Contributing to mailing lists
Contributing to testing and documentation
Contributing a plugin
How can we recruit committers?

Be active on mailing lists
Find and invite plugin developers and integrators
Code documentation/examples/tutorials
Community webinars?
"I found a bug, but I am not sure how to fix it"
→ Online mentoring?

... your suggestion?
Thanks!