ETICS: quality certification of open source cloud software

the ETICS approach and the way it's planned to be adopted within the Ocean project.

Gabriele Giammatteo
Open source

Often open source components developed in research projects are abandoned

- Poorly documented
- Not supported
- Confused development process
- Non-standard
  - Interfaces (interoperability)
  - Project' structure
  - Packaging
- Not properly tested

Low quality software, difficult to reuse and evolve
ETICS* approach

- ETICS is a system that supports software professionals during software life-cycle
- Helps to **manage software complexity** and improve quality
- Promotes, encourages and supports **standardization** of build, test and release processes
- Certificates **software quality** integrating metrics from various code analysis and testing tools

* E-infrastructure for Testing, Integration and Configuration of Software
ETICS Key Features

- Solution for building/testing/quality
  - Automation of build, test and QA verification
  - Dependency management
  - Creation of distribution packages
  - Easy extensible through plug-ins
- Independent from any language, platform, tools and project size
- Repository for metadata, packages, logs and reports
- web Graphical User Interface
Building

- **Predefined targets** to accomplish most common actions (e.g. clear, init, compile, install, package, etc...)
  - ETICS **execute** the commands specified for each target in a shell
  - Definition and use of **properties** and **env variables** to ease writing of commands

- Specification of **build/runtime dependencies**: e.g. static (v.1.4), dynamic (project-level), range (1.4 or newer), target OS

- **Built-in packaging system** automatically packages software according to the target platform (tarballs, RPMs, DEBs)

- Produced packages (sources and binaries) and reports published on central **repository**
Architecture

ETICS Infrastructure

Execution Engines (Metronome, glite, UNICORE, etc)

Physical Worker Nodes

Virtual OS Images

Configuration DB

Report/Metrics DB

Build/Test Artefacts

Repository Web Service

Configuration Web Service

Web Portal

Command Line User Interface
ETICS QA

• ETICS executes different types of testing at build-time:
  • **Unit tests**: automatic discovery and execution
  • **Static analysis tools**: activated according to the nature of project, language(s) and user preferences
  • **Functional testing**: automation of custom testing scripts for each module in the project
  • **Multi-node testing**: automation and synchronisation of testing scripts against components deployed on different nodes
    - e.g. client-server test
  • Measures for all metrics are stored in the repository for later analysis and trends
ETICS Plug-ins

- Plug-ins are usually wrappers around existing and very specialized tools (e.g. code analysis, testing, packaging, reporting, etc)

- Static analysis
  - Sloccount (Single line of code count)
  - Findbugs, PMD, CKJM, Checkstyle
  - CppCheck, CCCC

- Unit Testing:
  - PyUnit, Junit, CppUnit
  - Pycoverage, Jcoverage

- and more coming soon ...
Pacakges

The ETICS repository is the standard location where all the software artefacts of the ETICS Build and Test System: packages, metrics, build and test reports are publicly available.

Repository Information

Snapshot: http://gridb07.eng.irepository/pn/registered/repository/74dc05dc-1b62-4df8-8a85-f4e65d8d6b7/centos6_x86_64_pcc446

Repository Information

Submission ID: 74dc05dc-1b62-4df8-8a85-f4e65d8d6b7
Platform: centos6_x86_64_gcc446
Date: 22/11/2012 16:40:12

Project Name: org.etics
Configuration Name: org.etics.build-system.client.py-1.6e.15
Configuration Version: 1.6e.15-0

Author: CNA Gabriele Giammatteo, L="ENGINEERING ADLAB", OU="Personal Certificate", O="INFN", C="IT"

Build Reports: View

Repository Index Files

Yum repo file (permanent link to this build) [protect=0]: etics-registered-build-by-id.repo
Yum repo file (permanent link to this build) [protect=1]: etics-registered-build-by-id-protect.repo

Repmad file: repodata/repmad.xml
Primary index: repodata/primary.xml.gz
Filelists index: repodata/filelists.xml.gz
Other index: repodata/other.xml.gz

Package List (1)

etics-client (1.6e.15-0.centos6) [view contents] [package details]
QA Reports

Project: n/a (tepol)
Configuration: minisip.HEAD (minisip)
Date: 24/11/2012 19:58:02

Total Estimated Effort (Basic COCOMO)
59.93 Person/Years

Total Estimated Cost (Basic COCOMO)
8096149.0 EUR

Total SLOC grouped by language (dominant language first)

<table>
<thead>
<tr>
<th>Language</th>
<th>Total SLOC</th>
</tr>
</thead>
<tbody>
<tr>
<td>cpp</td>
<td>144730 (63%)</td>
</tr>
<tr>
<td>ansic</td>
<td>49019 (21%)</td>
</tr>
<tr>
<td>asm</td>
<td>21008 (9%)</td>
</tr>
<tr>
<td>sh</td>
<td>10747 (4%)</td>
</tr>
<tr>
<td>perl</td>
<td>1612 (0%)</td>
</tr>
<tr>
<td>python</td>
<td>1144 (0%)</td>
</tr>
<tr>
<td>cs</td>
<td>143 (0%)</td>
</tr>
</tbody>
</table>

SLOC by language for all modules

Procedural Metrics Summary

For descriptions of each of these metrics see the information preceding the project summary table. The label cell for each row in this table provides a link to the functions table in the detailed report for the module in question.

<table>
<thead>
<tr>
<th>Module Name</th>
<th>LOC</th>
<th>MVG</th>
<th>COM</th>
<th>L_C</th>
<th>M_C</th>
</tr>
</thead>
<tbody>
<tr>
<td>AccountDialog</td>
<td>544</td>
<td>38</td>
<td>26</td>
<td>20.923</td>
<td>1.462</td>
</tr>
<tr>
<td>AccountsList</td>
<td>74</td>
<td>4</td>
<td>42</td>
<td>1.762</td>
<td>1.462</td>
</tr>
<tr>
<td>AccountsListColumns</td>
<td>76</td>
<td>0</td>
<td>42</td>
<td>1.667</td>
<td>1.462</td>
</tr>
<tr>
<td>AccountsStatusWidget</td>
<td>248</td>
<td>14</td>
<td>46</td>
<td>5.391</td>
<td>0.304</td>
</tr>
<tr>
<td>AdvancedSettings</td>
<td>254</td>
<td>34</td>
<td>12</td>
<td>21.167</td>
<td>2.833</td>
</tr>
<tr>
<td>Animate</td>
<td>302</td>
<td>60</td>
<td>70</td>
<td>4.314</td>
<td>1.143</td>
</tr>
<tr>
<td>Atom</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td>Bounds</td>
<td>182</td>
<td>46</td>
<td>141</td>
<td>1.298</td>
<td>0.340</td>
</tr>
<tr>
<td>BorderLayout</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td>Button</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td>CallListStore</td>
<td>120</td>
<td>16</td>
<td>6</td>
<td>20.000</td>
<td>2.667</td>
</tr>
<tr>
<td>CallWidget</td>
<td>1848</td>
<td>262</td>
<td>264</td>
<td>7.000</td>
<td>0.765</td>
</tr>
<tr>
<td>CarrierDisplay</td>
<td>135</td>
<td>23</td>
<td>3</td>
<td>45.000</td>
<td>7.667</td>
</tr>
</tbody>
</table>
Open Cloud for Europe, JApan and beyoNd

- Build a **catalogue of open source cloud components**:  
  - foster **complementariness** and reduce **overlaps**  
  - contributing to the emergence of a sustainable open cloud business ecosystem  
  - Classification, **discoverability, comparison**

- Metadata associated to each component  
  - Authors, licenses  
  - Projects being used  
  - Relationships with other components  
  - Standards implemented  
  - ...

Oct 2012 – Sept 2014
ETICS in Ocean

- ETICS will be offered as a service for projects that join Ocean
- Running its plug-ins, ETICS will evaluate software quality
- Data collected will be used to enrich Ocean Catalogue's metadata
  - Publish values for re-usability, maintainability, documentation, ...
- Implementation of new plug-ins for cloud-specific components
  - e.g.
    - OCCI compliance
    - Cloud Benchmarks
Summary

- Low quality of open source components is an obstacle to reuse and evolution of such components
- ETICS tools tries to `standardize` build, test and release process and `check quality` by running testing tools plug-ins
- Ocean project will build a catalogue of open source cloud components
  - makes them discoverable, comparable
  - analyse relationships
  - Gives, through ETICS, an estimation for components' quality
Thank you!

mailto: gabriele.giammatteo@eng.it

ETICS : http://etics.web.cern.ch/
        https://grids06.eng.it/

Ocean : http://ocean-project.eu