



www.ow2.org



#ow2con




ETICS supporting compliance and interoperability

Gabriele Giammatteo
Engineering Group

14 November 2013

The OCEAN project

 is building a directory of open-source cloud-oriented projects focused on quality and interoperability of software components



ETICS tool is used to in OCEAN to:

- extract **quality** metrics from software
- verify **interoperability**
- check **compliance** to open cloud standards

ETICS

ETICS is a **build and test automation** tool



- **Web-based** portal for managing configuration, builds, tests, packages, reports
- **independent** from programming language, project structure and target platform:
 - C, C++, Java, Scala, Python, Ruby, ...
 - Make, Ant, Maven, SCons, Rake, ...
 - RedHat, Debian, Windows, ...

ETICS

ETICS is a **build and test automation** tool



- **Dependency management:** build/run-time, static/dynamic, ranges
- Built-in **packing system** to create distribution packages (tar.gz, deb, rpm, sources, ...)
- Central **repository** for packages, reports, logs compatible with yum/apt formats
- Testing capabilities extendible through **plug-ins**

Testing with ETICS - Overview

- **static analysis**

- integration of specialised tools for code analysis like *Checkstyle*, *Findbugs*, *PMD*, *CppCheck*, *CCCC*, *PyLint*, ...

Testing with ETICS - Overview

- **static analysis**

- integration of specialised tools for code analysis like *Checkstyle*, *Findbugs*, *PMD*, *CppCheck*, *CCCC*, *PyLint*, ...

- **unit testing**

- integration of *Junit*, *CppUnit*, *PyUnit*

Testing with ETICS - Overview

- **static analysis**

- integration of specialised tools for code analysis like *Checkstyle*, *Findbugs*, *PMD*, *CppCheck*, *CCCC*, *PyLint*, ...

- **unit testing**

- integration of *Junit*, *CppUnit*, *PyUnit*

- **functional testing**

- execution and reports of tests written by developers

Testing with ETICS - Overview

- **static analysis**

- integration of specialised tools for code analysis like *Checkstyle*, *Findbugs*, *PMD*, *CppCheck*, *CCCC*, *PyLint*, ...

- **unit testing**

- integration of *Junit*, *CppUnit*, *PyUnit*

- **functional testing**

- execution and reports of tests written by developers

- **multi-node testing**

- realisation of complex testing scenarios involving multiple nodes


Reports



CompatibleOne

compatibleone

Project homepage: <http://www.compatibleone.org/bin/view/Main/>

powered by  **eTICS**

Module Summary

name: compatibleone
configuration: compatibleone-1.0.0
description: CompatibleOne offers a simple and unique interface allowing for the description of user cloud computing needs, in terms of resources, and their subsequent provisioning on the most appropriate cloud provider
project: compatibleone
vendor:
license:

Execution Summary

worker node: etics-debian70-w126203.res.eng.it
platform: deb7_x86_64_gcc472-5
start time: 17/09/2013 14:25:30
end time: 17/09/2013 14:33:12
duration: 00:07:42
build result: **Success**

Quality Assurance

CBO (CCCC): 0.0 modules
Cppcheck: 3 Errors
Cppcheck rate: 100% Modules
DIT (CCCC): 0.0 level
IF4 (CCCC): 0.0 level
L_C (CCCC): 4.2102 lines
M_C (CCCC): 1.3183 lines
MVG (CCCC): 786.6 level
NOC (CCCC): 0.0 modules
SLOC: 43052 SLOC
WMC (CCCC): 51.3913 functions
Overall Metrics: Success

Modules

time	project	module	version	configuration	from	status	artefacts
14:25:44	OS	libtool		libtool 2.4.2-1.1 (OSD)	system	Checked-out (OSD)	
14:25:44	OS	autoconf		autoconf 2.69-1 (OSD)	system	Checked-out (OSD)	
14:25:44	OS	libuid-devel		libuid-dev 2.20.1-5.3 (OSD)	system	Checked-out (OSD)	
14:25:44	OS	automake		automake 1.11.6-1 (OSD)	system	Checked-out (OSD)	
14:27:57	compatibleone	sourcecode-initializer	1.0.0	sourcecode-initializer-1.0.0	src	Success	.tar.gz .deb
14:28:24	compatibleone	coxml	1.0.0	coxml-1.0.0	src	Success	.tar.gz .deb
14:28:32	compatibleone	CORDS	1.0.0	CORDS-1.0.0	src	Success	.tar.gz .deb
14:29:58	compatibleone	occi	1.0.0	occi-1.0.0	src	Success	.tar.gz .deb
14:30:53	compatibleone	cocarrier	1.0.0	cocarrier-1.0.0	src	Success	.tar.gz .deb
14:31:14	compatibleone	cocci	1.0.0	cocci-1.0.0	src	Success	.tar.gz .deb
14:31:29	compatibleone	copabr	1.0.0	copabr-1.0.0	src	Success	.tar.gz .deb
14:31:48	compatibleone	pubocci	1.0.0	pubocci-1.0.0	src	Success	.tar.gz .deb
14:31:57	compatibleone	corest	1.0.0	corest-1.0.0	src	Success	.tar.gz .deb
14:32:10	compatibleone	coips	1.0.0	coips-1.0.0	src	Success	.tar.gz .deb
14:32:18	compatibleone	ezvm	1.0.0	ezvm-1.0.0	src	Success	.tar.gz .deb
14:32:25	compatibleone	cocspi	1.0.0	cocspi-1.0.0	src	Success	.tar.gz .deb
14:32:38	compatibleone	slam	1.0.0	slam-1.0.0	src	Success	.tar.gz .deb
14:32:46	compatibleone	cosched	1.0.0	cosched-1.0.0	src	Success	.tar.gz .deb
14:32:59	compatibleone	cos	1.0.0	cos-1.0.0	src	Success	.tar.gz .deb
14:33:05	compatibleone	cosacs	1.0.0	cosacs-1.0.0	src	Success	.tar.gz .deb
14:33:12	compatibleone	compatibleone	1.0.0	compatibleone-1.0.0		Success	




Environment

LDLFLAGS: -L/var/lib/condor/execute/dir_14040/userdir/stage/lib
PYTHONPATH: /usr/lib/python2.7


Commands


client version: 1.9.2-1
etics-get-project: etics-get-project compatibleone
etics-checkout: etics-checkout --config compatibleone-1.0.0 --forcecheckout --verbose --continueonerror compatibleone
etics-build: etics-build --config compatibleone-1.0.0 --verbose --target postpublish --continueonerror --autopublish -p default.profile=cccc,ppcheck compatibleone
etics-test:

Terms of use Privacy Contacts

Reports



compatibleone
powered by 

Project homepage: <http://www.compatibleone.org/bin/view/Main/>

Project Summary

ETICS
VCS
Repository
Homepage
Description: The ETICS System

Created: Wed Mar 22 08:51:42
Modified: Sat Aug 01 01:15:06

AQCM Results:
Maintainability: 0.6553551
Portability: 0.2694004
Reliability: 0.2694004
Functionality: 0.04082789
[ETICS Complete Report](#)

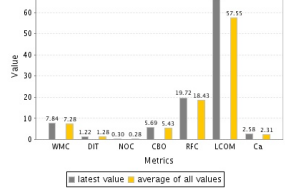
Portability Per Platform

Configuration	Platform	Status
etics-dev	stc4_la32_gcc346	Success
etics_int	stc4_la32_gcc346	Success
etics_R_2_4_RC3	stc4_la32_gcc346	Success

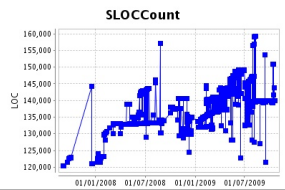
Jobs History

Type	Date	User	Project	Configuration	Platform	Status
build	20 October 2009 18:48:53	ETICS Scheduler	org.etics	etics_branch_2_5_0	Submitting	Submitting
build	20 October 2009 12:46:53	ETICS Scheduler	org.etics	etics_branch_2_5_0	Submitting	Submitting
build	20 October 2009 10:08:23	Marco Canaparo	org.etics	etics-dev	stc4_la32_gcc346	Success
build	20 October 2009 06:46:53	ETICS Scheduler	org.etics	etics_branch_2_5_0	Submitting	Submitting
build	20 October 2009 02:01:56	ETICS Scheduler	org.etics	etics-dev	stc4_la32_gcc346	Success
build	20 October 2009 02:01:56	ETICS Scheduler	org.etics	etics_int	stc4_la32_gcc346	Success
build	20 October 2009 00:46:51	ETICS Scheduler	org.etics	etics_branch_2_5_0	Submitting	Submitting
build	19 October 2009 18:46:54	ETICS Scheduler	org.etics	etics_branch_2_5_0	Submitting	Submitting
build	19 October 2009 15:01:12	Marco Canaparo	org.etics	etics-dev	stc4_la32_gcc346	Success
build	19 October 2009 14:10:21	Marco Canaparo	org.etics	etics-dev	stc4_la32_gcc346	Success

WMC DIT NOC CBO RFC LCOM Ca



SLOCCount



Quality Assurance

- BlockCheck: 188 Violations
- Checkstyle Rate: 100% Modules
- Checkstyle Thresholds Rate: 100% Thresholds
- CodingStyleCheck: 278 Violations
- DesignCheck: 437 Violations
- Findbugs: 171 Bugs
- Findbugs rate: 100% Modules
- ImportCheck: 264 Violations
- JavadocCheck: 1366 Violations
- LenghtCheck: 554 Violations
- MiscellaneousCheck: 430 Violations
- ModifierCheck: 7 Violations
- NamingCheck: 58 Violations
- SLOC: 10272 SLOC
- WhitespaceCcek: 41 Violations
- Overall Metrics: **Success**



Findbugs Reports

Project: choreos-1.0.0 (choreos)
Configuration: choreos-1.0.0 (choreos)
Date: 16/09/2013 16:08:23
Success rate: 100 %
Status: Success

Component name	Configuration name	Low	Medium	High	Total	Result
deployment-manager (summary - bugs)	deployment-manager-1.0.0	28	43	8	79	Success
enactment-engine-api (summary - bugs)	enactment-engine-api-1.0.0	10	36	5	51	Success
middleware-commons (summary - bugs)	middleware-commons-1.0.0	11	23	7	41	Success

ETICS Multi-node testing

ETICS provides a tool-kit for the definition and automation of complex test scenarios involving multiple nodes that interact each other in a synchronised way

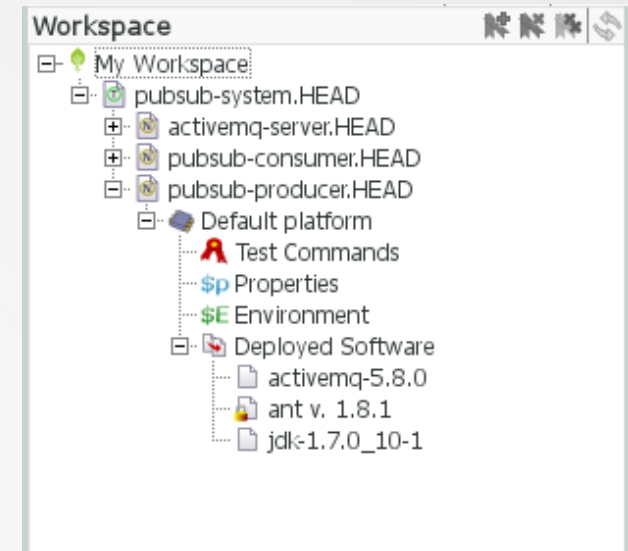
Applicability:

- **Deployment** testing of services
- Functional testing of **client-server** applications
- **Interoperability** testing between services



Multi-node characteristics

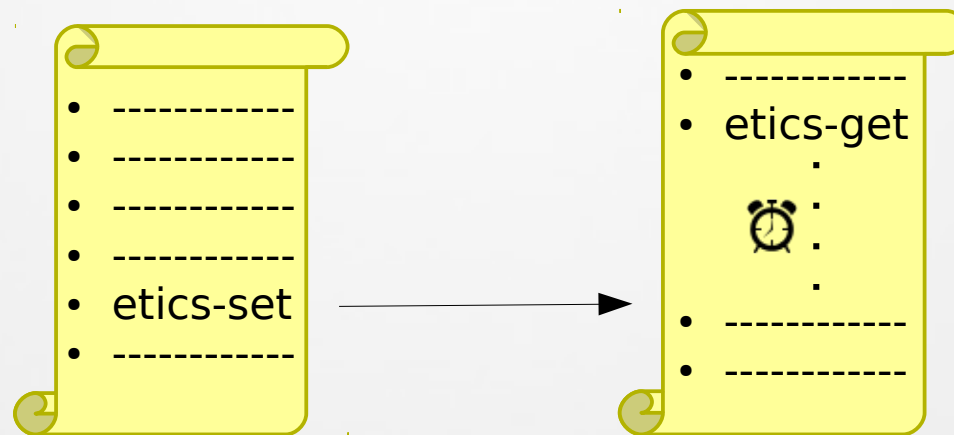
- tests definition through the web portal
 - number and characteristics of nodes
 - software to be installed on each node
 - testing scripts
- define once run multiple times:
 - on regular base, on every commit to source repository
 - on different versions of the software and platforms
 - with different values for parameters
- nodes dynamically created and destroyed for each test



Multi-node testing – Synch primitives

ETICS provides a **synchronisation primitives** to be used in testing scripts:

- **etics-set**: publish an information
- **etics-get**: retrieve an information (blocks until published)
- e.g.:
 - on server node: `etics-set SERVER_IP 192.168.100.12`
 - on client node: `etics-get SERVER_IP`



Multi-node testing – A simple example

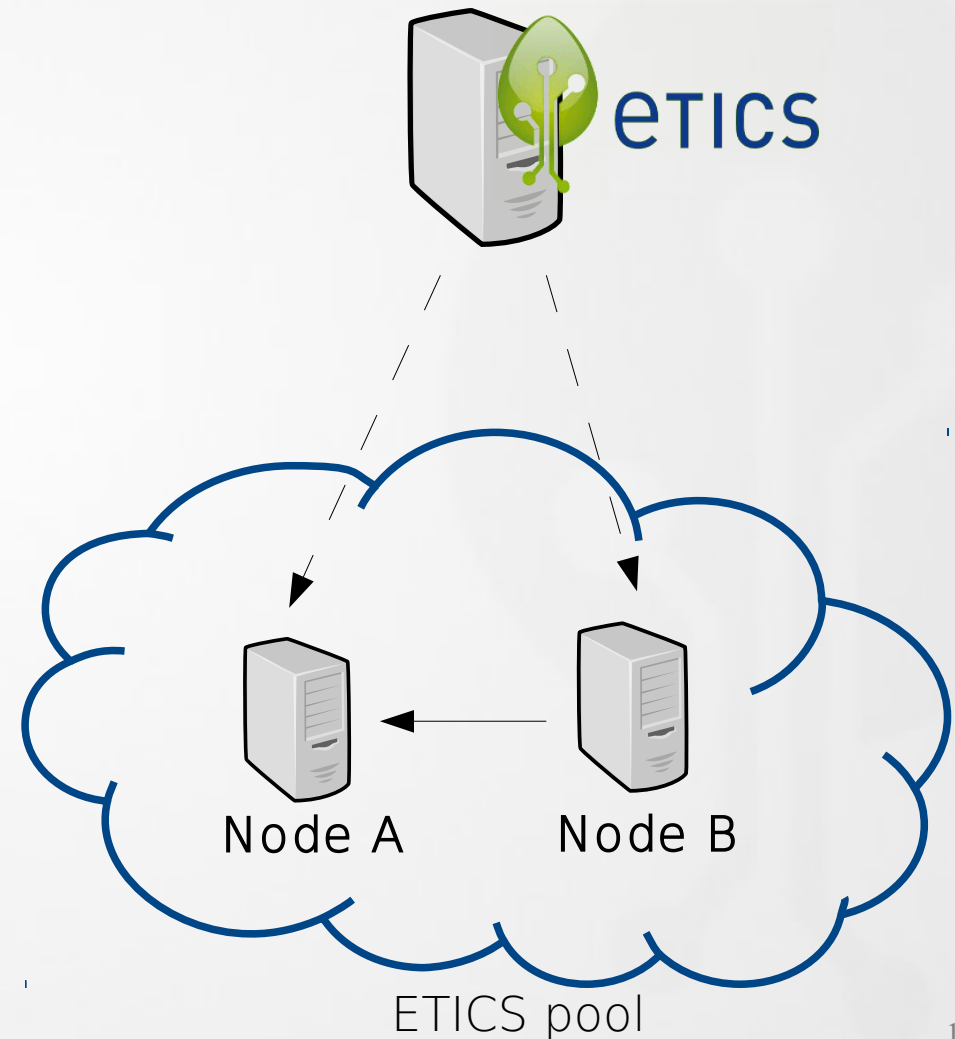
Test the deployment of a Java web application

Node A script:

1. deploy *Tomcat*
2. deploy *MyWebApp* in *Tomcat*
3. start *Tomcat*
4. publish node address

Node B script:

1. retrieve node A address
2. contact the web application and verify returned values are the expected ones



Multi-node testing – A complex example

Test deployment and set-up of a *Cassandra* cluster using four nodes:

1. *Cassandra* 0 (seeder) node
2. *Cassandra* 1 node
3. DB initialisation node
4. Test Client



The screenshot shows an IDE workspace with a tree view on the left and a 'Test Commands' panel on the right. The tree view shows a workspace named 'My Workspace' containing several test cases: 'cassandra-cluster-test.HEAD', 'cassandra-cluster-client.HEAD', 'cassandra-data-filler.HEAD', 'cassandra-node1.HEAD', and 'cassandra-seeder-node.HEAD'. The 'cassandra-cluster-client.HEAD' test case is expanded to show a 'Default platform' with 'Test Commands', 'Properties', 'Environment', and 'Deployed Software' (apache-cassandra-1.2.1). The 'Test Commands' panel shows a description and three command sections: 'clean:', 'init:', and 'test:'. The 'init:' section contains the command: `etics-get -b NODE0_READY`. The 'test:' section contains the command: `${apache-cassandra.location}/bin/nodetool --host `etics-get -b SEED` ring`. A red 'X' icon is visible next to the 'Test Commands' section in the tree view.

Interoperability Testing

Multi-node testing will be used to support interoperability tests in OCEAN project

- modelling software to test in ETICS
- writing tests to make services interact one each other

Benefit are multiple:

- **automate** execution, run tests remotely
- once the test is defined, it can be executed **multiple times using different combination** of software components available
- early **discovery** bugs that could **break the interoperability**

Compliance Verification

Focusing on OCCI for which some test-suites are already available:

- **OCCI Compliance Tool**
- **DoYouSpeakOcci (DYSO)**



Our goal is to:

- extend/complete these test-suites
- integrate them in ETICS:
 - execution simplified and automated
 - make them available to all OCEAN projects

Conclusion

- OCEAN aims at creating a directory for open-source cloud-oriented projects
- Projects information will be enriched with data extracted by ETICS
- Made possible by ETICS testing capabilities:
 - static analysis plug-ins
 - unit and functional testing plug-ins
 - multi-node testing mechanism to support
 - compliance and interoperability testing

Thank You!



<http://etics.res.eng.it/>



<http://ocean-project.eu/>