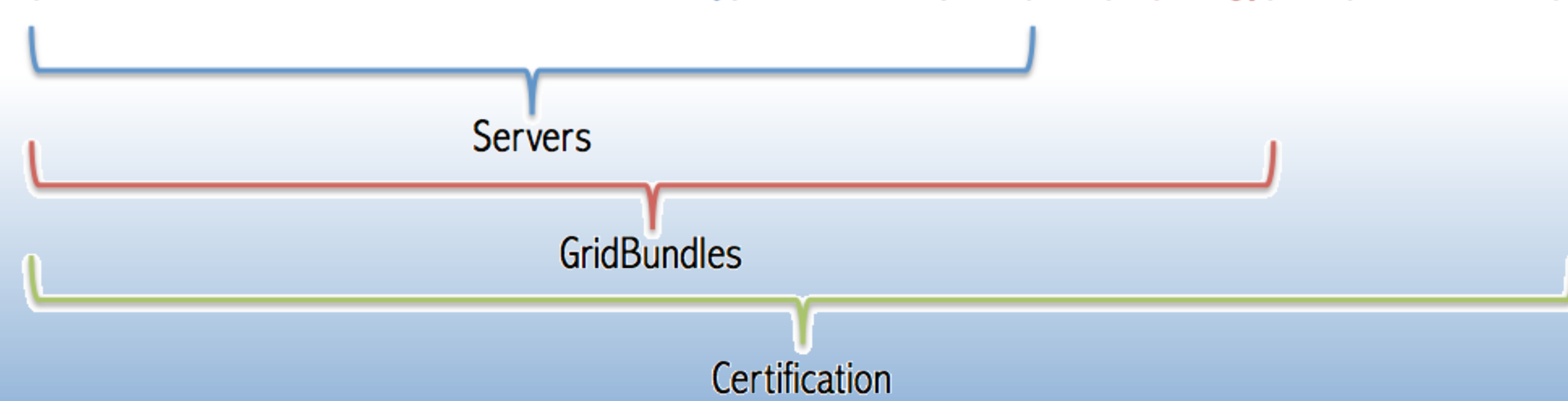


$$(\text{CPU} \times \text{OS} \times \text{iRODS} \times \text{DB} \times \text{ResourceType} \times \text{BuildOption}) \times (\text{Topology}) \times (\text{FeatureTest})$$


Through a replicable automated deployment of server combinations (consisting of cpu type, operating system and version, iRODS version, database type and version, resource type, and build options), grid topologies (both single zone and federation), and feature testing (iget, iput, irepl, etc.), the testing framework behind E-iRODS provides confidence moving forward as iRODS continues to increase its global footprint.

The open source framework is written in bash, Python, and JavaScript and builds on an existing toolset including RabbitMQ, Celery, nose, and node.js.

Future work includes expanding the reach of the server combinations, the feature tests, and the reporting generated by the framework. Additional goals include more virtualization of both the current system as well as the network links within a deployed network (for both disaster recovery and performance testing).

- Hudson launches task on Slave Pool
- Slave Pool runs script to “deploy a Gridbundle”
 - Gridbundle – topological definition of an n-zone iRODS network (json)
- Tests run against the resulting “live grid”
 - Automated and/or manual testing
 - Aggregates test results from various machines

```

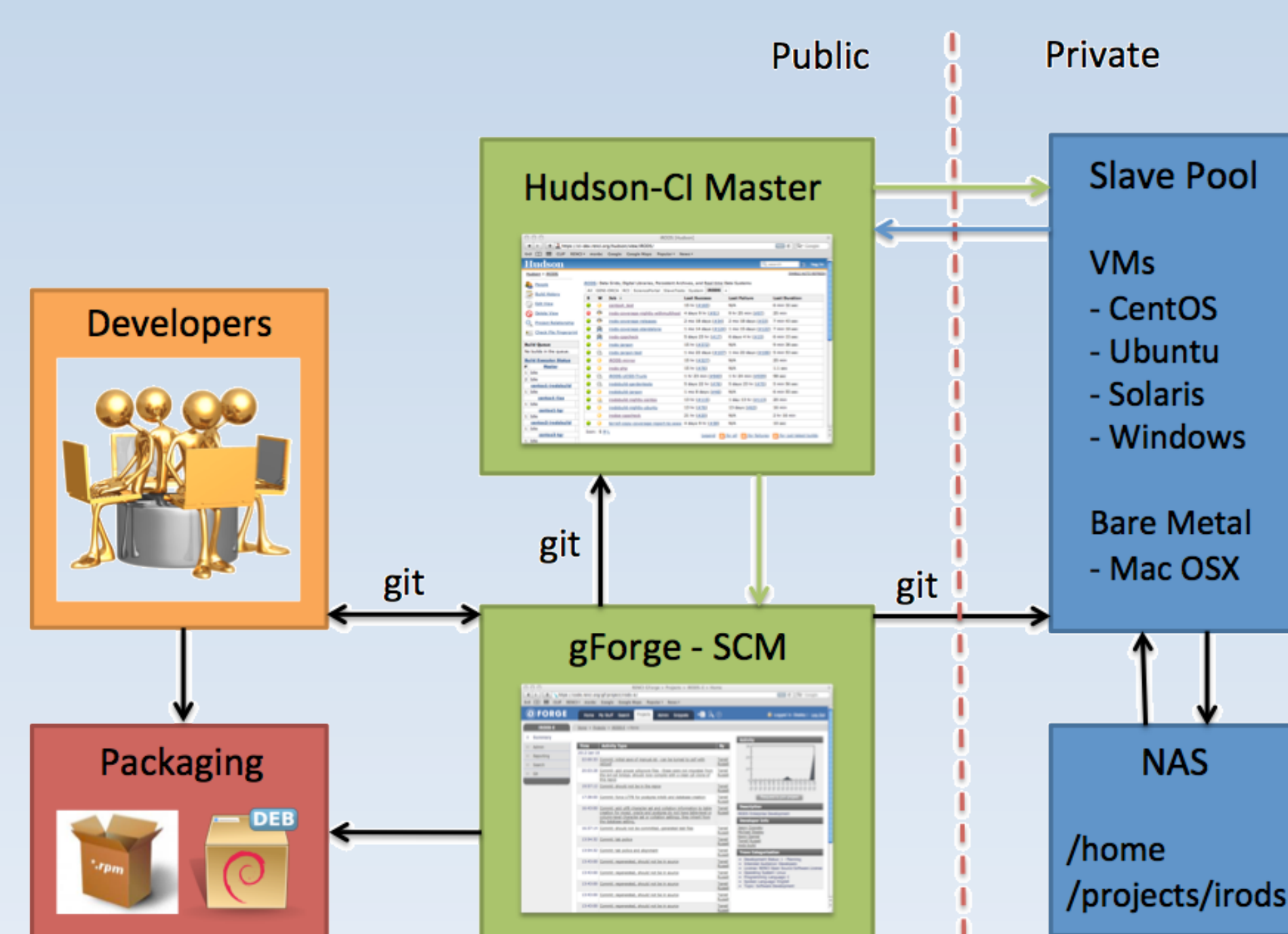
Validate gridbundle is well-formed
Validate testbed capacity is sufficient
For each iCAT server
- Send celery request, wait for success
- Populate gridbundle data structure (IP and hostname)
- For each resource server
  > Send celery request, wait for success
  > Populate gridbundle data structure (IP and hostname)
Write out populated gridbundle to livegrid.json

```

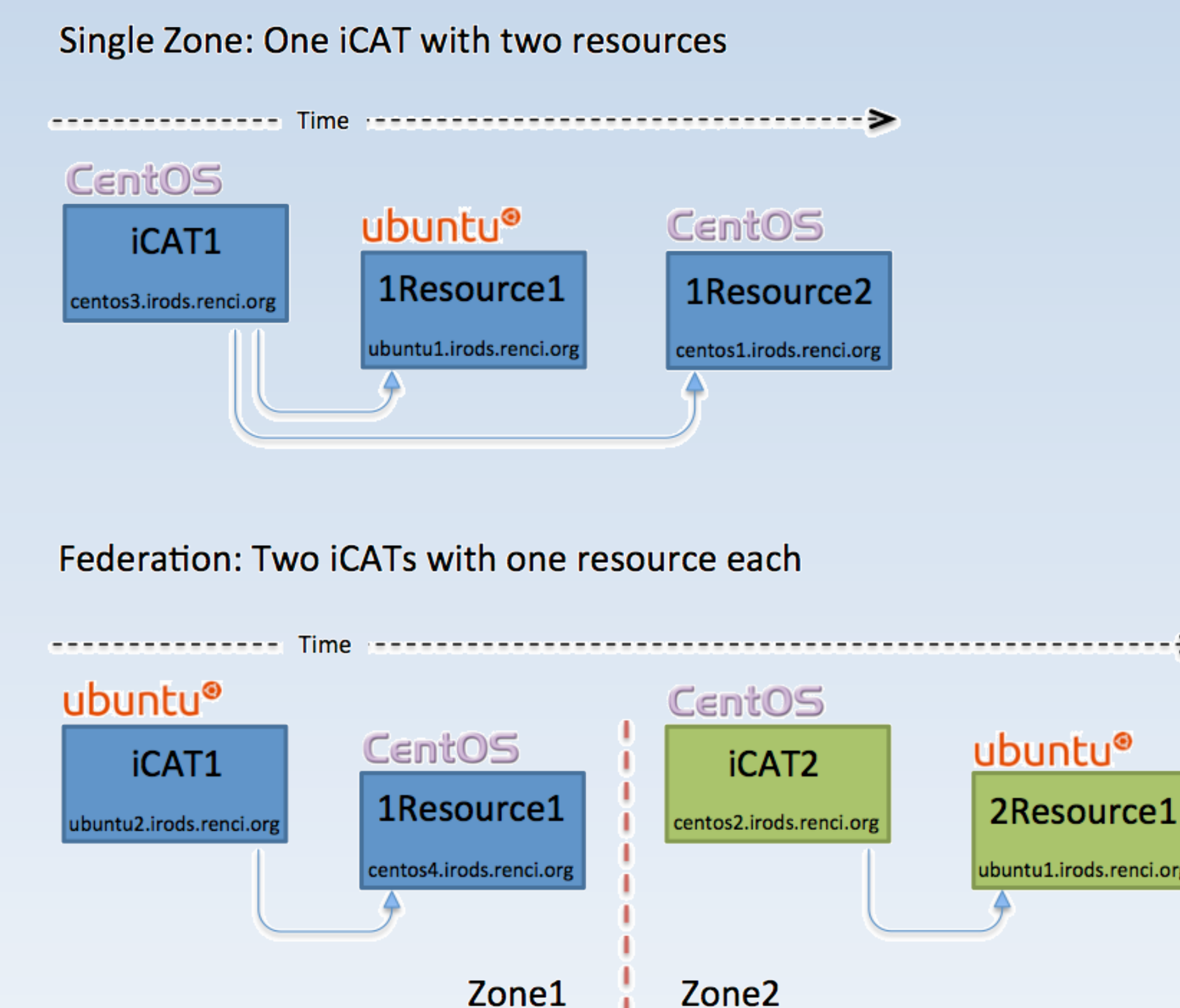
- 100% test coverage of server-side APIs: n-way testing across all combinations of selected platforms and topologies
- Platforms
 - CentOS 5.7
 - Ubuntu 11.04
- Topologies
 - Single Zone: iCAT server + 2 non-iCAT servers
 - Federation: two single zones

Downloadable binaries available at
<http://e-irods.com>
Initial release based on iRODS 3.0
Support contracts available upon
request

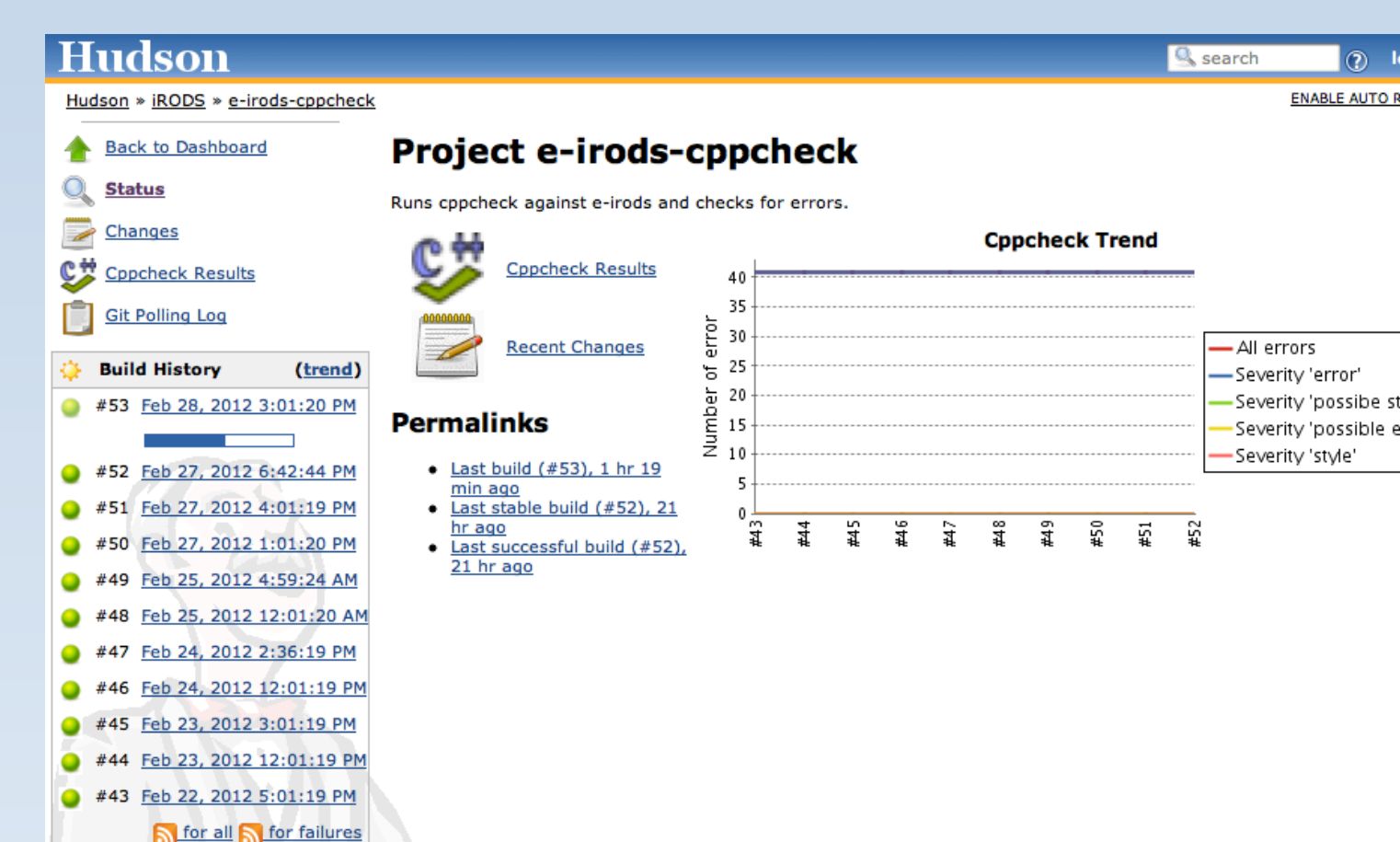
- Contact: Leesa Brieger,
leesa@renci.org



git	<u>Developed at RENCI</u>
python	
- celery	gridbundle
- nose	- schema.json
erlang	- validator.js
- rabbitmq	deploy_gridbundle.py
javascript	assertiCmd/assertiCmdFail
- node.js	
bash	



- Single Zone and Federated
- Resources – cache, compound, DBR (postgres, mysql)
- Ubuntu, CentOS (soon: MacOSX, Solaris, Windows)



LCOV - code coverage report

Current test: lgv_test - clartests/commands/

Test: E-RIIOD Code Coverage

Date: 2012-01-25

Lines: 3003

5634

57.72%

Functions: 81

106

56.60%

Branches: 1280

3678

70.34%

Filename	Line Coverage	Functions %	Branches %	Branches	
lsgb01.c	<div><div></div></div> 75.8%	871 / 753	95.8%	23 / 24	2203 / 200
lsgb02.c	<div><div></div></div> 65.3%	38 / 100.0%	0.0%	0 / 0	10 / 15
lsgb03.c	<div><div></div></div> 87.1%	44 / 77	100.0%	2 / 2	5634 / 14
lsgb04.c	<div><div></div></div> 65.3%	27 / 41	100.0%	2 / 2	11 / 18
lsgb05.c	<div><div></div></div> 80.0%	64 / 78	100.0%	2 / 2	11 / 24
lsgb06.c	<div><div></div></div> 65.3%	30 / 60	100.0%	2 / 2	13 / 22
lsgb07.c	<div><div></div></div> 100.0%	32 / 30	32.0%	1 / 3	12 / 6
lsgb08.c	<div><div></div></div> 15.4%	4 / 264	0.0%	0 / 0	1 / 18
lsgb09.c	<div><div></div></div> 82.5%	19 / 23	100.0%	2 / 2	17 / 6
lsgb10.c	<div><div></div></div> 98.5%	17 / 19	100.0%	0 / 0	6 / 6
lsgb11.c	<div><div></div></div> 45.5%	41 / 68	100.0%	2 / 2	4634 / 20
lsgb12.c	<div><div></div></div> 81.2%	26 / 33	100.0%	0 / 0	7 / 12
lsgb13.c	<div><div></div></div> 72.1%	31 / 43	100.0%	2 / 2	20 / 25
lsgb14.c	<div><div></div></div> 83.3%	30 / 47	100.0%	0 / 0	20 / 29
lsgb15.c	<div><div></div></div> 49.2%	61 / 124	75.0%	3 / 4	421 / 57
lsgb16.c	<div><div></div></div> 79.8%	31 / 33	100.0%	0 / 0	11 / 18
lsgb17.c	<div><div></div></div> 51.9%	130 / 260	75.0%	0 / 0	373 / 204
lsgb18.c	<div><div></div></div> 98.5%	1 / 1	100.0%	0 / 0	10 / 20
lsgb19.c	<div><div></div></div> 65.3%	60 / 86	89.5%	17 / 19	5401 / 2476
lsgb20.c	<div><div></div></div> 79.8%	31 / 43	100.0%	0 / 0	11 / 18
lsgb21.c	<div><div></div></div> 70.8%	27 / 38	100.0%	2 / 2	668 / 11
lsgb22.c	<div><div></div></div> 99.4%	26 / 42	100.0%	0 / 0	13 / 19
lsgb23.c	<div><div></div></div> 76.7%	79 / 103	100.0%	2 / 2	560 / 23
lsgb24.c	<div><div></div></div> 100.0%	14 / 14	100.0%	0 / 0	4 / 6
lsgb25.c	<div><div></div></div> 65.3%	27 / 41	100.0%	2 / 2	61 / 11
lsgb26.c	<div><div></div></div> 96.4%	60 / 61	80.0%	0 / 0	30 / 58
lsgb27.c	<div><div></div></div> 71.4%	30 / 42	100.0%	2 / 2	264 / 13
lsgb28.c	<div><div></div></div> 81.6%	19 / 22	100.0%	0 / 0	6 / 6
lsgb29.c	<div><div></div></div> 91.1%	40 / 60	75.0%	3 / 4	404 / 21
lsgb30.c	<div><div></div></div> 71.4%	40 / 60	75.0%	3 / 4	13 / 29
lsgb31.c	<div><div></div></div> 47.5%	77 / 162	80.0%	0 / 0	385 / 30
lsgb32.c	<div><div></div></div> 94.5%	88 / 121	83.3%	0 / 0	46 / 102
lsgb33.c	<div><div></div></div> 74.5%	11 / 14	33.3%	0 / 0	2 / 104
lsgb34.c	<div><div></div></div> 81.6%	19 / 22	100.0%	0 / 0	6 / 6
lsgb35.c	<div><div></div></div> 60.2%	30 / 46	100.0%	2 / 2	59 / 11
lsgb36.c	<div><div></div></div> 96.4%	26 / 27	100.0%	0 / 0	21 / 18
lsgb37.c	<div><div></div></div> 71.4%	28 / 39	100.0%	2 / 2	61 / 11
lsgb38					

E-IRODS is provided as a binary-only installable package for specific combinations of architecture and operating system. Please find your preferred combination below and click to download.

FILENAME	CHECKSUMS
e-irods-3.0B1-linux-3.0-x86_64.deb	md5 40ee0109f8be207d2c624c8182970a344 sha1 d5a141f71f113d5655a49747829a06127a91973ac

Please confirm any checksum values before installing.

(2010) iRODS Primer: integrated Rule-Oriented Data System (Synthesis Lectures on Information Concepts, Retrieval, and Services)
<http://www.amazon.com/dp/1608453332>

(2011) The integrated Rule-Oriented Data System (iRODS 3.0) Micro-service Workbook
<http://www.amazon.com/dp/1466469129>

E-iRODS Manual <http://e-irods.com>

iRODS Wiki <http://irods.org>