Sustainability: The iRODS Consortium
Economic
Technological

Mary Whitton
Project Manager

Brand Fortner (by telecon)
iRODS Consortium Executive Director, RENCI

Sustainability

- DFC produces open-source technology that facilitates federation of resources
  - Need to grow user and developer base
- DFC technology is built on and with iRODS technology
  - Technical dependence on iRODS for evolution
- The iRODS Consortium is dedicated to iRODS sustainability
  - Economic model for self-sufficiency
  - Engaged users to drive technical advances
Updates from the Consortium

• iRODS 4.0 and Beyond

• An update on iRODS

iRODS 4.0 and Beyond

Dan Bedard
iRODS Consortium, RENCI
Motivation and Goals

iRODS is open source middleware for:
• Data Discovery,
• Workflow Automation,
• Secure Collaboration, and
• Data Virtualization

The iRODS Consortium wants iRODS to be **sustainable**.

That means it has to be:
• Reliable
• Supportable
• Usable
• Scalable

Technical Framework for Sustainable iRODS

Components of a sustainable iRODS:
• Pluggable Architecture
• Configuration Management
• Grid Introspection
• Plugin Dependency Model

A feature set that remains **relevant**.
iRODS 4.0: A First Step

Pluggable Architecture
- Simplifies development, testing, and support
- Broadens the developer community

Makes the long-term relevance of iRODS more likely.

iRODS 4.0 Plugins, Presently

<table>
<thead>
<tr>
<th>Resources:</th>
<th>Authentication:</th>
<th>Network:</th>
<th>Database:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compound</td>
<td>Native</td>
<td>TCP</td>
<td>PostgreSQL</td>
</tr>
<tr>
<td>Deferred</td>
<td>PAM</td>
<td>SSL</td>
<td>Oracle</td>
</tr>
<tr>
<td>Load Balanced</td>
<td>OSAuth</td>
<td></td>
<td>MySQL</td>
</tr>
<tr>
<td>MSO</td>
<td>GSI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Non-Blocking</td>
<td>Kerberos</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Passthru</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Random</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Replication</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Round Robin</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Universal MSS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Unix File System</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WOS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HPSS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S3</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
iRODS 4.0: Continuous Integration

Why Build and Test?
- Transparency (in both process and product)
- Use existing industry best practices
- Coverage - Confidence in Refactoring
- Packaging - Ease of installation and upgrade
- Test framework idempotency
- Test independence
- Topology awareness
- Automation, Automation, Automation

iRODS 4.0 transitioned legacy Perl-based test framework to a Python-based framework using unittest.
- We have increased code coverage from ~20% to ~57%
- Status always visible at http://ci-dev.renci.org/hudson/view/iRODS

The iRODS Ecosystem:
DFC Contributions

The DataNet Federation Consortium is creating national-scale research data federations.

Ongoing iRODS Development:
- Clients and Client Interfaces
  - Jargon and iDrop Web 3
  - Modeshape (plus WebDAV) plugin
- Messaging Interface
  - Integration with Elastic Search
- Metadata Templating and Ontology Discovery

This material is based upon work supported by the National Science Foundation under Cooperative Agreement OCI-0940841.
Update on iRODS

Terrell Russell

iRODS Consortium, RENCI

New iRODS releases

• 4.0.0  March 2014
• 4.0.1  June 2014
• 4.0.2  June 2014
• 4.0.3  August 2014
Release History (since Feb 2014)

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Release Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2014-03-28</td>
<td>4.0.0</td>
<td>Merged Codebase</td>
<td>4th major release of iRODS and first merged open source release from RENCI.</td>
</tr>
<tr>
<td>2014-06-05</td>
<td>4.0.1</td>
<td>1st Point Release</td>
<td>Memory leak fixes, security fixes, --run-in-place, MacOSX support, schema update mechanism.</td>
</tr>
<tr>
<td>2014-06-17</td>
<td>4.0.2</td>
<td>2nd Point Release</td>
<td>Random and RoundRobin resource plugin fix, memory leak fixes, microservice fixes, security fixes, large collection recursive operations, and better server-server authentication setup.</td>
</tr>
<tr>
<td>2014-08-20</td>
<td>4.0.3</td>
<td>3rd Point Release</td>
<td>More flexible installation options (service account name/group), block storage operation fix, impostor resource, memory leak fixes, and security fixes.</td>
</tr>
</tbody>
</table>

Highlights of iRODS 4.x

- Architecture Change
  - Pluggable Core
  - Composable Resources
- Build and Test
  - Binary Packages
  - Code Coverage
  - Static Analysis
  - Continuous Integration
    - Across six(+) supported Operating Systems
**Architecture**

- iRODS Core is a substrate upon which new functionality may be added via seven interfaces. The core is designed to be a small, stable broker of extensible services.
- Interfaces for Extensibility:
  - Authentication, Database, Network, Microservices, Objects, Resources, RPC API
- Plugins extend the functionality of iRODS relevant to a given interface. They are self contained, dynamically loadable, and could be proprietary.
- iRODS will include a plugin dependency model.
  - Plugins may be inter-dependent and provide new functionality via multiple plugins.
- A Bundle of plugins can provide a set of features to support newly created first-class objects within iRODS such as Tickets or Workflows.

---

**Plugins**

- **29 plugins shipping**
  - 1 API
  - 3 authentication
  - 3 database
  - 6 microservice
  - 2 network
  - 14 resource
- **12 supported add-on plugins**
- **1 externally developed storage plugin**
Composable Resources

Resource Plugins
- Tree Metaphor
- Virtualization
- Extensible

Available
- 14 shipping
  - compound, mso, passthru, roundrobin,
  unixfilesystem, deferred, mssofile, random,
  structfile, load_balanced, mockarchive,
  nonblocking, replication, univmss
- 4 additional
  - WOS, S3, directaccess, HPSS

Supporting DFC

Indexing Framework
- Rules and microservices
- Messaging based
- Event driven
- Continued hardening

Upgrading/Supporting DFC installations
- Migrations from 3.x to 4.x
Future enhancements

• Release of new plugins
  – Don’t require a core code release
  – User extensions welcome
    • May be added to testing and “certified”
    • May be tested, certified, and added to shipping code
    • May remain uncertified, distributed independently
• Release 4.1 – API compatible, refactoring
• Release 5.0 – Major new functionality

www.datafed.org
www.irods.org

National Science Foundation Cooperative Agreement: OCI-0940841