iRODS 4.0 and Beyond
Presented at the iRODS & DDN User Group Meeting 2014

September 8, 2014
Motivation and Goals

iRODS is open source middleware for:

- Data Discovery,
- Workflow Automation,
- Secure Collaboration, and
- Data Virtualization

We want it to be sustainable.

That means it has to be:

- Reliable
- Supportable
- Usable
- Scalable
A Framework for Sustainable iRODS

- Consortium
- Messaging
- **Technology**
The Vision for iRODS Administration

• Knowing Your Requirements, Map Out Your Grid:
  – Where does the user interact with the system?
  – Where does the system interact with your storage?
  – What functionality is required from each node?

• Translate Your Map into a Configuration

• Updating the Configuration Updates Your Map

• When You Need Help...
A Technology Framework for 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, support
  - Broadens developer community

Makes long-term relevance more likely.
# iRODS 4.0: Plugins, Presently

## Resources:
- Compound
- Deferred
- Load Balanced
- MSO
- Non-Blocking
- Passthru
- Random
- Replication
- Round Robin
- Universal MSS
- Unix File System
- WOS
- HPSS
- S3

## Authentication:
- Native
- PAM
- OSAuth
- GSI
- Kerberos

## Network:
- TCP
- SSL

## Database:
- PostgreSQL
- Oracle
- MySQL
iRODS 4.0: Look What You Can Do with Composable Resources!

Live storage migration

```
iadmin mkresc $newResc unixfilesystem $newPath
iadmin mkresc replResc replication
iadmin addchildtoresc replResc $demoResc
iadmin addchildtoresc replResc $newResc
iadmin modresc replResc rebalance
iadmin rmchildfromresc replResc $demoResc
iadmin rmchildfromresc replResc $newResc
iadmin rmresc replResc
itrim -rM -N 1 -S $demoResc /$zoneName
iadmin rmresc $demoResc
```
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: Continuous Integration

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

• Currently in Continuous Integration
  – OSes: Ubuntu 10 and 12, CentOS 5 and 6, SuSE 11 and 12
  – Databases: PostgreSQL, MySQL, Oracle
  – Microservice plugins
  – Authentication plugins: Native, PAM, OSAuth
  – Network plugins: TCP, SSL
  – Resource plugins: Compound, Random, Replication, RoundRobin, Passthru, Deferred, Unixfilesystem, UnivMSS, S3, WOS, MockArchive, NonBlocking
Continuous Integration: Coverity

Enterprise code defect analysis tool

- Catches memory leaks, potential attack vectors, and unintended behavior
- Assigns severity
- Not all detected defects are necessarily defective

We have addressed all high severity issues. Evaluating and eliminating remaining defects.

Status available at https://scan.coverity.com/projects/2605
iRODS 4.0.x: Point Releases

• Fixing things...
  – Security issues
  – Memory leaks
  – Bugs

• Implementing useful features...
  – Run-in-place and OSX support
  – Additional options on initial configuration
  – Microservice templates
  – C API support
  – Review and testing of system microservices
iRODS 4.1: More Groundwork

Operating roadmap: https://github.com/irods/irods/issues?q=is%3Aopen+is%3Aissue+milestone%3A4.1.0

• First Hints of Configuration Management and Grid Introspection
  – JSON-based configuration describe a grid
  – Infrastructure to support grid report

• New API Interface
  – Easier implementation, support for more languages

• Specific User Community Requests
  – Data/metadata ingest as an atomic operation
  – Key-value passthrough btw. iCommands and plugins
  – Improved support for file streaming and PEPs

• Additional Useful Features
  – Pluggable parallel transfer
  – Inter-zone metadata copying
  – Making configuration more straightforward
iRODS 4.1: Enabling Grid Report

- Query the Entire Grid (Privileges Determine Level of Detail)
  - What nodes are connected?
  - What storage resources are connected?
  - Are the storage resources alive?
  - What plugins are installed?
  - What iRODS component versions are installed?

- Goes Hand-in-Hand with Configuration Management
  - Grid report can be used to replicate the grid
Beyond iRODS 4.1

Further steps toward the vision:
• Registry to track installed plugins
• Plugin dependency model
• Visual interfaces for configuration and reporting

Additional important features:
• Infrastructure to support metadata templating
• Bugfixes
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.
Beyond Technical: Documentation

By Spring 2015...

Reference implementations:
• Genomics
• Research Library/Archive

System Architecture and Administration Manuals

System Administration Videos

Training and Certification Curriculum
• System Architect
• Datagrid Administrator
• Support Technician
• Developer
Beyond Technical: Consortium Activities

Products:
• Membership
• Professional Services
• Support Services
• Training and Certification
Thank you!

Dan Bedard
iRODS Market Development Manager
danb@renci.org