



# iRODS Status

## Integrated Rule-Oriented Data System

Reagan Moore

Mike Wan

{moore,mwan@diceresearch.org}

Jean-Yves Nief

[nief@cc.in2p3.fr](mailto:nief@cc.in2p3.fr)



# Seeking Feedback

- What is important to the iRODS user community?
- Which features will be most useful (justifying further development)?
- How can user support be improved?
  - Tutorials
  - iRODS reference book
  - User scenarios
  - Reference implementations (policies and procedures)

# DICE (Data Intensive Cyber Environments)



- Center for Advanced Data Intensive Cyber Environments: CA-DICE
  - Institute for Neural Computation
  - University of California, San Diego
    - Mike Wan
    - Wayne Schroeder
    - Lucas Gilbert
    - Bing Zhu
    - Sheau-Yen Chen
    - Paul Tooby
- National Center for Data Intensive Cyber Environments: NC-DICE
  - School of Information and Library Science
  - University of North Carolina at Chapel Hill
    - Reagan Moore
    - Arcot Rajasekar
    - Richard Marciano
    - Antoine de Torcy
    - Chien-Yi Hou



# Data Intensive Cyber-environments Foundation

- Non-profit, open source foundation to promote international use and development of the iRODS technology
  - Modeled after the Fedora Commons and DSpace Foundation
  - Completing formation of the foundation this year
  - Plan to assign copyright to the foundation
  - Plan to promote collaborations on iRODS use and development



## Funding:

- NSF Office of Cyberinfrastructure OCI-0848296 grant, “NARA Transcontinental Persistent Archive Prototype”, (2008-2012).
- NSF Software Development for Cyberinfrastructure SDCI-0721400, "SDCI Data Improvement: Data Grids for Community Driven Applications" (2007-2010).
- Thirteen other proposals/collaborations through NSF, NHPRC, IMLS, NASA, NOAA, DOD, UNC, North Carolina



# iRODS Data Grid

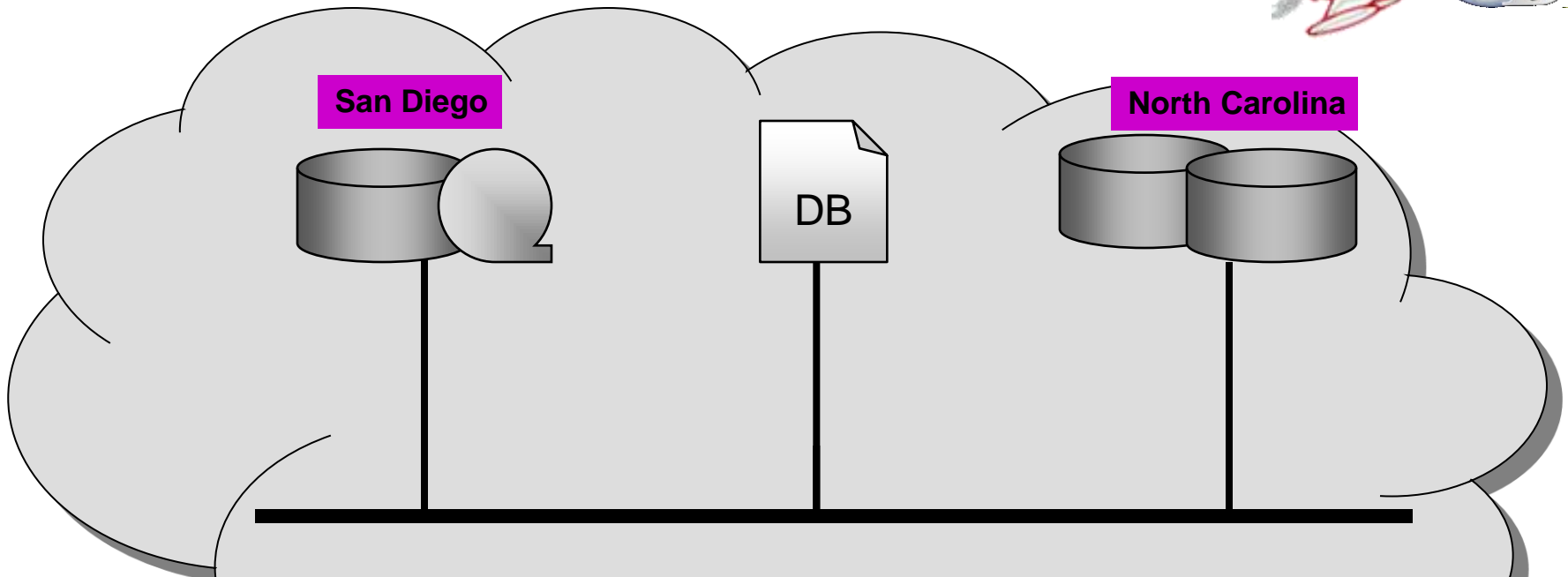
- Organize distributed data into a shared collection
  - Implement mechanisms needed for infrastructure independence
- Automate enforcement of management policies
  - Map policies into computer actionable rules
  - Map procedures into computer executable micro-services
- Automate validation of assessment criteria
  - Query state information and audit trails to verify compliance with policy



## iRODS - integrated Rule-Oriented Data System

<b>Data Management Environment</b>	<b>Control Properties Assessment Criteria</b>	<b>Control Mechanisms Management Policies</b>	<b>Remote Operation Management Processes</b>
<b>Management Functions</b>	<b>Data Grid Management Virtualization</b>	<b>Rules</b>	<b>Metadata</b>
<b>Data Management Infrastructure</b>	<b>Persistent State</b>	<b>Data Grid Bandwidth Virtualization</b>	<b>Metadata</b>
<b>Physical Infrastructure</b>	<b>Database</b>	<b>Rule Engine</b>	<b>Storage System</b>

# Building a Shared Collection



Have collaborators at multiple sites, each with different administration policies, different types of storage systems, and different naming conventions,

**Assemble a self-consistent, persistent distributed shared collection**





# High Level Objectives

- Working with multiple communities to develop examples of:
  - Management policies
    - NARA Electronic Records Archives
  - Assessment criteria
    - ISO MOIMS-rac (Mission Operations and Information Management Services; Repository Audit & Certification)
  - Reference implementation
    - SUN PASIG - Preservation and Archiving Special Interest Group



## Status: High Level Objectives

- Building prototypes through collaborations to test each objective
  - NARA Transcontinental Persistent Archive Prototype
    - Preservation policies and procedures
  - SHAMAN - Sustaining Heritage Access through Multivalent Archiving
    - Integration with parsing and analysis micro-services
  - Fedora/DSpace - storage and policy level integration
    - Carolina Digital Repository
    - French National Laboratory
    - Duke Medical Archives



## iRODS Infrastructure Status

- Release 0.5 Dec 20, 2006
- Release 0.9 June 1, 2007
- Release 0.9.1 June 7, 2007
- Release 0.9.2 June 8, 2007
- Release 0.9.2 (Solaris) June 27, 2008
- Release 1.0 (pre-release) Jan 18, 2008
- Release 1.0 Jan 23, 2008
- Release 1.1 June 27, 2008
- Release 2.0 Dec 2, 2008
- Release 2.0.1 Jan 26, 2009



# iRODS Release 2.0

## (NARA, ARCS, BIRN, Textgrid)

- \* Federation. Zone Federation has been added which provides facilities for two or more independent iRods systems to interact with each other and allow for seamless access of data and metadata across these iRods systems. These systems are called iRODS Zones, with each Zone running its own iCat and administrative domain. iRODS Federation requires less synchronization between zones than the similar SRB Federation system. See Federation and Federation Administration for more information.
- \* Master/Slave iCat with Oracle. An iRods installation or Zone can be configured to run with a single Master iCat plus zero or more Slave iCats. The purpose of the Slave iCat is to improve responsiveness of queries across a Wide-Area-Network. The Slave iCats are used for "read only" type queries. The following icommands have been converted to use the Slave iCat by default: icd, iget, ils, ilsresc and iqstat. This makes use of RDBMS functionality to synchronize the ICAT databases and so is available when using Oracle as the ICAT RDBMS.



# iRODS Release 2.0

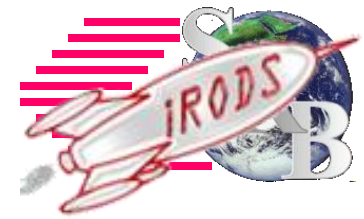
## (NARA, UK e-Science data grid, ARCS)

- \* Initial SRB to iRODS Migration tool. There is now a preliminary version of a tool to help convert an SRB Instance to an iRods one. The 'm2icat.pl' script uses Spullmeta to get SRB-MCAT information and creates and executes sets of commands for iadmin, psql, and imeta to create resources, collections, dataObjects, and users in the iRods instance. The iRODS system can then access former-SRB data without moving the physical files. This is still incomplete and cannot handle many of the features of SRB, but you may wish to experiment with it. See the script for more information and contact us to help us plan additional extensions.
- \* Grid Security Infrastructure (GSI) Improvements. A significant problem in the iRODS interface to GSI was corrected, allowing regular iRODS users to authenticate with GSI. Users can also now set the environment variable SERVER\_DN to authenticate the server via the GSI system (perform mutual authentication).



# iRODS Release 2.0 (OOI, NARA, CDR)

- \* iRods FUSE improvements. iRods FUSE now works with the latest versions of FUSE instead of only version 2.7.0. Caching files and directories query results to improve the performance of the getattr call which is called frequently by FUSE. Small files are cached to improve the I/O performance on small files. File modes of files has been enabled so that chmod of files now works. You should see noticeable performance improvement in commands such as ls or cp of small files.
- \* iRODS Explorer for Windows. The new iRODS Explorer for Windows has been available since mid-September. As described on the windows page, this is an iRODS browser that runs as a native Windows binary and provides a rich Graphical User Interface and a fast navigation of the hierarchical collection-file structure inside iRODS. In addition, users can add, modify, and view metadata with long string values through a user-friendly metadata dialog.



# iRODS Release 2.0 (NARA, TDLC)

- \* DataMode preserved. A "dataMode" metadata item was added so that the file mode of files uploaded to iRods and downloaded from iRods, can be preserved. "dataMode" is the Read-Write-Execute status for user, group, other, of a file.
- \* New bundling. A new 'ibun' command is used to handle the bundling of small files into structured files such as tar files. It can be used for the uploading, downloading, and archival of a large number of small files. For example, to upload a large number of small files, a user can use the normal UNIX tar command to tar these files into a single tar file. This single tar file can then be uploaded to iRods using the iput command. The "ibun -x" command can be used to request the iRods server to untar this file into many small files and register these small files with the iCat. Similarly the "ibun -c" command can be used to efficiently download a large number of small files.



# iRODS Release 2.0 (TDLC, Odum)

- \* New 'ipasswd' Command. 'ipasswd' allows users to change their iRODS password. As with the corresponding iadmin command, the password is obfuscated for network transfer.
- \* Rule-oriented Data Access (RDA) ported to Oracle. RDA is now supported on Oracle as well as the previous PostgreSQL. RDA provides access to arbitrary databases through the iRODS system, somewhat like the SRB DAI (Database Access Interface) but implemented via rules and micro-services.
- \* Other RDA improvements. A msiRdaRollback micro-service was implemented; some memory leaks were fixed; and an obfuscated form of the RDA password can be set in the RDA configuration file.





# iRODS Release 2.0

## (BNF, ARCS, NARA)

- \* Rule-language 'break'. A 'break' statement is now accepted in the iRODS Rule language to break out of for, while, and foreach loops.
- \* Federation User and Administrative Changes. In the iadmin and imeta commands, users are now represented with an optional Zone name (user[#zone]), where the local Zone is default. iadmin has mkzone, modzone, and rmzone subcommands to manipulate remote-zone information. ilsresc, imeta, and irmtrash now have '-z zoneName' options to work with remote Zones. Internally, user authentication uses an optional Zone name.
- \* Federation Server Authentication. A capability was added where the iRODS Server which is authenticating a Client for a remote Zone, is itself authenticated. This is optional, but highly recommended for Federated Zones to improve security.



# iRODS Release 2.0

## (NARA, CineGrid, JPL)

- \* Zone Renaming. The iadmin tool can now be used to rename your local Zone, handling the conversion of the Zone and user information and renaming user home collections (via a new Rule and micro-services). This may be useful with iRODS Zone Federation.
- \* Timed connection -T option. A new -T option was added to the iput, iget, irepl and icp commands which renews the socket connection between the client and server after 10 minutes of connection. This gets around the problem of sockets getting timed out by the firewall as reported by some users.



# iRODS Release 2.0

## (Cinegrid, Teragrid, NARA)

- \* New RDUDP data transfer mode. A new data transfer mode - RBUDP (Reliable Blast UDP) was added, in addition to the existing the sequential (single TCP stream) and parallel (multi TCP streams) modes currently supported by iRODS. RBUDP is developed by Eric He, Jason Leigh, Oliver Yu and Thomas Defanti of U of Ill at Chicago.
- <http://www.evl.uic.edu/cavern/RBUDP/Reliable%20Blast%20UDP.html>
- It uses the UDP protocol for high performance data transfer. A new option -Q has been added to the iput, iget, irepl and icp command to specify the use of the RBUDP protocol.
- \* HDF5/iRods Improvements. The HDF5/iRods client can now be built without linking to the HDF5 library. A JNI capability for JAVA client such a hdfView, was added. A memory leak was fixed.
- \* Inherited Access Permissions. An inheritance attribute can be set on a collection to cause new data-Objects and sub-collections created under it to acquire the access rights (ACLs) of the collection. See 'ichmod -h' and 'ils -A' for more information.



# iRODS Release 2.0

## (Odum, NARA)

- \* ICAT Improvements. The iRODS Metadata Catalog interface software (ICAT) was improved in various ways, in addition to changes for iRODS Zone Federation. Access to the Audit tables via queries is now restricted by default. A bug was fixed dealing with recursively setting access control on replicated data-objects. Problems in getting the totalRowCount (when requested) were resolved (for both Oracle and Postgres). In the General-Query, any number of compound conditions (separated by || or &&) can now be handled (instead of just two). After various errors, the ICAT functions (when using PostgreSQL) will do an automatic rollback to allow subsequent SQL to function. Some ICAT-Oracle memory leaks were found and fixed. imeta and the ICAT AVU queries can now accept multiple conditions separated by 'and' and a single 'or'. 'isysmeta' can now set the data-type of a data-object. For improved long-term maintenance, internal changes were made in the way that the ICAT general queries are structured.



# iRODS Release 2.0 (BNF, NARA, SDCI)

- \* Additional Micro-services. A number of new micro-services have been added to enable new functionality, including some of the features described in these release notes. These can also be used in your own rules and as examples for developing your own micro-services. See Released Micro Services for the current list. Special thanks to Romain Guinot of the French National Libraries for providing some of the new micro-services, both for the core and as a separate module ('guinot').
- \* Testing Improvements. As before, the IRODS development version is continually built and tested on local DICE hosts and occasionally on various platforms at the NMI Build & Test facility. ICAT tests were expanded to cover new ICAT functionality. Some heavy-load tests were developed.



# iRODS Release 2.0 (User communities)

- \* Installation/Control Improvements. A variety of changes were made
- to the iRODS installation and control scripts to handle specific
- error situations; also, finishSetup.pl will now update ~/.odbc.ini
- to include the [PostgreSQL] section that is also stored in the
- .../pgsql/etc/odbc.ini file, to avoid problems on some hosts,
- etc. Also, the Make scripts will now automatically re-link modules
- that have updated source files. And 'iinit' will create the
- ~/.irods directory if it is not present and prompt for and store
- the needed .irodsEnv items if they are missing. The install
- scripts now default to Postgres 8.3.5.



# iRODS Release 2.0

## (NARA, Odum)

- \* New Transfer Logging. When enabled (manually), the transfer operations of get, put, replicate, and rsync are logged. See the comments in rsDataObjClose.c for more information.
- \* imkdir can create parent collection. A -p option was added to the imkdir command to cause it to create parent collections if they don't already exist.
- \* Other Bug Fixes. As always, other bugs have been fixed and many small improvements made; too numerous to describe.



# iRODS Release 2.0.1 (NARA, Users)

- iRODS 2.0.1 is primarily a stability release, as it includes a number
- of bug fixes and smaller improvements over our fairly recent 2.0
- release (December 1). It also includes a new capability for Windows,
- the ability to run Windows servers.
  
- Major Extension
  - \* Storage Resources on Windows Servers Microsoft Windows systems can
  - now be used as a storage resource host. A binary version of
  - Windows servers can be downloaded from the iRODS web site. Or, as
  - with Unix systems, it can be built from the 2.0.1 released
  - source. For installation instructions see the server section of
  - the windows page on the iRODS web site. You will need to install
  - the 2.0.1 version of the ICAT-Enabled Server too.





# iRODS Release 2.0.1

## (Users)

- Improvements
  - \* iput to replica behavior improved. The behavior iput has been changed, in the case where a copy of the file exists but not on the iput target resource. Now, the file is uploaded into the target resource and registered as a replica. Previously, one of the existing physical copies of the file was overwritten even though it was not stored in the target resource.
  - \* Processing rule for 'ireg' added. A rule has been added, "acPostProcForFilePathReg", which is executed after the registration of a file path (ireg). Admins can update this rule for site-specific post-processing. Default is a no-op.



# iRODS Release 2.0.1

## (Users, ARCS, IN2P3)

- \* Avoiding postgresQL errors in log. irodssetup will now add an option in a configuration file ("Ksqa=0" in the ~/.odbc.ini) to avoid a warning error (ERROR: unrecognized configuration parameter "ksqa", STATEMENT: set ksqa to 'ON') in the postgres log (pgsql.log) with each connection. Without this, pgsql.log could grow quite large on busy iRODS systems. Thanks to Graham Jenkins of the Australian Research Collaboration Service for this.
- \* GSI libraries added. Makefiles were updated to include two additional libraries needed for certain versions of GSI. These are globus\_callout\_\$(GSI\_INSTALL\_TYPE) and ltdl\_\$(GSI\_INSTALL\_TYPE). This fix was provided by Jean-Yves Nief of IN2P3, France.



# iRODS Release 2.0.1 (ARCS)

- \* GSI irodsServerDn added. An iRODS variable, irodsServerDn can be defined to specify the server's GSI Distinguished Name (DN). When irodsServerDn, or SERVER\_DN, is defined the client will authenticate the server (making sure it's talking to the real server). irodsServerDn takes precedence over SERVER\_DN. Like others iRODS variables, irodsServerDn can be defined in the user .irodsEnv file or as an environment variable.
- \* GSI authentication can be without irodsUserName. Clients can now connect to iRODS and authenticate with GSI without first setting the irodsUserName. If the client's irodsUserName is blank, the server, during the authentication sequence, will check the iCAT for users matching the authenticated DN. If one, and only one, irods user matches, the current session will be established as that user.



# iRODS Release 2.0.1 (ARCS)

- \* Additional GSI DN-processing configurations. If, as above, no irodsUserName is provided during GSI authentication, but if no matching DN is found, a rule, 'acGetUserByDN', will be executed. By default this is a no-op but the rule can be configured to run a command (or Micro-service). If that command returns an iRODS user name in stdout, that user name will be used for the session. If not, a check will be made again in the ICAT. So the external command can either return an iRODS user name to use, or it can create the new user itself (via iadmin). This is useful for Virtual Organizations. See the acGetUserByDN section in core.irb for an example. This was developed in collaboration with Shunde Zhang and Graham Jenkins of the Australian Research Collaboration Services (ARCS).
- \* remoteExec port number. A port number can now be included on the remoteExec call.



# iRODS Release 2.0.1

## (Users)

- Bug fixes
  - \* ils bug fix. A bug in ils was fixed where with no command-line options, it would show the same file multiple times if the file had multiple copies (replica).
  - \* irsync memory leak fixed. An a irsync memory leak was fixed which could cause the process to run out of memory and died after synchronizing a few hundred files.
  - \* ibun -x bug fixed for large numbers of files. A bug was fixed where 'ibun -x' would fail when the number of files in the tar file was larger than 500 (MAX\_SQL\_ROWS).



# iRODS Release 2.0.1

## (Users, ARCS, NARA)

- \* iRODS Fuse deadlocks fixed. A couple of deadlock problems were fixed in the iRods Fuse interface where the iRODS connection lock was not being released. Now users can use the normal Unix file browser ("finder" for mac) to browse files stored in iRODS.
- \* GSI segfault bug fix. A bug was fixed in the iRODS-GSI interface code where, when using GSI authentication and the environment variable SERVER\_DN was not defined, the i-command would die (segfault).
- \* m2icat.pl bug fix (SRB to IRODS migration tool). A problem was fixed in handling some versions of th SRB MCAT. Logic was added to parse and use the header in the Spullmeta-data log file to get the indexes of the various fields, since they can vary.



# Planned Extensions

- 1) Enhance the iRods infrastructure to support production use of rules and client workflows. We need to make it more robust and easy to use.
  - Multi-tasking the batch server (irodsReServer) so that multiple jobs can run concurrently. One long running job cannot hold up other jobs.
  - Job scheduling (much later)
  - Job restart capability (this will be hard)
  - Continue development of the rule engine and the rule programming language
  - Add more helper routines to make it easier to write micro-services.
  - Restructure the procedures for adding micro-services to make it easier to add micro-service modules.
- 2) Build a comprehensive set of rules and micro-services for digital preservation



## Planned Demonstrations

- Sun PASIG reference implementation
  - DSpace, Fedora, iRODS
- SHAMAN preservation platform
  - Cheshire, media micro-services, iRODS
- Medical patient record management
  - UNC
- NARA Transcontinental Persistent Archive Prototype





## iRODS Extensions

- Capabilities are now being driven and implemented by the user community:
  - GSI integration
  - Shibboleth integration
  - HPSS drivers
  - WebDAV interface
  - Python client libraries
  - Cheshire and media micro-services
  - mySQL port



## Tutorial (Tuesday and Wednesday)

- Possible topics
  - Elementary use of iRODS data grid
  - Reference manual components
  - Using structured information interface
  - Transport mechanisms
  - Example rules
  - Writing a micro-service
  - Windows iRODS Explorer Browser



# Information

- Wiki
  - <http://irods.diceresearch.org>
- iRODS Chat
  - <http://groups.google.com/group/iROD-Chat>
- E-mail
  - irods @ irods.org
- Bugzilla
  - <https://www.irods.org/bugzilla/index.cgi>