jargon.update();

iRODS User Meeting 2013
Garching, Germany
Mike Conway - DICE (www.irods.org)
 michael_conway@unc.edu
Let’s talk about client development.

‘Client’ includes API for both Java and PHP, and also includes interfaces for the desktop and webtop, as well as mobile devices.
Desired Topics?
Items

- Current State
- Strategy
- Release Highlights
  - Jargon 3.0 and iDrop 2.0
- Roadmap
  - Jargon
  - iDrop
- PHP
Current State

- Jargon 3.3 is released and recommended for all versions of iRODS.
- iDrop 2.0 is released and recommended for all versions of iRODS.
- Jargon ‘classic’ (Jargon Trunk) is at EOL and will only get bug fixes.
- PRODS/PHP API is alive and kicking and getting community attention.
Jargen Core

- Jargen is the current client API. Jargon-core is the main library
  - Low level implementation of iRODS protocol, which you should never see
  - ‘Public’ API with POJO’s in and out that represents the ICAT domain model as well as core iRODS services
Jargon Core in the stack

Jargon is a driver for the grid, and should intuitively present iRODS services across federations.
Jargon services above core

Jargon-core is used to develop higher level services. Keep the core as small as possible!

Everything above core can be tested by using mocks!
iDROP

iDrop is a suite of interfaces and web API that provide a supported, out-of-the-box set of clients.
Release!

- Jargon 3.3.0 Feature Release
  - Lots of features and fixes to support iDrop Web2
- iDrop Suite 2.0 (Phase I)
  - Swing GUI redesign (with iPlant) Phase I
- Web interface redesign
  - Twitter Bootstrap
Projects and Code

http://code.renci.org

Codebase available via GIT and Maven
Demos of iDrop
Jargon Core Roadmap

- 3.3.1 tune up release in 45 days
  - Tune ups for ‘listing’ methods needed
- 3.3.2 feature release 90 days
  - Mounted collections
  - Workflows and MSOs
  - Kerberos
  - Expanded GenQuery support
- 3.3.3 Optimization and Refactoring only release
  - JMeter automated benchmarks against eIRODS grid
iDrop Roadmap

- iDrop Swing GUI Phase II
- Grid Memory
- New ‘transfer engine’ with restructured queue manager functions and better management of restarts
- Client side rule engine
iDrop Keystore

- “Keystore” functionality can remember all grids accessed, and can pre-seed grids with deployment.
iDrop Transfer Database

- Adjust local database
- Transfer
  - Transfer Attempt
  - Transfer File
Client Side Action Engine

- Transfer Callback mechanism exists to provide hooks
  - Before transfer
    - Before file
    - Within file
    - After file
  - After transfer
Client Side Action Engine

- Attach handlers to events by host/zone/action
  - a.host.name:zone:put
  - another.host.name:zone:put
- Handlers implement interface that
  - injects context
  - provides condition hook
  - provides execution hook
iDrop new components

- iDrop new components
- iDrop-REST - basic put, get, metadata, rule functions for files and collections. This has been prototyped using Spring REST services
- iDrop-Mobile - client of REST API. We’ve been evaluating PhoneGap for this effort.
PHP and PRODS

- PHP is not going away, getting community support
  - Tickets
  - PAM
- PHP efforts need to focus on the SUBSET of capabilities required for PRODS and eventual Drupal/Wordpress module support
iRODS is a platform

- The challenge is to view iRODS as a platform for application development
- Grid
- Catalog
- Metadata
- Policy
Future Imperatives

- Going forward, these are strategic considerations
Client feature development needs to be more closely linked to server feature plans.
A server feature is not a feature unless clients can access it.
Client communication patterns are changing - one connection per operation giving way to multiple operations per connection
Discovery and Introspection of heterogeneous capabilities on a grid is required
New classes of metadata?
Client operations need to migrate to server
Migration to Protobuf? Per invocation protocol specification
Hot-spot detection and migration/resolution. Evolution of new API on server, via Protobuf?
Efficient transfer of row/column data a-la TDS?