2013 iRODS User Group Meeting - Garching, Germany

Requests for iRODS Features	**Priority**
Access full server capabilities from clients - use cases	20
Data encryption at client side	20
Improved connection pooling to handle client overload	20
Generic micro-services	17
How to archive system information	17
Scalability for large number of communities	17
Testing of EiRODS at scale (petabytes) for composable resources	17
Provide support for WebDavis and Griffin	16
Common protocol for Java, Python, PHP to simplify bindings	15
Rebalancing across composable resource (micro-service)	15
Set lifetime on .irodsA similar to GSI proxy certificate	15
Sychronization of rules across federation of data grids	15
Update web site to remove old information	15
Sandbox plugins or validate plugins	14
Manage client communication (connection pools)	13
Certificate attributes available to rule engine and micro-services	12
File names with extended character sets - IN2P3 tweaks	12
Publish policies - RDA mediawiki	12
Stage command for manipulating composable resources	11
IPv6	10
S3 driver - support for >5GB files, openstack driver	10
Support for search on metadata within iDrop	10
Discover heterogeneous capabilities in grid from client	8
iRODS Debian and RPM packaging (Distributed Bio)	8
Add Cheshire indexing	7
S3 interface to iRODS	7
Application of rules to users from another data grid (dynamically load rules)	5
Integrate Bit Torrent with iRODS	5
PyRODS distribution with iRODS, combined install	5
Description of Shibboleth integration with iDrop-web	4
Efficient transfer of row/column data	4
Push client operations to the server	4
Remotely control an iDrop transfer	4
Hot-spot detection and resolution on client	3
Shibboleth support within the core	3
Populate triple store	2
Protobuf for per invocation protocol specification	2
JSR portlet support in iDrop for workflows	1
How migrate composable resources to iRODS	-
SRB migration, need Scommand-icommand emulator	-
Stop a running agent - versus cancel	-