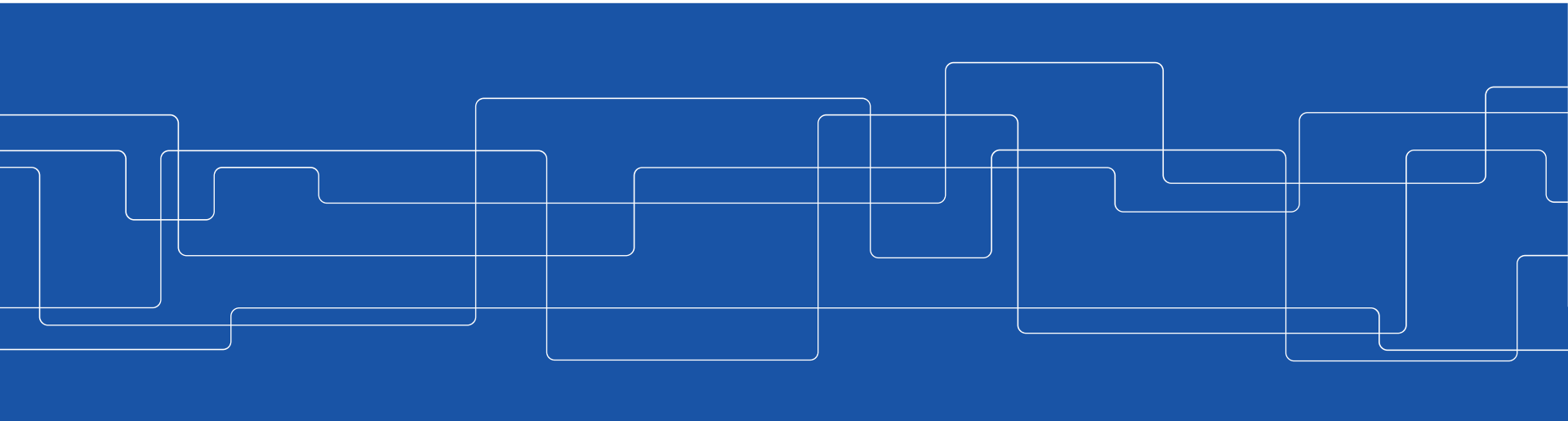
KTH ROYAL INSTITUTE
OF TECHNOLOGY

Swedish National Storage Infrastructure for Academic Research with iRODS

Ilari Korhonen
KTH PDC Center for High Performance Computing
iRODS UGM 2017
June 15th, 2017



SNIC iRODS as a Distributed Storage System

- We operate the core infrastructure from two SNIC centers, KTH PDC and Linköping NSC.
- Core infrastructure:
  - PostgreSQL Database
  - iRODS Catalog Services (iCAT)
  - Identity Management (FreeIPA)
- At the beginning the two centers will host all the storage resources as well, later more centers might be involved.

Redundant (Geo) Replicated iCAT

- We use PostgreSQL streaming replication to maintain two replicas of the iCAT database, one at NSC and one at PDC.
- Due to latency, we have to resort to asynchronous replication, meaning that in case of a catastrophic failure, the very last transaction(s) could be lost and manual repair required.
- Originally we attempted to use bidirectional replication with PostgreSQL but this turned out to be very problematic.
- Our current approach is a simple and elegant solution with not very many moving parts.

Redundant (Geo) Replicated iCAT

- Due to the geographically distributed nature of the database replication, the usual HA with heartbeats etc. will not work.
- We use a secondary DNS zone rods.swestore.se to manage the pointer to the master node (short TTL = failover time).

```
irods-login.swestore.se. 3599 IN CNAME    i.rods.swestore.se.

i.rods.swestore.se. 299 IN  A     130.236.100.88
```

- We have a set of scripts which update the DNS, promote the slave PostgreSQL server into master and start the warm standby iCAT server.

Distributed Storage with SNIC iRODS

- We use (geo) replication as a safeguard against data loss and to provide high availability.
- All data objects in SNIC iRODS are required to exist in at least 2 replicas at different centers.
- The simplest scenario: 2 copies on disk resources at different centers (we will begin production use with this).
- Later: one copy on disk + one copy on tape, 2 copies on different tape libraries of two centers, …
- Currently we use iRODS (synchronous) replication of objects.

Distributed Storage with SNIC iRODS

```
$ ilsresc SNICDisk

SNICDisk:passthru

└── Replication:replication

    ├── NSC:random

    |   ├── nscR01

    |   ├── nscR02

    |   └── nscR03

    └── PDC:random

        ├── pdcR01

        └── pdcR02
```

Tape Library Access with IBM Spectrum Protect (TSM) API

- At the SNIC centers we rely heavily on the IBM Spectrum Protect, previously known as Tivoli Storage Manager for managing our tape libraries.
- For this, a member of our team at NSC Janos Nagy, wrote an interface for iRODS against the TSM API, which enables the use of TSM native `univmss` class compound resources in iRODS.
- This work has been released as open source with the Simplified BSD License.
- Source available at: https://github.com/KTH-PDC/irods-dsmarc

Distributed Storage with SNIC iRODS

```
$ ilsresc t01
t01:compound
├── a01:univmss
└── c01
$ ilsresc t02
t02:compound
├── a02:univmss
└── c02
```

User Authentication and Authorization

- We chose FreeIPA for the Identity Management (IdM) solution, since it is based on standard technologies and provides a robust Kerberized platform with an API.
- Our iRODS servers authenticate against FreeIPA Kerberos, both via PAM (`pam_krb5`) and iRODS KRB Auth.
- This way users can leverage single sign-on via Kerberos and regular password authentication against the same backend.
- We also evaluated GSI authentication for iRODS, but since the Globus Toolkit support is ending, we will reconsider on it.

User Authentication and Authorization

- For PAM authentication we (Janos) also developed an alternate PAM executable for iRODS, which has useful debugging features.
- We have been also using Ubikeys for authentication successfully with iRODS. We will begin production use with only Kerberos.
- Also we are investigating into expanding the feature set of the PAM authenticator, by enabling different auth methods for different users, specified at LDAP.
- Published with the Simplified BSD License.
- Source code available at: https://github.com/KTH-PDC/irods-pamauth

Integration with SNIC Services

- SNIC has processes for approval of new projects, which is managed via SUPR (SNIC User and Project Repository).
- We integrated SUPR into our choice of Identity Management (IdM) solution – FreeIPA, to create new user accounts from SUPR approvals.
- More over, the SUPR integration also deploys the user accounts and project groups into iRODS.
- This module was written in Python against the Python iRODS Client library by Krishnaveni Chitrapu at NSC.
- Later on we will build a self service portal for users to request an account, and set their initial password via external authentication.
- In the first stage we will send out initial (random) passwords to new users.

Federated Access to Other iRODS Grids

- We federate against the Swedish EUDAT zone (also operated by KTH PDC) to enable distinct users for additional resources.
- In addition, we are building a local iRODS grid for KTH PDC, which is intended as a landing zone for HPC users of PDC.
- Thus we are in a position to deploy an iRODS path from local parallel filesystems at HPC resources to national to European resources.
- We hope that the PDC iRODS could serve as a reference model for other Swedish HPC centers willing to build a local zone.

User Interfaces

- We recommend iRODS native clients to our users:
  - iCommands
  - Cyberduck
  - Kanki
- iRODS environment module to be available at compute clusters
- We offer a WebDAV gateway (Davrods) at both NSC and PDC
- Also, we offer an EMC MetaLnx Web UI

Provisioning from GitHub

- Since we have a distributed operation, we use GitHub to host our repositories. That way we can also interact with the iRODS community and share.
- https://github.com/KTH-PDC
- We are consolidating our configurations into an Ansible package, published with the BSD License, available at: https://github.com/KTH-PDC/irods-provisioner
- This enables us (and you) to spin up (virtual) iRODS grids for testing.

KTH PDC Local iRODS Grid (zone: pdc.kth.se)

- For the local PDC iRODS grid we set out different goals than for the national grid. Performance is the primary goal.
- iCAT performance: 10,500 PostgreSQL transactions/s
- We are aiming at high performance data transfers from the local InfiniBand fabric at our pre/postprocessing cluster (EDR, 100 Gbit/s).
- For this we need an equally fast storage solution for the iRODS resource servers and backing filesystems.
- Our (PoC) solution for resources: an InfiniBand SAN hosting multiple ZFS pools accessed via multipathing over redundant IB links and fabrics.
- Performance so far, ZFS read : ~ 6,200 MB/s (avg), ZFS write: ~ 5,500 MB/s (avg), iput -N 64 w/ 100 GbE: ~2,700 MB/s avg, ~ 3,300 MB/s max

Our testbed at PDC

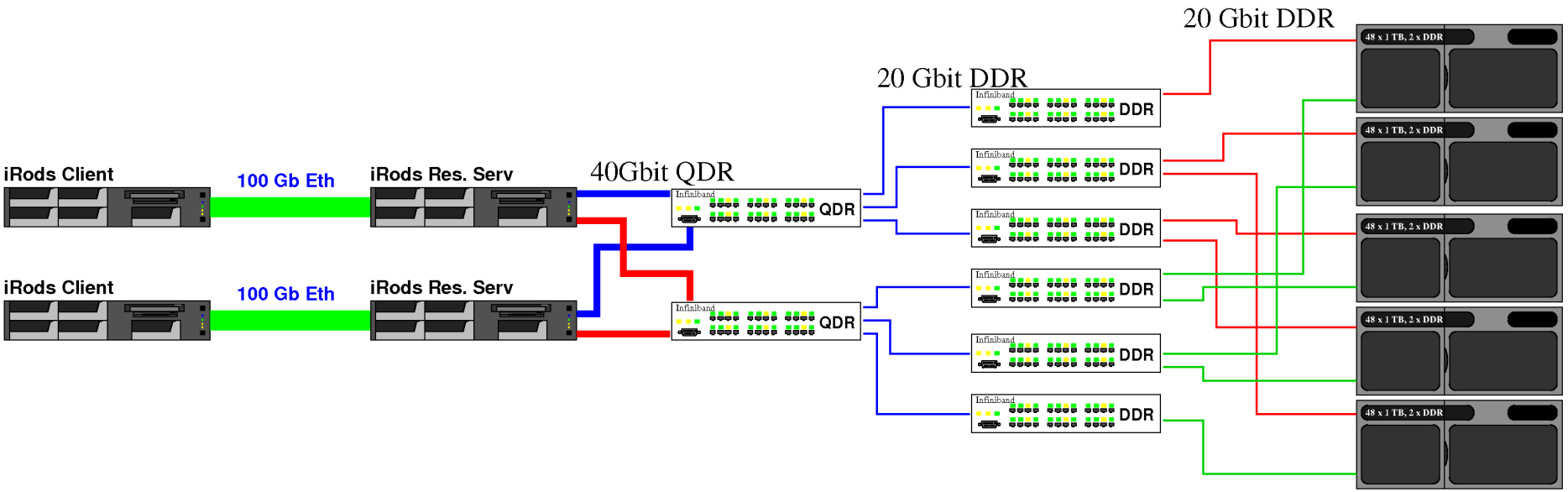
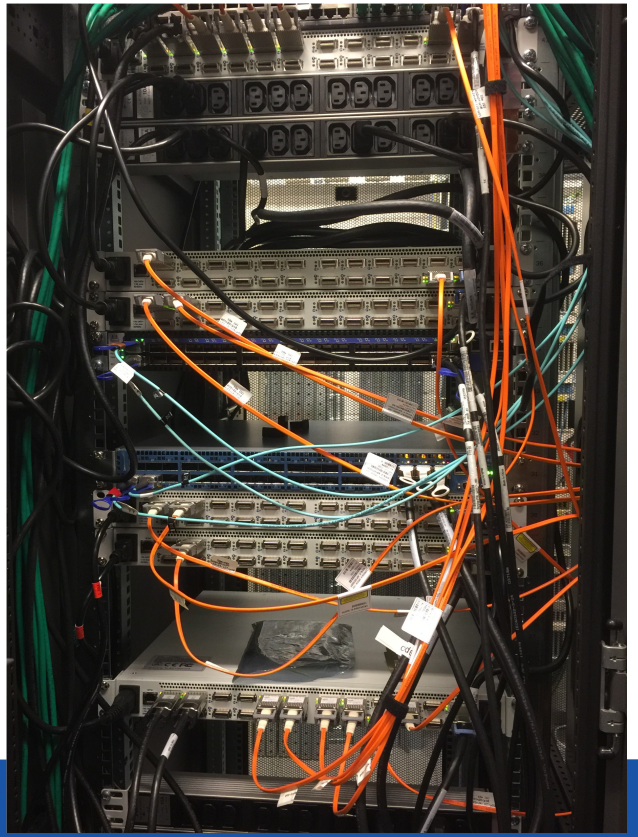


Diagram by Ilker Manap

Having fun with our PoC



Thank You !

- I would like to acknowledge the work of my colleagues, who have contributed greatly to our presentation and upcoming iRODS UGM 2017 conference paper:
  - Janos Nagy, NSC
  - Krishnaveni Chitrapu, NSC
  - Dejan Vitlacil, PDC
  - Ilker Manap, PDC

Thank You !

- And last but not the least, thank you all for listening!
- Questions?
- Contact information:
  - Ilari Korhonen, email: ilarik@kth.se
  - Dejan Vitlacil, email: vitlacil@kth.se
  - Janos Nagy, email: fconagy@nsc.liu.se
  - Krishnaveni Chitrapu, email: krishnaveni@nsc.liu.se
  - Ilker Manap, email: manap@kth.se