Safe Data Replication Service
Technical Insights and Demonstrations

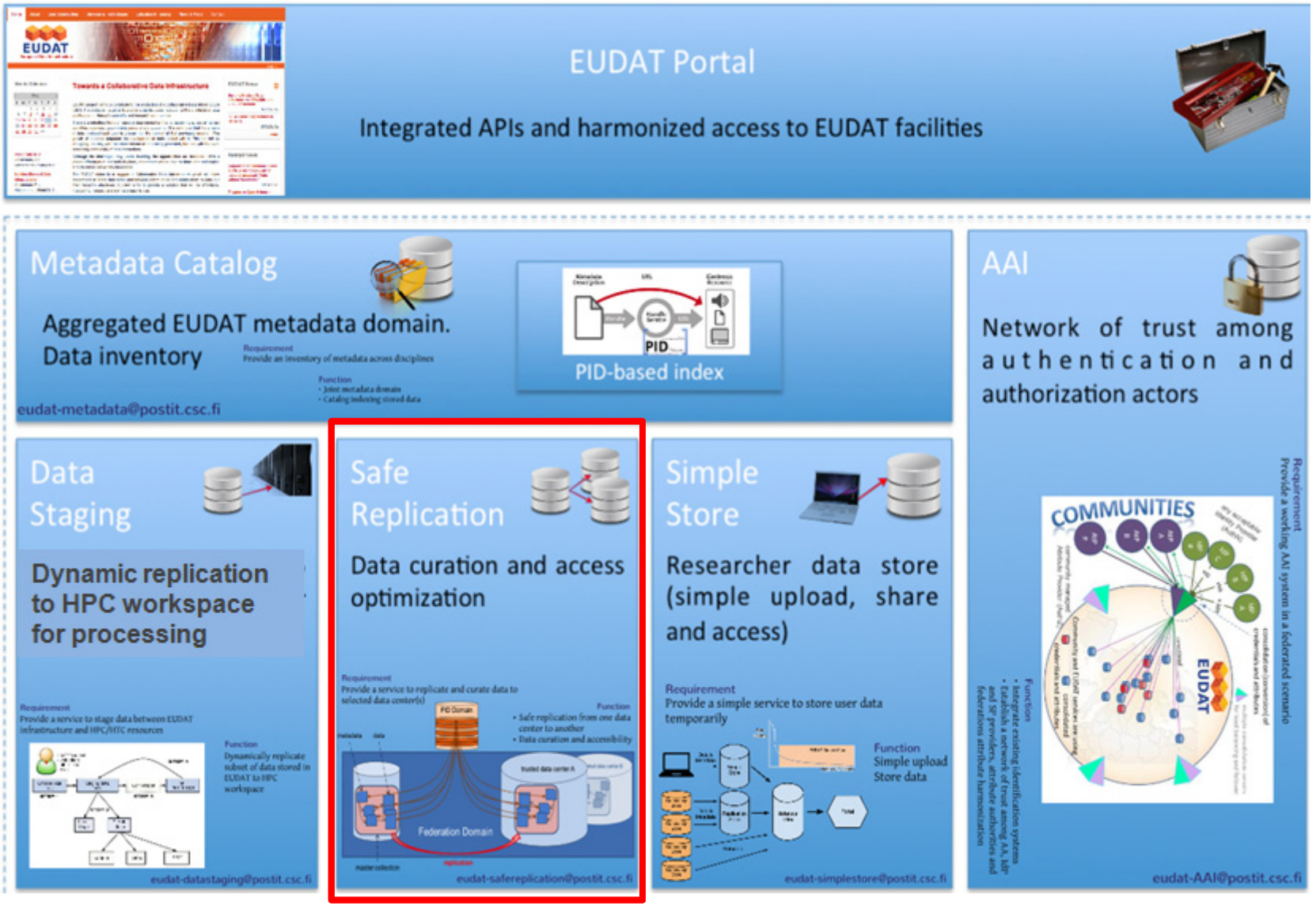
How you and your community benefit from a simple and reliable EUDAT service

Morris Riedel* and Willem Elbers' et al.
*Juelich Supercomputing Centre
' MPI for Psycholinguistic Nijmegen
EUDAT Conference, Barcelona

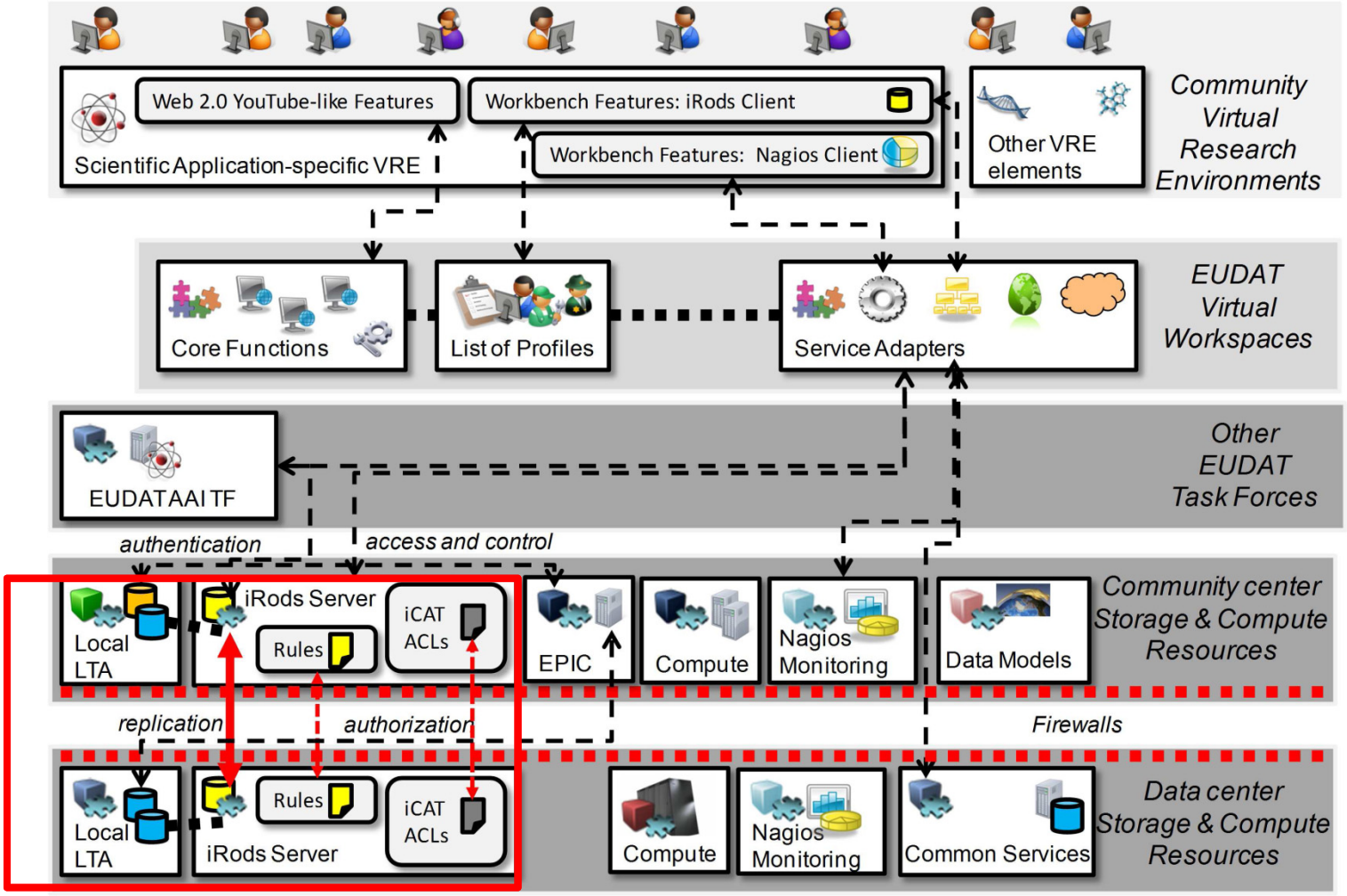
SEVENTH FRAMEWORK PROGRAMME

EUDAT

Date: 23th October 2012


EUDAT Portal
Integrated APIs and harmonized access to EUDAT facilities
Metadata Catalog
Aggregated EUDAT metadata domain.
Data inventory
eudat-metadata@postit.csc.fi
PID
PID-based index
AAI
Network of trust among authentication and authorization actors
COMMUNITIES
EUDAT
eudat-AAI@postit.csc.fi
Data Staging
Dynamic replication to HPC workspace for processing
eudat-datastaging@postit.csc.fi
Safe Replication
Data curation and access optimization
Federation Domain
eudat-safereplication@postit.csc.fi
Simple Store
Researcher data store (simple upload, share and access)
eudat-simplestore@postit.csc.fi

Building Blocks for Your Data Replication

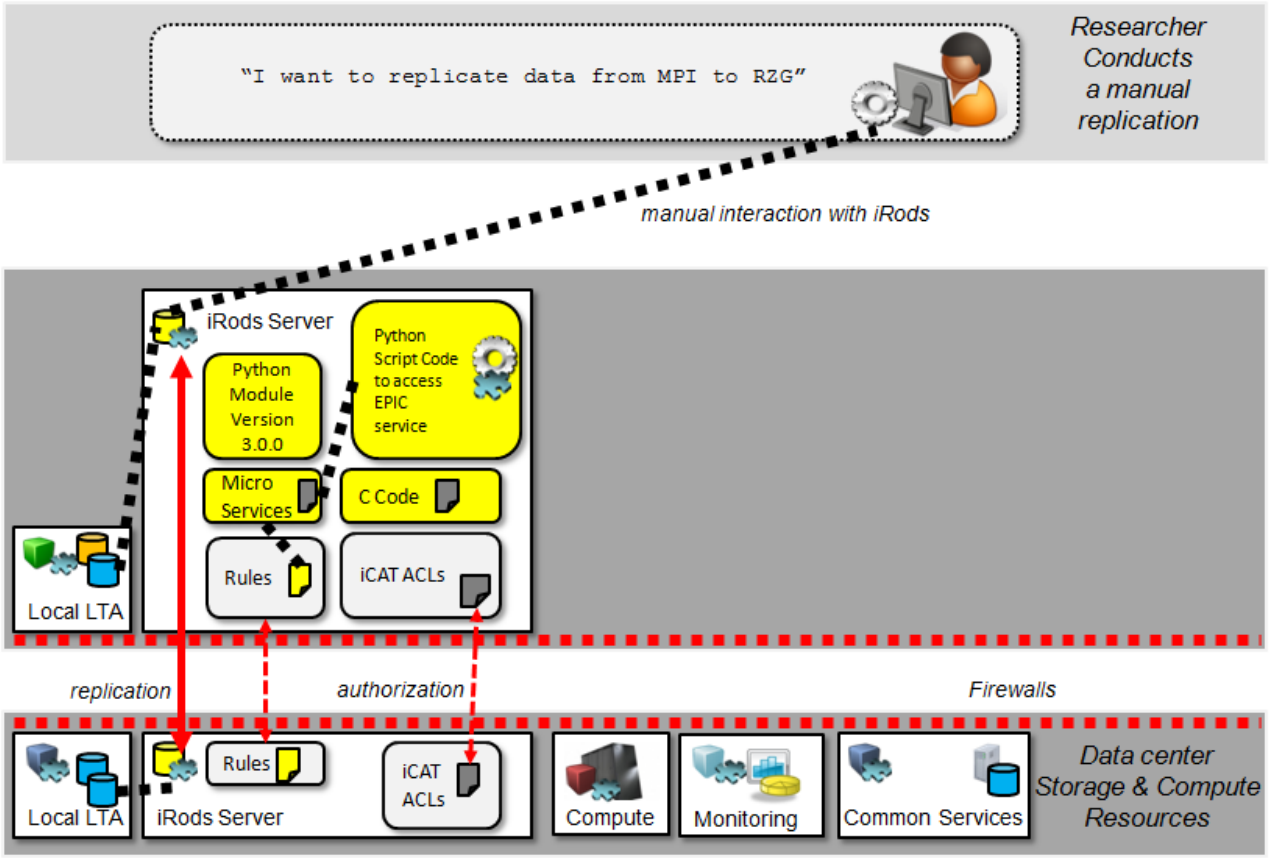


Why your community needs replication?

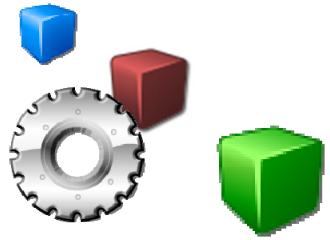
- Common Problem:
  - Scientific Dataset and data models need to be preserved
- Selected Data Replication Needs
  - Better accessibility of scientific data
  - High degrees of reliability and trust
  - More optimal data curation
  - Make data referencable
- EUDAT Safe Replication Service
  - Integrated set of ‘core building blocks’ to satisfy needs

Overview: Manual Upload Replicated File

- Need to understand federations and zones in iRods



Demo: Manually Upload Replication File



- Willem Elbers (MPI-PL, Nijmegen)
  - Show acPostProcForPut in core.re
  - Show collection in source zone (resource is there)
  - Show shared collection (where replication file will be uploaded)
  - Show destination collection (resource is not there)
  - Show content of replication file
  - Using command-line iput replication file
  - Show destination collection (resource is there)

Demo Slides

```
acPostProcForPut {
        ON($objPath like "\*.replicate") {
                processReplicationCommandFile($objPath);
        }
}
acPostProcForPut {
        ON($objPath like "\*.pid.create") {
                processPIDCommandFile($objPath);
        }
}
```

Demo Slides

```
corpman@lux115:/srv/irods/current> ils -l /vzMPI/bin
/vzMPI/bin:
  latuser           0 mpi-vault1                 22296 2012-10-22.12:50 & ladfc11a_1.imdi
  latuser           0 mpi-vault1                 22377 2012-10-22.12:50 & ladfc12a_1.imdi
  latuser           0 mpi-vault1                 21550 2012-10-22.12:50 & ladfc13c_1.imdi
corpman@lux115:/srv/irods/current>
```

```
corpman@lux115:/srv/irods/current> ils -l /vzRZG/home/latuser#vzMPI
/vzRZG/home/latuser#vzMPI:
corpman@lux115:/srv/irods/current>
```

Demo Slides

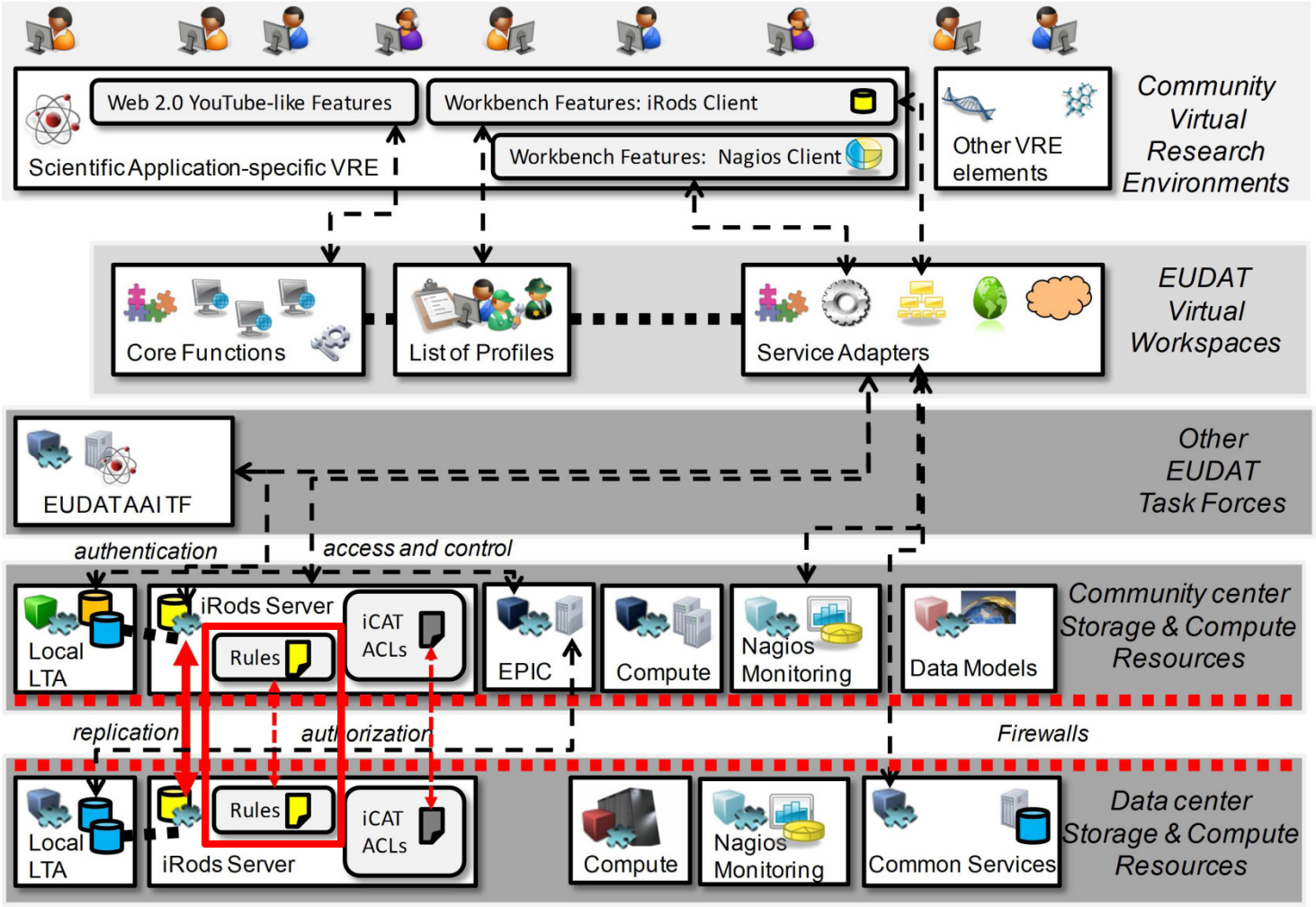
```
corpman@lux115:/srv/irods/current/modules/EUDAT-PID/rules> cat science_data.replicate
842/394d442e-1c5c-11e2-95ff-005056ae635a,/vzMPI/bin/ladfc11a_1.imdi,/vzRZG/home/latuser#vzMPI/ladfc11a_1.imdi
corpman@lux115:/srv/irods/current/modules/EUDAT-PID/rules>
```

```
corpman@lux115:/srv/irods/current/modules/EUDAT-PID/rules> iput science_data.replicate /vzMPI/replicate
corpman@lux115:/srv/irods/current/modules/EUDAT-PID/rules> ils /vzMPI/replicate
/vzMPI/replicate:
  science_data.replicate.20121023T120149.success
corpman@lux115:/srv/irods/current/modules/EUDAT-PID/rules> ils /vzRZG/home/latuser#vzMPI
/vzRZG/home/latuser#vzMPI:
  ladfc11a_1.imdi
corpman@lux115:/srv/irods/current/modules/EUDAT-PID/rules>
```

Bottom Line: Advantages using the Service

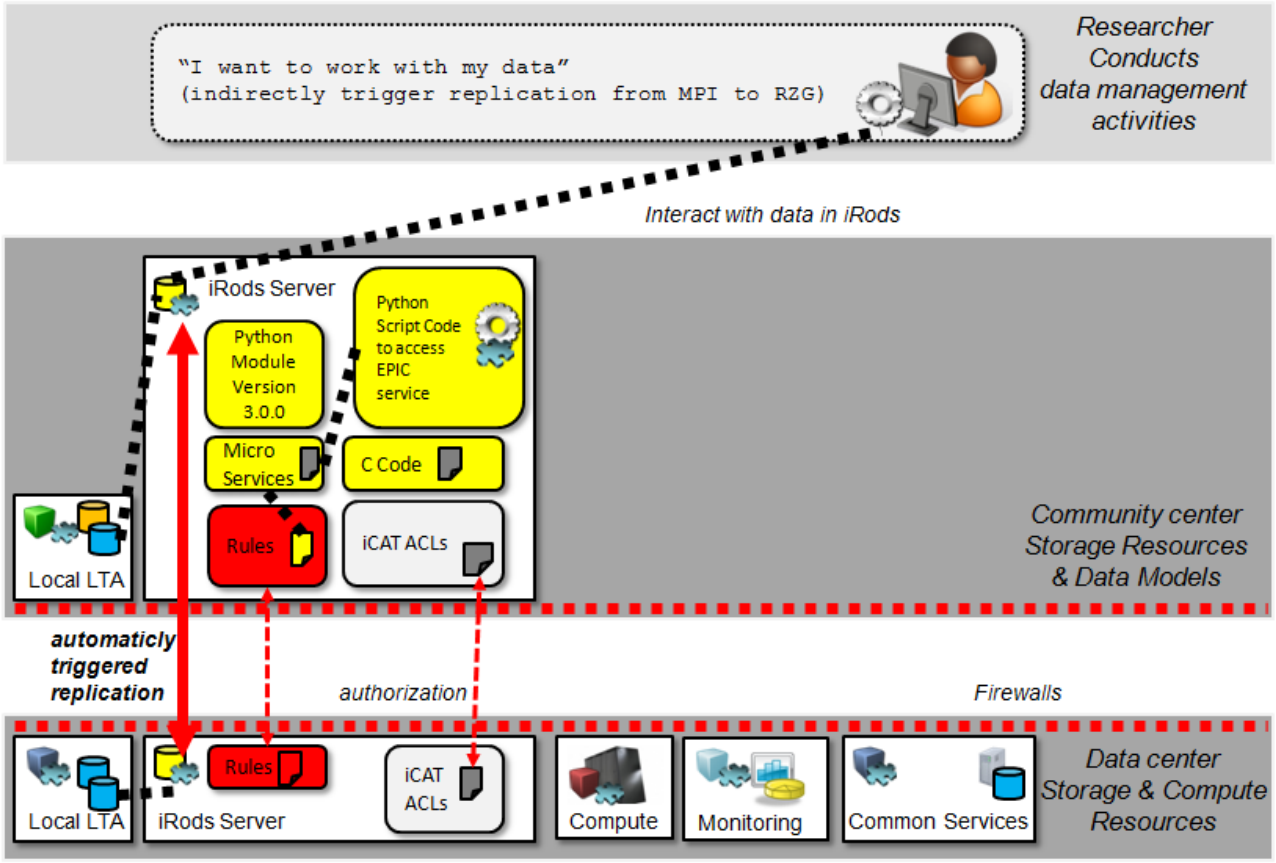
- Enable replication between community centers and data centers
- Control manually replication if needed
- Benefit from know-how and expertise around iRods in real production environments

Create your own data replication policies

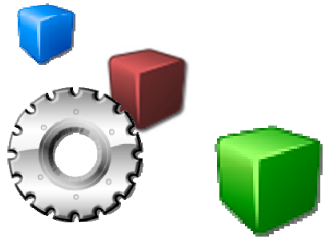


Overview: Rule-based data management

- Need to understand rules & micro-services in iRods



Demo: Replication Rule to Trigger Event



- Willem
  - Show rules in eudat.re
  - Show contents of replication rule
  - Show collection in source zone (resource is there)
  - Show destination collection (resource is not there)
  - Trigger replication rule (irule –vF …)
  - Show destination collection (resource is there)

Demo Slides

```
#
# Process a .replicate file and perform the replication
#
# Parameters:
#       *cmdPath    [IN]    the path to the .replicate file
#
# Author: Willem Elbers, MPI-TLA
#
processReplicationCommandFile(*cmdPath) {
        msiWriteRodsLog("processReplication(*cmdPath)", *status);

        readFile(*cmdPath, *out_STRING);

        *list = split(*out_STRING, ",");
        if(size(*list)==3) {
                *pid = elem(*list,0);
                *source = elem(*list,1);
                *destination = elem(*list,2);
                doReplication(*pid,*source,*destination,*status);
                updateCommandName(*cmdPath,*status);
        } else {
                msiWriteRodsLog("incorrect list", *status);
        }
}
```

Demo Slides

```
#
# Start a replication
#
# Parameters:
#       *pid            [IN]    pid of the digital object
#       *source         [IN]    source path of the object to replicate
#       *destination    [IN]    destination path of the object to replicate
#       *status         [OUT]   status, 0 = ok, <0 = error
#
# Author: Willem Elbers, MPI-TLA
#
doReplication(*pid,*source,*destination,*status) {
        msiWriteRodsLog("doReplication(*pid,*source,*destination)", *status);

        #rsync object (make sure to supply "null" if dest resource should be the default one)
        msiDataObjRsync(*source, "IRODS_TO_IRODS", "null", *destination, *rsyncStatus);

        #trigger pid management in destination
        getSharedCollection(*destination,*collectionPath);
        msiSplitPath(*destination, *parent, *child);
        triggerCreatePID("*collectionPath*child.pid.create",*pid,*destination);
        updateMonitor("*collectionPath*child.pid.update");
}
```

Demo Slides

```
corpman@lux115:/srv/irods/current/modules/EUDAT-PID/rules> cat science_data_replication1.r
replicate {
        msiWriteRodsLog("starting replication", *status);
        getSharedCollection(*source,*collectionPath);
        msiWriteRodsLog("source = *source", *status);
        msiWriteRodsLog("collectionPath = *collectionPath", *status);
        msiWriteRodsLog("shared collection = *collectionPath*commandFile", *status);
        triggerReplication("*collectionPath*commandFile",*pid,*source,*destination);
}
INPUT *pid="842/b1ffb6f4-1c75-11e2-9618-005056ae635a",*source="/vzMPI/bin/ladfc12a_1.imdi"
,*destination="/vzRZG/home/latuser#vzMPI/ladfc12a_1.imdi",*commandFile="ladfc12a_1.imdi.replicate"
OUTPUT ruleExecOut
```

Demo Slides

```
corpman@lux115:/srv/irods/current/modules/EUDAT-PID/rules> irule -vF science_data_replication1.r
rcExecMyRule: replicate {
        msiWriteRodsLog("starting replication", *status);
        getSharedCollection(*source,*collectionPath);
        msiWriteRodsLog("source = *source", *status);
        msiWriteRodsLog("collectionPath = *collectionPath", *status);
        msiWriteRodsLog("shared collection = *collectionPath*commandFile", *status);
        triggerReplication("*collectionPath*commandFile",*pid,*source,*destination);
}

outParamDesc: ruleExecOut
ExecMyRule completed successfully.    Output
```

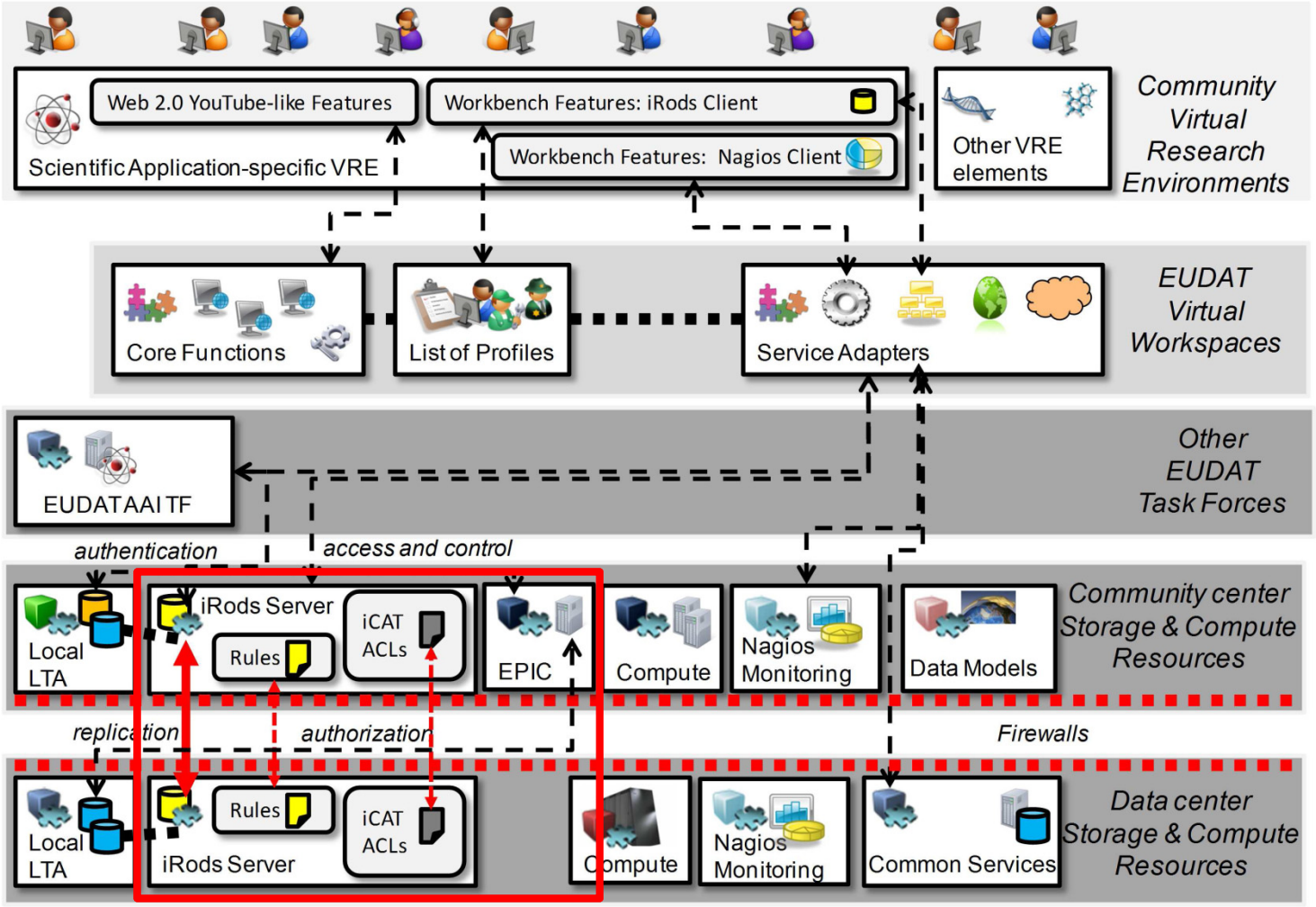
Demo Slides

```
corpman@lux115:/srv/irods/current/modules/EUDAT-PID/rules> ils /vzRZG/home/latuser#vzMPI
/vzRZG/home/latuser#vzMPI:
  ladfc11a_1.imdi
  ladfc12a_1.imdi
corpman@lux115:/srv/irods/current/modules/EUDAT-PID/rules>
```

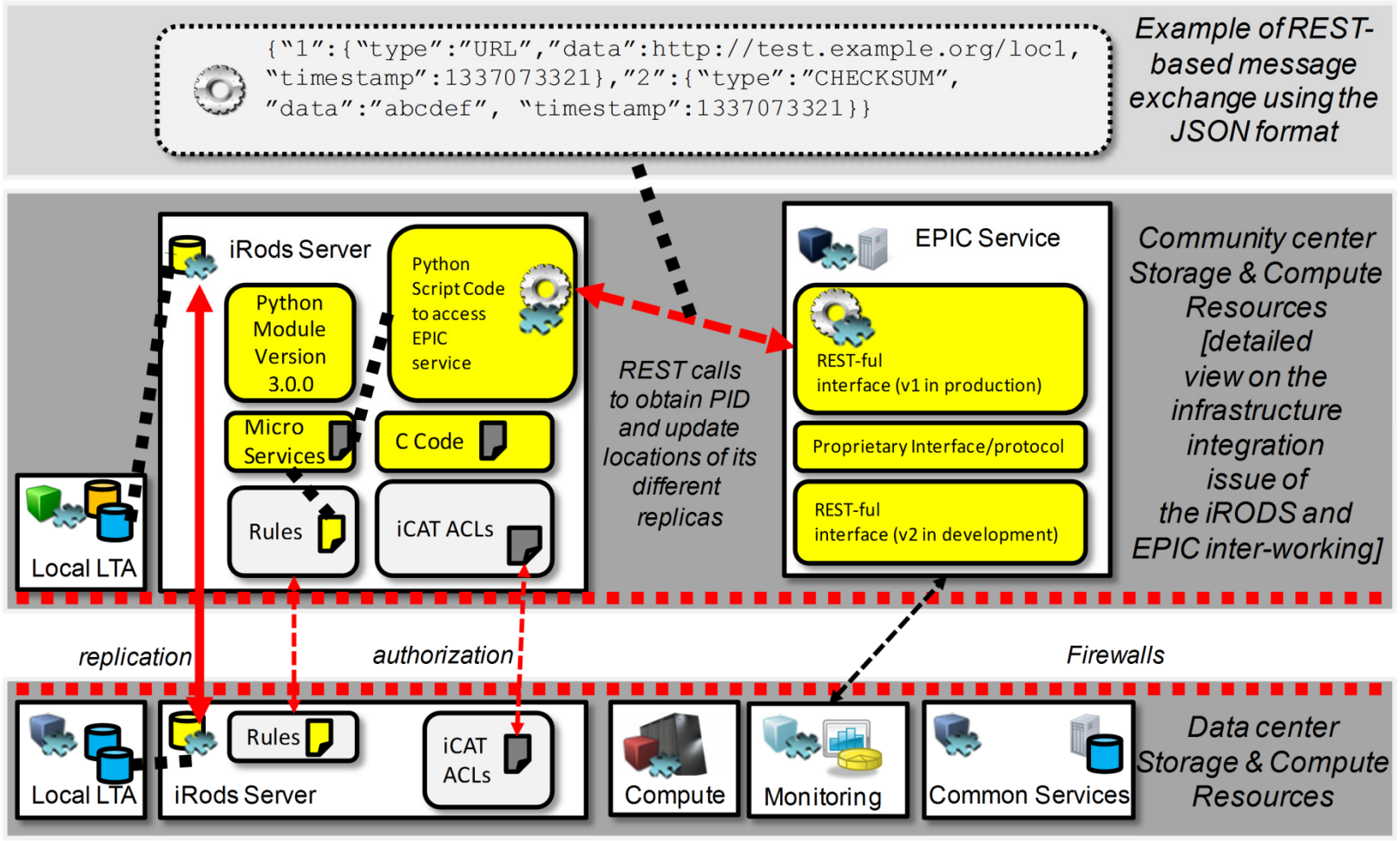
Bottom Line: Advantages using the Service

- Enable replication between community centers and data centers
- Control manually replication if needed
- Benefit from know-how and expertise around iRods deployments in real production ecosystems
- Take-up initially defined replication policies & tune them to your automatic replication demand

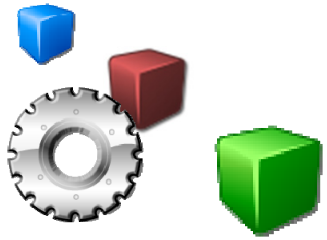
Persistent Identifier for Replicated Data



Persistent Identifier (PID) Service



Demo: Interaction iRods and PID Service



- Willem
  - CreatePID policy
  - Trigger replication rule (irule –vF …)
  - Show destination collection (resource is there)
  - Show result
    - https://epic.sara.nl/epic1/842

Demo Slides

```
#
# Generate a new PID for a digital object.
# Fields stored in the PID record: URL, ROR and CHECKSUM
#
# Parameters:
#   *rorPID     [IN]    the PID of the repository of record (RoR), should be stored with all child PIDs
#   *path       [IN]    the path of the replica to store with the PID record
#   *newPID     [OUT]   the pid generated for this replica
#
# Author: Willem Elbers, MPI-TLA
#
createPID(*rorPID, *path, *newPID) {
        msiWriteRodsLog("create pid for *path and save *rorPID as ror", *status);

        #check if PID already exists
        msiExecCmd("searchHandle.py", *path, "null", "null", "null", *out);
        msiGetStdoutInExecCmdOut(*out, *existing_pid);

        if(*existing_pid == "empty") {
            # create PID
            msiExecCmd("createHandle.py", *path, "null", "null", "null", *out);
            msiGetStdoutInExecCmdOut(*out, *newPID);
            msiWriteRodsLog("created handle = *newPID", *status);

            # add RoR to PID record
            msiExecCmd("modifyHandle.py","*newPID ROR https://epic.sara.nl/epic1/*rorPID", "null", "null", "null", *out2);
            msiGetStdoutInExecCmdOut(*out2, *response2);
            msiWriteRodsLog("modify handle response = *response2", *status);

            # add CHECKSUM to PID record
            retrieveChecksum(*path, *checksum);
            msiExecCmd("modifyHandle.py","*newPID CHECKSUM *checksum", "null", "null", "null", *out3);
            msiGetStdoutInExecCmdOut(*out3, *response3);
            msiWriteRodsLog("modify handle response = *response3", *status);
        } else {
            *newPID = *existing_pid;
            msiWriteRodsLog("PID already exists (*newPID)", *status);
        }
```

Demo Slides

Metadata for handle 843/97398870-1c81-11e2-a6d0-001636f0c2c6

idx	type	data	timestamp
1	URL	/vzRZG/home/latuser#vzMPI/ladfc11a_1.imdi	Mon, 22 Oct 2012 19:49:30 GMT
2	ROR	https://epic.sara.nl/epic1/842/394d442e-1c5c-11e2-95ff-005056ae635a	Mon, 22 Oct 2012 19:49:30 GMT
3	CHECKSUM	3f1d47b7303dfbf18125e45e0f5a67cd	Mon, 22 Oct 2012 19:49:30 GMT

Powered by sara

Metadata for handle 842/394d442e-1c5c-11e2-95ff-005056ae635a

idx	type	data	timestamp
1	URL	/vzMPI/bin/ladfc11a_1.imdi	Mon, 22 Oct 2012 19:50:20 GMT
2	CHILD	https://epic.sara.nl/epic1/843/97398870-1c81-11e2-a6d0-001636f0c2c6	Mon, 22 Oct 2012 19:50:20 GMT
3	CHECKSUM	3f1d47b7303dfbf18125e45e0f5a67cd	Mon, 22 Oct 2012 19:50:20 GMT

Powered by sara

Bottom Line: Advantages using the Service

- Enable replication between community centers and data centers
- Control manually replication if needed
- Benefit from know-how and expertise around iRods deployments in real production ecosystems
- Take-up initially defined replication policies & tune them to your automatic replication demand
- Use an inter-working set of products (iRODS&EPIC) to make data referencable

Selected References

- Documentation on iRODS and EPIC/Handle system available on the Web
- 1st EUDAT Conference Training Day - Many training sessions yesterday!
  - PID handling & services, iRODS policies, rules, micro-services, etc.
- EUDAT Newsletter April 2012
  - Check the EUDAT WebSite
- M. Riedel and P. Wittenburg et al.
  'A Data Infrastructure Reference Model with Applications - Towards Realization of a ScienceTube Vision with a Data Replication Service',
  Journal of Internet Applications,
  to be published early 2013
- Contact to specialists:
  eudat-safereplication@postit.csc.fi


A Data Infrastructure Reference Model with Applications
Towards Realization of a ScienceTube Vision with a Data Replication Service