CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

ETH zürich

# Spectrum Scale Object at CSCS

HPCXXL – Summer 2017

Giuseppe Lo Re, CSCS

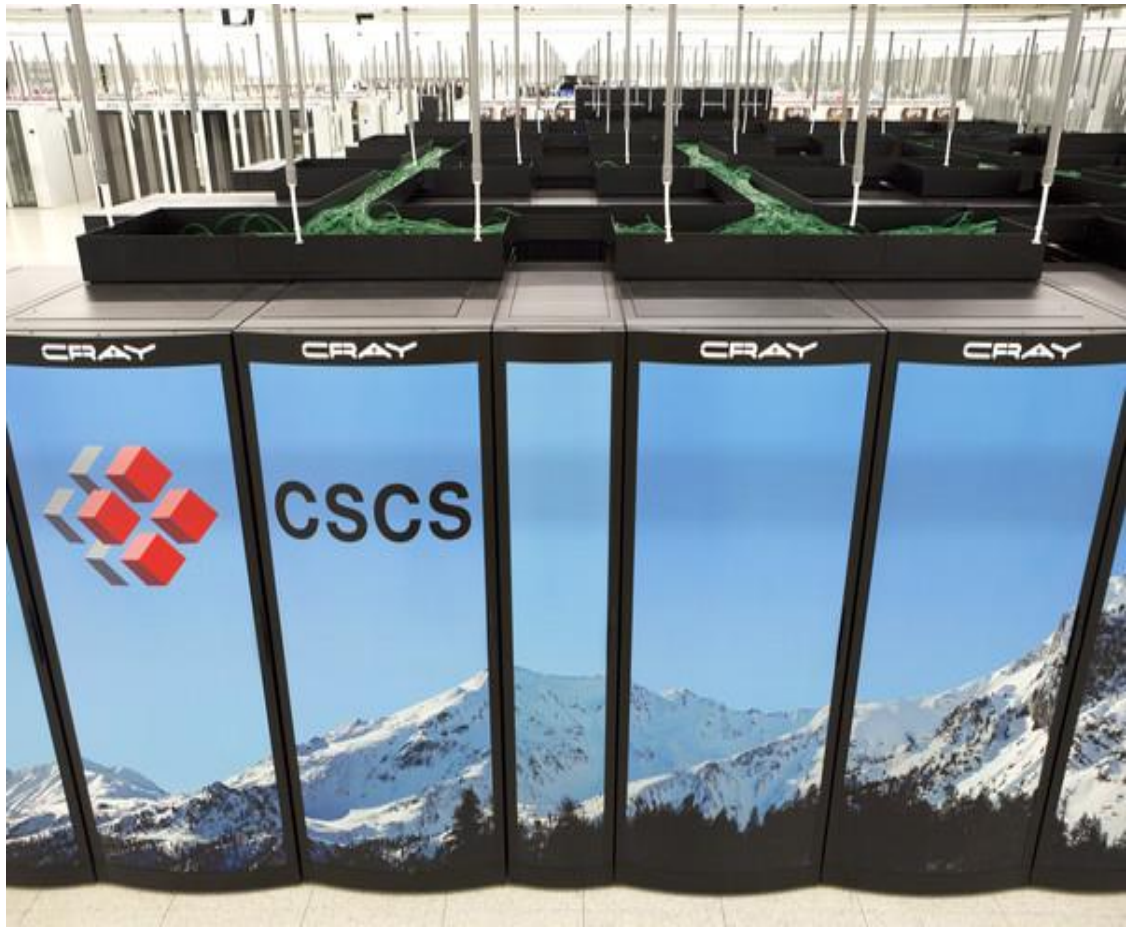
Sep 28, 2017

# Outline

- CSCS overview
- Storage services
- IAAS motivations/use cases/constraints
- Openstack facility
- GPFS CES Object cluster
- Conclusion

# CSCS overview

- CSCS is the Swiss National Supercomputing Centre
- Unit of the Swiss Federal Institute of Technology in Zurich (ETH Zurich), located in Lugano
- CSCS's resources are open to academia, industry and the business sector
- 2000 $m^2$ machine room with no single supporting pillar or any partitioning
- Some operational HPC supercomputers:
  - Piz Daint (Cray XC40/XC50)
  - Kesch + Escha (Meteoswiss, Cray CS-Storm)
  - Mönch (NEC Cluster)
  - Phoenix (LHC CERN, Grid Cluster)
  - Monte Leone (High-memory cluster)
  - Gran Tavé (KNL R&D)



# Storage services

- Posix file systems
  - GPFS
  - Lustre
- Data movers
  - GridFTP
  - Slurm queue
  - Active File Management
- TSM Backup/Archive
  - mmbackup
  - GPFS/HSM
  - Arema (BA API)
- NAS
  - GPFS CES for NFS and SMB
- dCache
- **Object store**

| Filesystem | Size (TiB) | Type |
|---|---|---|
| /users | 86 | GPFS |
| /apps | 58 | GPFS |
| /project | 5940 | GPFS |
| /store | 3891 | GPFS |
| /scratch/lcg | 642 | GPFS |
| /scratch/shared | 1434 | GPFS |
| GSS-BBP | 3800 | GPFS |
| /scratch/snx3000 | 6349 | Sonexion |
| /scratch/snx2000 | 904 | Sonexion |
| /scratch/snx1600 | 2765 | Sonexion |
| /scratch/monch | 350 | Lustre |
| Escha /scratch | 73 | Lustre |
| Kesch /scratch | 73 | Lustre |
| dCache | 2877 | dCache |
| NAS | 248 | GPFS |
| /object | 165 | GPFS |

# **IAAS motivations**/use cases/constraints

- Cloud ideas today very popular among users
- Pay as you go approach vs submission-approval process: good for small scientific projects, with variable duration
- Fast resource availability
- Dynamic scaling
- Clear distinction of layers and responsibilities
- Easy access/management through Rest API

# **IAAS** motivations/**use cases**/constraints

- Project which cannot run or don’t need to run in Daint
- Web portals
- DBs
- users-to-users services
- Distributed scientific platforms:
  - Neuroinformatics
  - Material science
  - Data science

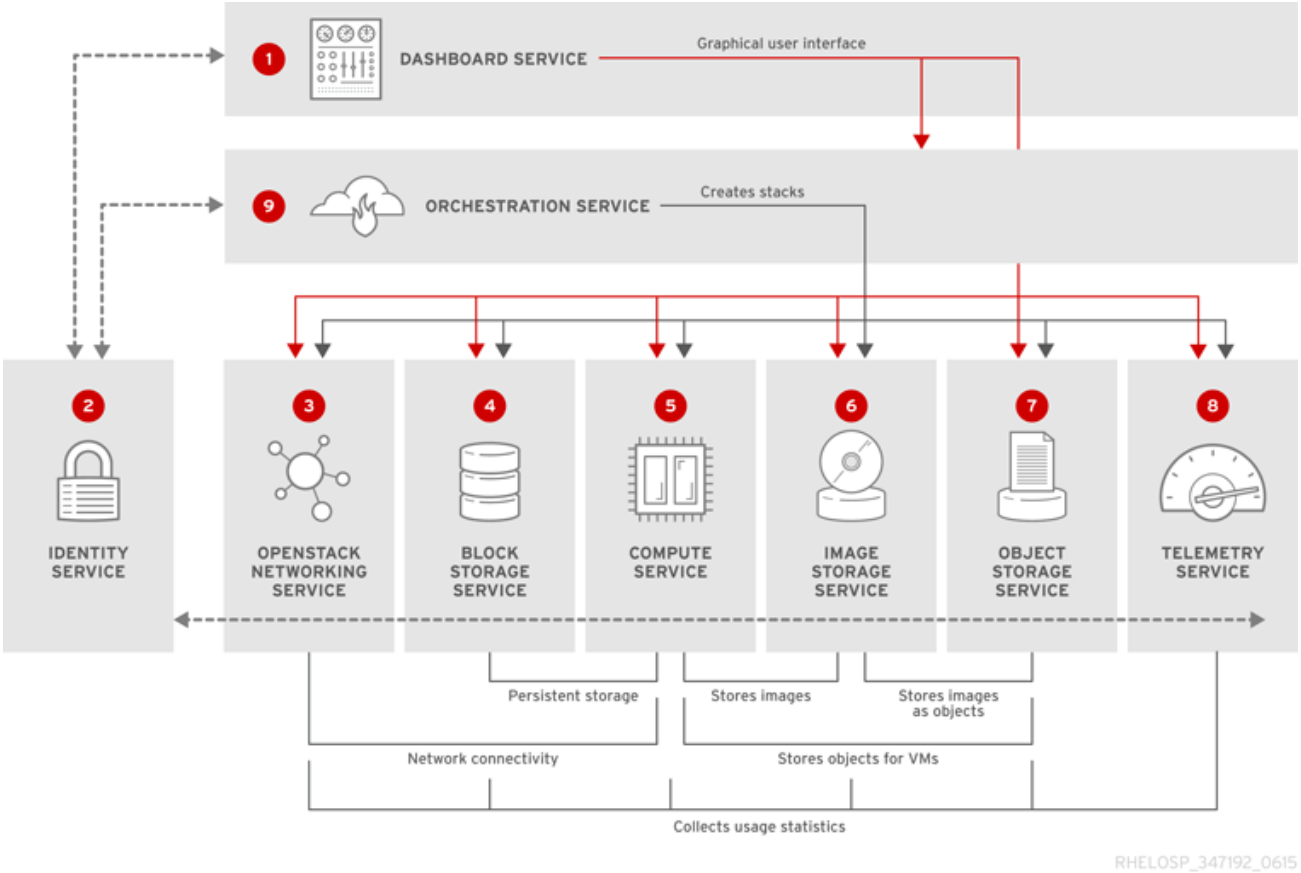
# **IAAS** motivations/use cases/**constraints**

- Reuse of existing LDAP/Kerberos infrastructure for authentication
  - Avoids creating an isolated OpenStack “island”
- Be prepared to **Federate services** with other external IdPs
  - Beta users part of European initiatives
- Integration with CSCS storage infrastructure
  - SAN
  - GPFS
  - TSM
- Object store able to scale to PBs, and millions of objects, high bandwidth

# Openstack facility

- Redhat OpenStack Platform
- Keycloak
- GPFS CES Object

# Openstack facility (RHOSP)




**Redhat Openstack Platform** | **Director (TripleO)**

| Central services | | | |
|---|---|---|---|
| pollux-controller-1 | Keystone | Cinder | Glance |
| pollux-controller-2 | Heat | Horizon | Galera |
| pollux-controller-3 | Ceilometer | Neutron | Nova |
| | Haproxy | Mistral | … |

| Compute services | | |
|---|---|---|
| pollux-compute-1 | KVM | Nova |
| … … | | |
| pollux-compute-7 | Neutron | Ceilometer |
| pollux-compute-8 | | |
| pollux-compute-9 | | |

| Storage service | |
|---|---|
| pollux-cephstorage-1 | Ceph |
| pollux-cephstorage-2 | |
| pollux-cephstorage-3 | |

# Openstack facility (RHOSP)

- Fast deployment
- Starting point to build knowledge
- Enterprise grade support

**Redhat Openstack Platform**

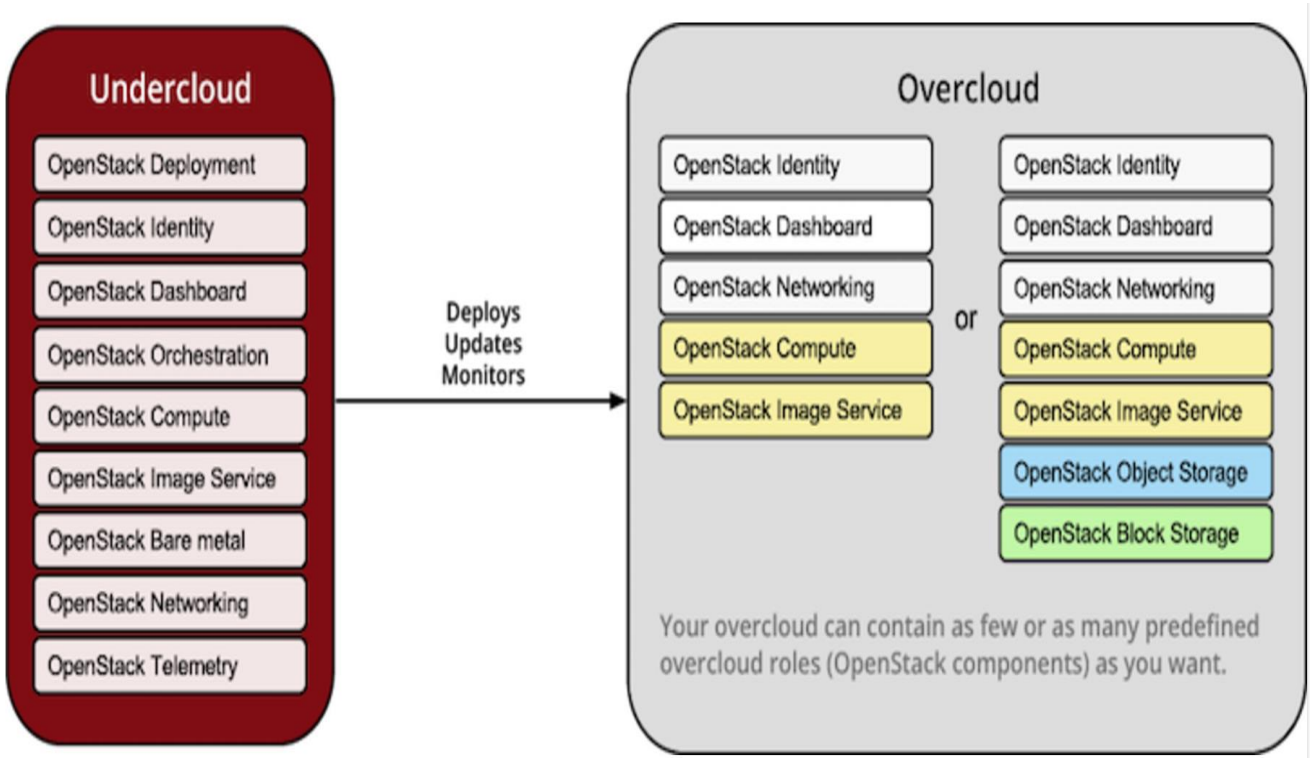
**Director (TripleO)**

| Central services | | | |
|---|---|---|---|
| pollux-controller-1 | Keystone | Cinder | Glance |
| pollux-controller-2 | Heat | Horizon | Galera |
| pollux-controller-3 | Ceilometer | Neutron | Nova |
| | Haproxy | Mistral | … |

| Compute services | | |
|---|---|---|
| pollux-compute-1 | KVM | Nova |
| … ... | | |
| pollux-compute-7 | Neutron | Ceilometer |
| pollux-compute-8 | | |
| pollux-compute-9 | | |

| Storage service | |
|---|---|
| pollux-cephstorage-1 | Ceph  |
| pollux-cephstorage-2 | |
| pollux-cephstorage-3 | |

# Openstack facility (RHOSP)




**Redhat Openstack Platform**

**Director (TripleO)**

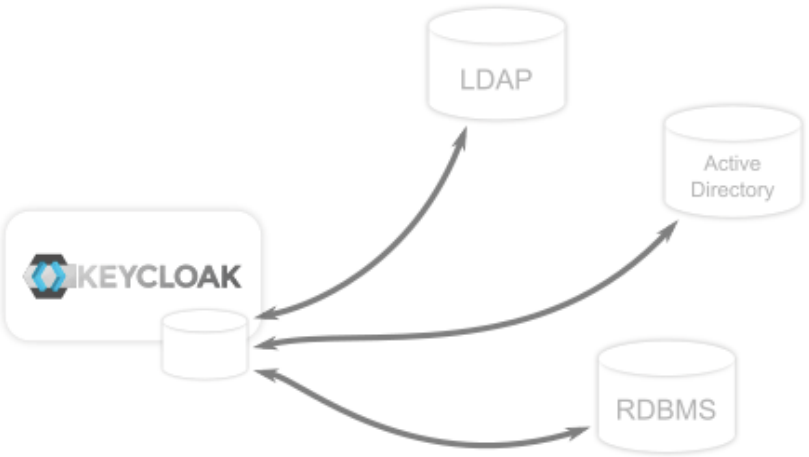
| Central services | | | |
|---|---|---|---|
| pollux-controller-1 | Keystone | Cinder | Glance |
| pollux-controller-2 | Heat | Horizon | Galera |
| pollux-controller-3 | Ceilometer | Neutron | Nova |
| | Haproxy | Mistral | … |

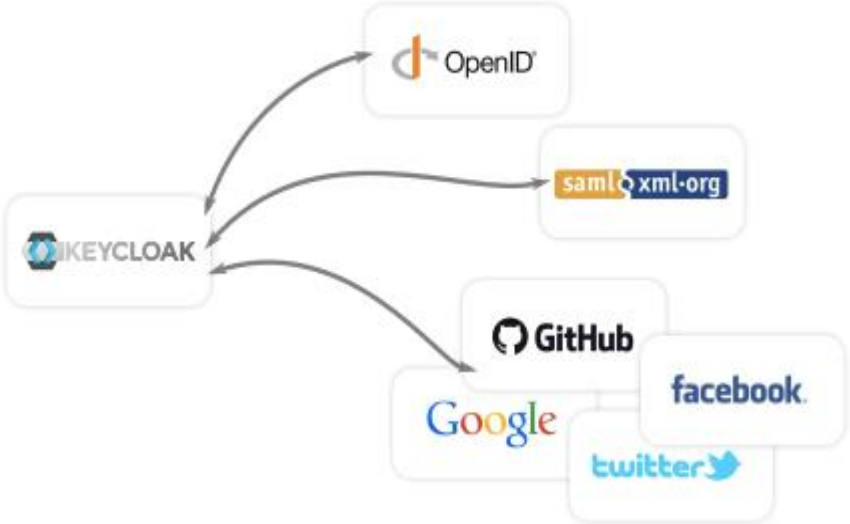
| Compute services | | |
|---|---|---|
| pollux-compute-1 | KVM | Nova |
| … … | | |
| pollux-compute-7 | Neutron | Ceilometer |
| pollux-compute-8 | | |
| pollux-compute-9 | | |

| Storage service | |
|---|---|
| pollux-cephstorage-1 | Ceph  |
| pollux-cephstorage-2 | |
| pollux-cephstorage-3 | |

# Openstack facility (Keycloak)

- Identity and Access Management solution aimed at modern applications and services
- Based on standard protocols
- Need to maintain our users accounting unchanged (LDAP username and Kerberos password) → keystone natively don't allow this configuration.
- Be prepared to Federate services with other external IdPs



# GPFS CES Object cluster

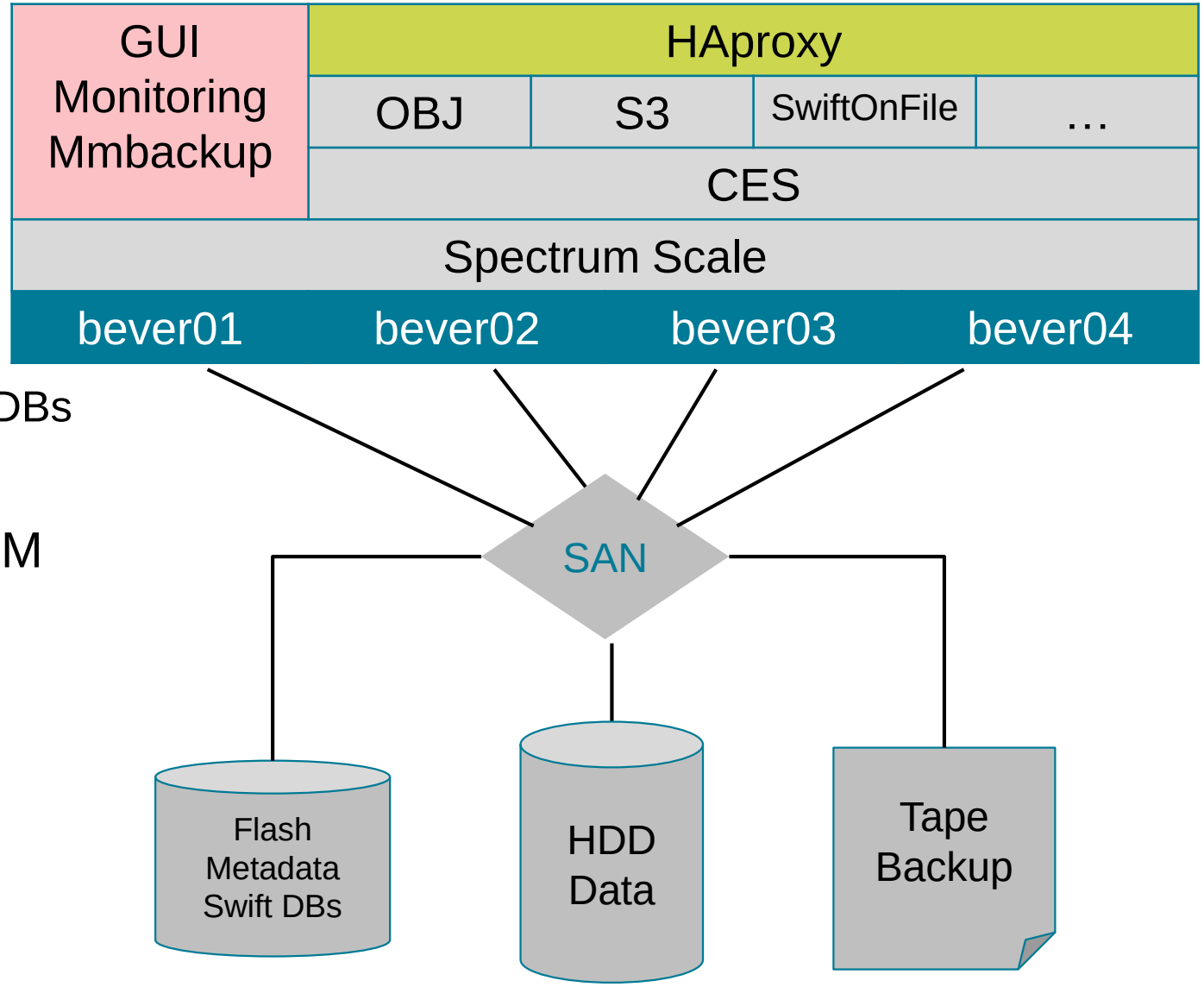
- CSCS storage strongly based on SAN
  - Servers decoupled from disks
  - No need for extra replicas
  - Automation/support from vendor for hardware replacement

- GPFS features
  - Posix access
  - ILM: mmbackup, reporting/accounting
  - HSM
  - Quota
  - Snapshot
  - Single platform for several protocols :Swift, S3, SMB, NFS

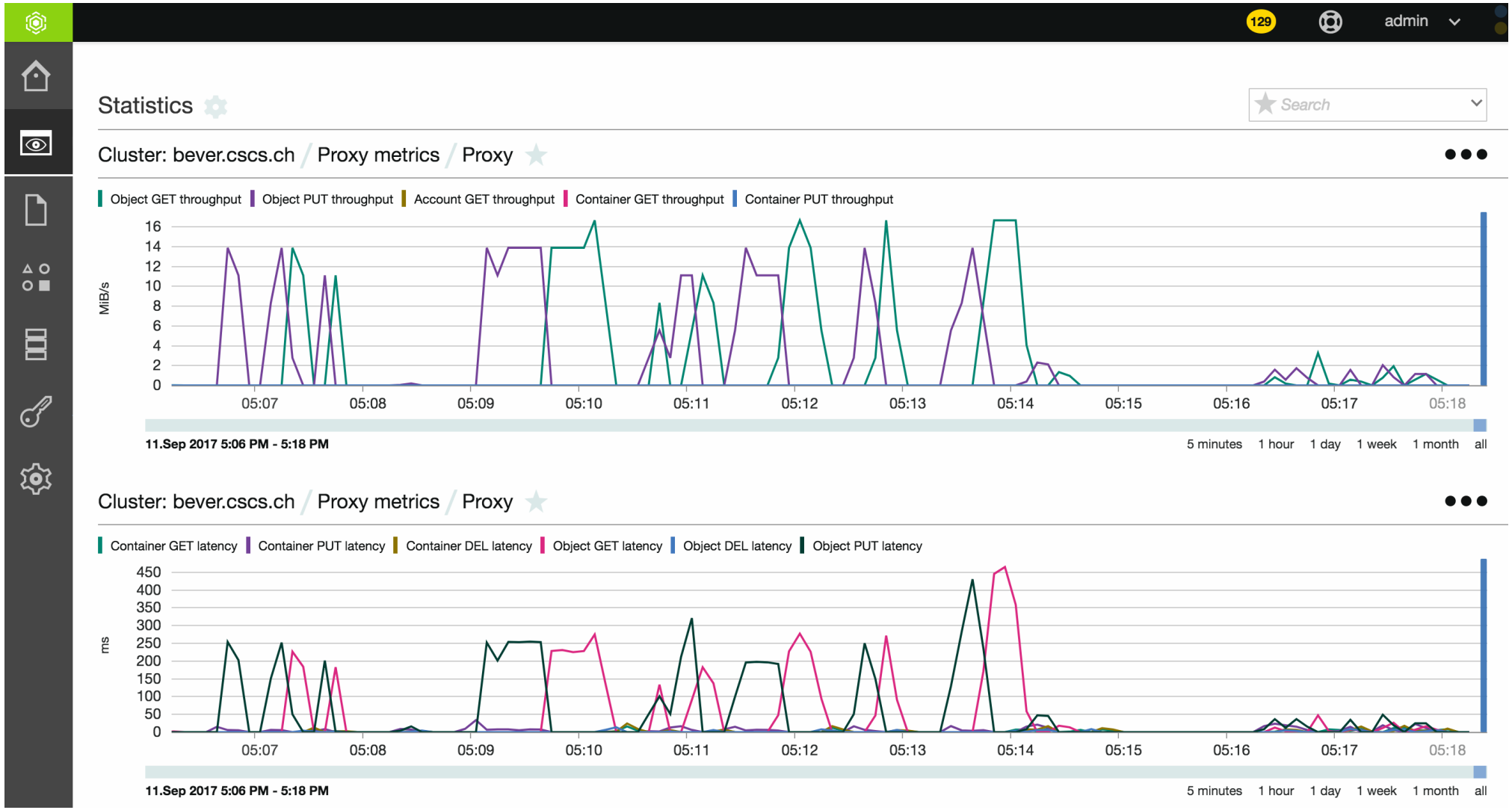
- Promising features
  - SwiftOnFile
  - SwiftHLM (early testing)

# GPFS CES Object cluster

- https://object.cscs.ch:443
- HAproxy added for SSL support
- Storage pools
  - Data pool on HDD
  - System pool for metadata on Flash
  - Dedicated pool for account/container DBs
- Backed up to TSM with snapshot
- Used to backup Cinder volumes, TSM plugin for Cinder/Chep not working
- GUI runs only  on port 443
- GUI/monitoring setup issues
- S3 support problematic
- SMB dependency with ext-keystone



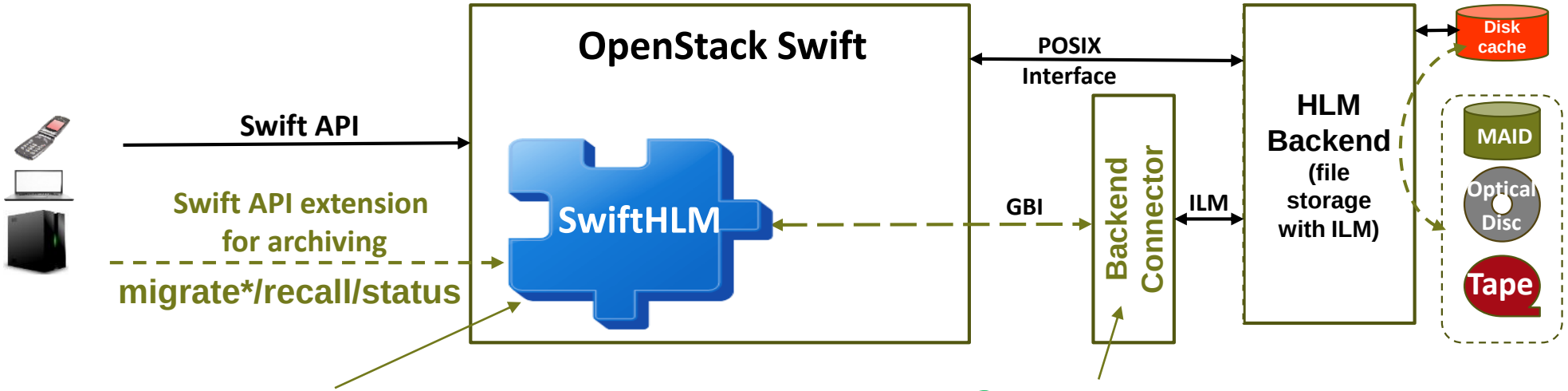
# GPFS CES Object cluster



# Features evaluation

- Unified file and object interface
  - Neuroscience community interested to access data both from Object and Posix interface
- SwiftHLM
  - Material science community interested to have a the same interface for both online data and archived (tape) one

# SwiftHLM: An Extension to the OpenStack Swift Object Storage to Support High-Latency Media (such as tape)




## SwiftHLM

- **Main/generic HLM function/component, open sourced** https://github.com/ibm-research/swifthlm
- Extends Swift API
- Maps&Distributes Swift API requests to backend requests across storage nodes and replicas

## Connector

- **Small/simple backend-specific component**
- Maps SwiftHLM generic backend interface (GBI) HLM requests to backend-specific ILM operations
- Connectors created so far: Spectrum Archive, Spectrum Protect (trial versions); OpenLTFS (prototype/demo)

## SwiftHLM is a Community supported effort:

- Design discussion regular conf. calls: IBM, BDT, Fujitsu, Amethystum; Reviews at OpenStack events: Redhat, NTT, HP, RackSpace, SwiftStack
- Interest for integration with different backends: IBM w/ its Spectrum Archive** and Spectrum Protect**, BDT w/ BDT's Tape Library Connector, Fujitsu w/ its Optical Storage DA700, Amethystum w/ its NFS-mounted optical library
- Official status: SwiftHLM is an OpenStack Swift Associated Project https://docs.openstack.org/developer/swift/associated_projects.html#alternative-api



** Offloads data from disk to HLM media, does not change the Swift name space*

*** Trial software is available, the usage is documented in an IBM Red Book: http://www.redbooks.ibm.com/abstracts/redp5430.html?Open*

# SwiftHLM user/application API (extension of Swift API)

*** Migrate/Recall**

```
POST http://<host>:<port>/hlm/v1/<action>/<account>/<cont>/<obj>
POST http://<host>:<port>/hlm/v1/<action>/<account>/<cont>

    <action> is MIGRATE or RECALL (case insensitive)
    return code: 202 (ok), or an error code
```

*** Status of submitted requests (query pending/non-completed requests)**

```
GET http://<host>:<port>/hlm/v1/REQUESTS/<account>/<cont>/<obj>
GET http://<host>:<port>/hlm/v1/REQUESTS/<account>/<cont>

    return code: 200 (ok), or a standard error
    return value: JSON-encoded list of pending requests for object or container
```

-> faster and more efficient than “Status”

*** Status of objects (query status of object or container)**

```
GET http://<host>:<port>/hlm/v1/STATUS/<account>/<cont>/<obj>
GET http://<host>:<port>/hlm/v1/STATUS/<account>/<cont>

    return code: 200 (ok), or a standard error
    return value: JSON-encoded list of objects and their states
```

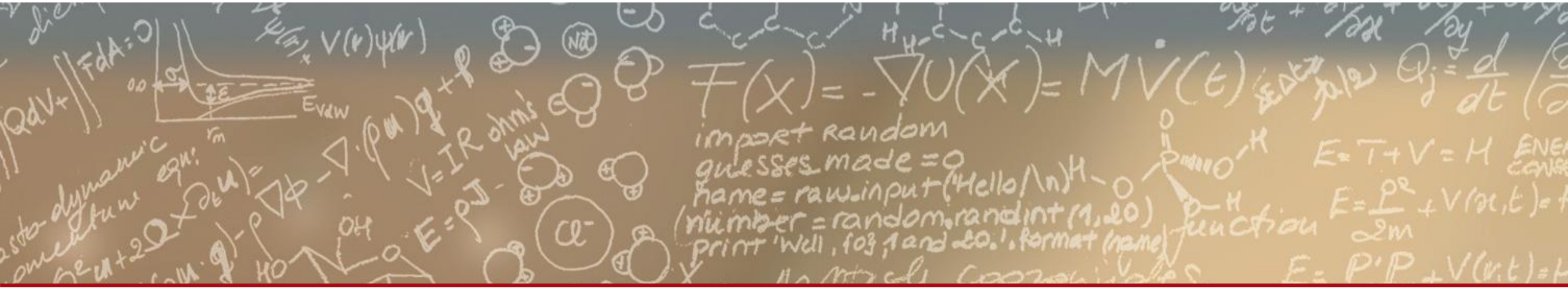
-> likely needs limiting number of objects per request (ranged requests)

# Conclusion

- GPFS Object store service deployed and now in production phase at CSCS
- Part of a wider Openstack facility
- Integration challenging: different vendors
- GPFS Object is based on Liberty/Mitaka
  - 3 releases older than RedHat
  - 4 releases older than community

| Release | Date |
|---|---|
| Liberty | Oct 2015 |
| Mitaka | Apr 2016 |
| Newton | Oct 2016 |
| Ocata | Feb 2017 |
| Pike | Aug 2017 |

- This can be a problem, and it was for us when we tried to us the S3 API
  - Broken in Ocata
  - Fixed in Pike
  - Bug fix port from Pike to Ocata ok. To Mitaka not easy.
- Deploying a current Swift software is key for us
- Exposing the same endpoint/Rest API for both disks and tapes very important



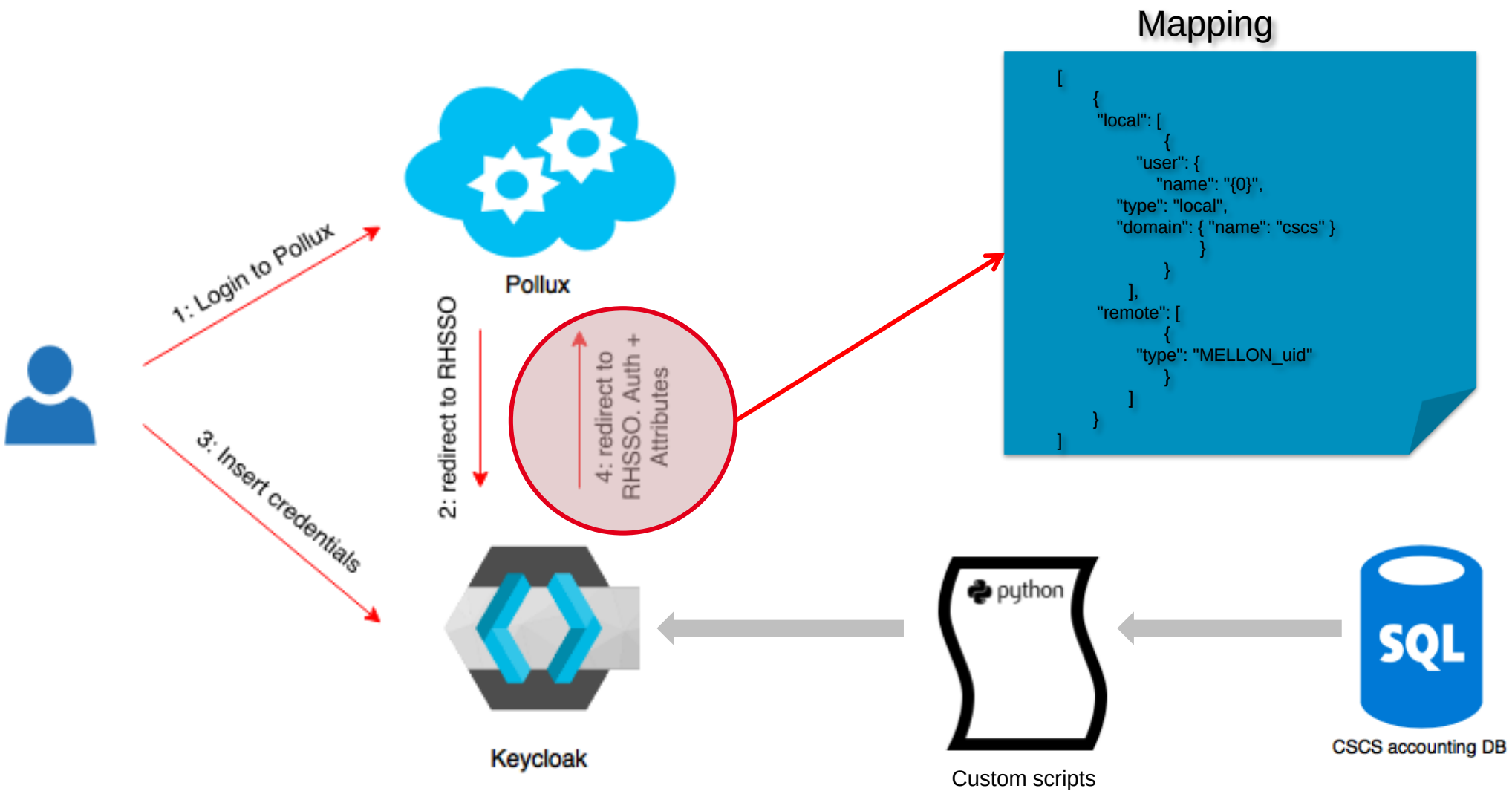

# Thank you for your attention.

# Chapter Title

# Quick Styles

| | | | | | | |
|---|---|---|---|---|---|---|
| Abc | Abc | Abc | Abc | Abc | Abc | Abc |
| Abc | Abc | Abc | Abc | Abc | Abc | Abc |
| Abc | Abc | Abc | Abc | Abc | Abc | Abc |
| Abc | Abc | Abc | Abc | Abc | Abc | Abc |
| Abc | Abc | Abc | Abc | Abc | Abc | Abc |
| Abc | Abc | Abc | Abc | Abc | Abc | Abc |

# Mapping


Mapping
[
{
"local": [
{
"user": {
"name": "{0}",
"type": "local",
"domain": { "name": "cscs" }
}
}
],
"remote": [
{
"type": "MELLON_uid"
}
]
}
]
Pollux
1: Login to Pollux
2: redirect to RHSSO
3: Insert credentials
4: redirect to RHSSO. Auth + Attributes
Keycloak
python
Custom scripts
SQL
CSCS accounting DB