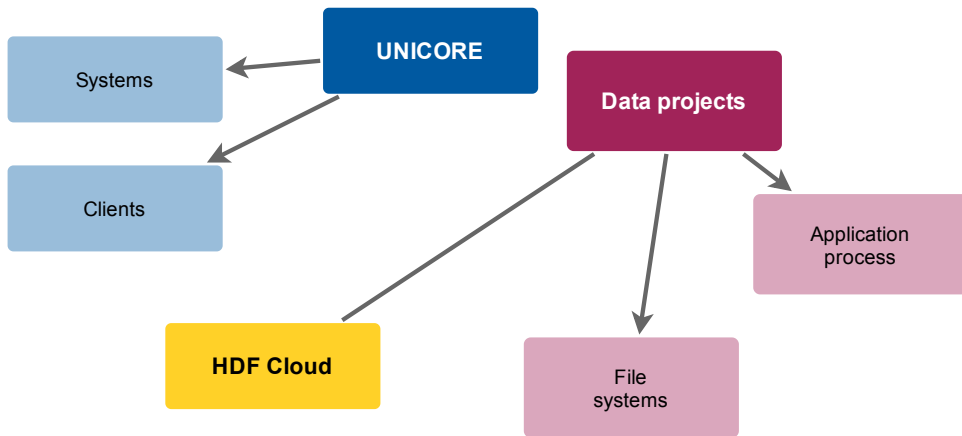




Uniform Resource Access Compute, Data and Cloud Resources at JSC

2021-11-25 | Björn Hagemeier | Juelich Supercomputing Centre

Overview





Workflows



Jobs

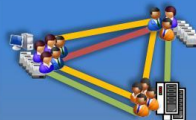


Data Management



Discovery

Services



Federations



Part I: UNICORE

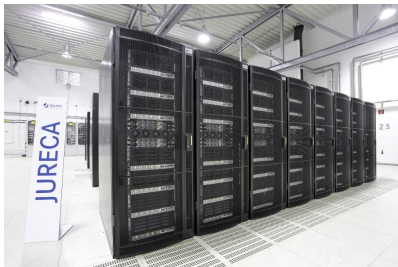
Motivation

Differences of systems

Uniform Interface to Computing Resources

Various RMS on systems

- JUQUEEN: IBM LoadLeveler
- JURECA: Slurm
- Different job description languages for specifying # of nodes, memory requirements, wall time, ...
- Different parameters on the command line
- Unify and simplify supercomputer access



Slurm



Load Leveler

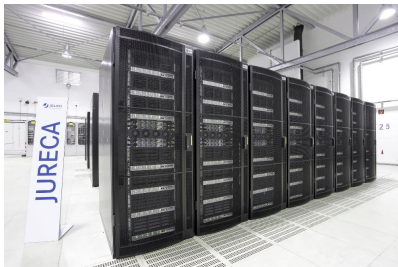
Motivation

Differences of systems

Uniform Interface to Computing Resources

Various RMS on systems

- JUQUEEN: IBM LoadLeveler
- JURECA: Slurm
- Different job description languages for specifying # of nodes, memory requirements, wall time, ...
- Different parameters on the command line
- Unify and simplify supercomputer access



Slurm



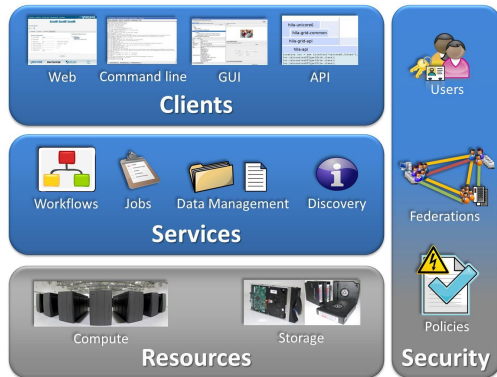
Slurm

Why UNICORE

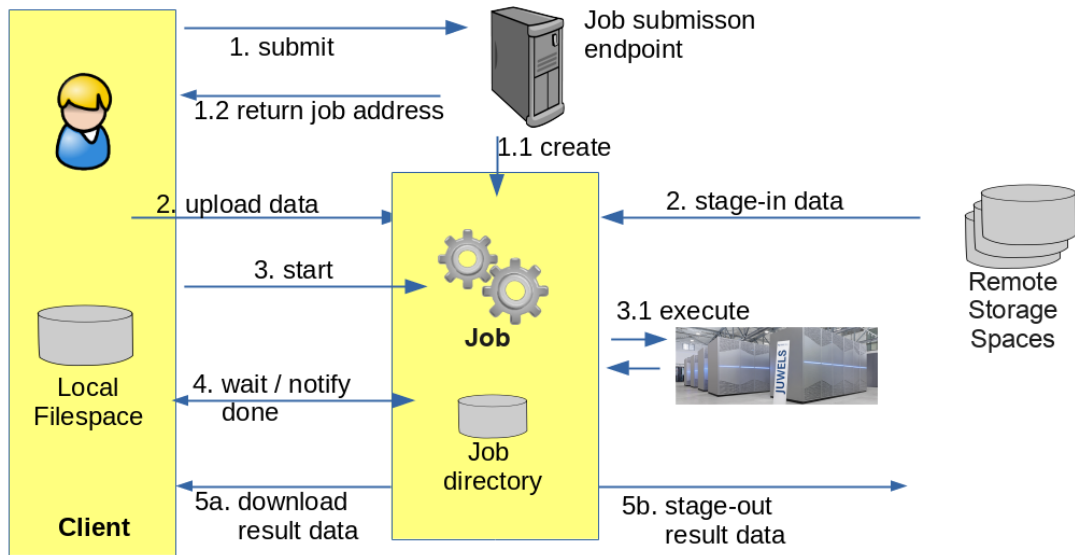
Advantages

- Hide system specific commands
- Create, submit and monitor jobs
 - Seamless, secure, and intuitive access to distributed compute and data resources
- Multiple clients
- Integrated **data management**
- **Federated identities**
- Open Source:
<https://github.com/UNICORE-EU>

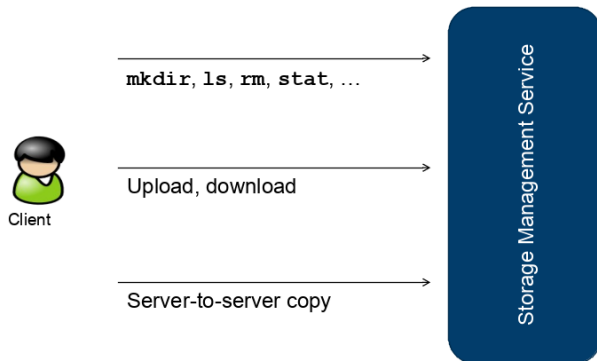
UNICORE



Job execution model



Data management and file transfer



- File Systems

- Apache HDFS



- S3



- iRODS



Efficient file transfer

UFTP

- Data **streaming** library and **file transfer** tool
- Fully integrated into UNICORE
- Standalone (non-UNICORE) client available
- Client to server and server to server data transfers
- Data staging among UFTP-enabled sites
- Efficient **synchronization** of individual local and remote files using the **rsync** algorithm
- Optional **compression** and **encryption** of data streams

Efficient file transfer

UFTP

- Data **streaming** library and **file transfer** tool
- Fully integrated into UNICORE
- Standalone (non-UNICORE) client available
- Client to server and server to server data transfers
- Data staging among UFTP-enabled sites
- Efficient **synchronization** of individual local and remote files using the **rsync** algorithm
- Optional **compression** and **encryption** of data streams
- Awarded “best systemic approach” in SC Asia Data Mover Challenge 2020



Source: SC Asia web site

Clients and APIs

- Commandline tools

- UNICORE Commandline Client (UCC): [https://sourceforge.net/projects/ \unicore/files/Clients/Commandline%20Client/](https://sourceforge.net/projects/unicore/files/Clients/Commandline%20Client/)
- UFTP client for high-performance data access:
<https://sourceforge.net/projects/unicore/files/Clients/UFTP-Client/>

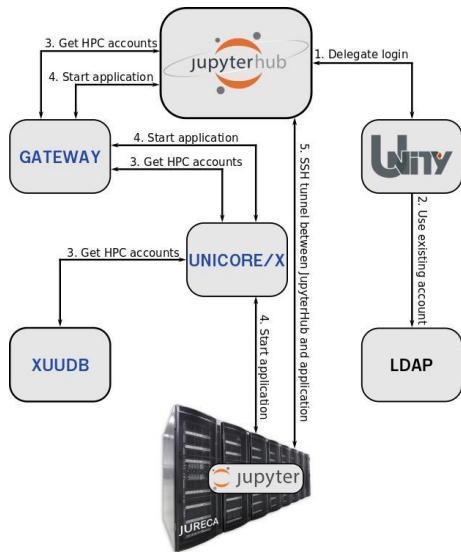
- RESTful APIs

- curl, Python Requests
- https://sourceforge.net/p/unicore/wiki/REST_API/
- PyUNCIORE client library: <https://github.com/HumanBrainProject/pyunicore>

Jupyter Hub @JSC

HPC in your web browser

- UNICORE is an integral part of the Jupyter offering at JSC
- Start Jupyter Labs on JUWELS, JURECA-DC, JUSUF, DEEP, HDFML, or a cloud based VM
- <https://jupyter-jsc.fz-juelich.de/>



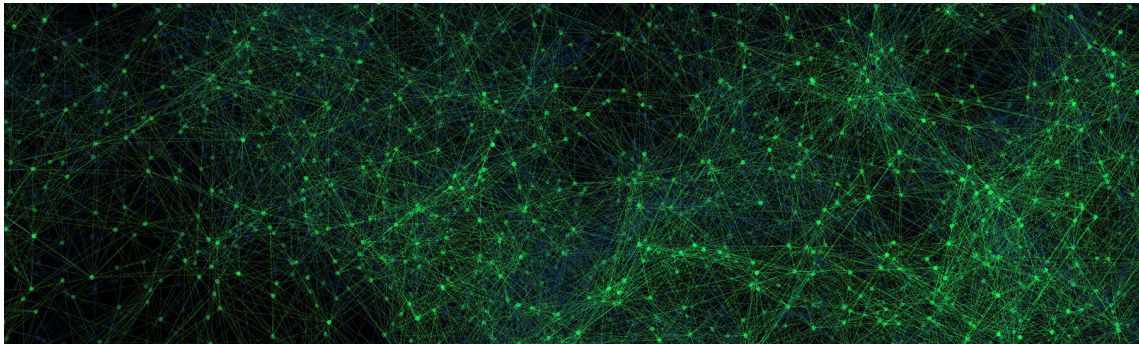
Additional information and support

UNICORE

- Project web site: <https://www.unicore.eu/> for downloads and documentation
- Product support: unicore-support@lists.sourceforge.net

UNICORE at FZJ

- User support email: ds-support@fz-juelich.de
- Registry: https://fzj-unic.fz-juelich.de:9112/FZJ/rest/registries/default_registry
- Documentation: https://www.fz-juelich.de/ias/jsc/EN/Expertise/Support/Software/UNICOREProduction/unicore_production_node.html



Part II: Data Projects

Data projects

Overview

- Additional storage resources
- Started in early 2019
- Rolling call, apply at any time
- So far, we have received 360 applications (2021-04-12)
 - duplications due to annual renewal
- Other data as of Dec. 2020

	# Projects	Capacity	Volume		Inodes	
			used	allocated	used	allocated
ARCHIVE	167	150PB	38PB	52PB	74M	406M
(LARGE)DATA	118	40PB	9PB	22PB	81M	265M
FASTDATA	14	8.6PB	7PB	10PB	54M	254M
USERSOFTWARE	10	33TB	.2 TB	1TB	1M	9M

Data project

Eligibility

- ARCHIVE
 - current or past compute project, in which data has been generated or used
- DATA
 - current, recent, or upcoming compute project
 - data sharing across compute projects
 - access to data from cloud environment
- FASTDATA
 - current compute project absolutely required
 - minimum size of 100TB with strong evidence that this will be utilized
- USERSOFTWARE
 - common software installations **across** compute projects

Data project

Application process

- 1 Prepare application at the application server (JARDS)
- 2 Check fact sheet (also linked on login page of application server) or general storage documentation
- 3 Finalize application
- 4 Print, sign, and send application form **and** PI agreement to ds-support@fz-juelich.de
- 5 Be patient, we try to answer requests within 2-4 weeks

Quick finder

- > JUWELS
- > JURECA
- > JUSUF
- > System messages
- > Computing time
- > User Support
- > JSC online, application forms
- > Data projects
- > Software
- > Industry Relations

Data projects

A day in the life of a reviewer

Application form asks for (per file system)

- **Description of the use case(s)**
- **Justification for resource requirements**
- **Expected data growth pattern**

Please be specific!

- An empty description has no value
 - We try to understand your use case to match it to our infrastructure
- “default” is not a justification
 - Our “defaults” are default **upper** limits. Anything beyond that goes through a more thorough review process, but anything below that is also reviewed and must be justified.
- An empty growth pattern does not help us

Data projects

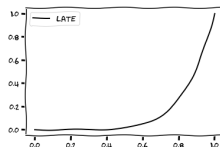
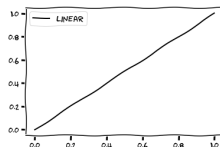
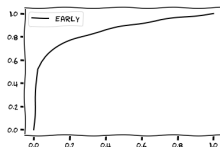
A day in the life of a reviewer

Application form asks for (per file system)

- **Description of the use case(s)**
- **Justification for resource requirements**
- **Expected data growth pattern**

Please be specific!

- An empty description has no value
 - We try to understand your use case to match it to our infrastructure
- “default” is not a justification
 - Our “defaults” are default **upper** limits. Anything beyond that goes through a more thorough review process, but anything below that is also reviewed and must be justified.
- An empty growth pattern does not help us
 - Essentially three options: a) early, b) linear, c) late



Data projects

Common pitfalls

General

- Data projects are always granted for a duration of **one year** only
- Apply for the new **total** volume and inodes

ARCHIVE

- Avoid high numbers of inodes, or you may not see all your data again. Rule of thumb: stay **below 250 inodes per TB** of volume or **» 4GB per file**
- Know how much volume and how many inodes are already taken by your project

DATA

- Currently, all data is duplicated, which is also shown by `df`, `du` and our accounting
- Please apply for the **net value** (your payload) only

FASTDATA

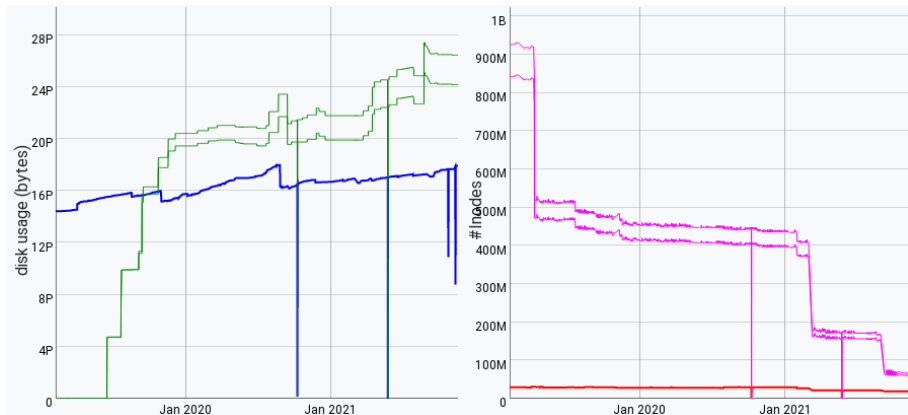
- The entry barrier is **high** on purpose
- It is **not** a convenience store to quickly drop your data and be happy with it
- High volatility, temporary usage, or expected “growth and contraction” are anti-patterns for this resource

Data projects

Effects so far

ARCHIVE

- Decreased number of inodes by about 30 million while increasing volume by 8 PB



Source: JSC KontView screenshot



Part III: HDF Cloud

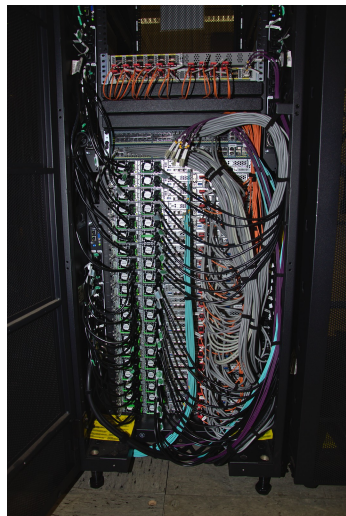
Overview

- OpenStack Infrastructure-as-a-Service (IaaS) environment
 - Compute, storage, network, orchestration, load balancing
 - Run VMs to provide services **linked to DATA**
 - Orchestration using OpenStack Heat
 - Load Balancer as a Service (LBaaS) using OpenStack Octavia
- Phase 1: 16 compute nodes, 768 VCPUs, 6.1TB RAM
- *Phase 2: 10 compute nodes, 480 VCPUs, 7.7TB RAM*
- **Total: 26 compute nodes, 1248 VCPUs, 13.8 TB RAM**
- 10GbE storage uplink per node up to 80Gb total
- 40GbE internal links
- Further information and reference:
<https://go.fzj.de/hdf-cloud>



Overview

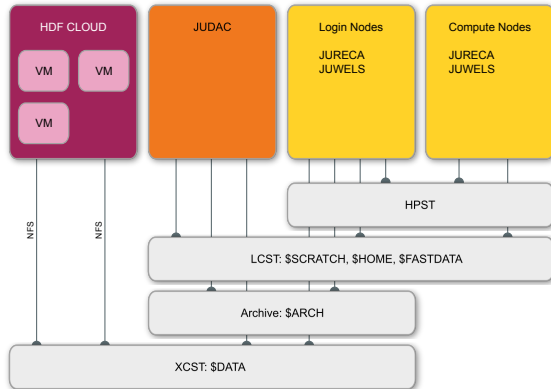
- OpenStack Infrastructure-as-a-Service (IaaS) environment
 - Compute, storage, network, orchestration, load balancing
 - Run VMs to provide services **linked to DATA**
 - Orchestration using OpenStack Heat
 - Load Balancer as a Service (LBaaS) using OpenStack Octavia
- Phase 1: 16 compute nodes, 768 VCPUs, 6.1TB RAM
- *Phase 2: 10 compute nodes, 480 VCPUs, 7.7TB RAM*
- **Total: 26 compute nodes, 1248 VCPUs, 13.8 TB RAM**
- 10GbE storage uplink per node up to 80Gb total
- 40GbE internal links
- Further information and reference:
<https://go.fzj.de/hdf-cloud>



JSC Storage Landscape

Availability of file systems

- XCST
 - \$DATA on JUDAC and login nodes
 - dedicated NFS export to VMs
- Archive
 - \$ARCH on JUDAC and login nodes
- LCST
 - \$SCRATCH, \$HOME, \$FASTDATA on JUDAC, login and compute nodes
- HPST



Data access

VMs and the DATA file system

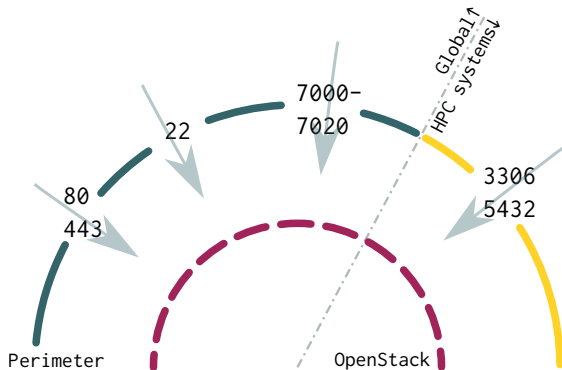
- Helmholtz Data Federation (HDF) Cloud / OpenStack cluster
 - Hosts virtual machines (VMs) for communities
 - Potentially administered by externals, bound by acceptable use policy
- Enable access to data beyond perimeter of SC facility
 - Web interfaces, databases, post processing, ...
 - Users of service likely unknown to SC directory information service
- Access Method
 - POSIX file systems (\$DATA) accessible in VMs via NFS mount from CES servers
 - Server side UID squashing
 - ensures consistency
 - requires services to manage data accordingly
 - read-write or read-only

/p/largedata/slai
/p/largedata/slbid
/p/largedata/slchem
/p/largedata/slcm
/p/largedata/slfire
/p/largedata/slfsfse
/p/largedata/slkit
/p/largedata/slmet
/p/largedata/slnpp
/p/largedata/slns
/p/largedata/slpp
/p/largedata/slqip
/p/largedata/slts



Network setup

- OpenStack firewall freely configurable
- Restrictions apply for inbound connections in perimeter firewall
 - Globally available services and ports: HTTP (80), HTTPS (443), SSH (22), 7000–7020
 - Available from HPC systems: MySQL (3306), PostgreSQL (5432)
- Outbound connections: anything but MTA (25) aka. SMTP



Summary

