



HDF Cloud

Overview and usage scenarios

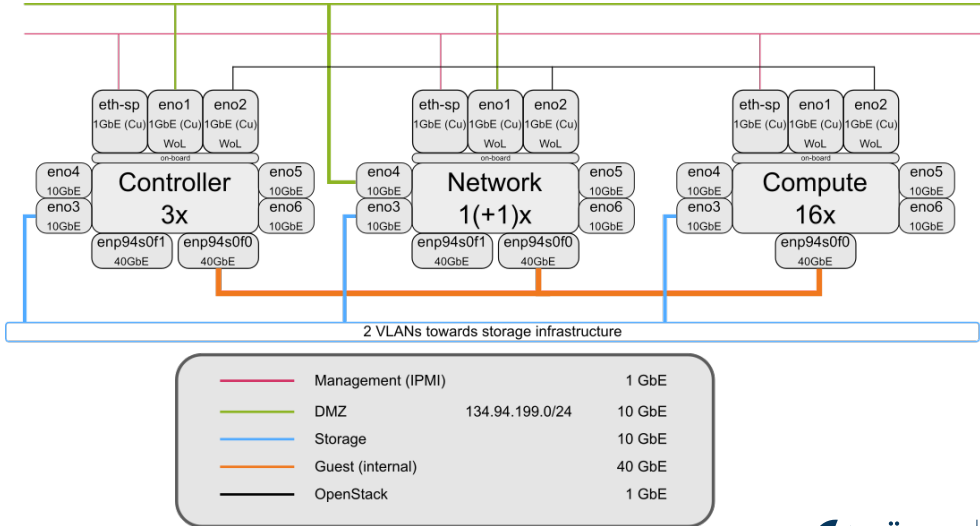
2020-02-05 | Björn Hagemeyer | Juelich Supercomputing Centre

HDF Cloud summary

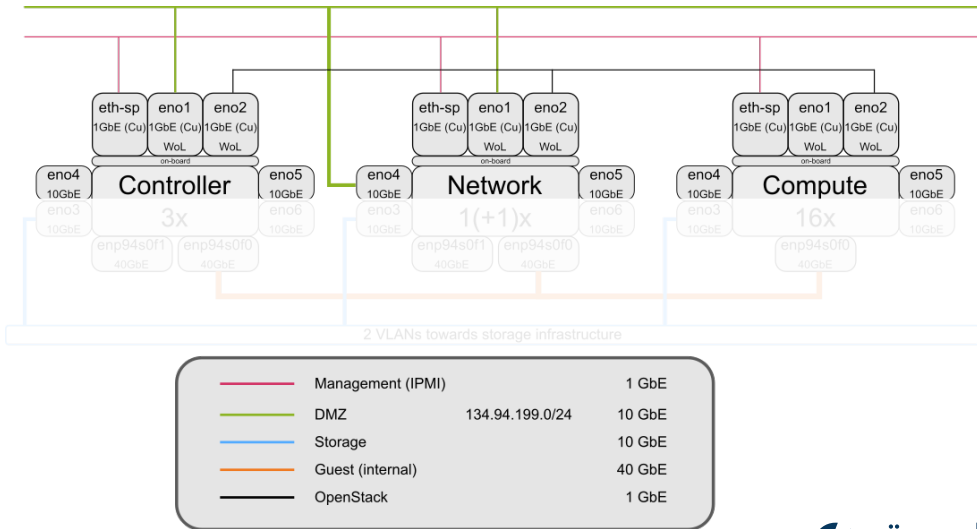
- 16 compute nodes: 24 cores (48 VCPUs), 384 GB RAM
- 3 management + 1 network nodes
- project networks: VXLAN tunnels in 40Gbps
- 10Gbps provider level network towards XCST aka. LARGEDATA
 - physical link shared with storage access to JUSTCOM, separated by VLANs
 - aggregated infrastructure limit of 80Gbps
- default OpenStack overcommit ratios for CPU (16x) and RAM (1.5x)
 - for more compute intensive work loads, we can allow CPU pinning
- Details: Jülich Supercomputing Centre. (2019). **HDF Cloud – Helmholtz Data Federation Cloud Resources at the Jülich Supercomputing Centre**. Journal of large-scale research facilities, 5, A137. <http://dx.doi.org/10.17815/jlsrf-5-173>



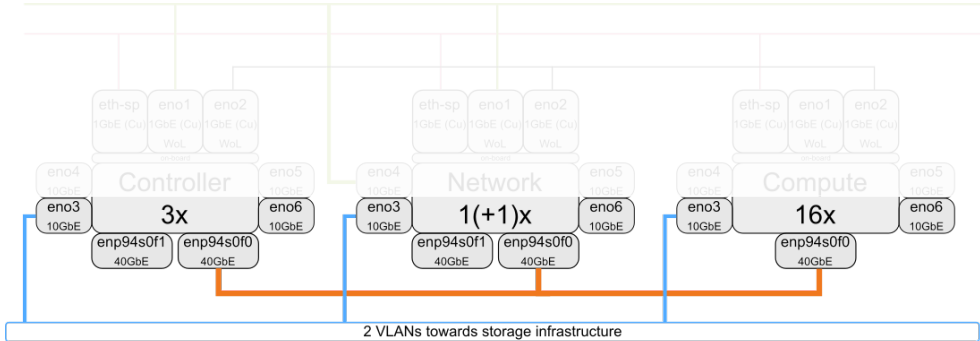
Network setup



Network setup

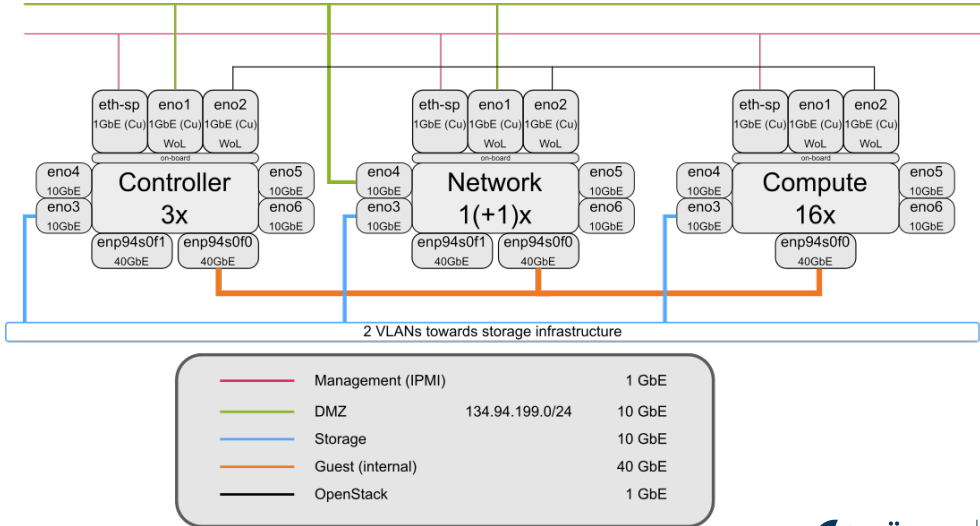


Network setup



—	Management (IPMI)	1 GbE
—	DMZ 134.94.199.0/24	10 GbE
—	Storage	10 GbE
—	Guest (internal)	40 GbE
—	OpenStack	1 GbE

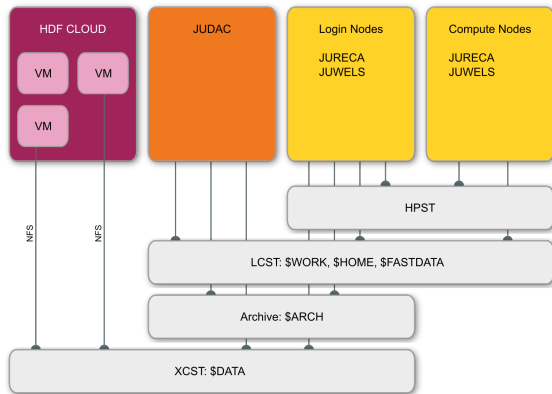
Network setup



JSC Storage Landscape

Availability of file systems

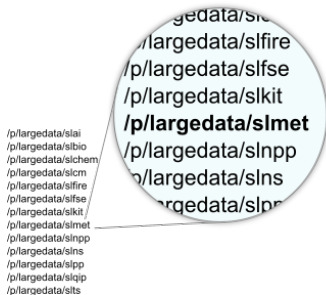
- XCST
 - \$DATA on JUDAC and login nodes
 - dedicated NFS export to VMs
 - 14PB usable space
- Archive
 - \$ARCH on JUDAC and login nodes
 - 1.9PB usable disk space
- LCST
 - \$SCRATCH, \$FASTDATA on JUDAC, login and compute nodes
 - 9.1PB usable space each
- HPST
 - Login and compute nodes



XCST

Cloud Use Case

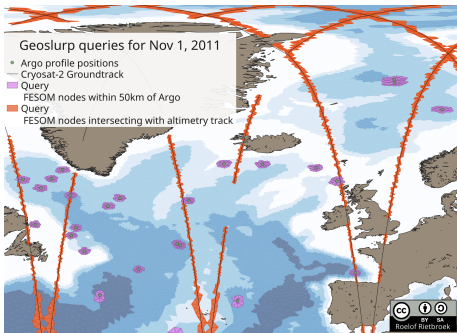
- NFS export of sub-trees of \$DATA provides a window into the file system
- typical example: making data available to the general public, possibly filtered or processed through a service
- web access
- read-only or read-write
- server side UID/GID squashing



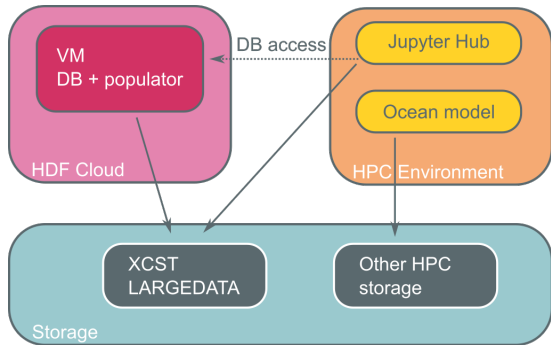
Geoslurp

Greenland ice sheet ocean interaction

- HPC project
- LARGEDATA access from both HPC and cloud
- Postgres DB on LARGEDATA

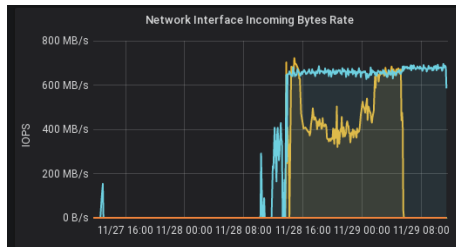


Setup



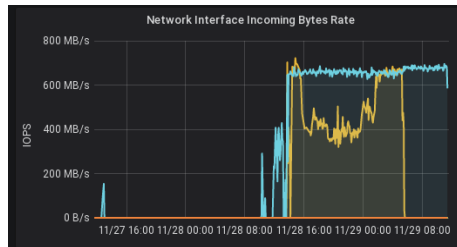
Postprocessing

- a project wanted to do post processing of data generated in the HPC domain
- we offered 2 VMs of flavor m16 (16 cores / 32 GB RAM) as a proof of concept
- crunched approximately 80TB over 24h at one occasion and another 20TB over 10h at another; CPU utilization at the time: approx. 40%
- VMs otherwise idle



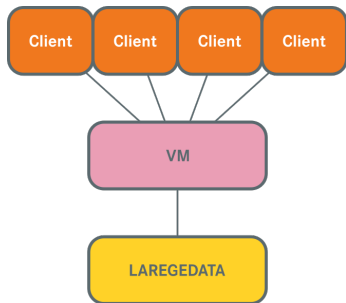
Postprocessing

- a project wanted to do post processing of data generated in the HPC domain
- we offered 2 VMs of flavor m16 (16 cores / 32 GB RAM) as a proof of concept
- crunched approximately 80TB over 24h at one occasion and another 20TB over 10h at another; CPU utilization at the time: approx. 40%
- VMs otherwise idle
- argued that such jobs would not utilize an HPC node to its full potential
- $\Rightarrow$ user assessment is not always correct 😊



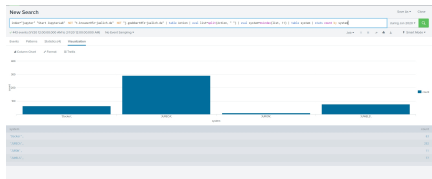
NFS forwarding

- NFS will be re-exported to further clients through NFS
- restrictions
 - server-side user squashing $\Rightarrow$ all access mapped to single UID/GID
 - technically, a simple protocol pass through using `iptables`
 - if only IBM SpectrumScale exported NFS 4.1, something more versatile would be possible using NFS Ganesha as a proxy



Jupyter @JSC

- `https://jupyter-jsc.fz-juelich.de` is powered by HDF Cloud
- access **JURECA**, **JUWELS**, **JURON**, or an HDF Cloud based generic system



Other use cases

- EUDAT services
 - B2SHARE
 - B2ACCESS
 - soon: B2DROP
- JSC Cordex data exchange server
- HBP Collab, NestApp and AppDev
- INM-1
- INM-7
- EOSC



Eligibility

HDF

- will receive all requested resources.
- resources will be extended if needed for this use case.

HAF

- will receive all requested resources
- resources may be extended if needed and after review

FSD

- will receive resources as available
- redistribution of resources when exhausted

Other

- will receive resources temporarily
- may be replaced by use cases of higher priority (cf. above)