Best Practices in HPC Cluster Deployments and How to Achieve it with OpenCATTUS

Eiji Kawahira
Product and Customer Development Team

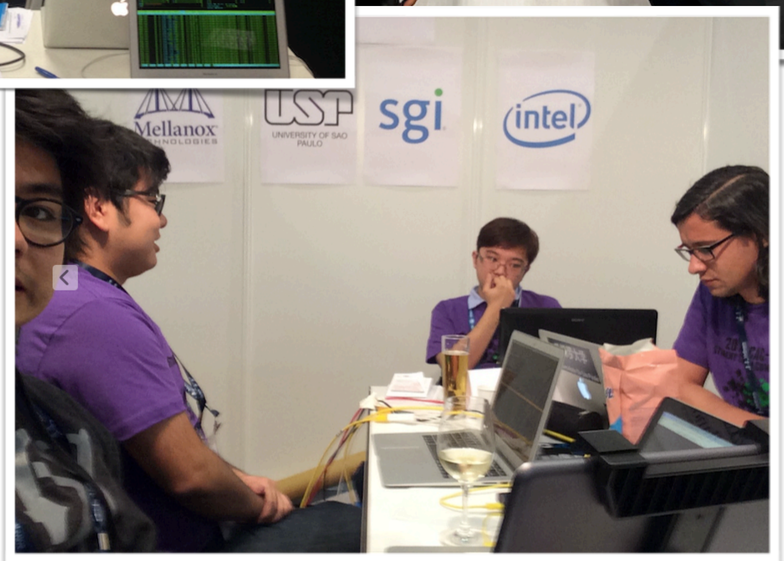
Lenovo

Intro


SC19
Denver, CO | hpc is now.


inspur

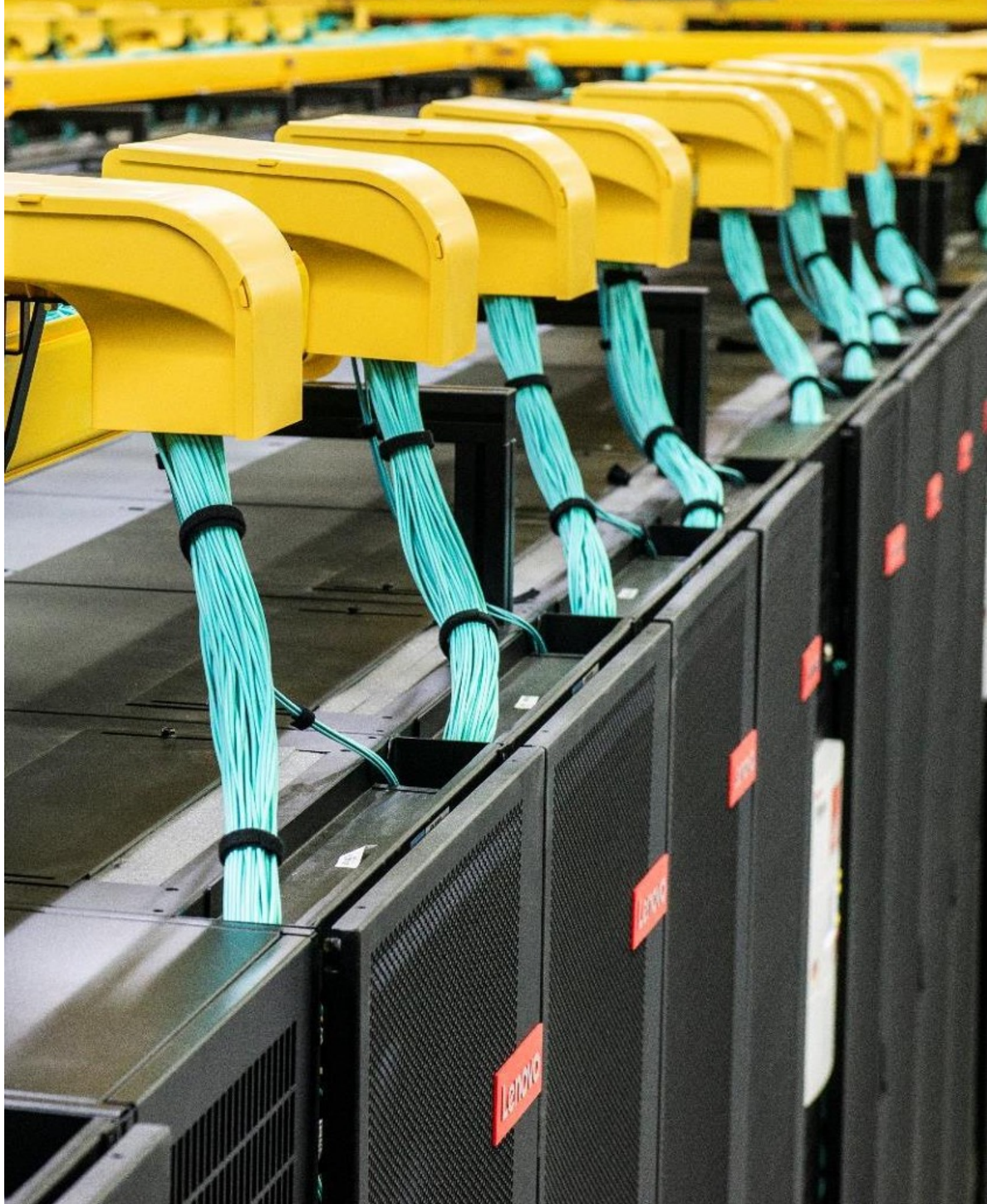

Ohio Supercomputer Center


Mellanox
USP
UNIVERSITY OF SAO PAULO
sgi
intel


天河
天河二号

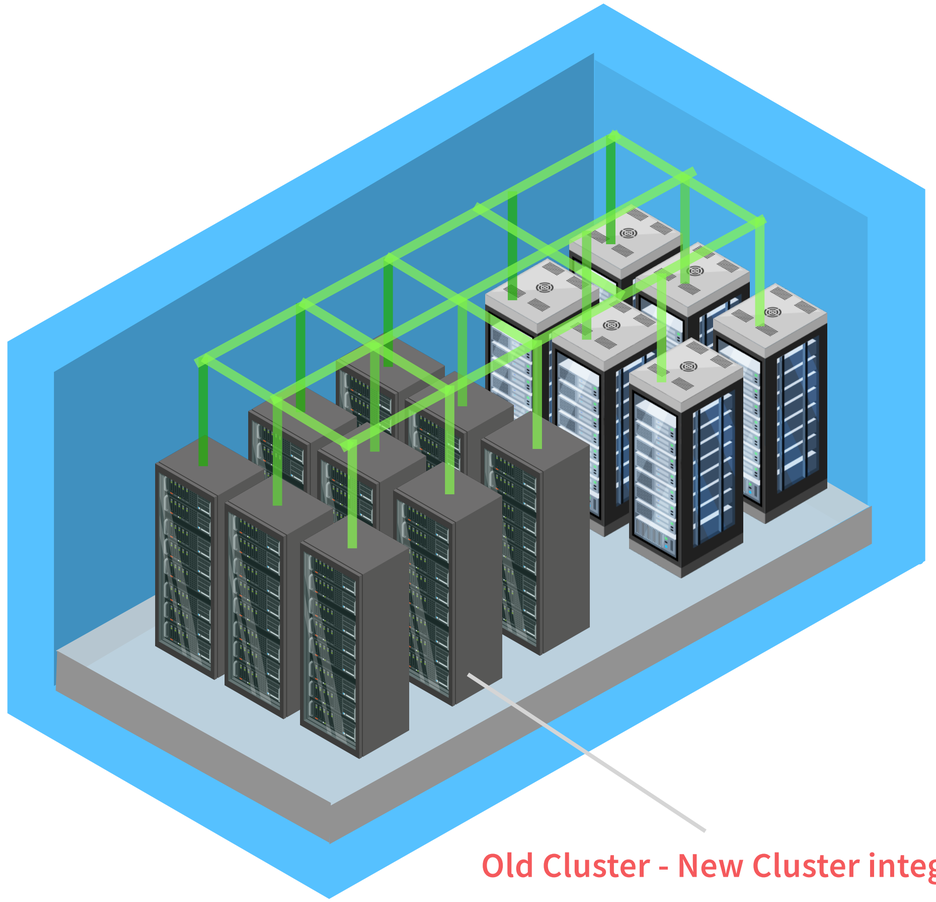
100% focused on HPC

versatushpc.com.br



Versatus HPC

More than 15 years in Professional Services for HPC



Old Cluster - New Cluster integration

- New systems can be appended to an already running cluster

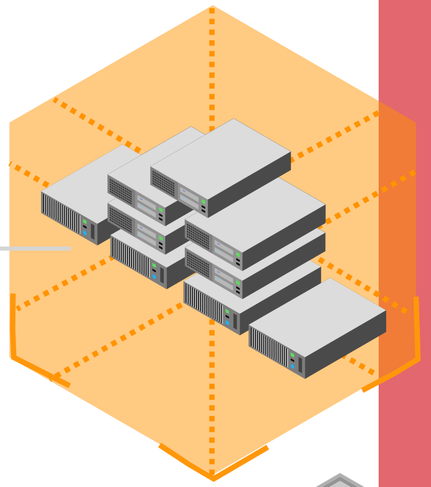
Parallel File System Deployment

Lustre ● GPFS ● BeeGFS

Deployment

Maintenance

Tuning



Remote Administration Service

VHPC-RAS

Environmental Monitoring

Management

Software Installation

User account administration

Help Desk Service

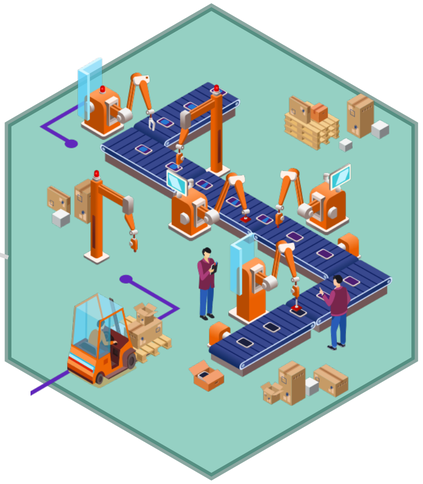
Job Scheduler

OpenPBS ● SLURM ● LSF

Management -

Tuning -

Trouble-shooting -



Success Cases

Latin America

Servicio Meteorológico Nacional
Argentina

Senamhi
SERVICIO NACIONAL DE METEOROLOGÍA E HIDROLOGÍA DEL PERÚ

IGP Instituto Geofísico del Perú
SISMOS PERÚ

Universidad de los Andes

UNIVERSIDAD DE COSTA RICA

Perfomance

Importance of a planned installation

versatus
HPC

Versatus otimiza plataforma HPC do CETENE aumentando seu desempenho em até 30 vezes

Cliente

CETENE
Centro de Tecnologias Estratégicas do Nordeste

CETENE
CENTRO DE TECNOLOGIAS ESTRATÉGICAS DO NORDESTE
UNIDADE DE PESQUISA DO MCTI

Desafios

Centro de referência para o desenvolvimento tecnológico e científico no Nordeste do país, o CETENE depende para diversas de suas pesquisas de cluster computacional que suporte uma elevada carga de processamento e entregue resultados no menor período possível.

Seu cluster, utilizado em projetos de química computacional (adsorção/difusão de medicamentos e dinâmica de reações químicas), aprendizagem de máquina (modelagem de redes neurais com deep learning), análise de estruturas genômicas (detecção de variações genéticas em plantas) e modelagem molecular é fundamental para o bom andamento de grande parte dos projetos científicos da instituição.

Formado por um head node e 6 compute nodes com 4 GPUs em cada, o cluster possuía os nós alocados de

Saiba mais em **versatushpc.com.br**

1

"One of the experiments in the genomic structure analysis project managed to reduce the processing time from 5 days to 4 hours just by using a cluster queue that shared 4 GPUs."

- Jarley Palmeira Nóbrega, Coordinator CETENE

Best practices in HPC?

Best practices in HPC?

- **What are they?**
- **What are they like?**
- **Where are they?**
  - Best practices are not necessarily technical components, such as software and hardware, but also communication and common sense components for the maintenance of your HPC cluster.

Best practices in HPC?

- Architecture
- Hardware
- Network
- User management
- Security
- Queue system
- Process Handling
- Monitoring and observability
- Configurations Headnode
- Modules
- Filesystems
- Compute Nodes
- Communications
- Users relations

Architecture

“The practice of aggregating computing power, so that the performance capacity is immensely greater than a common desktop”

Scientific Application

Libraries

- Numerical
- Communication (MPI)
- I/O
- CUDA

Compilers and Debbugers

Job Scheduling

Computational resources manager, distribute jobs across the system

O.S.

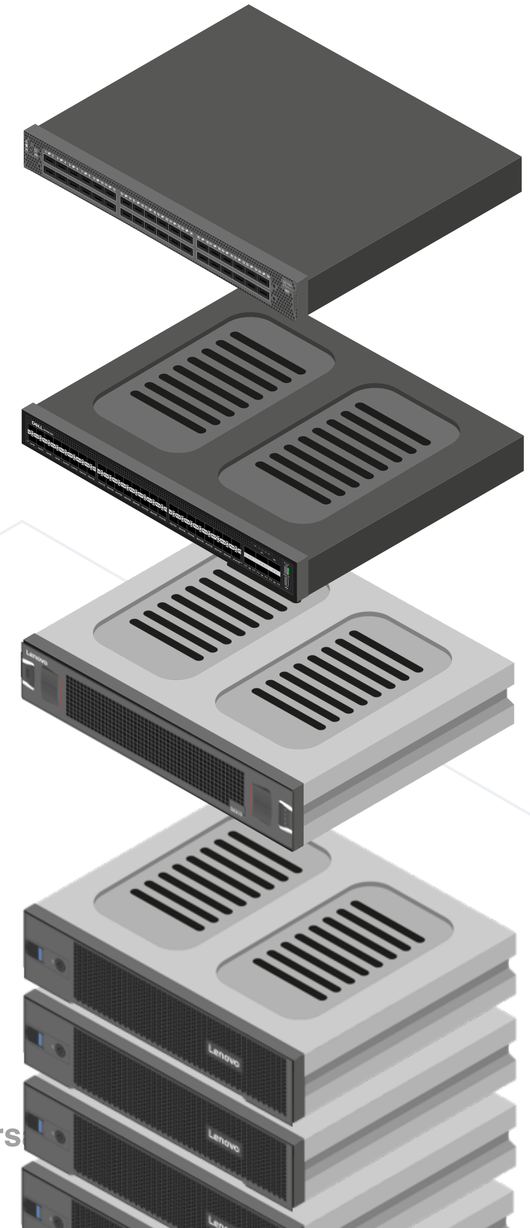
Orchestrator

S.O. image management, provisioning and base networking

Parallel File System

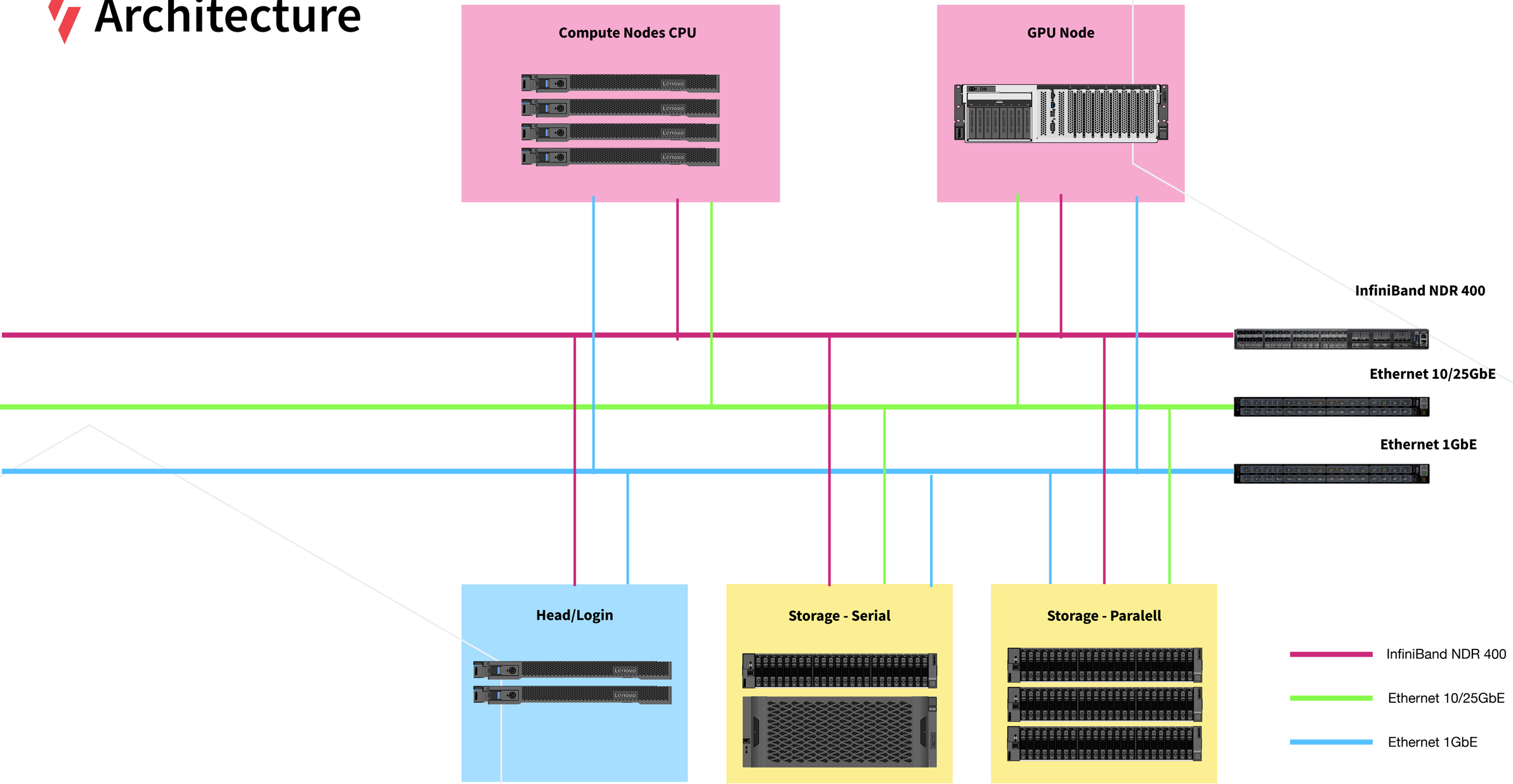
Specialized SDS for HPC environments, High Throughput and Low Latency

Architecture

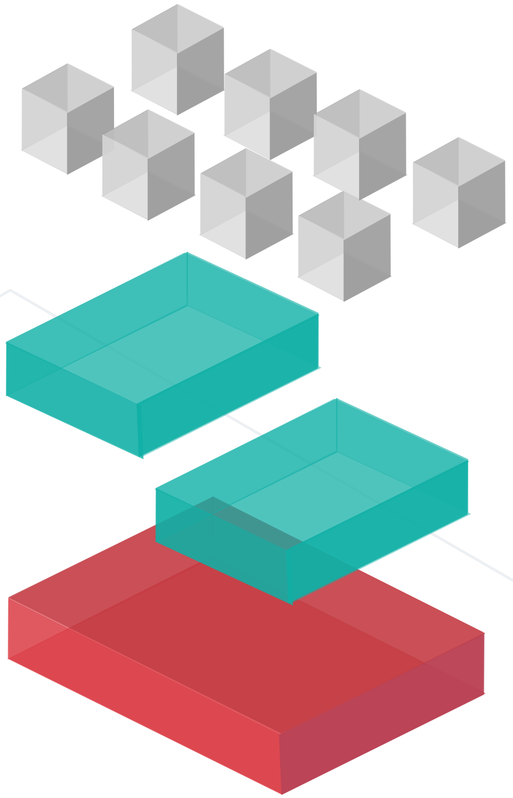


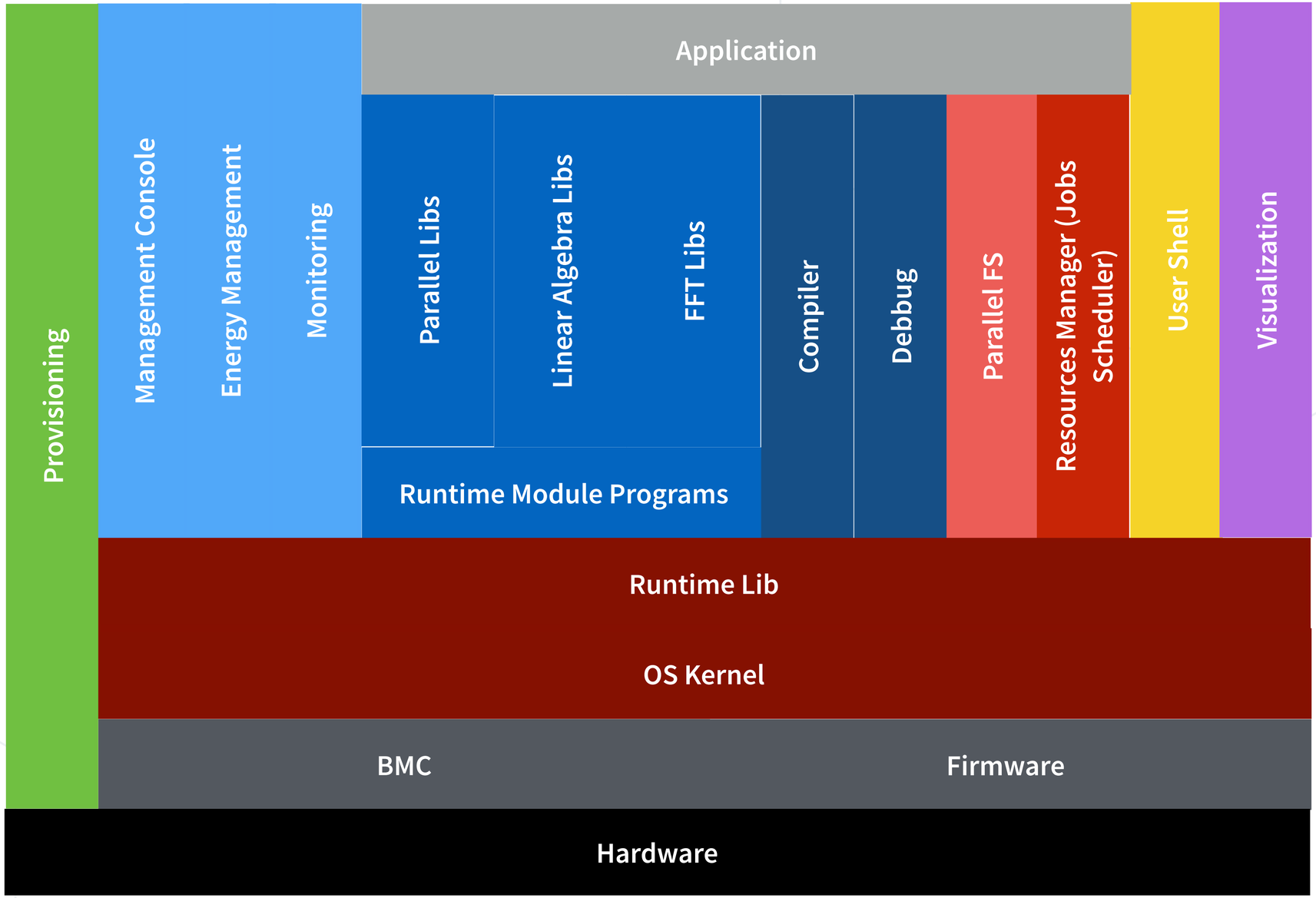
Low Latency Network	Calculation and Storage (PFS)
Management Network	Provisioning, Monitoring, Management
Head Node	Image Management, Job Scheduler, Overall Management and Control
Compute Node	Calculation

Architecture



Intro

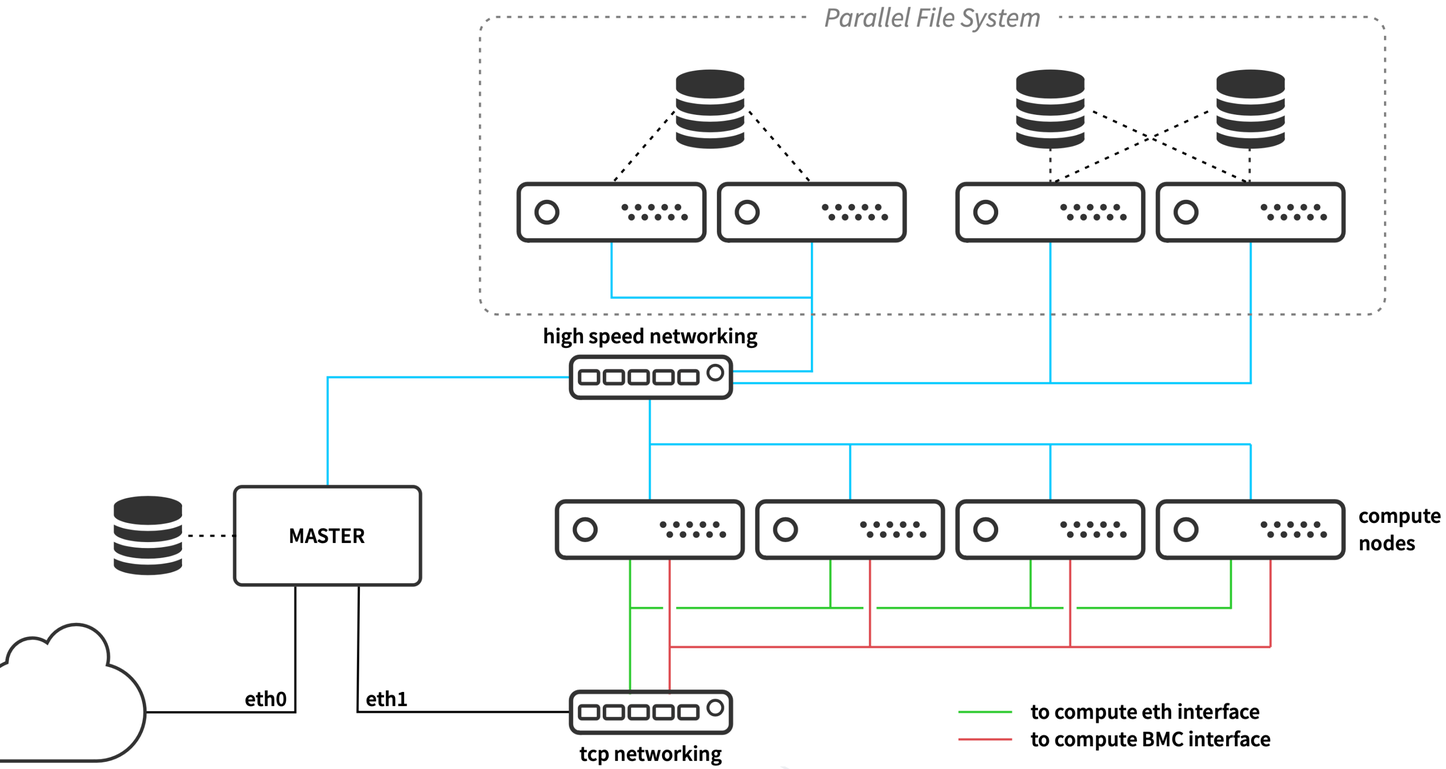



Application
Provisioning
Management Console
Energy Management
Monitoring
Parallel Libs
Linear Algebra Libs
FFT Libs
Runtime Module Programs
Compiler
Debbug
Parallel FS
Resources Manager (Jobs Scheduler)
User Shell
Visualization
Runtime Lib
OS Kernel
BMC
Firmware
Hardware

Hardware

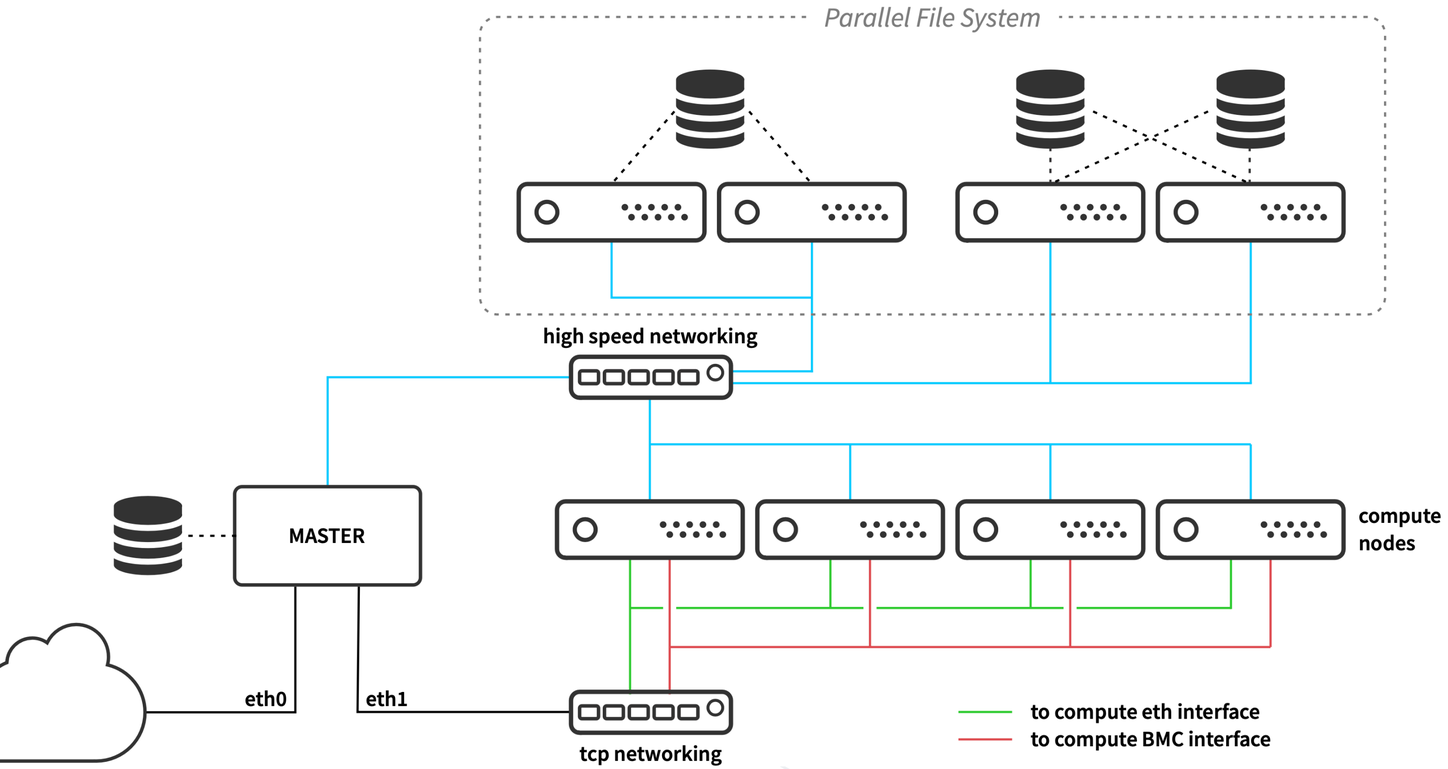
- **Use Enterprise Hardware**
- If possible, use different servers for the main node and the login node. You don't want your entire cluster to collapse because a user ran an inappropriate application.
- Use the BMC extensively, if available.
- Redundancy: Use RAID and redundant power supplies on critical systems: main nodes, login nodes, and storage nodes.

Network



- Use an internal domain of your external domain.
- Do not use external DNS servers.
- Integrate DHCP and DNS.
- In situations where you have multiple management nodes, have multiple DNS servers.
- Delegate the zone (NS) from your external domain to your cluster.

Network



- Do not expose internal IP addresses
- Network segregation: Service, Management, Application, External, Storage.
- Use network bonding on critical servers: main node, login nodes, visualization nodes.
- Do not use Jumbo Frames in management and service networks. Consider its use for storage networks over Ethernet.
- Use a low latency network only where it really shines: MPI, RDMA.

User management

- Do not use local Unix authentication.
- Have an LDAP system in operation.
- Use MFA (multi-factor authentication) if possible.
- Include the email and phone number of your users in LDAP.

User management

- Do not overcomplicate the LDAP system.
- Establish clear procedures to aggregate and remove users to maintain security and compliance: LDAP, Queue System, Mailing List, Resource Monitoring, Web Portals, Observability Platform.

Communications

- Communications in /etc/motd.
- Use the wall command to announce shutdowns, whether emergency or planned.
- Have a mailing list for your internal users.
- Have examples of job scripts in a shared NFS directory, preferably in /opt.
- Keep documentation and frequently asked questions in an internal user Wiki or have a basic user manual.

Referencia completa del comando "`sacct`" (en inglés): https://slurm.schedmd.com/sacct.html. Com o comando "`sreport`" é possível gerar estatísticas conjuntas de uso dos jobs e do *cluster*. Para entrar no modo interativo dos comandos "`sreport`" basta executar o comando:

```
[root@mmgt01 ~]# sreport
sreport:
```

Como ejemplo de uso del comando, puedes ejecutar el siguiente comando para mostrar el uso total del clúster, en términos de minutos de CPU, del día anterior:

```
sreport: cluster utilization
```

Referencia completa del comando "`sreport`" (en inglés): https://slurm.schedmd.com/sreport.html.

13.3 Colas

Las colas son las que permiten un buen rendimiento, distribución y programación del uso de recursos. En HPC UCR solo tenemos las colas configuradas que se enumeran a continuación:

Colas	Tiempo límite	Nodos asociados
cpu	Ilimitado	cn[001-016]
gpu	Ilimitado	cngpu[001-002]

13.4 *Scripts* de Envío

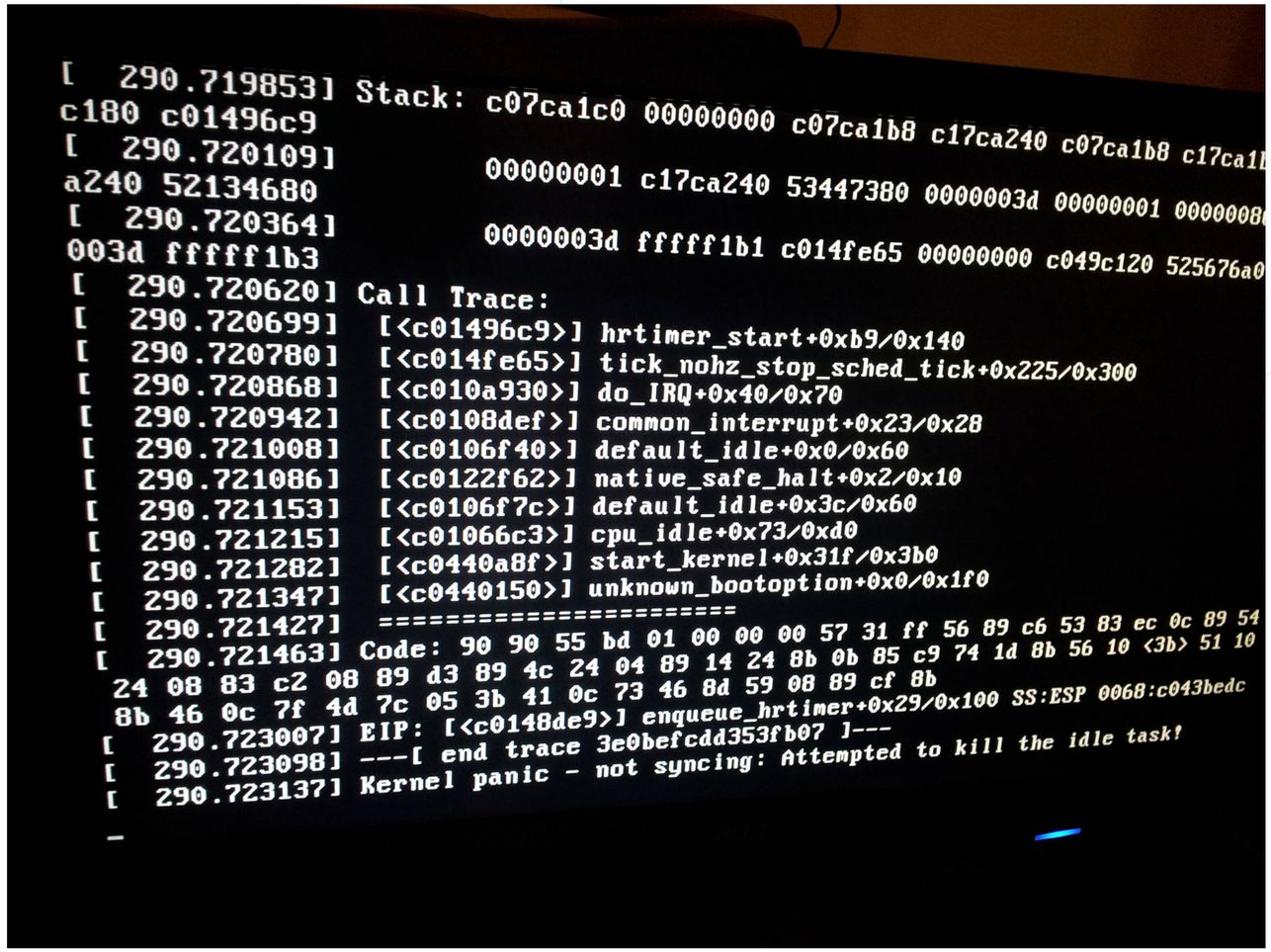
Un ejemplo de *script* de envío de trabajos está ubicado en:

`/opt/versatushpc/data/jobscripts/`

El *script* `template.slurm` puede ser utilizado como modelo para la creación de *scripts* personalizados para los *softwares*.

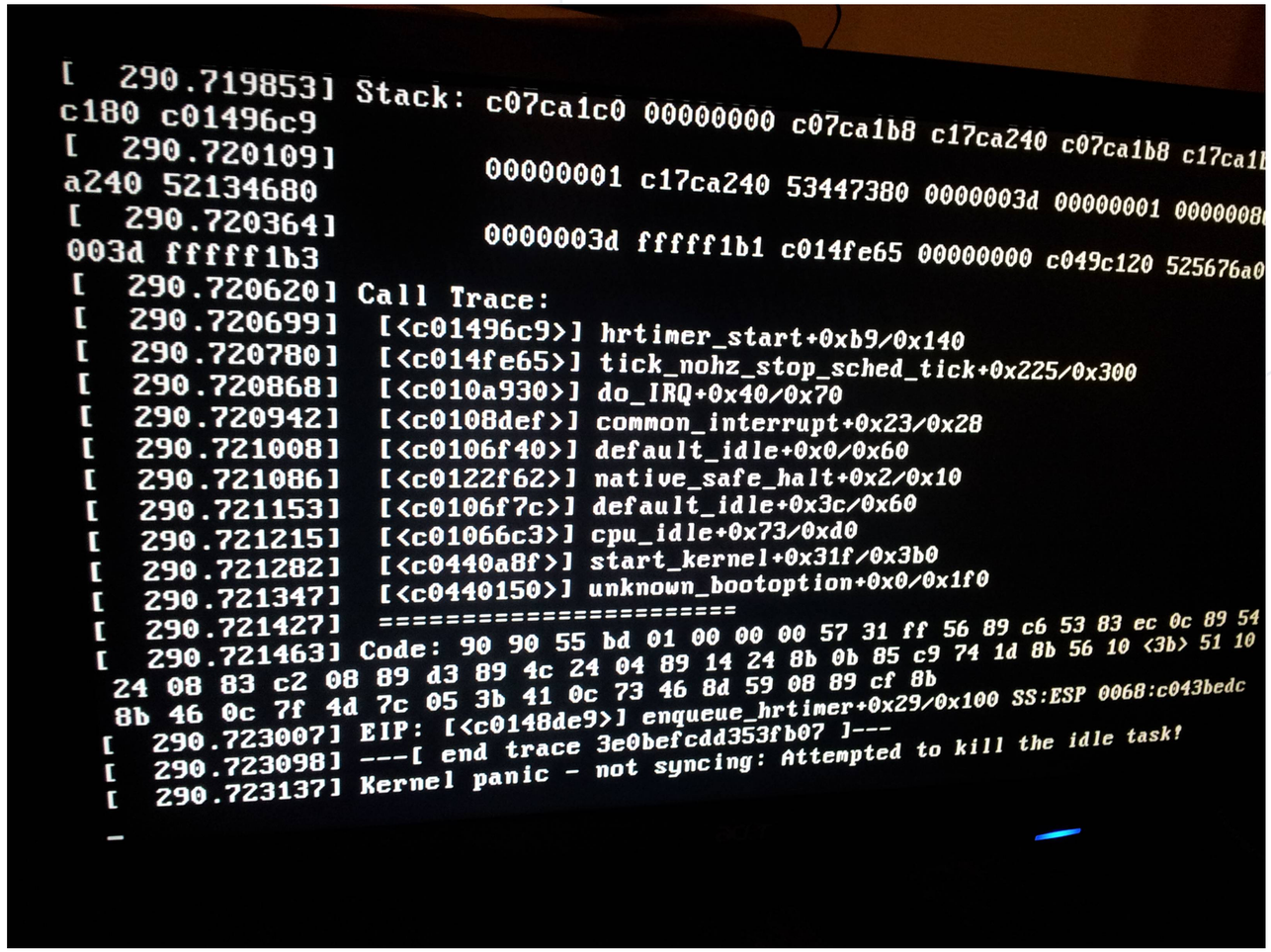
Security

- Firewall on the main and login nodes.
- Use fail2ban.
- SELinux on the main node and login nodes. Makes custom rules made specifically for the system. We don't just put everything as public_content_rw_t.
- Do not overshadow the services: change the SSH port, change the queue manager port, block users from logging in. This will only add complications and unavailability for its users.



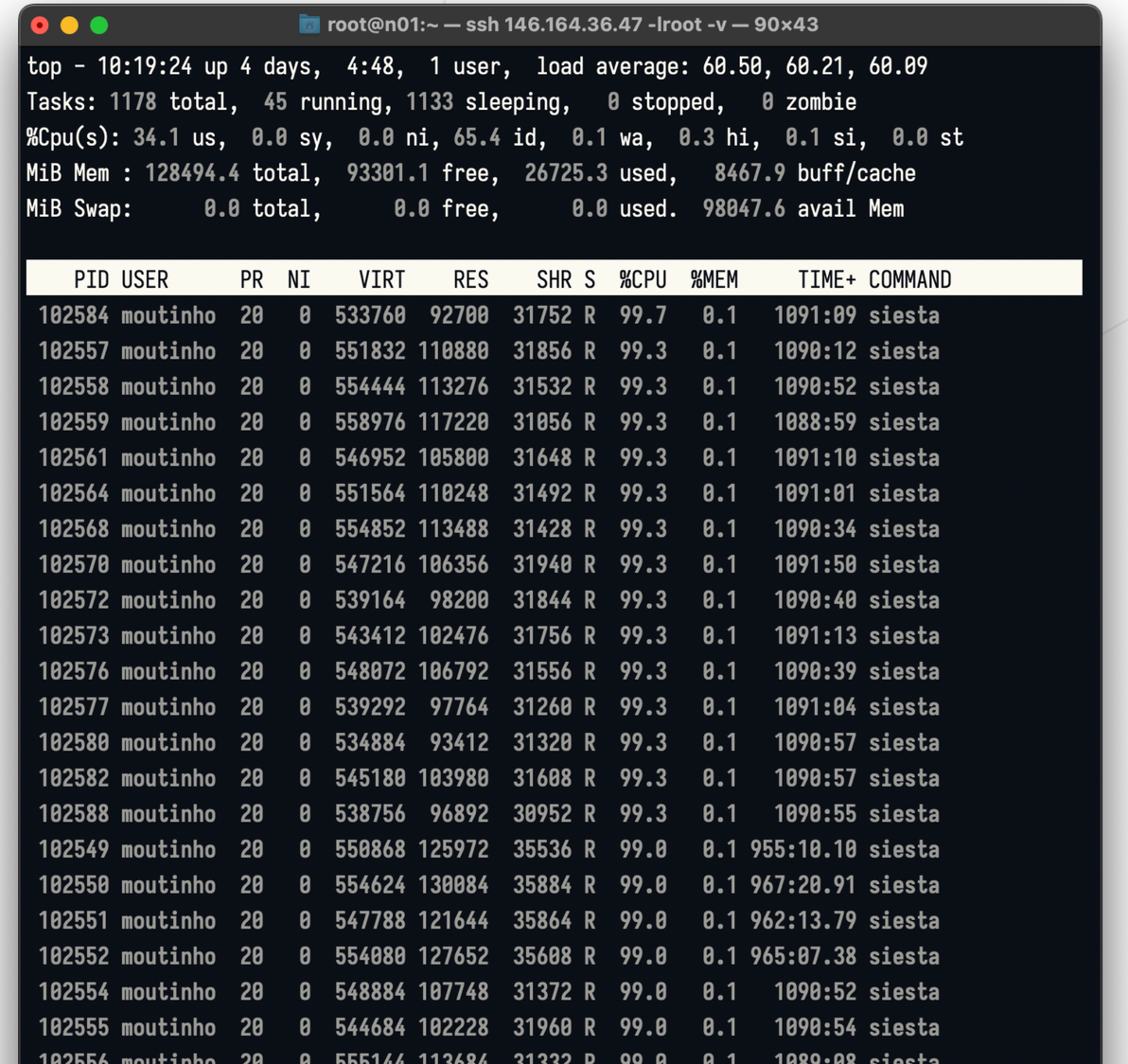
Security

- Use SSH key pairs, disable password authentication for SSH. But allow password change for other services.
- Today certificates are free. Use Let's Encrypt.
- If you are in a shared root environment, use root audit.
- If you are in a high-security environment, also enable the audit of user processes and commands.

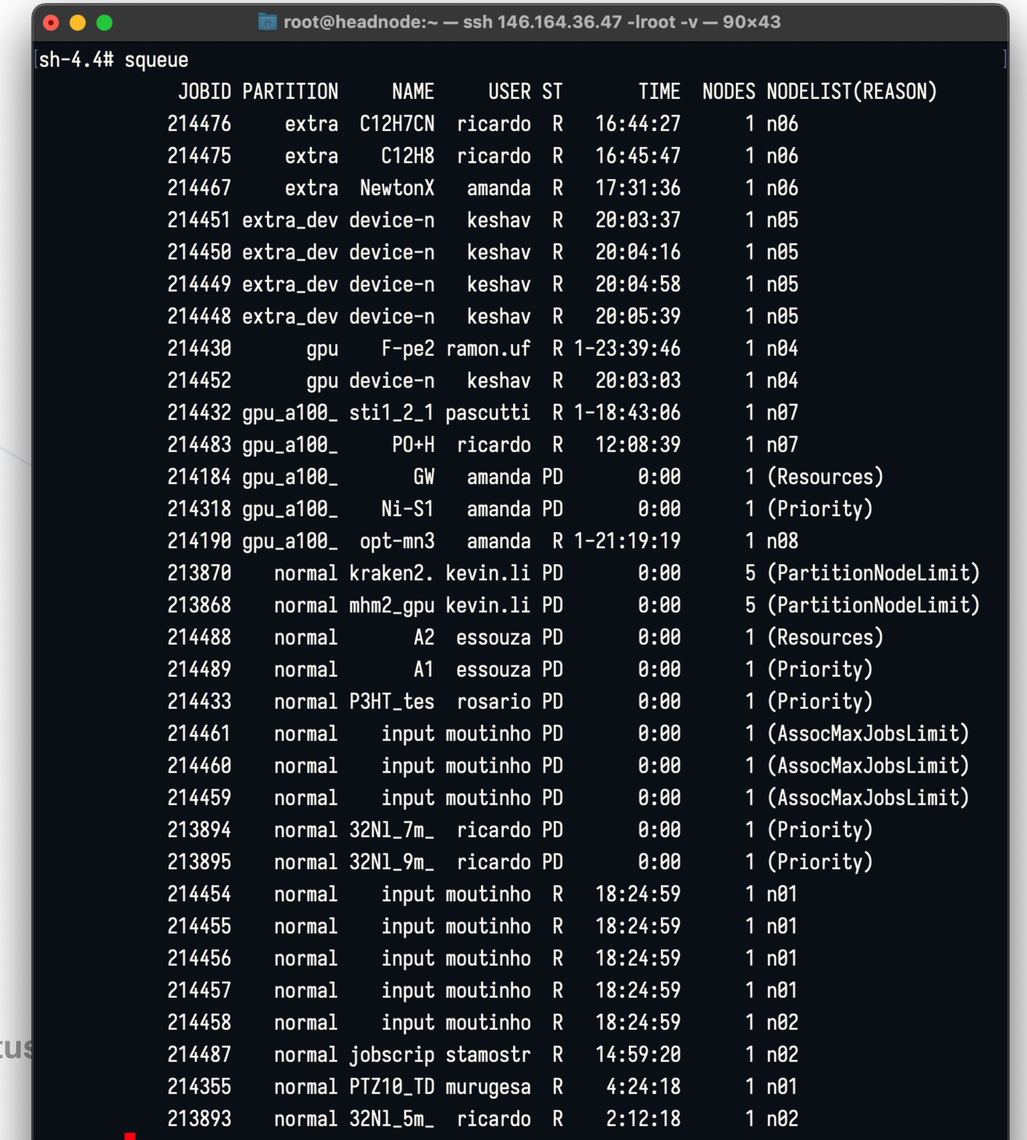


Process Management

- Cgroups at login nodes to avoid abuse
- Ulimit on login nodes to limit resources, but allow unlimited on compute nodes
- Hiding processes at login nodes: mount -or remount,rw,hidepid=2 /proc

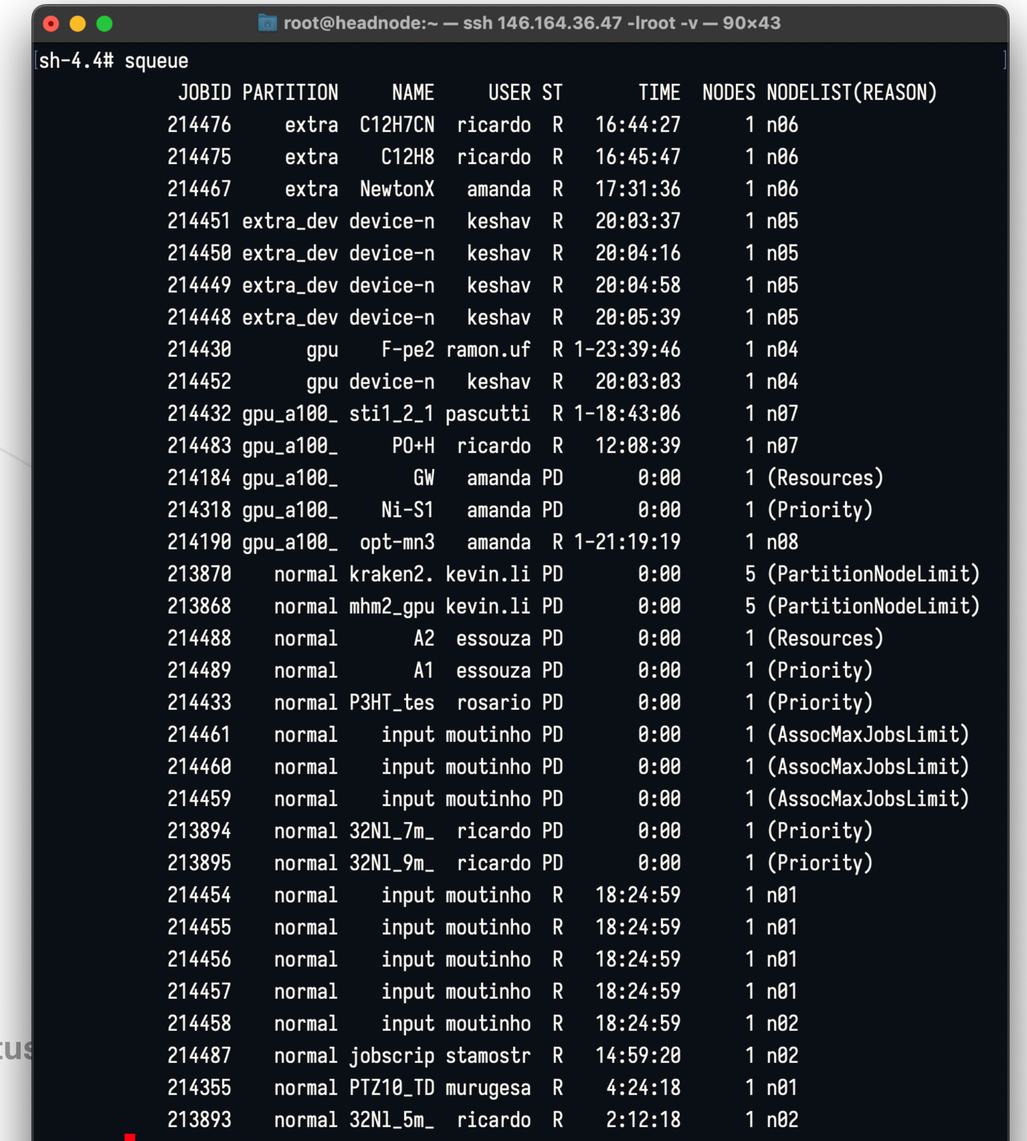


Job Scheduler



- Have priority queues for small and fast jobs, and enforce your time limit.
- Ensure that everything runs through the queue system: disable SSH to compute nodes without jobs or assignments.
- Consider the applications your users are running to adapt your queue strategy.

Job Scheduler



- Enable accounting to keep track of what is used.
- Use at least one fairshare model. FIFO is not enough.

Monitoring and Observability

- For administrators and maybe users
  - Graphics system
  - Dashboards

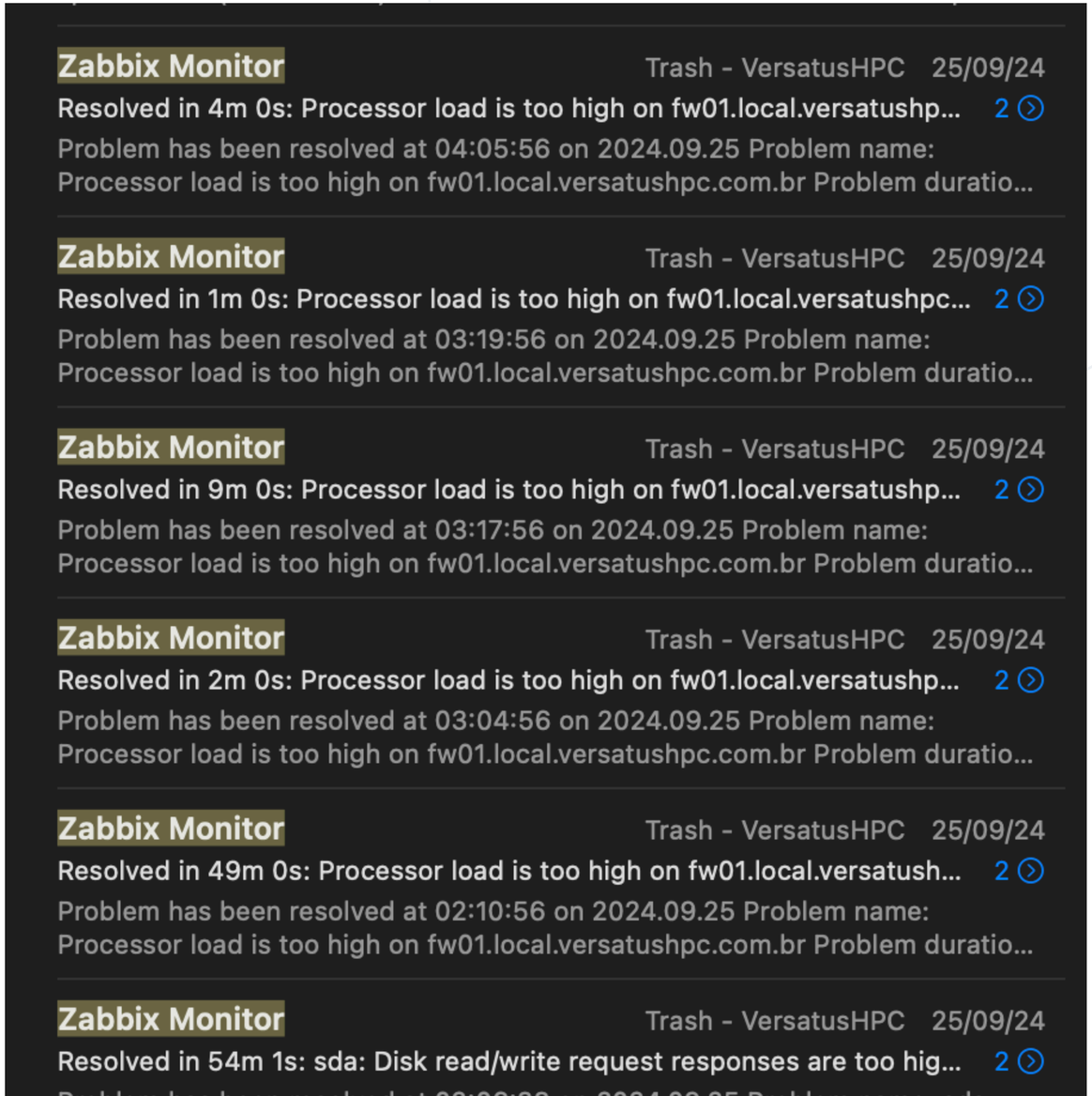
- Only for administrators
  - Alerts

- Traditional monitoring is not effective in HPC environments! You have to make customizations

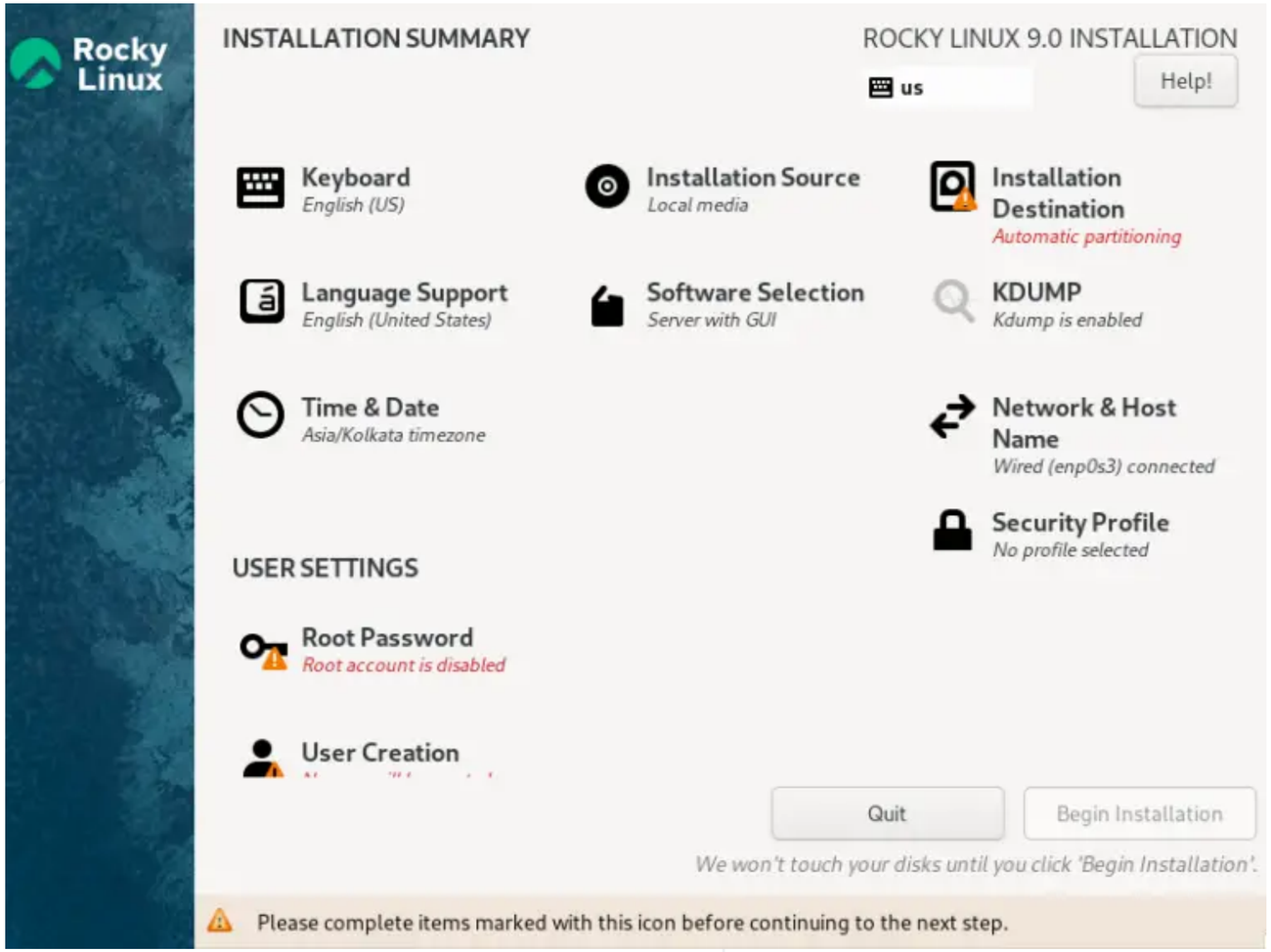
- Use remote syslog and preferably with a log analysis tool.

Monitoring and Observability

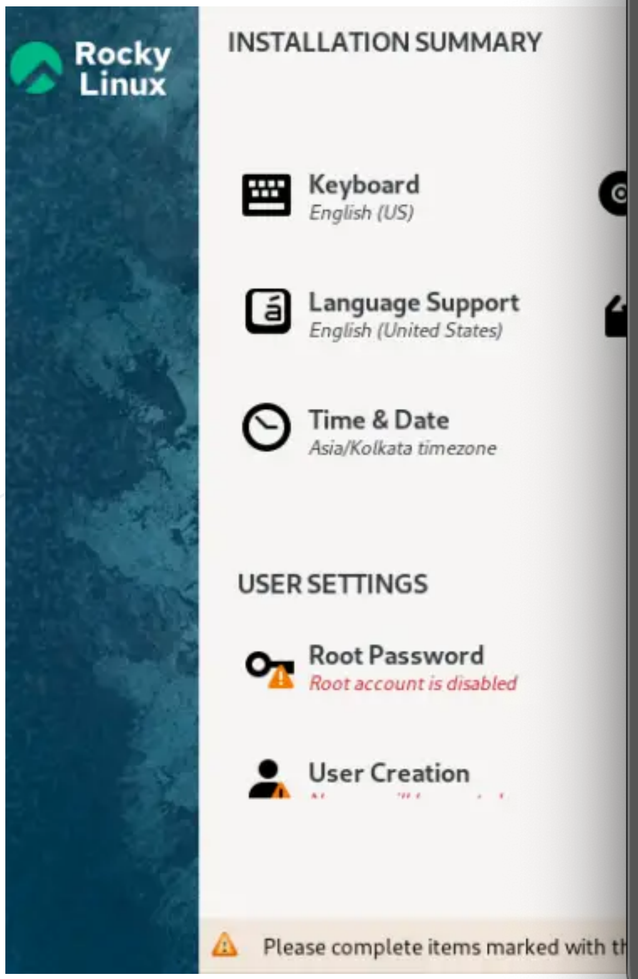
- Don't exaggerate/saturate notifications!
- Yes, the CPU will reach 100% utilization in an HPC system.
- Yes, the average load will be high for long periods of time.
- Yes, the server will run with little free memory for extended periods.
- Yes, sometimes the OOM killer will be activated and that's fine.



Main Node Settings



- Use LVM's "thin provisioning" to easily manage your available space.
- Snapshot of your main node with LVM.
- Do not allocate all disk space of the main node for no reason.
- Don't be a perfectionist with partitioning. Fewer partitions is better.



Example Sizing for a 1TB Disk:

1. **/boot** – 1GB
2. **/** (root) – 15GB
3. **/home** – 100GB
4. **/var** – 20GB
5. **/var/log** – 20GB
6. **/tmp** – 5GB
7. **/usr** – 20GB
8. **/opt** – 10GB
9. **/srv** – 50GB (if running web, FTP, or other services)
10. **swap** – 16GB (if you have 8GB RAM, adjust based on system memory)
11. **/data** – 743GB (for primary data storage, encrypted if needed)

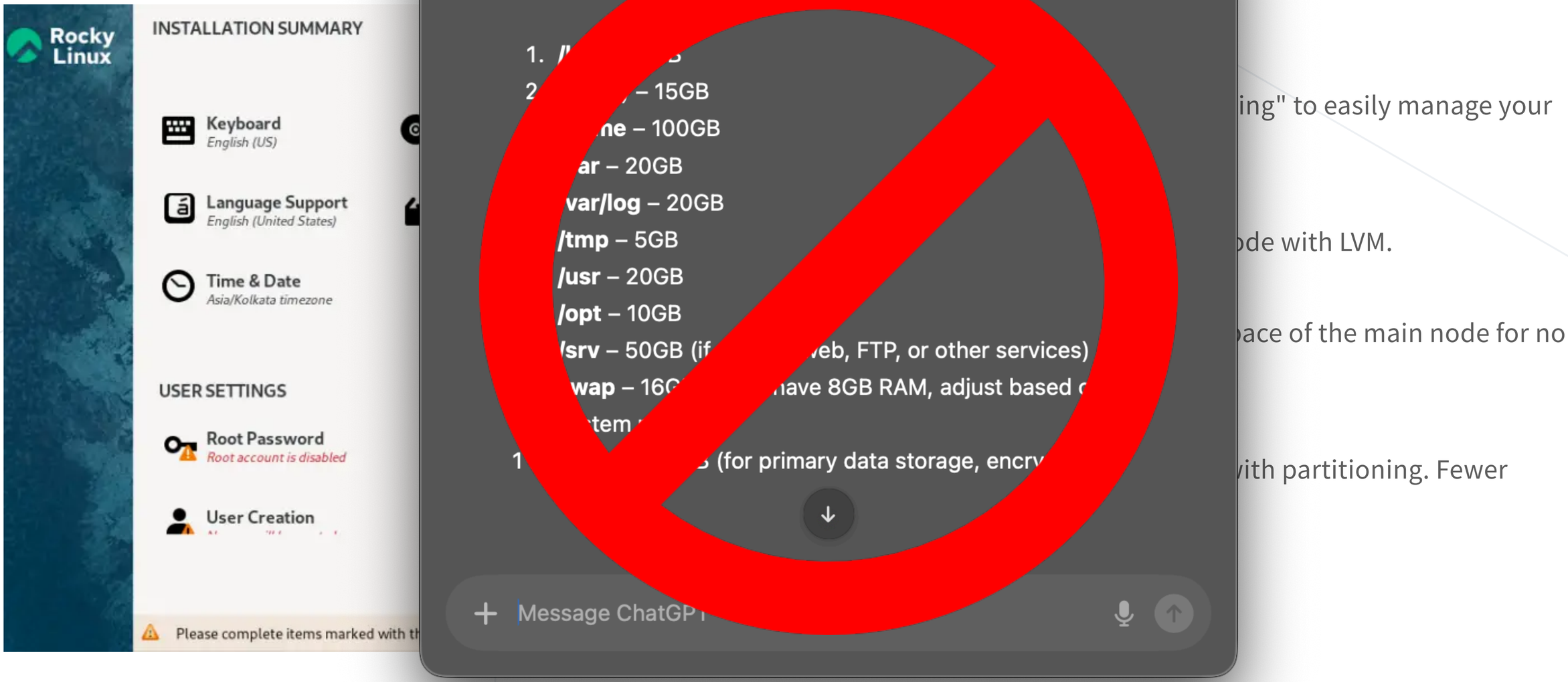
Message ChatGPT

ing" to easily manage your

ode with LVM.

pace of the main node for no

vith partitioning. Fewer



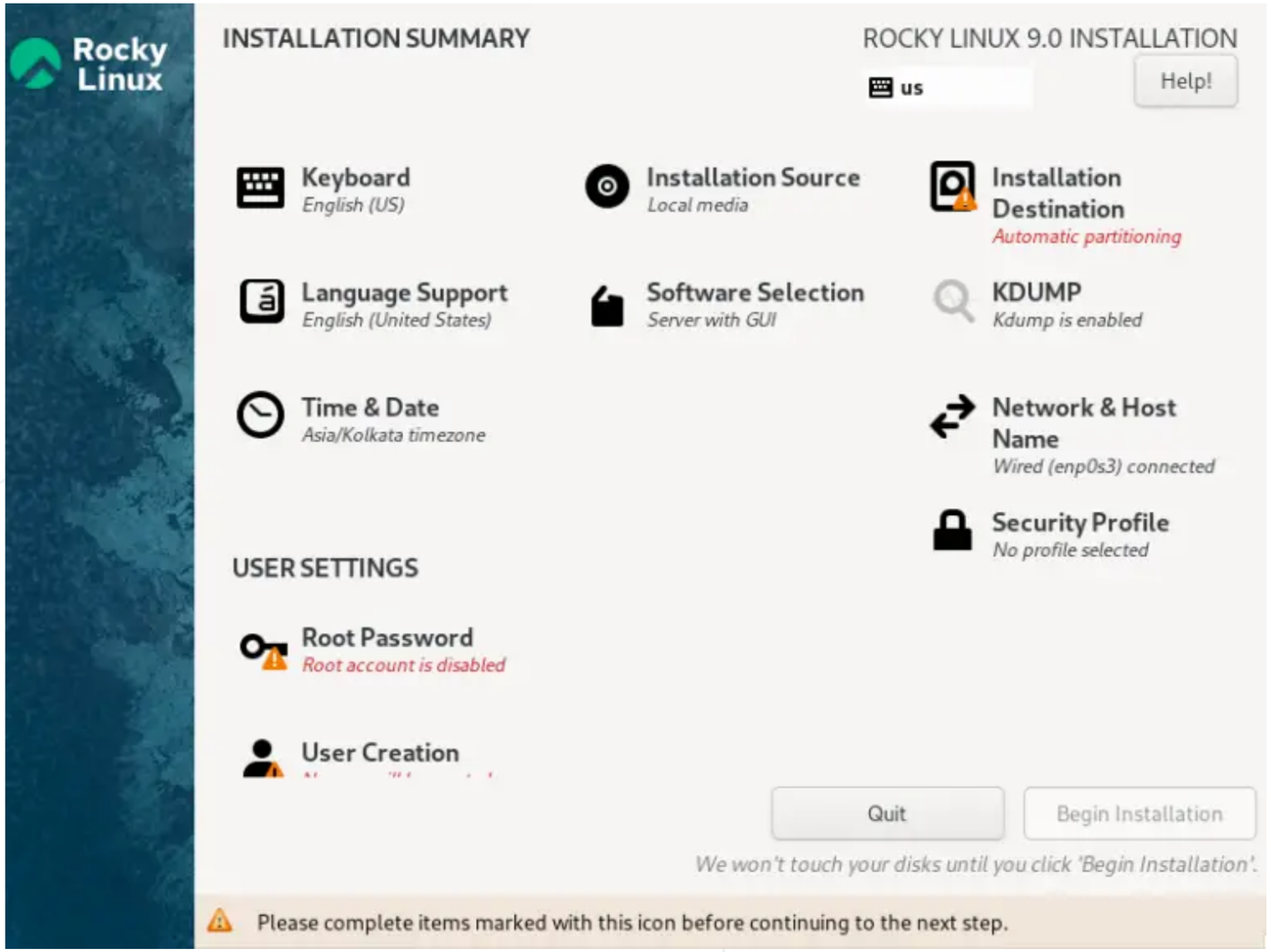
ing" to easily manage your

ode with LVM.

pace of the main node for no

vith partitioning. Fewer

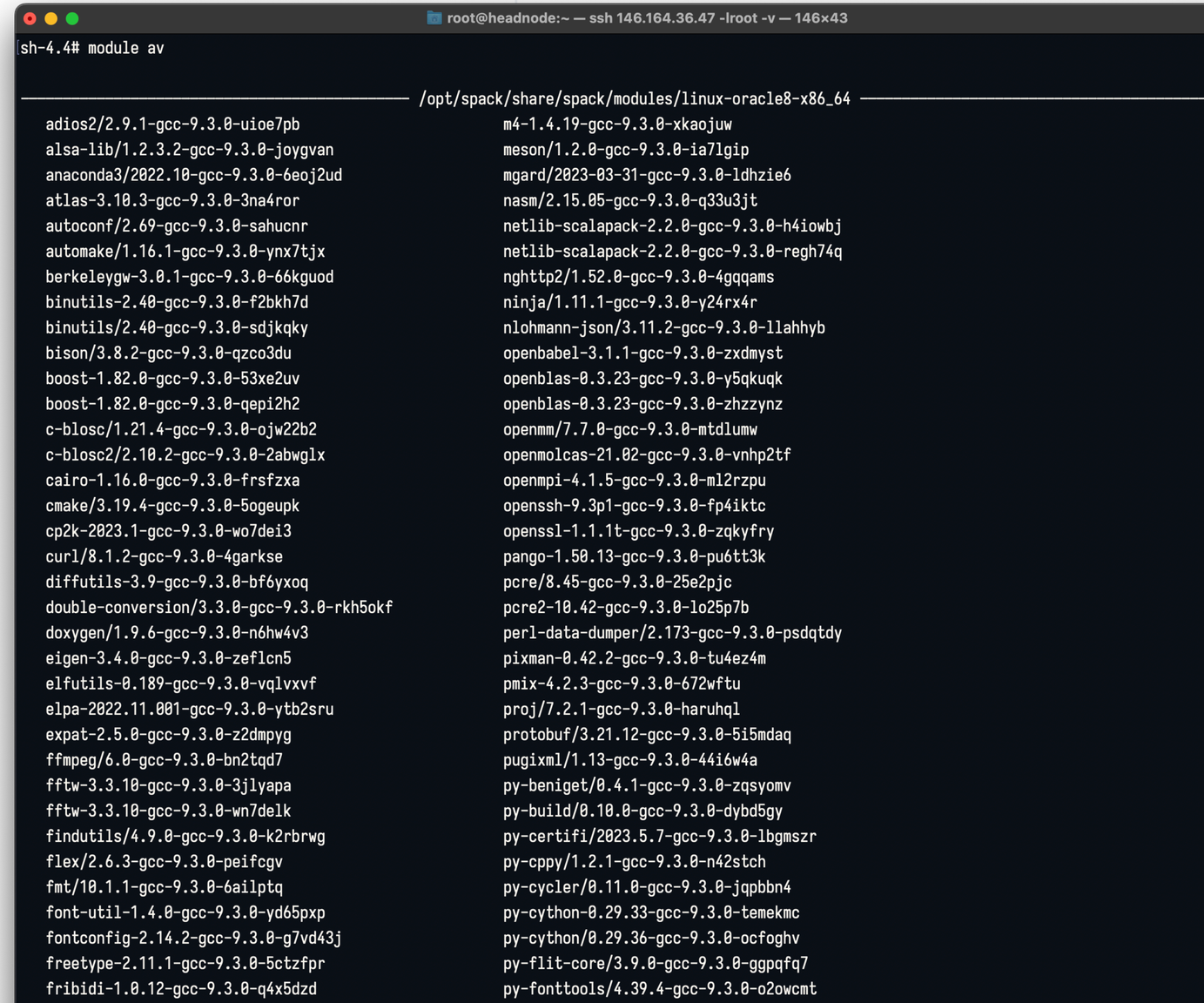
Main Node Settings



- NTP clock synchronization with a local layer; we don't care if the clock is not 100% perfect, but it must be consistent throughout the cluster.
- Correct mail system integrated in the queue manager.
- Have a local mirror for your distribution.

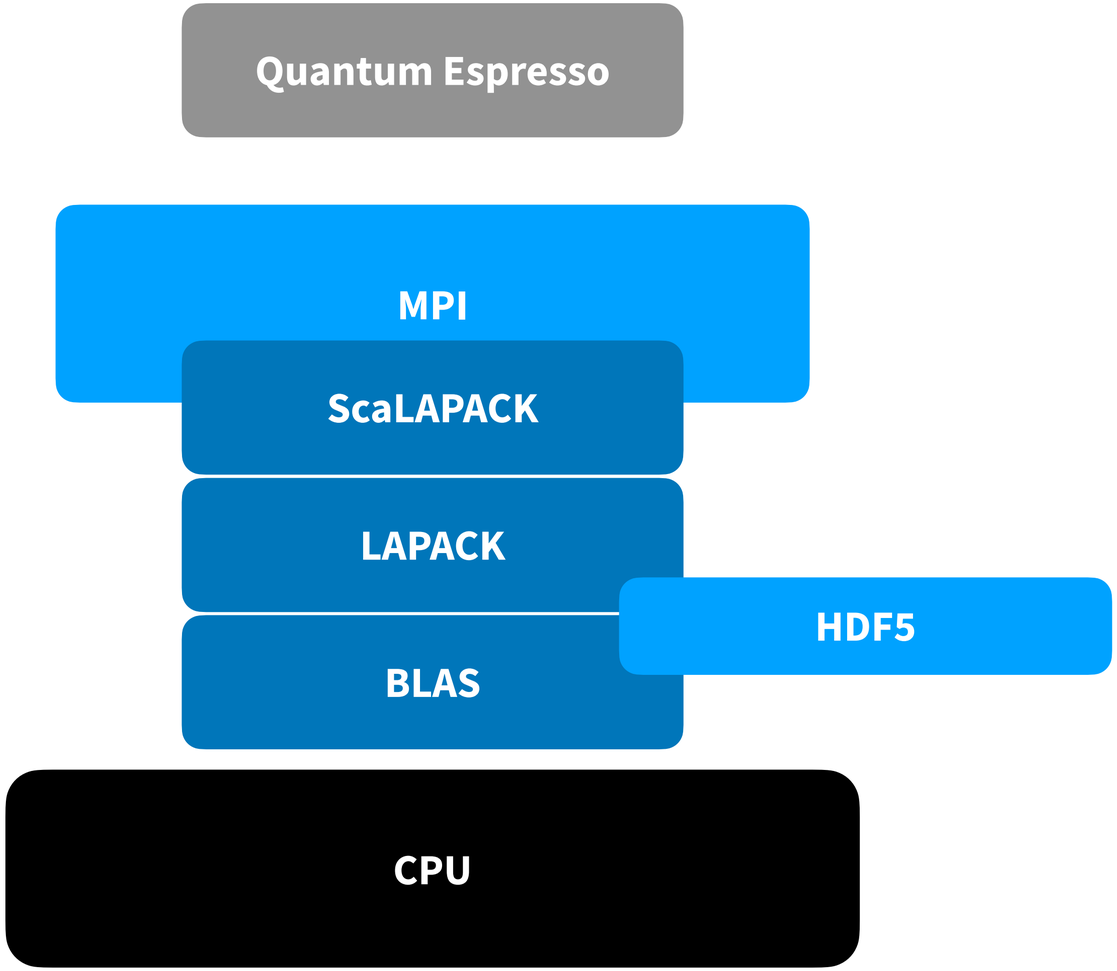
Modules

- Use modules, period.
- Configure default HPC modules to be enabled by default.
- Always add software through modules.
- Avoid using the system package manager (yum / dnf / apt) for user applications.

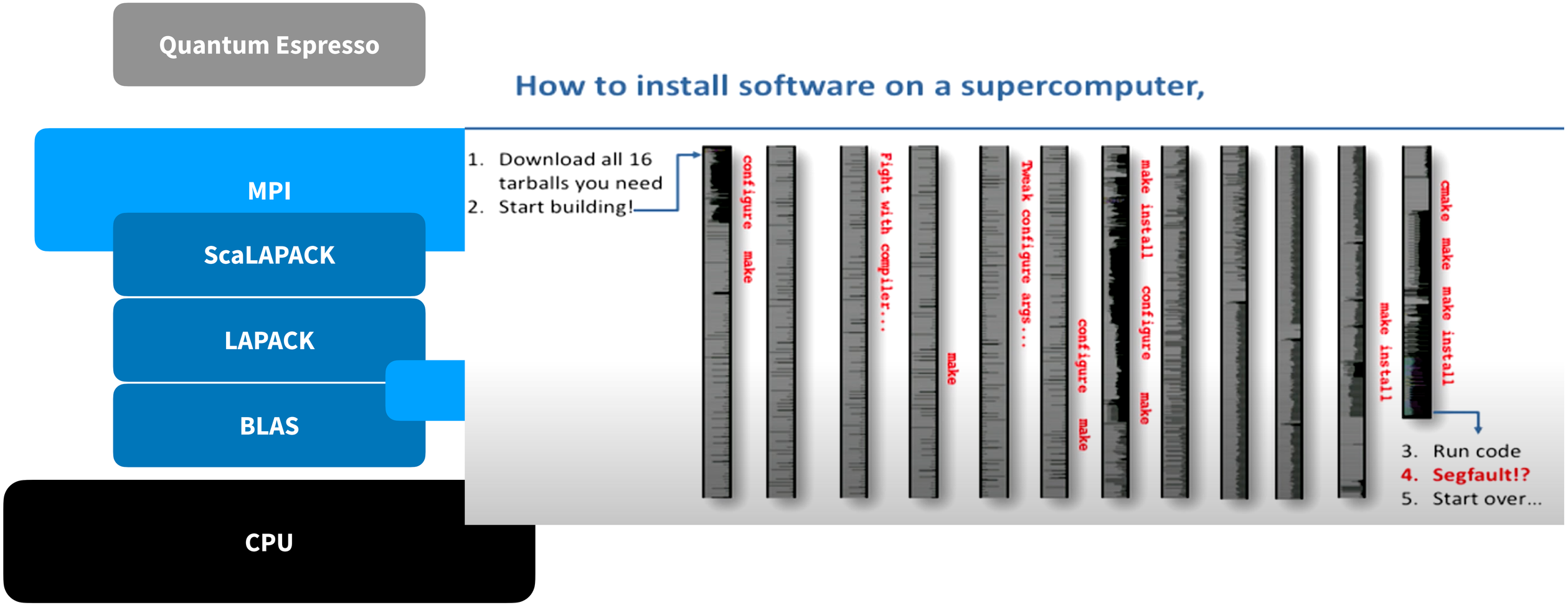


Compilation Example

Scientific Application

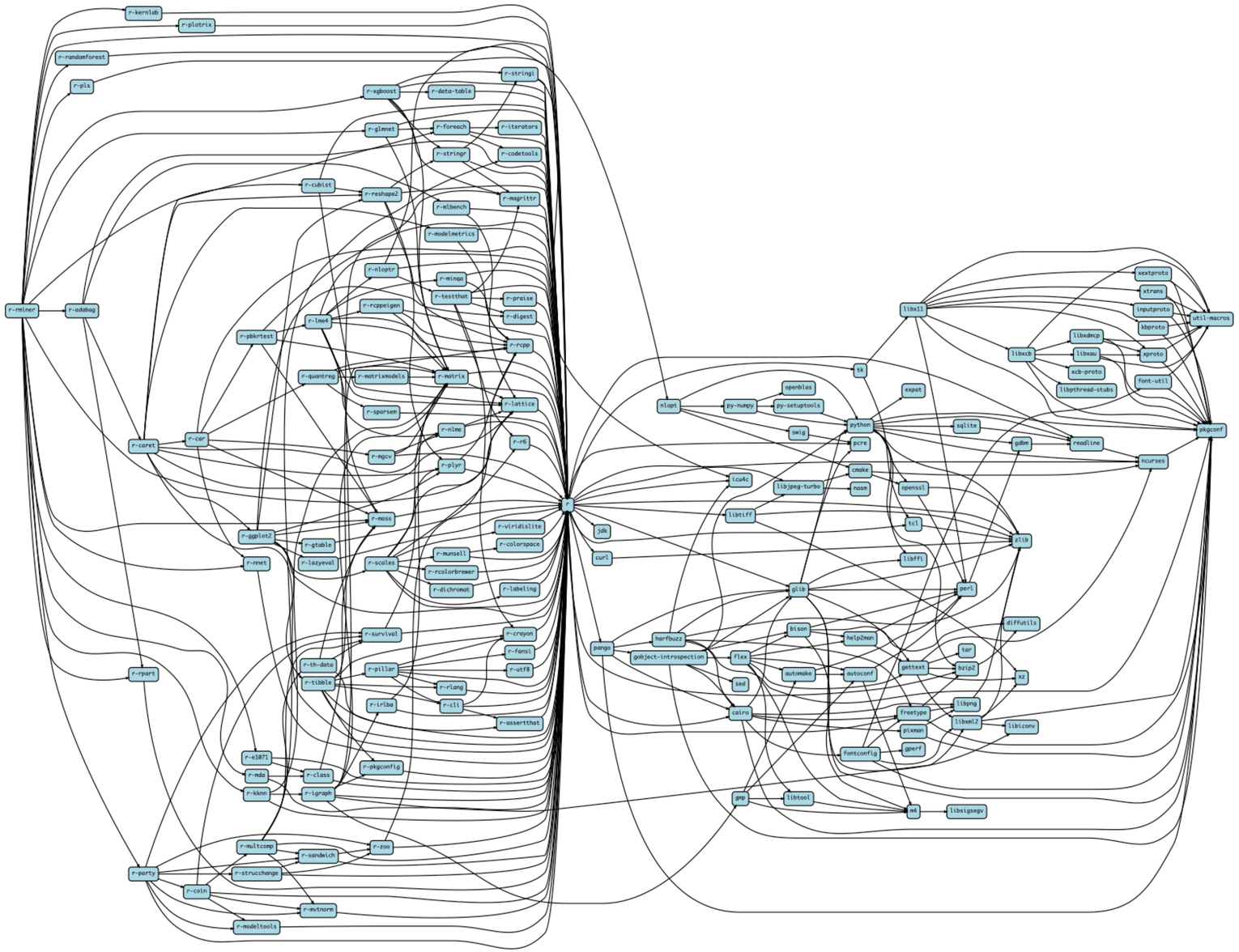


Compilation Example



Scientific Application Management

Example



150 dependencies.

Scientific Application Management

SPACK
Scientific application manager for HPC environments

- Target
- Compiler
- MPI
- Linear Algebra
- Flags
- “Variants"

Scientific Application Management

SPACK

Scientific application manager for HPC environments

Query packages

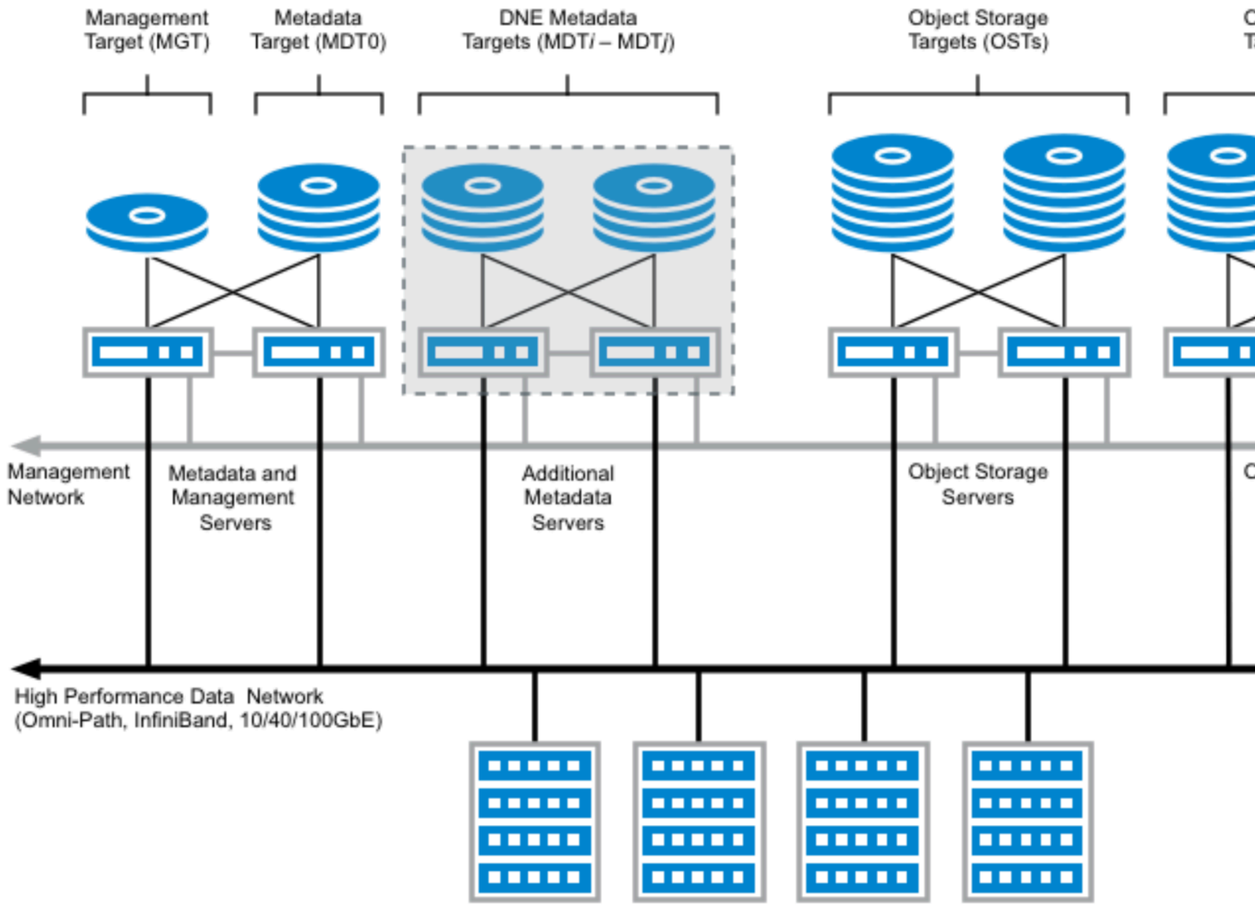
`spack info` package	get detailed information on a particular packages are on disk as installed
`spack dependencies` package/package_specs	show dependencies of a package
`spack dependents` package	show packages that depend on another
`spack find [-ldvf] [package]`	list and search installed packages
`spack graph` package/package_specs	generate graphs of package dependency relationships
`spack list`	list and search available packages
`spack location [-i, --install-dir] [-p, --package-dir]` package	print out locations of packages and spack directories
`spack providers` virtual_package	list packages that provide a particular virtual package

Build packages

`spack clean [-a, --all] [-s, --stage] [-p, --python-cache] [package(s)]`	remove temporary build files and/or downloaded archives
`spack gc [-y, --yes-to-all]`	remove specs that are now no longer needed
`spack install [-j, --jobs JOBS] [-v, --verbose] [--cache-only] [--source]` package_specs	build and install packages
`spack setup [-v, --verbose]` package_specs	create a configuration script and module, but don't build
`spack spec [-l, --long] [-I, --install-status]` package(s)	show what would be installed, given a spec
`spack uninstall [-R, --dependents] [-y, --yes-to-all]` package_specs	remove installed packages

File System

- Use NFS with automount to enable multiple connections.
- NFS with RDMA if available.
- Use NFSv4. Do not use NFSv3 or NFSv2.
- If possible, have separate devices for /home and /scratch (avoid resource depletion).
- Do not use a non-HPC file system in an HPC environment: Use traditional file systems such as ext4, xfs, and NFS or parallel file systems such as pNFS, Lustre, BeeGFS…



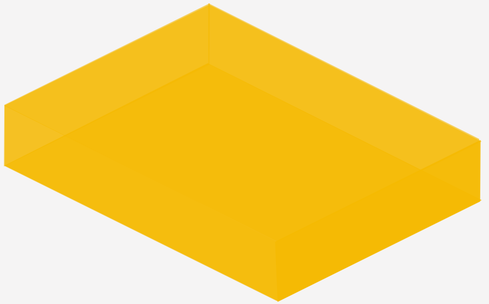
https://wiki.lustre.org/images/a/a3/Lustre_File_System_Overview_(DNE)_lowres_v1.png

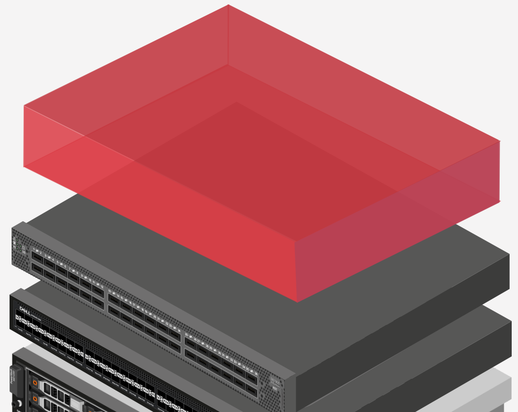
File System

- High performance read / write
- High capacity
- Low latency
- Parallel file access
- Scalability
- Cost efficiency (price per TB | price per GB/s)

Parallel File System

Specialized for HPC, High Throughput and Low Latency



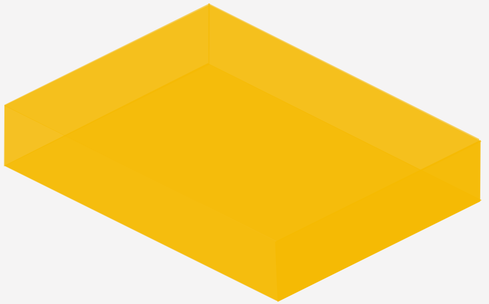


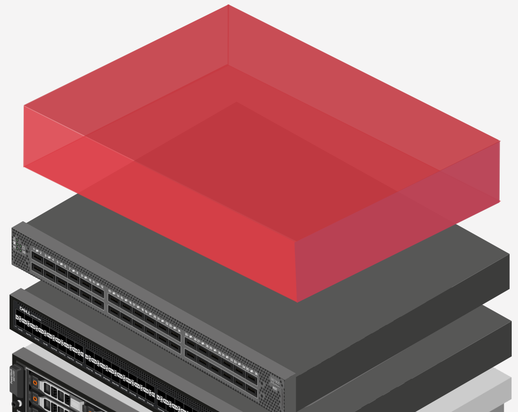
File System

- Compute Nodes need access to the same data at the same time
- Best option for scale out - capacity and performance
- Less bandwidth bottlenecks
- Specialized protocols optimized for performance and parallel file access

Parallel File System

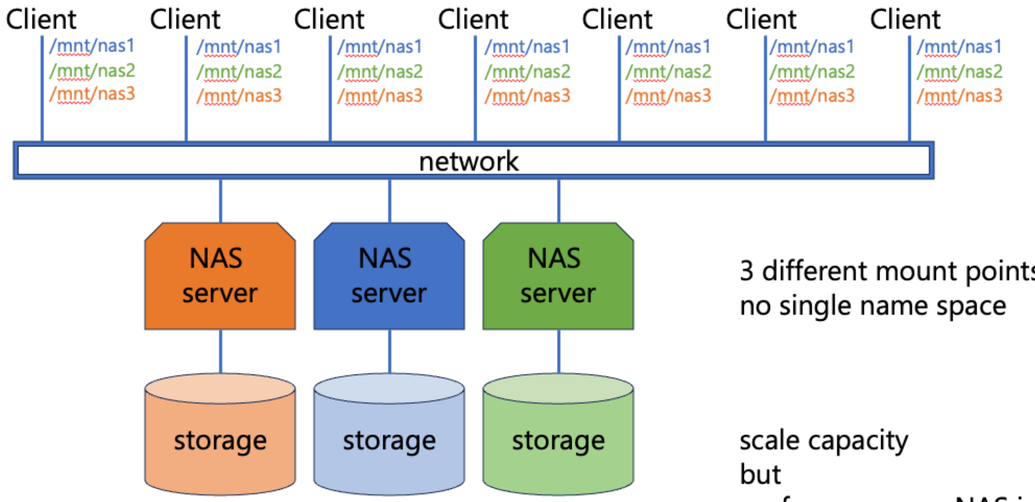
Specialized for HPC, High Throughput and Low Latency





NAS vs PFS

What are the differences?

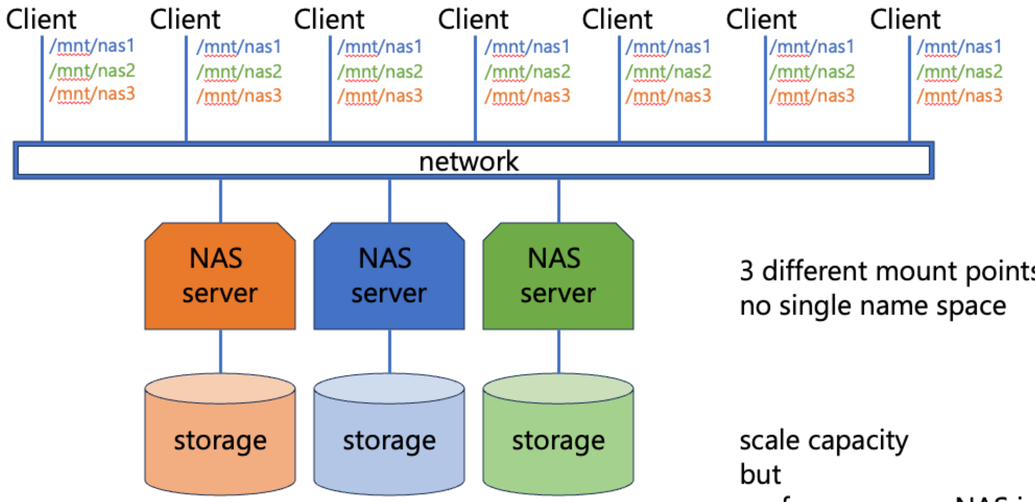


3 different mount points
no single name space

scale capacity
but
performance per NAS is limited

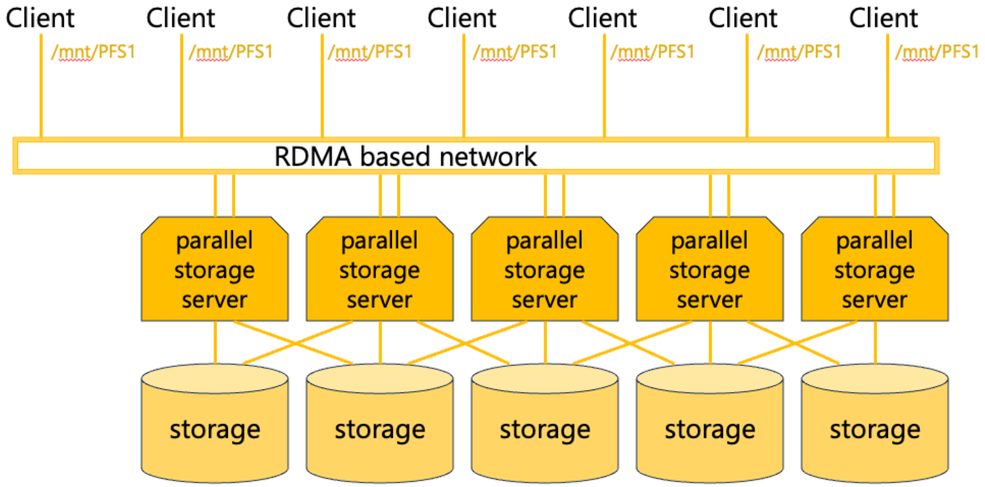
NAS vs PFS

What are the differences?



3 different mount points
no single name space

scale capacity
but
performance per NAS is limited



one namespace
one mount point

scale in capacity

and

performance in
one namespace

NAS vs PFS

What are the differences?

Feature	NAS	Parallel file system
Architecture	centralized	distributed
Protocol	NFS, SMB/CIFS	BeeGFS, GPFS, Lustre
Access	file level	file and block level
Scalability	limited	high
Performance	moderate	high
Ease of Use	simple	complex (most of them)
Cost	depend on solution	higher initial invest
Use case	general purpose storage	HPC , I/O intensive workloads

Conclusion

NAS: Best for environments where simplicity and moderate performance are sufficient.
Ideal for small to medium-sized businesses and general-purpose file sharing.

Parallel File Systems: Essential for HPC environments where high performance, scalability, and handling of large datasets are critical. Suitable for scientific computing, big data analytics, and other demanding applications.

Architecture

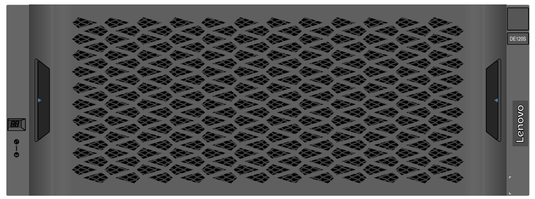
What composes a PFS?

Metadata

Data describing the structure and contents of the file system. Its role includes keeping track of the directories in the file system and the file entries in each directory.

Storage

Stores the distributed user file contents, known as *data chunk files,* in its storage targets.



Management

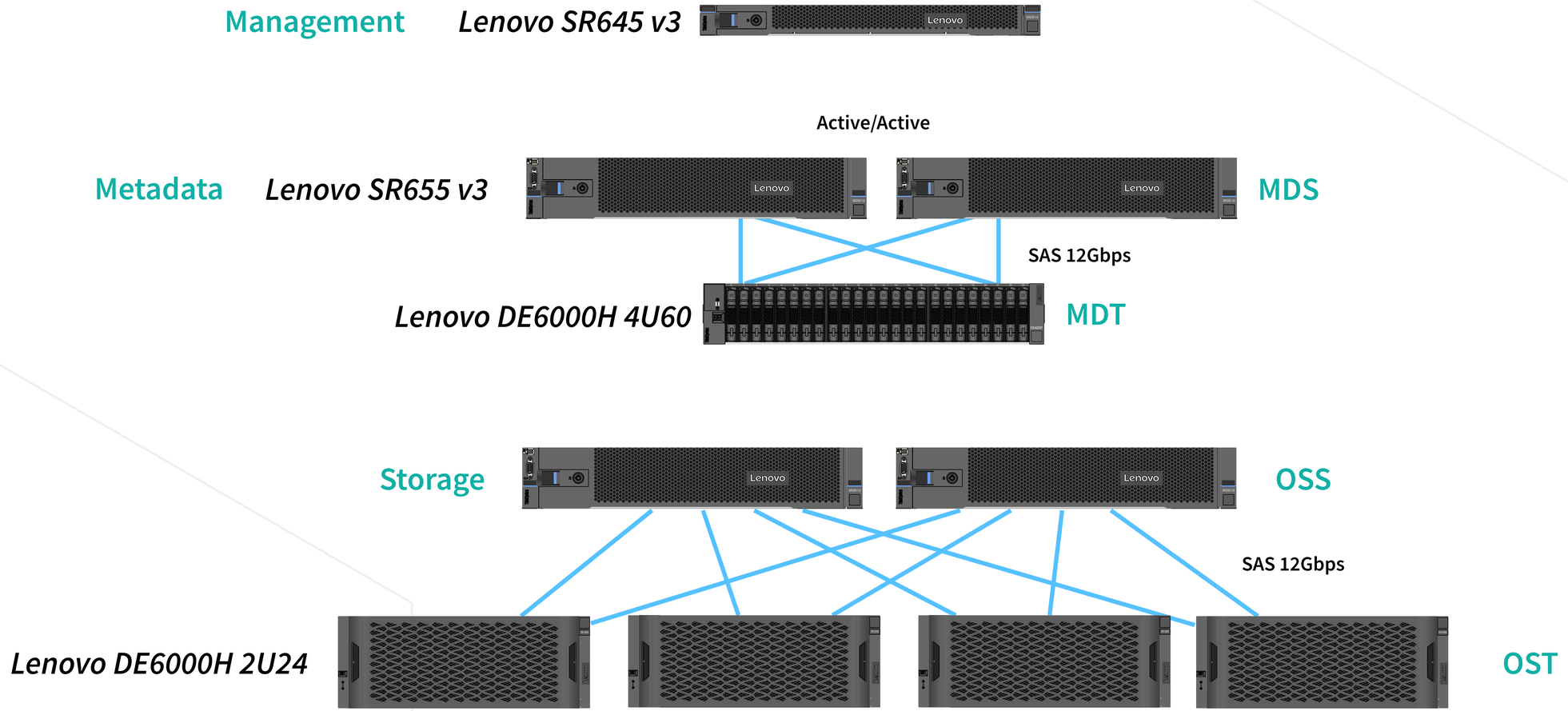
The storage cluster “Head Node”, responsible for logging, managing and reporting events.

Client

Compute Nodes that write and read on the parallel storage

Architecture

What composes a PFS?



Architecture

What composes a PFS?

Management *Lenovo SR645 v3*

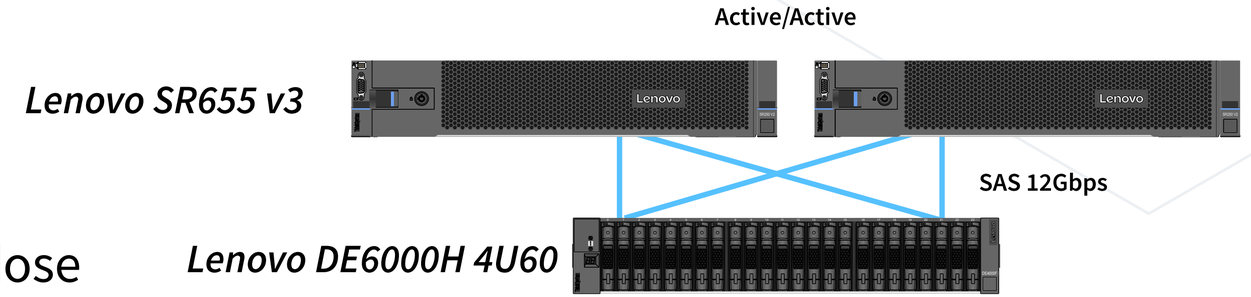
1. Meeting point for servers and clients
2. Watches registered services and checks their state
3. Not critical for performance, stores no user data

Architecture

What composes a PFS?

Metadata

1. Stores information about the data
   1.1. Directory information
   1.2. File and directory ownership
   1.3. Location of user data files on storage targets
2. Not involved in data access between file open/close
3. Faster CPU cores improve latency
4. Manages one metadata target
   4.1. In general, any directory on an existing local file system
   4.2. Typically a RAID1 or RAID10 on SSD or NVMe devices
   4.3. Minimal 0,5% of raw Storage space
5. Stores complete metadata including file size

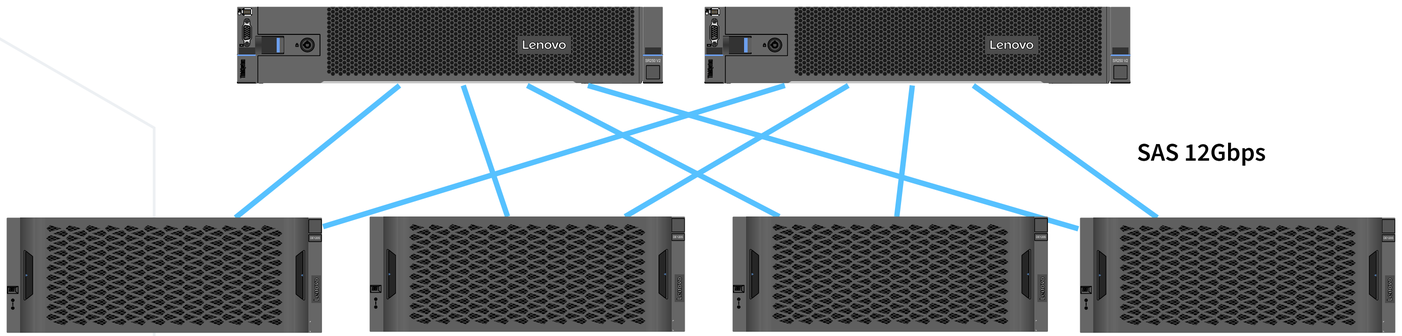


Architecture

What composes a PFS?

Storage

1. Stores striped user file contents (data chunk files)
   1.1. One or multiple storage services per instance
   1.2. Manages one or more storage targets
      1.2.1. In general, any directory on an existing local file system
      1.2.2. Typically a RAID-6 (8+2 or 10+2) or zfs RAIDz2 volume, either internal or externally attached
      1.2.3. It can also be a single HDD, NVMe, or SSD device
2. Multiple RDMA interfaces per server possible
   2.1. Different storage service instances bind to different interfaces
      2.1.1. Different IP subnets for the interfaces for the routing to work correctly



Lenovo DE6000H 2U24

Compute Nodes

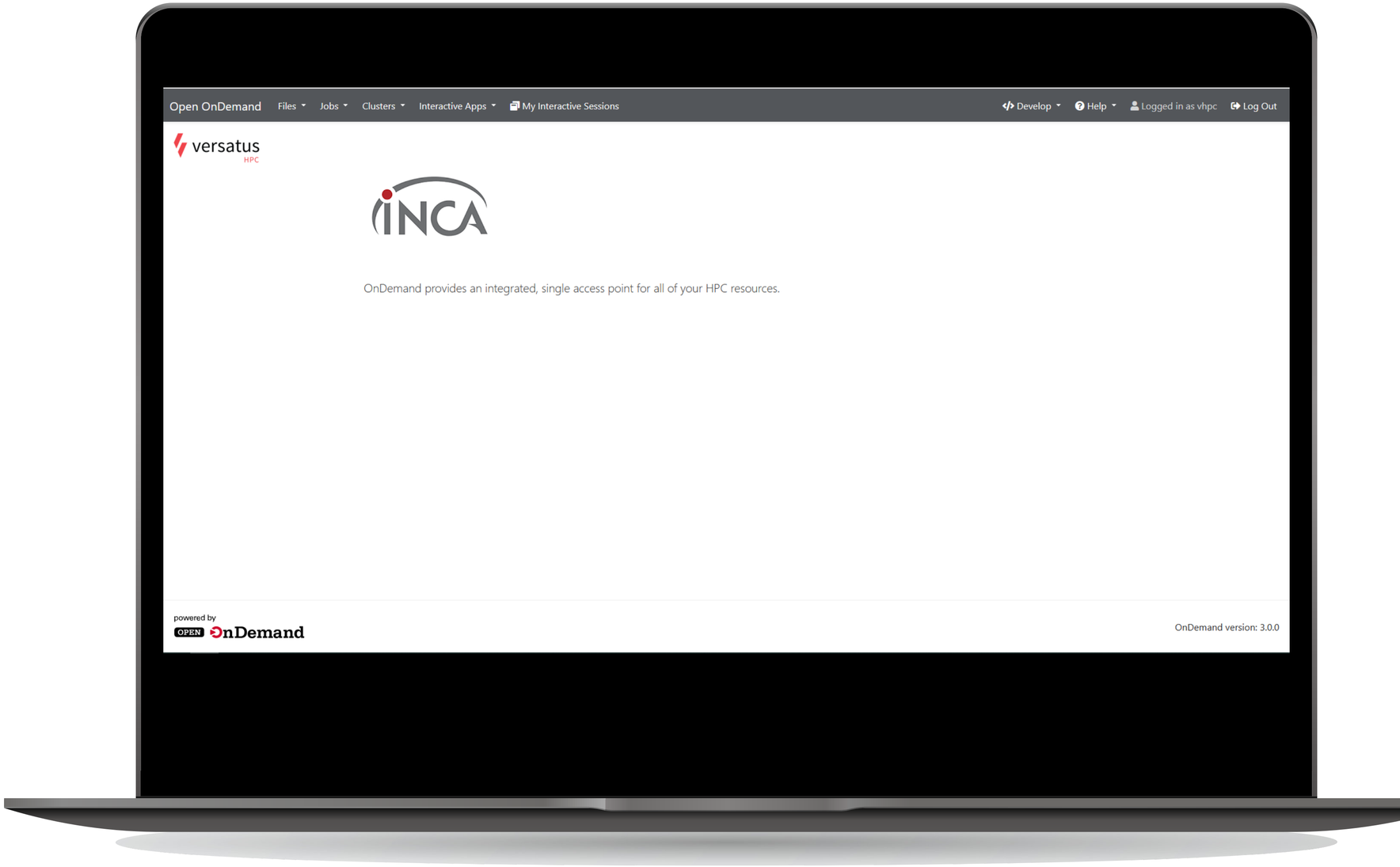
- Do not overload the computer node image with unnecessary packets.
- Install scientific packages in /opt and share them via NFS in the management network.
- Make it disposable. Easy to reinstall, easy to reuse.
- Don't use swapfile. Unso that it is required by any user.
- If you need to write locally and temporarily, use a local scratch device.

Compute Nodes

- Be Workload oriented!
- InfiniBand is not always required
- GPU is not always better
- Consult other University and research in the same field
- Example of Gold Ratios:
  - Material Sciences: 4GB/CPU Core
  - CAD/CAE: 4~6GB/CPU Core
  - Life Science: >8GB/CPU Core

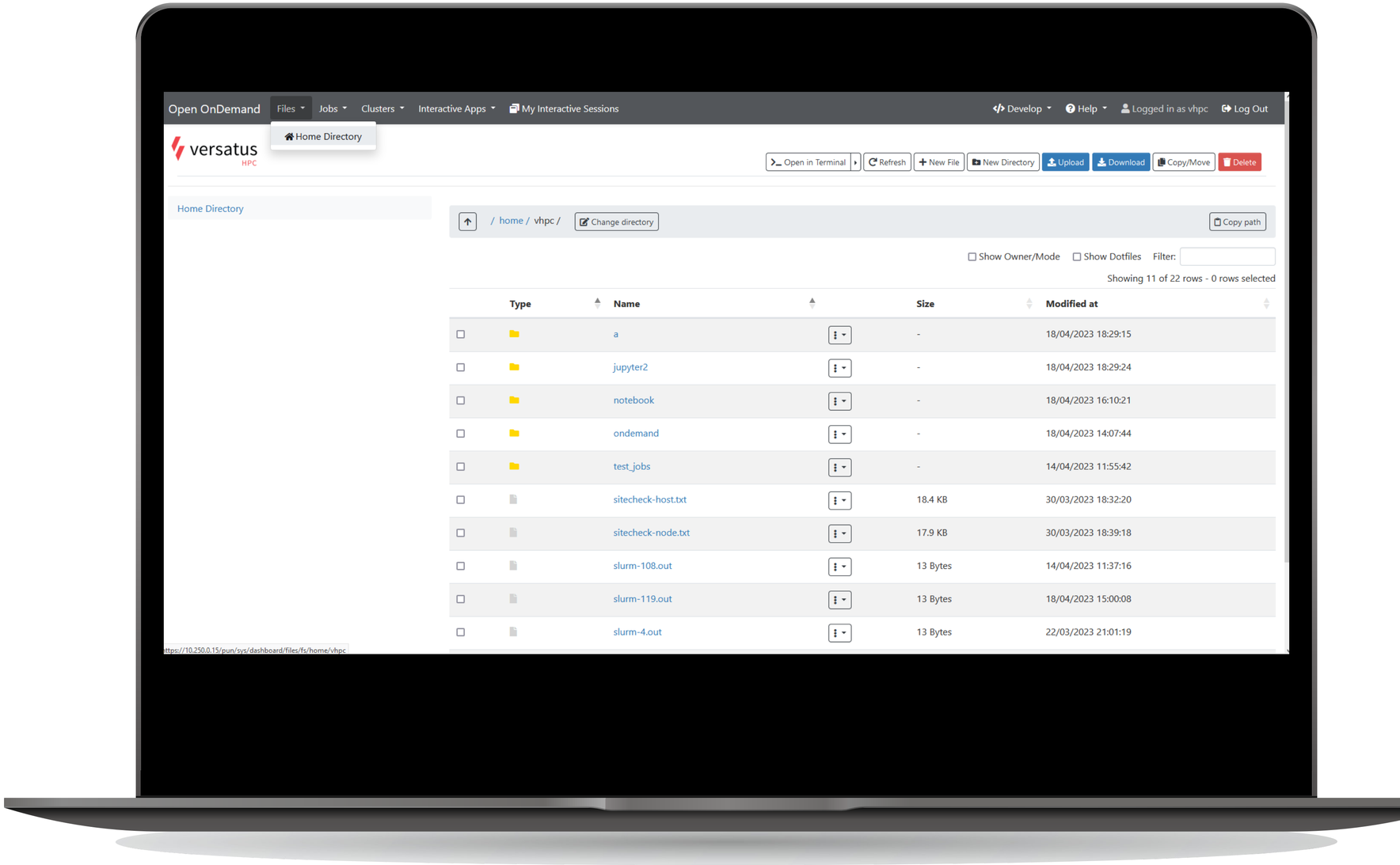
User Interface

Analytical tools and interactive applications are now easily integrated with the high-performance computing environment. New HPC users can take advantage of the ease of graphical interfaces to operate jobs regardless of the system architecture; the ease of use of a cluster is the same as that of a simple workstation. coordinator



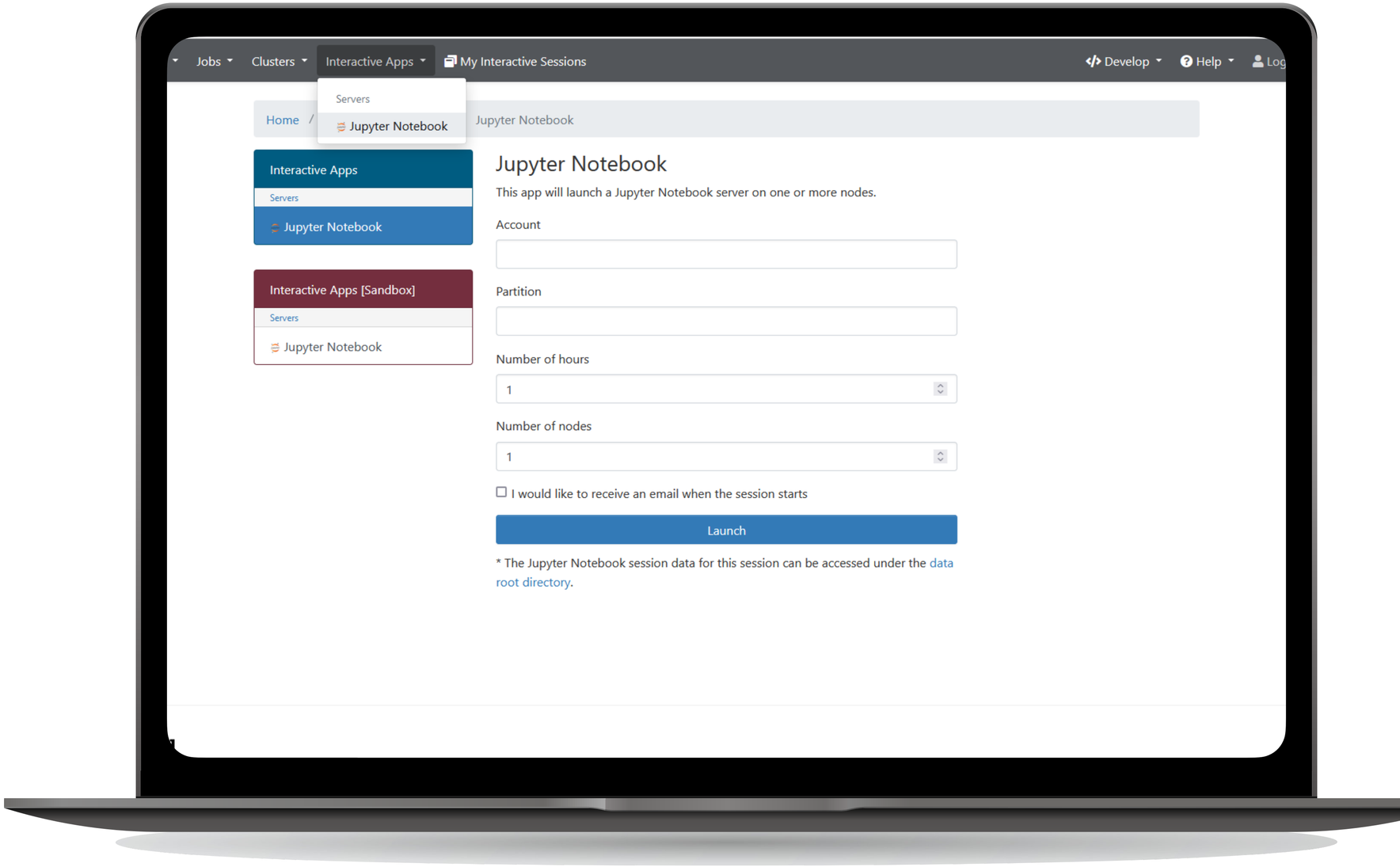
User Interface

Analytical tools and interactive applications are now easily integrated with the high-performance computing environment. New HPC users can take advantage of the ease of graphical interfaces to operate jobs regardless of the system architecture; the ease of use of a cluster is the same as that of a simple workstation. coordinator



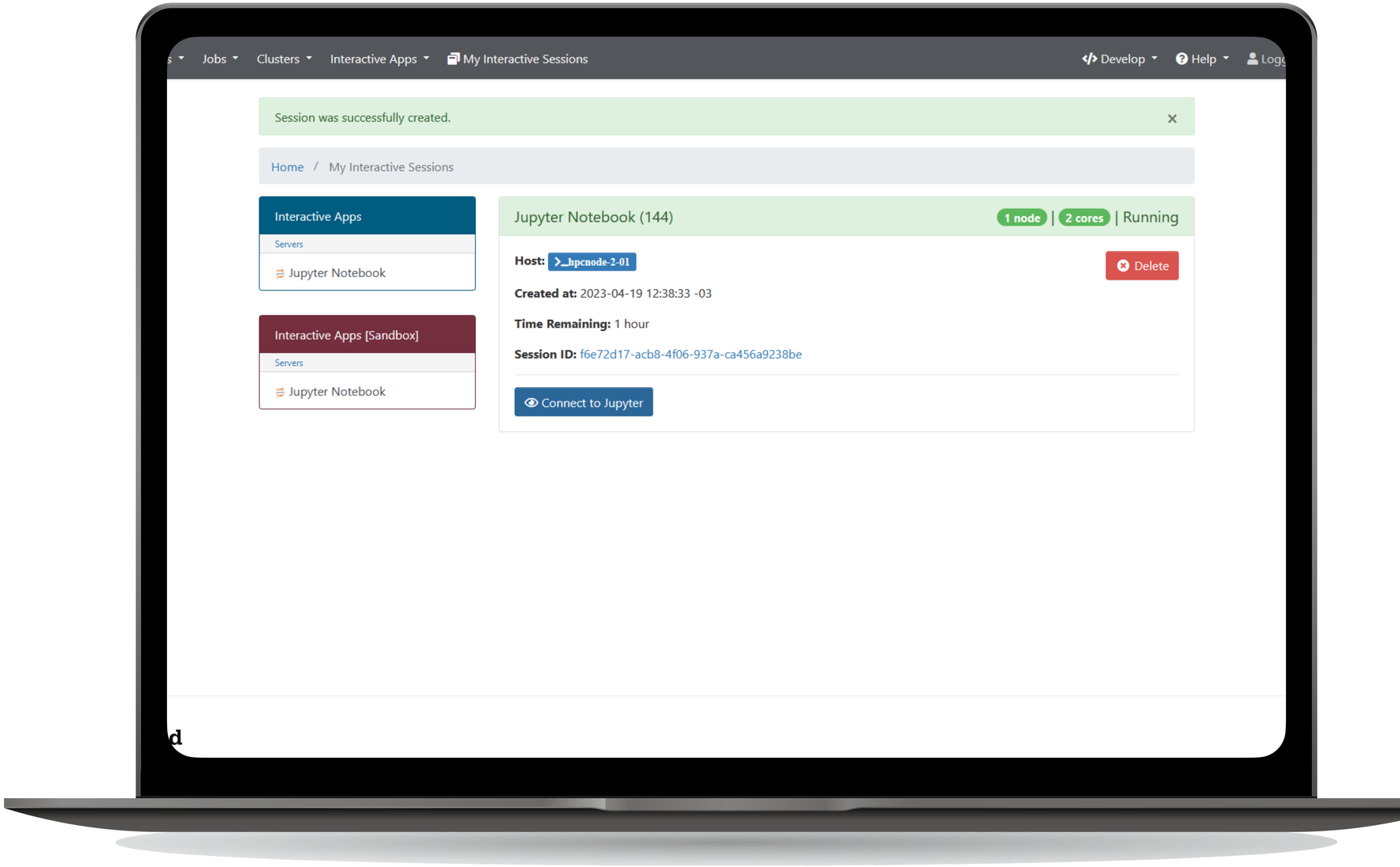
User Interface

Analytical tools and interactive applications are now easily integrated with the high-performance computing environment. New HPC users can take advantage of the ease of graphical interfaces to operate jobs regardless of the system architecture; the ease of use of a cluster is the same as that of a simple workstation. coordinator



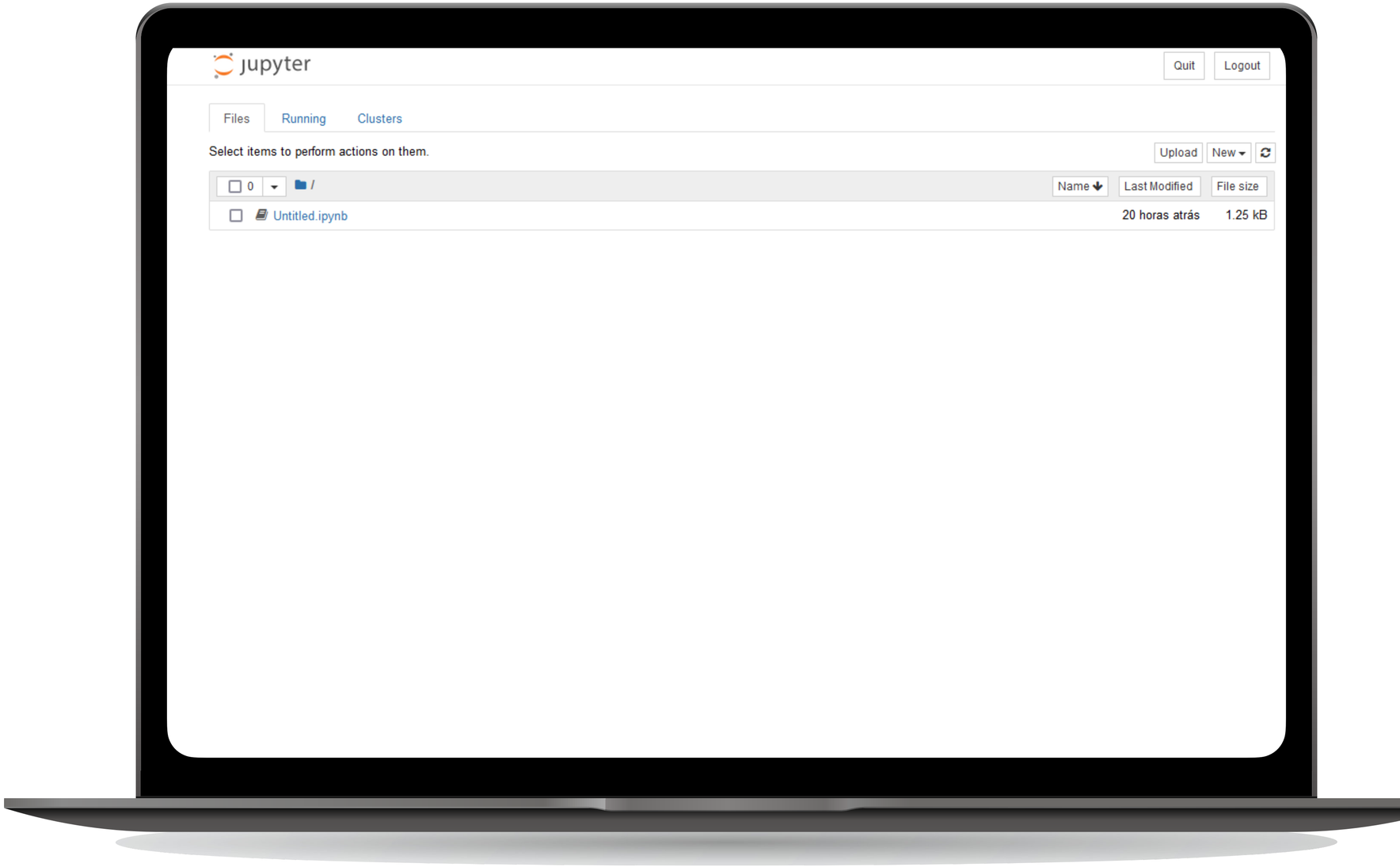
User Interface

Analytical tools and interactive applications are now easily integrated with the high-performance computing environment. New HPC users can take advantage of the ease of graphical interfaces to operate jobs regardless of the system architecture; the ease of use of a cluster is the same as that of a simple workstation. coordinator



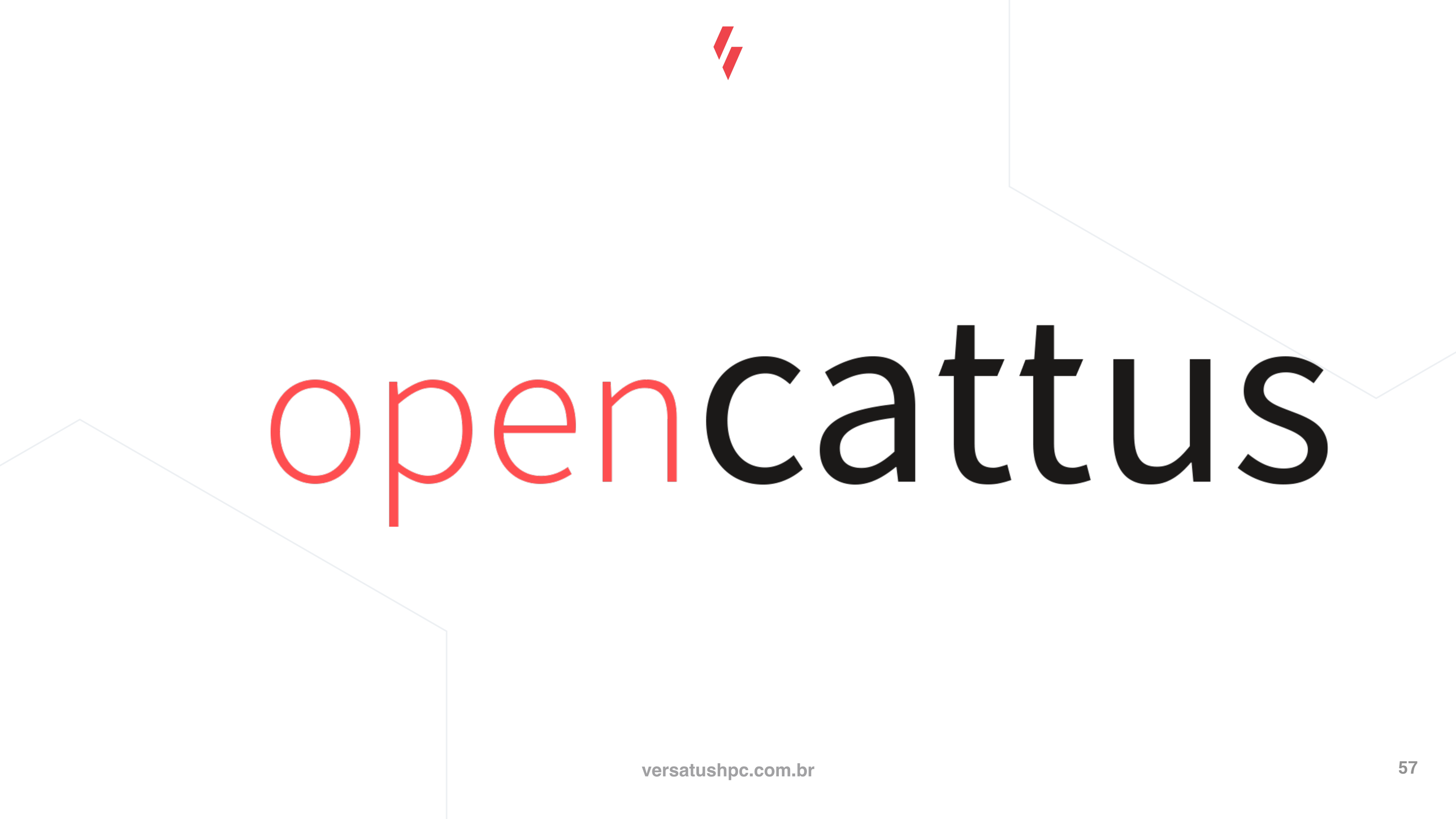
User Interface

Analytical tools and interactive applications are now easily integrated with the high-performance computing environment. New HPC users can take advantage of the ease of graphical interfaces to operate jobs regardless of the system architecture; the ease of use of a cluster is the same as that of a simple workstation. coordinator



Remember

- Its users are not its enemies.
- Yes, they will cause problems, but almost always without bad intentions.
- Yes, they will ask a lot of questions and consume your time.
- Yes, they may sound arrogant at times, but many times they feel insecure.
- But you need them, so be an excellent HPC cluster manager.


opencattus

opencattus

Cluster Management Suite

OpenCATTUS

Redefining Cluster Deployment Efficiency

Introducing OpenCattus, an open source cluster deployment system forged from the real experience of the industry. Developed by Versatus HPC - a company specialized in building, deploying and managing High Performance Computing (HPC) clusters - OpenCattus represents the culmination of years of practical experience and collaboration with leading manufacturers such as Dell Technologies and Lenovo. By releasing parts of its internal toolkit as open source, Versatus HPC brings enterprise-level cluster management solutions to organizations of all sizes and industries.

opencattus

Cluster Management Suite

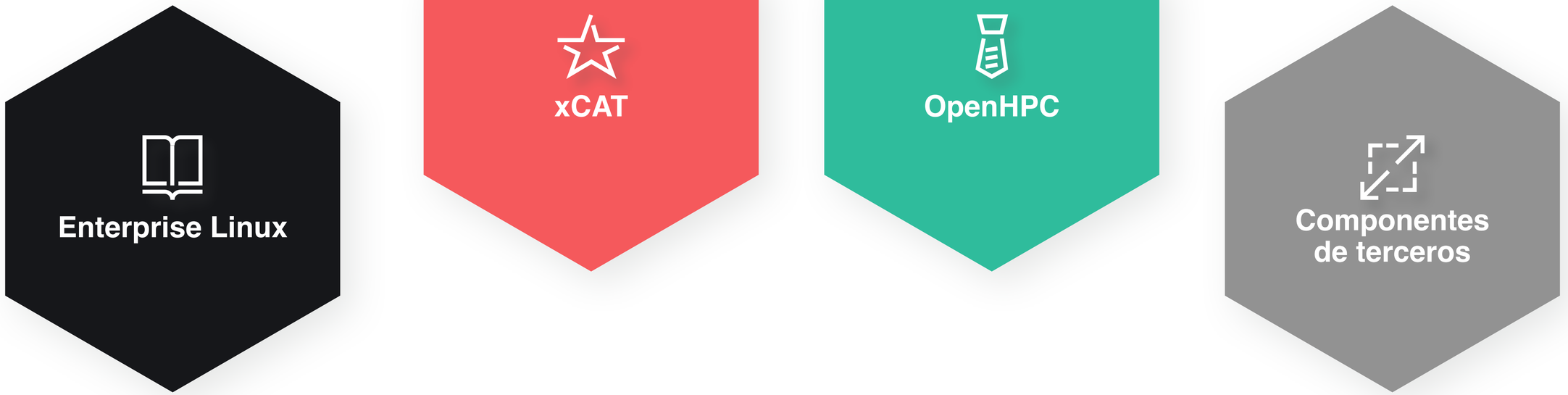
OpenCATTUS

Why have we done this?

The HPC software market in 2016 was stable. The free and open source standard software was Rocks Clusters and the state of the project was stagnant. Without updates for a long time, Enterprise Linux 7 had been available for two years and nothing had changed in the HPC scenario. Other initiatives such as OpenHPC emerged, but are incomplete compared to commercial offerings and what Rocks Clusters used to offer. OpenHPC is basically a recipe with a good amount of well-tested and packaged software, but lacks other auxiliary tools.

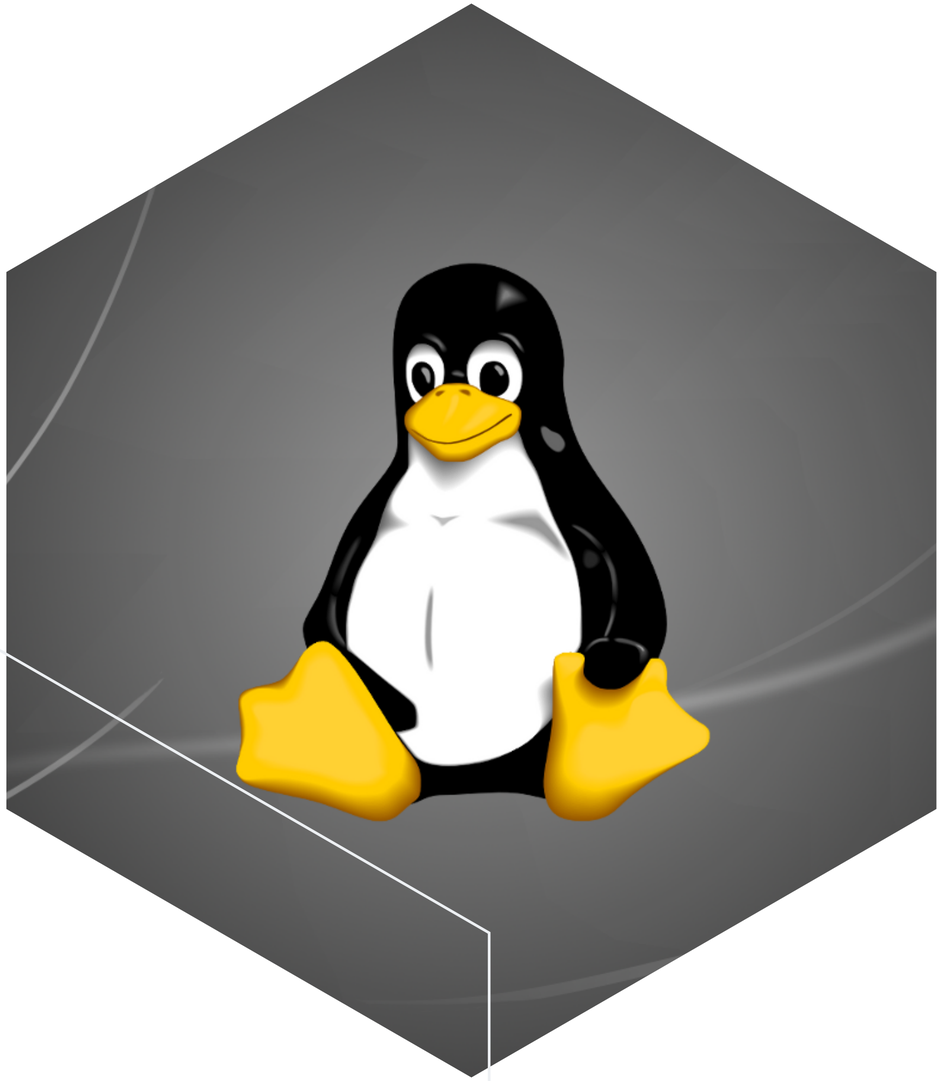
So we needed to do something about it, and the idea of openCATTUS came up.

So we needed to do something about it, and vxCAT came up



The base of the software

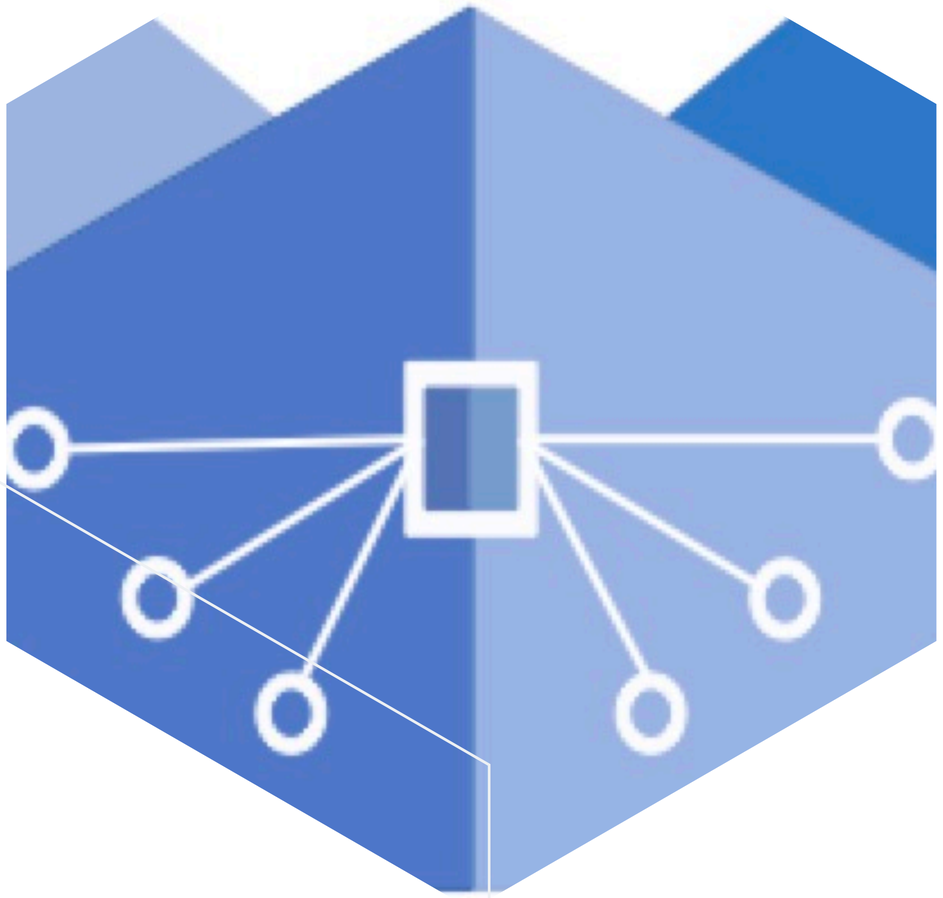
A solid base with upstream components to always keep the product in a safe life cycle



Enterprise Linux

OS

Standardized in Enterprise Linux as the default Linux flavor, with this we can have almost identical distributions for customers with or without upstream support.



xCAT

Orchestrator

The product core is based on xCAT provisioning software. With customized modifications to make the provisioner more business-friendly and with many improvements in security features and the ability to integrate with third-party tools.

OpenHPC

Common Foundation of HPC

We rely on the community to have a solid foundation to run scientific applications that are standard and well known to all in the HPC area.

With improvements and tight integration with additional custom packages, component consumption is extremely easy without breaking the upstream.

Third-party components

Everything else

With a large number of third-party components, the usability of the product is improved. It ranges from core functions such as complete domain and identity management to optional functions such as document composition with LaTeX integrated into the system.

What does the product offer?

- **Environment in operational cluster in a few minutes (20 ~ 30 minutes)**
- System easy to install through a terminal interface or with a response file
- Full support for Enterprise Linux and its variants
- Enterprise features such as SELinux and Firewall
- Close integration of management software
- Extended functionality within upstream components
- Different installation paths with a selection of components to install
- Stateless compute nodes to reduce hardware solution requirements
- Metapackage system configured (Spack) for builds to take advantage of CPU-specific architecture

What are the base components?

- Enterprise Linux 8.10 o 9.4 con:
  - Red Hat Enterprise Linux
  - Rocky Linux
  - Alma Linux
  - Oracle Linux
- xCAT 2.16
- OpenHPC
  - Version 2.8 for EL8
  - Version 3.1 for EL9
- Mellanox OFED modified by VersatusHPC for EL 8.10 and EL 9.4
- Zabbix Release 7.0 LTS
- OpenZFS 2.2
- VersatusHPC Pre-configured Spack Version

What are the scientific components sent?

- Directly from OpenHPC:
- Debuggers and profilers with the GNU and LLDB debugger
- MPI Libraries
  - OpenMPI, MPICH, MVAPICH2 with adequate PMIx or TM support, depending on the task scheduler
- Mathematical libraries
  - FFTW, GMP, GotoBLAS, ScaLAPACK.
- IO Performance Libraries
- Serial and Parallel Libraries

openHPC

What are the scientific components sent?

- Directamente desde Intel:
  - Intel oneAPI
    - Intel Base Kit
    - Intel HPC Kit
      - Intel MPI
      - Intel MKL

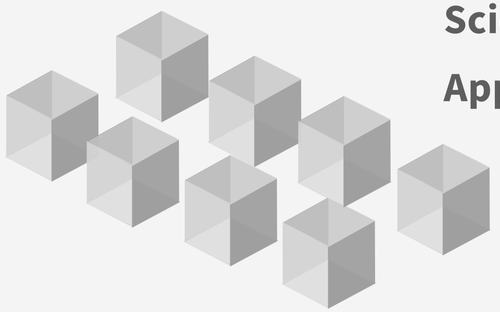
** opcionalmente aceptando el EULA de Intel*

What are the scientific components sent?

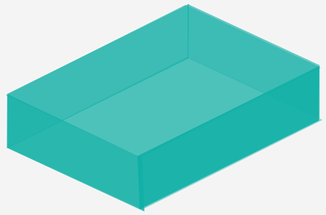
- Directly from NVIDIA:
  - NVHPC
  - Includes the old PGI compilers
  - CUDA
  - HPC-X with OFED previously manufactured by Mellanox
- Optionally, we also add a MLNX OFED patch for old boards and to solve problems with OpenHPC

- Http://github.com/viniciusferrao/mlnxofed-patch
- * optionally accepting the NVIDIA EULA

Software Stack



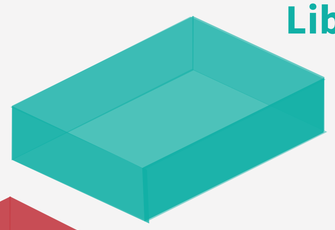
Scientific Application



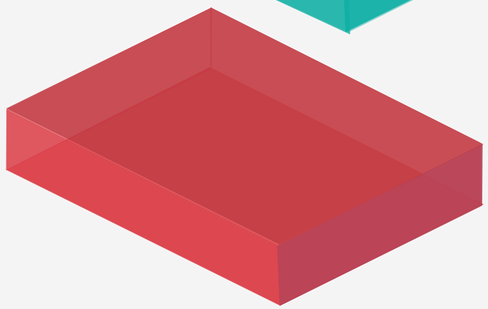
Compilers and Debuggers

Libraries

Job Scheduling



Parallel File System



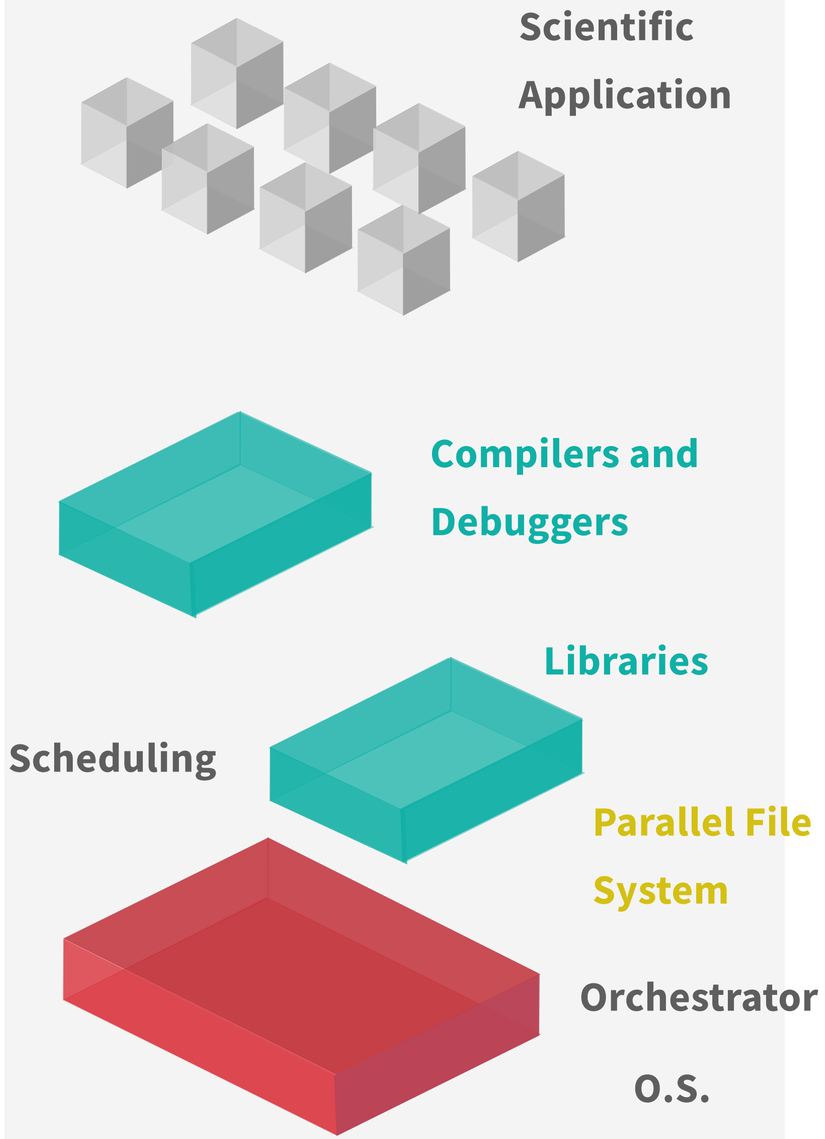
Orchestrator

O.S.

Python	mpi4py	NumPy	SciPy
I/O Lib	ADIOS	HDF5	NetCDF
	PHDF5	PnetCDF	
Numerical Lib	Boost	FFTW	HYPRE
	MFEM	MUMPS	OpenCoarrays
	PETSc	PT-Scotch	ScaLAPACK
	SLEPc	SuperLU-DIST	Trilinos
	GSL	METIS	OpenBLAS
	R	Scotch	SuperLU
	Intel oneAPI MKL		
Benchmarks	Dimemas	Extrae	LIKWID
	mpiP	OSU Micro-Benchmarks (OMB)	PAPI
	Scalasca	TAU	Score-P
	HPL		HPCG

Bibliotec Mensage	
Debugge	
Compila	
Job Sche	
Monitora	
Console	
Provisio	
Sistemas Paralelo	
Sistema	
Containe	
Sistema	

Software Stack



MPI Library	OpenMPI	MPICH
	Intel oneAPI MPI	MVAPICH
Debuggers	ARM Forge	
	GNU Debugger (gdb)	Intel oneAPI Debbuger (idb)
	LLDB	NVIDIA HPCSDK Debugger (pgdb)
Compilers	GCC	-
	Intel oneAPI Compilers	NVIDIA HPC SDK
Job Scheduling[1]	SLURM	PBS Professional
Monitoring	Zabbix/Grafana	XClarity
Console Management	goconserver	
Orchestrator	OpenCATTUS	xCAT-
Parallel File System	BeeGFS	Lustre
File System	XFS	
	NFS	
Container HPC	Singularity	
OS	Enterprise Linux	

We have made it suitable for the 'enterprise'

- Enterprise Linux support with Red Hat Enterprise Linux and variants
- Minimum factory-enabled firewall configuration to protect the machine exposed to the outside
- Creation of trust zones
  - Specific security policies in public areas
  - SELinux applied and enabled in the management machine
- Custom rules made specifically for the system. We don't just put everything as public_content_rw_t.

We have made it suitable for the 'enterprise'

- Proper configuration of security features such as brute force blockers and SSH strengthening
- Identity, Policy and Audit with FreeIPA / Red Hat Identity Manager to apply rules to users and hosts
- Proper handling of SSH user and host keys included in the LDAP directory
- Easy configuration of sudoers with customized policies per user if you need
- Integrated PKI with the possibility of certifying HPC cluster services
- Support to hide processes within the file system /proc to protect user privacy

We have made it suitable for the 'enterprise'

- Integration of xCAT with FreeIPA/IdM to handle DNS domain zones
- Integration of custom OFED distributions with OpenHPC packages
- Custom SELinux policies adapted for xCAT with permissions only for the necessary areas of the file system
- Consumption of OpenHPC compilers and tools in our extended version of the Spack packaging system
- Network Time Protocol service that feeds the entire HPC environment without conflicts between services

We have made it suitable for the 'enterprise'

- SSH host-based authentication between compute nodes with host keys stored within the FreeIPA directory
- Integration of OpenHPC network file system requirements with FreeIPA self-assembly features
- Postfix local mail service with Smarthost
- Stateless compute images fully integrated with identity services and generated with the appropriate configuration

Why should I consider it?

	openCATTUS	OpenHPC	Rocks Clusters	Commercial Offering
Easy to install	Yes	No	Yes	Yes
LVM Snapshot	Yes	No	No	No
Preconfigured repositories	Yes	No	No	Yes
Multiple networks	Yes	Yes	Yes	Yes
Firewall with custom rules	Yes	No	Yes	Yes
Digital certificate management	Mid 2025	No	No	Yes
SELinux	Mid 2025	No	No	No
Security improvements in the main node	Yes	No	No	Yes
Identity management	Mid 2025	Unix Files By Default	Unix Files By Default	LDAP Only
Stateless Nodes	Yes	Yes	No	Troublesome
Multiple OFED stack	Yes	No	Custom Rolls	Yes
Automatic mount (autofs)	Mid 2025	No	Yes	No
Host-based SSH authentication	Mid 2025	No	Yes	No
NFSv4 with RDMA	Yes	No	No	No

Why should I consider it?

	openCATTUS	OpenHPC	Rocks Clusters	Commercial Offering
DNS integrated in the directory	Yes	No	No	No
Open source components	Yes	Yes	Yes	No
HPC Containers	Yes	Yes	No	No
Support for built-in applications	Yes	Community	Community	No
Multiple queue systems	Yes	Yes	Yes	Yes
Open source monitoring	End of year	Unstable	Yes	No
Preconfigured MTA Postfix	Yes	No	No	No
"Mirror" tools for repositories	Yes	No	Partial	No
Root Account Audit	Future	No	No	No
Limit resources with cgroups	Future	No	No	Yes
Parallel file system installation	Future	Solo cliente	No	Yes
Health and Compliance Verification Tool	Future	No	Yes	Yes
Advanced queue system rules	Future	No	No	Yes

opencattus

Finally:

As you have seen, OpenCATTUS is a powerful tool designed to give agility and consistency to cluster administration, facilitating day-to-day tasks.

Currently, the software is in constant development. We have made great progress, refactoring various parts of the code to increase its robustness and at the same time delivering new functionalities according to our roadmap.

opencattus

Finally:

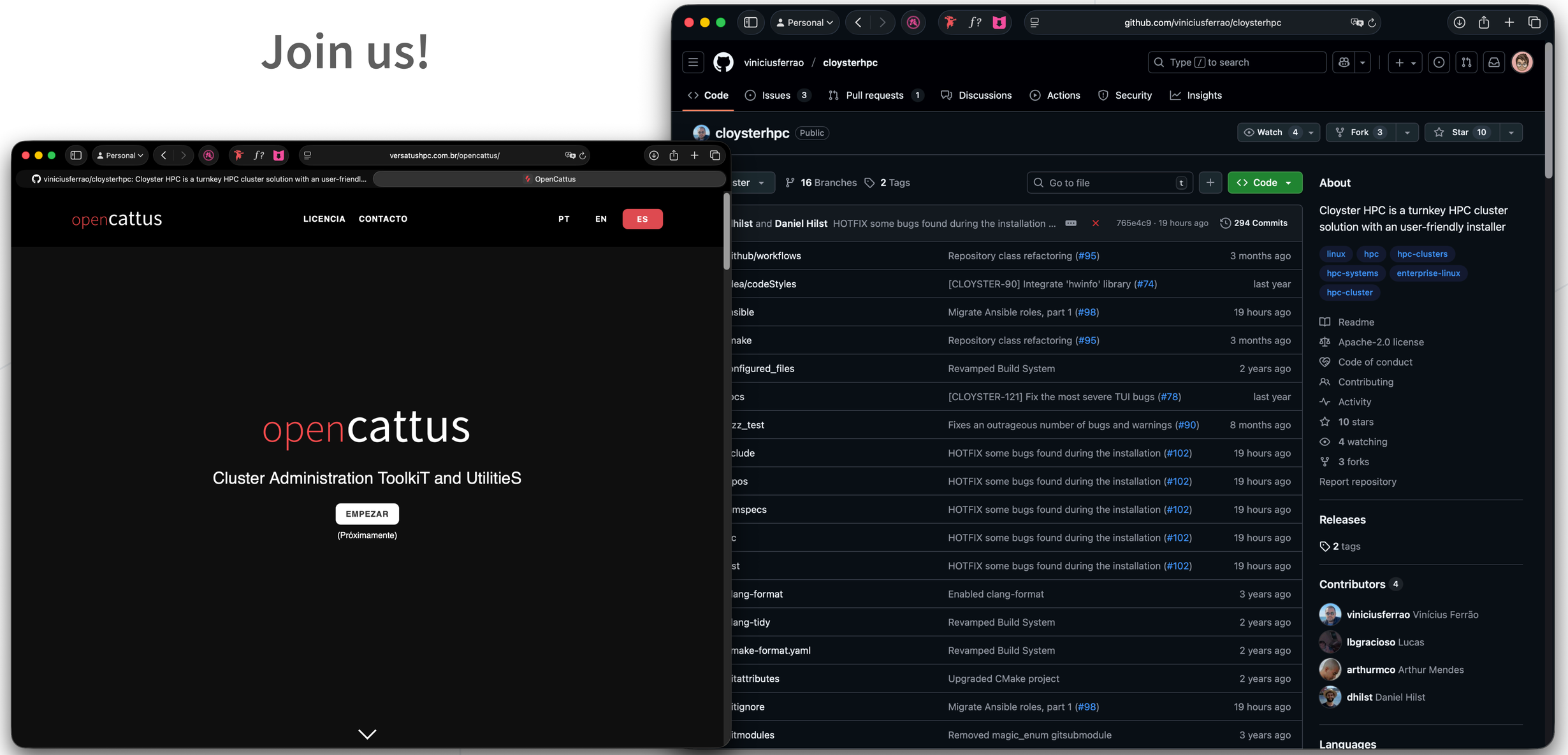
OpenCATTUS is a **free software and, like any open source** project, community participation is essential to accelerate development and ensure its quality. We want to invite each of you to join the OpenCATTUS community. Whether suggesting or voting for the most important functionalities, reporting bugs or difficulties, suggesting improvements, or even contributing with patches. The contribution of the first adopters is especially valuable to ensure that the software meets real needs and evolves quickly.

opencattus

Finally:

We know that the strength of free software lies in the community around it. Therefore, our development team is always available for discussions and support in our mailing list, creating a valuable exchange network. You will have direct contact with the developers and, at the same time, you will help us with important comments to identify and correct problems, in addition to improving OpenCATTUS.

Join us!


github.com/viniciusferrao/cloysterhpc
viniciusferrao / cloysterhpc
Type / to search
Code
Issues 3
Pull requests 1
Discussions
Actions
Security
Insights
cloysterhpc
Public
Watch 4
Fork 3
Star 10
16 Branches
2 Tags
Go to file
Code
About
Cloyster HPC is a turnkey HPC cluster solution with an user-friendly installer
linux
hpc
hpc-clusters
hpc-systems
enterprise-linux
hpc-cluster
Readme
Apache-2.0 license
Code of conduct
Contributing
Activity
10 stars
4 watching
3 forks
Report repository
Releases
2 tags
Contributors 4
viniciusferrao Vinícius Ferrão
lbgracioso Lucas
arthurmco Arthur Mendes
dhilst Daniel Hilst
Languages
hilst and Daniel Hilst HOTFIX some bugs found during the installation ...
765e4c9 · 19 hours ago
294 Commits
ithub/workflows
Repository class refactoring (#95)
3 months ago
lea/codeStyles
[CLOYSTER-90] Integrate 'hwinfo' library (#74)
last year
sible
Migrate Ansible roles, part 1 (#98)
19 hours ago
make
Repository class refactoring (#95)
3 months ago
nfigured_files
Revamped Build System
2 years ago
cs
[CLOYSTER-121] Fix the most severe TUI bugs (#78)
last year
zz_test
Fixes an outrageous number of bugs and warnings (#90)
8 months ago
clude
HOTFIX some bugs found during the installation (#102)
19 hours ago
pos
HOTFIX some bugs found during the installation (#102)
19 hours ago
mspecs
HOTFIX some bugs found during the installation (#102)
19 hours ago
c
HOTFIX some bugs found during the installation (#102)
19 hours ago
st
HOTFIX some bugs found during the installation (#102)
19 hours ago
lang-format
Enabled clang-format
3 years ago
lang-tidy
Revamped Build System
2 years ago
make-format.yaml
Revamped Build System
2 years ago
itattributes
Upgraded CMake project
2 years ago
itignore
Migrate Ansible roles, part 1 (#98)
19 hours ago
itmodules
Removed magic_enum gitsubmodule
3 years ago
versatushpc.com.br/opencattus/
viniciusferrao/cloysterhpc: Cloyster HPC is a turnkey HPC cluster solution with an user-friendl...
OpenCattus
opencattus
LICENCIA
CONTACTO
PT
EN
ES
opencattus
Cluster Administration ToolkiT and UtilitieS
EMPEZAR
(Próximamente)

Thanks · Obrigado · Gracias · ありがとう

eiji@versatushpc.com.br versatushpc.com.br +55 11 3436-0664