



# The convergence of HPC and BigData

What does it mean for HPC sysadmins?

damienfrancois



Scientists are never happy



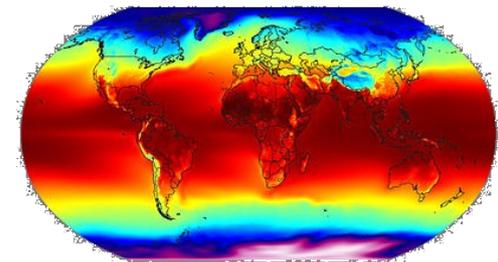
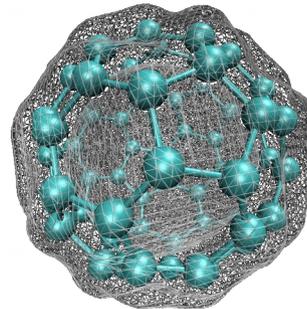
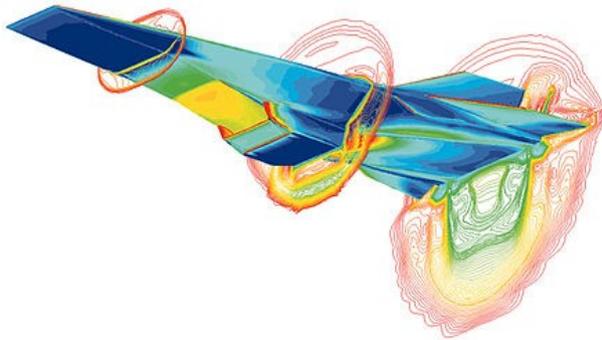
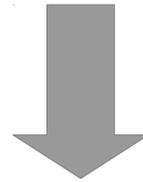
# Some have models but they want data



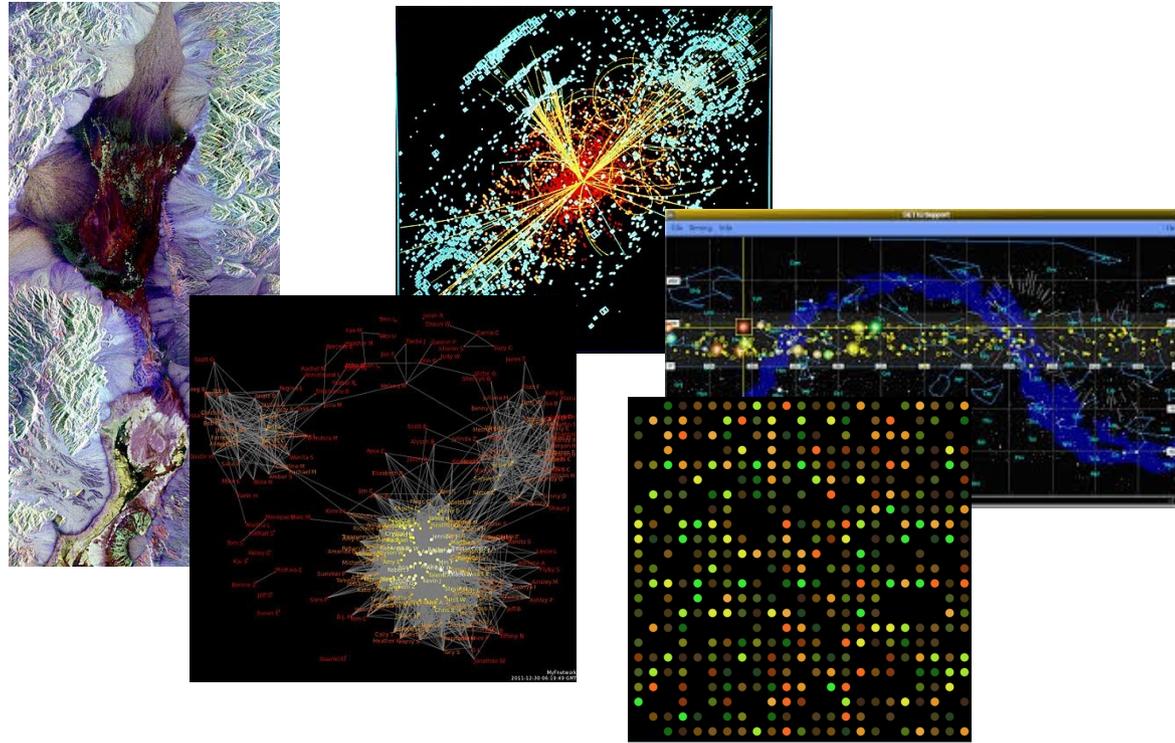
$$\rho \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} = \rho \bar{f}_i + \frac{\partial}{\partial x_j} \left[ -\bar{p} \delta_{ij} + \mu \left( \frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - \overline{\rho u'_i u'_j} \right]$$

$$\hat{H}\Psi = [\hat{T} + \hat{V} + \hat{U}] \Psi = \left[ \sum_i^N \left( -\frac{\hbar^2}{2m_i} \nabla_i^2 \right) + \sum_i^N V(\vec{r}_i) + \sum_{i<j}^N U(\vec{r}_i, \vec{r}_j) \right] \Psi = E\Psi$$

$$\Delta T = f(T, \eta) := \frac{1}{R} \left( Qs(y)(1 - \alpha(\eta, y)) - (A + BT(n, y)) - C \left( T(n, y) - \int_0^1 T(n, y) dy \right) \right)$$



# Others have data but they want models



$$h_{\theta}(X) = \frac{1}{1 + e^{-\theta^T X}} = Pr(Y = 1|X; \theta)$$

$$\varphi(\mathbf{x}) = \sum_{i=1}^N a_i \rho(\|\mathbf{x} - \mathbf{c}_i\|)$$

$$\arg \min_{\mathbf{S}} \sum_{i=1}^k \frac{1}{2|S_i|} \sum_{\mathbf{x}, \mathbf{y} \in S_i} \|\mathbf{x} - \mathbf{y}\|^2$$

Please do not ask me to explain the equations. Thanks. Pictures courtesy of NASA and Wikipedia.



## Data intensive (BigData Ogres)

PageRank, Collaborative Filtering, Linear Classifiers, Outlier Detection, Clustering, Latent Dirichlet Allocation, Probabilistic Latent Semantic Indexing, Singular Value Decomposition, Multidimensional Scaling, Graphs Algorithms, Neural Networks, Global Optimisation, Agents, Geographical Information Systems

Fox, G et al Towards a comprehensive set of big data benchmarks.  
In: BigData and High Performance Computing, vol 26, p. 47,  
February 2015



## Compute intensive (HPC Dwarfs)

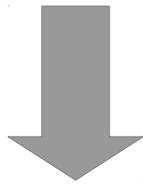
Dense and Sparse Linear Algebrae, Spectral Methods, N-Body Methods, Structured and Unstructured Grids, MonteCarlo

The Landscape of Parallel Computing Research: A View from Berkeley  
Krste Asanović et al EECS Department University of California, Berkeley  
Technical Report No. UCB/EECS-2006-183 December 18, 2006

I did not invent that. Pictures courtesy of Disney and DreamWorks.



Data intensive  
(BigData)



Cloud



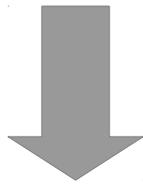
Compute intensive  
(HPC)



Clusters



Data intensive  
(BigData)



Cloud

👍 Instant availability  
Self-service or Ready-made  
Elasticity, fault tolerance



Compute intensive  
(HPC)

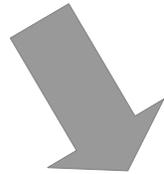


Clusters

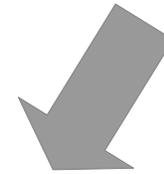
👍 Close to the metal  
High-end/Dedicated hardware  
Exclusive access to resources



Data intensive  
(BigData)



Compute intensive  
(HPC)



👍 Cloudster(?) 👍

# Now all Cloud providers offer HPC services

IBM | Cloud | Why IBM | Products | Solutions | Garage | Mo

## Solve compute-intensive problems quickly with high-performance computing (HPC) on IBM Cloud

Find the IBM Cloud HPC solution that best meets your needs — from IaaS to SaaS

Microsoft Azure

Why Azure | Solutions | Products | Documentation | Pricing | Training

## Big Compute: HPC & Batch

Large-scale cloud computing power on demand

Azure provides on-demand compute resources that enable you to run large jobs when you need more capacity, or run work entirely in Azure. Scale easily and take advantage of the results you want, when you need them.

Google Cloud | Why Google | Products | Solutions | Pricing | Security | Documentation | Customers | Partners | Support

High performance computing

## High performance computing

Powerful, flexible infrastructure to support scalable workloads.

ORACLE Cloud

Sign In | Contact | Chat | English | Estimate | Buy | Try for Free

Applications | Platform | Infrastructure | Resources | Search



## High Performance Computing

Try for Free | Buy Now

Overview | Products | Solutions | Documentation | Pricing | Jump Start | Customers | Getting Started

aws

Contact Sales | Products | Solutions | Pricing | Getting Started

## High Performance Computing (HPC)

Imagine the problems you can solve with virtually unlimited infrastructure

What should Academic HPC centers do?

Answer on next slide. Please be patient.

The screenshot shows the CSCS website header with the logo and name: "CSCS Centro Svizzero di Calcolo Scientifico Swiss National Supercomputing Centre". The ETH logo is in the top right. A navigation bar contains "User Portal", "Getting Started", "Scientific Computing", "Storage", "Tools", and "My Projects". Below this, a "TOOLS" sidebar is visible, and the main content area is titled "OpenStack" with a sub-header: "CSCS offers an Infrastructure as a Service (IaaS) on the POLLUX facility and a cloud object storage".

The screenshot shows a CERN article page with a dark, starry background. The CERN logo is on the left. The navigation menu includes "About CERN", "Students & Educators", "Scientists", and "CERN community", with a secondary row for "Accelerators", "Experiments", "Physics", "Computing", "Engineering", "Updates", and "Opinion". The main heading is "The importance of OpenStack for CERN" by Andrew Purcell.

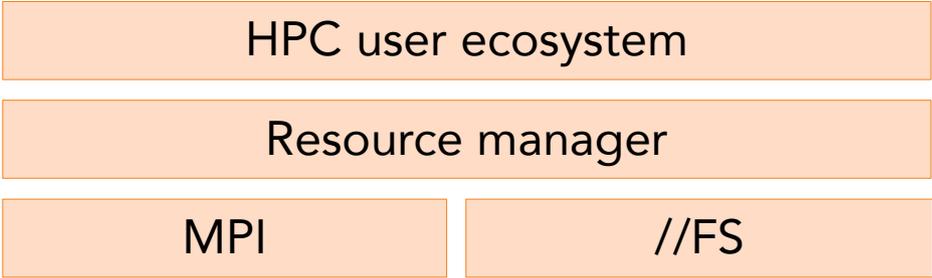
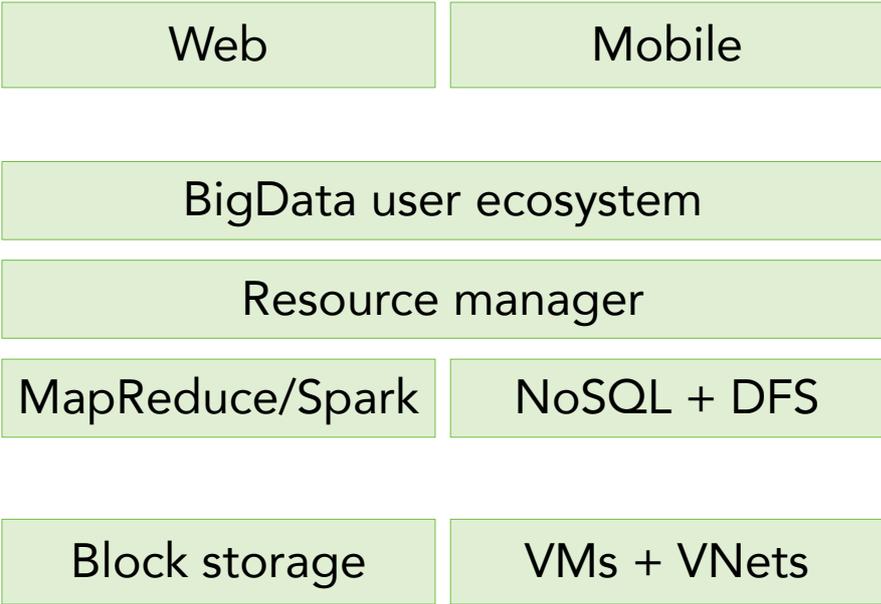
The screenshot shows the SURF website header with the logo and a navigation bar: "Home", "Services and products", "Knowledge base", and "Agenda". Below the navigation, a breadcrumb trail reads "Home > Agenda > Festive launch of the new HPC Cloud infrastructure". The main heading is "Festive launch of the new HPC Cloud infrastructure".

They should add Cloud-related technologies to their offering.

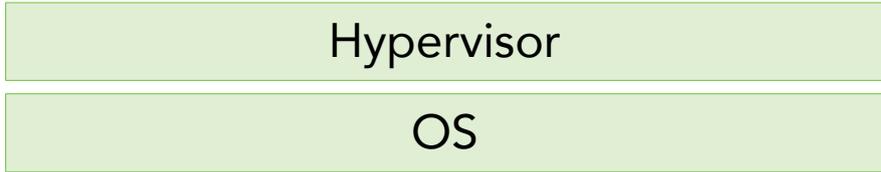
# Cloud stack

# Cluster stack

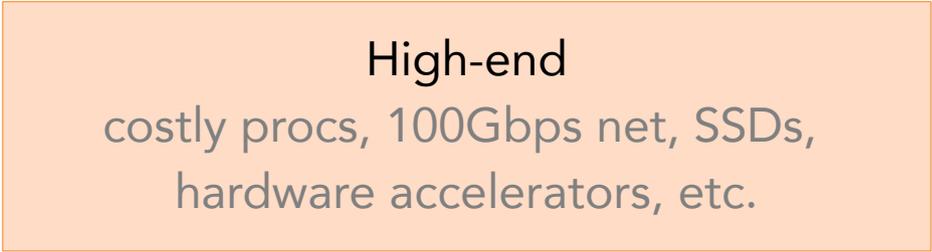
Infra., Platform, Soft.



System



Hardware



# Closing a Gap between Big Data and Big Computing

## Ecosystems\*:

## Big Data

Orchestration	Crunch, Tez, Cloud Dataflow	➔
Libraries	MLlib/Mahout, R, Python	➔
High-Level Programming	Pig, Hive, Drill	➔
Platform as a Service	App Engine, BlueMix, Elastic Beanstalk	➔
Languages	Java, Erlang, SQL, SparQL	➔
Streaming Parallel Runtime	Storm, Kafka, Kinesis	➔
Coordination	MapReduce	➔
Caching	Memcached	➔
Data Management	Hbase, Neo4J, MySQL	➔
Data Transfer	Sqoop	➔
Scheduling	Yarn	➔
File Systems	HDFS, Object Stores	➔
Formats	Thrift, Protobuf	➔
Virtualization	Openstack	➔

## Big Computing

➔	Kepler, Pegasus
➔	Matlab, Eclipse, Apps
➔	Domain-specific Languages
➔	XSEDE Software Stack
➔	Fortran, C/C++
➔	MPI/OpnMP/OpenCL
➔	iRODS
➔	GridFTP
➔	Slurm
➔	Lustre
➔	FITS, HDF
➔	Docker, SR-IOV



## Motivation:



## New Frontiers



## Leaders:

**Spark**

**MPI**

\*G. Fox et al. HPC-ABDC High Performance Computing Enhanced Apache Big Data Stack, CCGrid, 2015

5 paths to follow

# 1 Virtualization

More user control, more isolation

## 1.a Private Cloud on HPC

Deploy virtual machines inside a job allocation with, for instance, [pcooc](#).

## 1.b HPC On Demand & HPC as a Service

Deploy a cloud and install the HPC stack inside virtual machines allocated for each project/user with, for instance, [TrinityX](#).

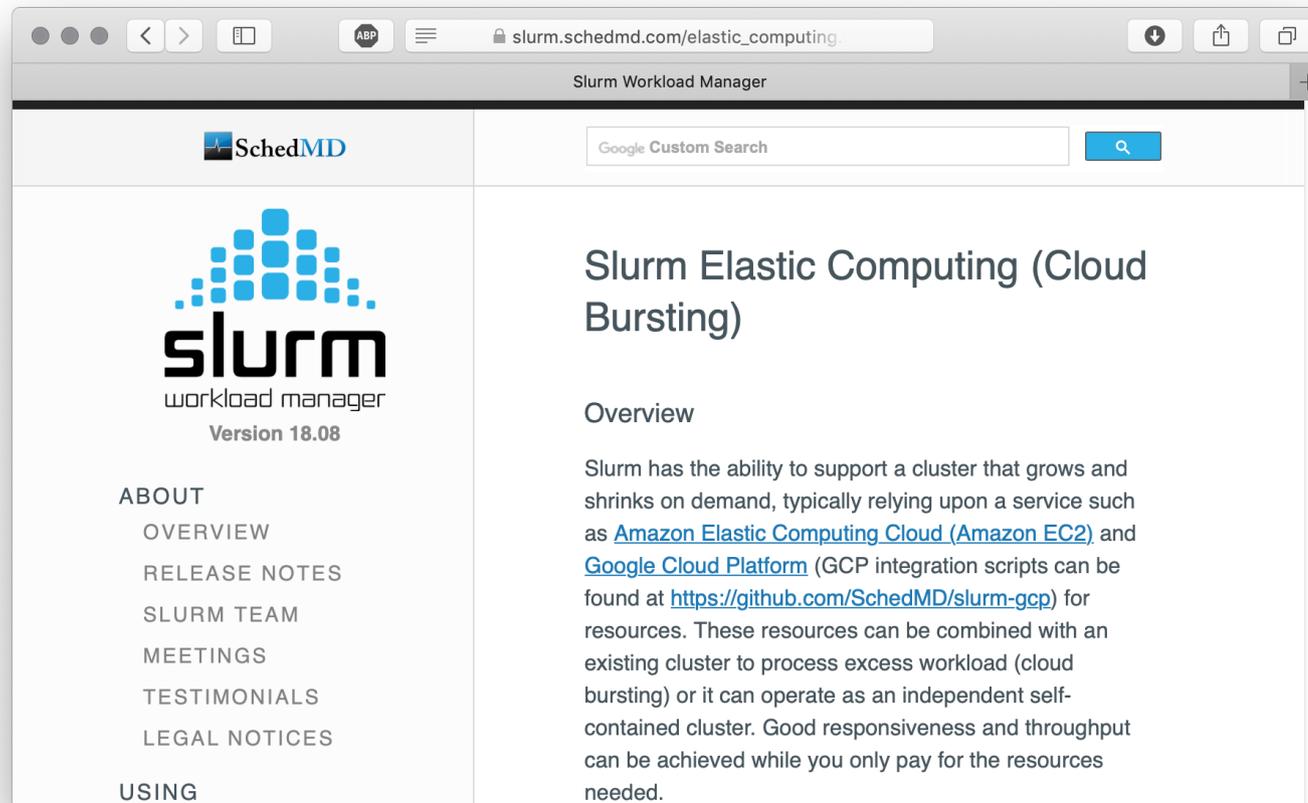
## 1.c Containers

Run jobs in containers, with for instance [Singularity](#), [Shifter](#), or [CharlieCloud](#).

# 2 Cloud bursting

## Elasticity for the cluster

Provision virtual machines in a cloud and append them to the cluster resources.  
Example with the **Slurm resource manager**:



# 3 Additional storage paradigms

Solve the ZOT files problem and increase external share-ability

## 3.a Object storage

Deploy an object store, e.g. **HDFS**, but also **Swift** or **Ceph**, either on a dedicated set of machines close to the cluster and with external connectivity or on the hard drives of the compute nodes.

## 3.b Hadoop connectors

Install a 'connector' on top of **BeeGFS**, **Gluster**, **Lustre**, etc. to offer a HDFS interface.

## 3.c NoSQL

Deploy an **ElasticSearch**, a **MongoDB**, a **Cassandra**, a **InfluxDB**, and a **Neo4j** cluster on separate hardware close to the cluster.

There are many more other options for NoSQL databases.

# 4 Additional programming paradigms

Offer new libraries, mid-way between MPI and job arrays: HPDA

## 4.a Standalone MapReduce or Spark

Not the most user friendly.

## 4.b Deploy a Hadoop framework inside allocation

Using for instance [MyHadoop](#), a “Framework for deploying Hadoop clusters on traditional HPC from userland”

## 4.c Disguise the scheduler as a Hadoop platform

Using a tool that deploys a Hadoop framework by submitting jobs, then report back to the user and allow them to submit MapReduce jobs, for instance [HanythingOnDemand](#), [HAM](#), or [Magpie](#)

# 4 Additional programming paradigms

Offer new libraries, mid-way between MPI and job arrays: HPDA

## 4.d HPC and BigData scheduler colocation

Take advantage of the elasticity and resilience of the Hadoop framework to **deploy Yarn on the idle nodes of a cluster** and update the Yarn node list upon job start or termination. Or dedicate a portion of the cluster to Yarn/Mesos.

## 4.e Unified BigData/HPC stack

One day? Intel, IBM working on that. Will it be FOSS?

<Spoiler> Probably not. But generates a lot of fuss. </Spoiler>

# 5 Web and Apps

Going beyond SSH and the command line, adding interactivity

## 5.a Web-HPC

Allow users to submit jobs through web interfaces, but also to use Web-based interactive scientific interpreters such as [RStudioServer](#) and [JupyterLab](#), and notebooks, etc.

## 5.b Ubiquitous access to data

Let the user access data and results from the Web, an App, or a Desktop client, with for instance [NextCloud](#).

I personally prefer my terminal.

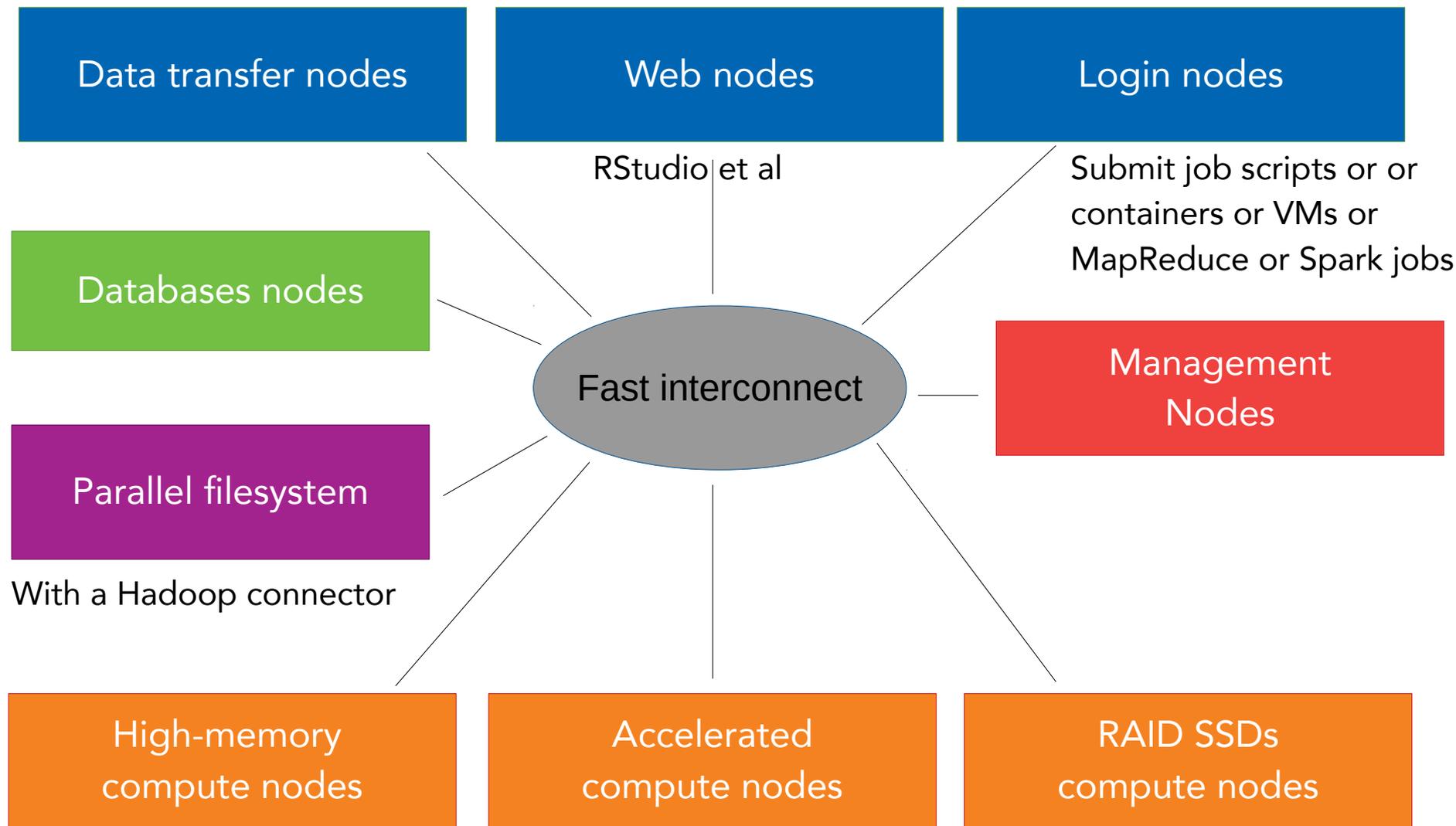


# The "Cloudster"



Outbound

GridFTP, Sqoop  
NextCloud



Run baremetal or  
container or VM

The Ultimate Machine.

# Scientists will be happy



Well, I hope. Thank you for your attention.

