Mixing and matching virtual and physical HPC clusters

Paolo Anedda
paolo.anedda@crs4.it

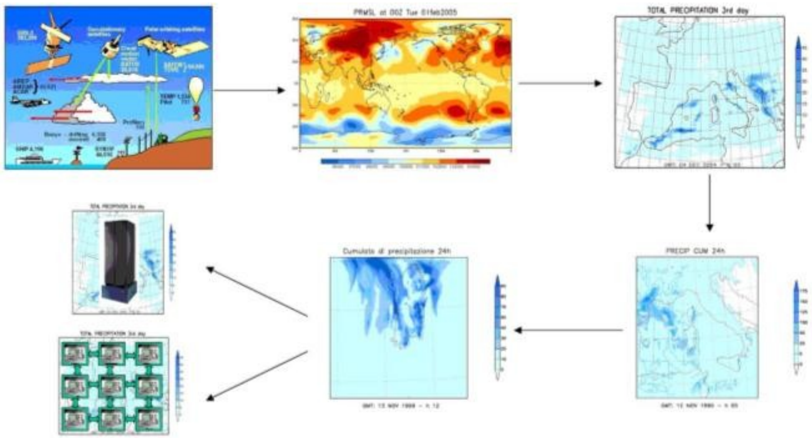
HPC 2010 - Cetraro 22/06/2010

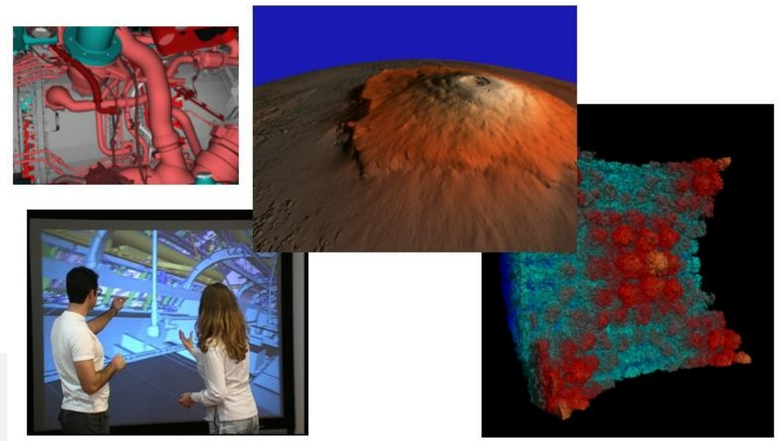
Outline

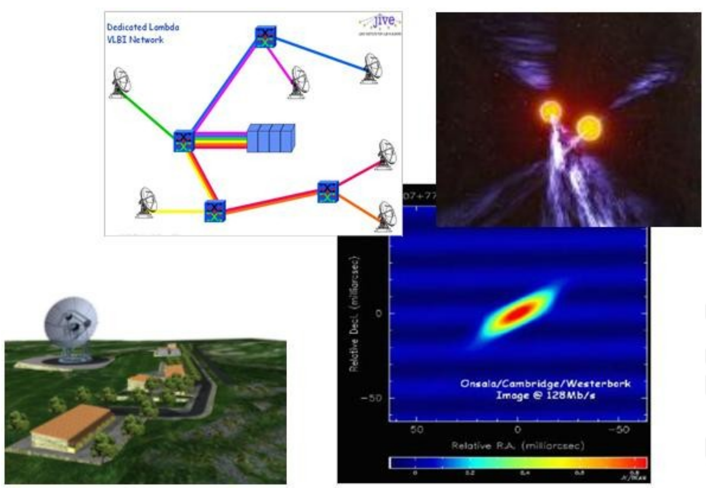
- Introduction
- Scalability Issues
- System architecture
- Conclusions & Future Works

- Introduction
- Scalability Issues
- System architecture
- Conclusions & Future Works

Supporting High Data Producing Applications







HPC: The Traditional Way

- Based on a (almost) fixed hardware/software platform.
- Good for standard production environments.
- Unsuitable for research and development enviroments.
- It lacks *flexibility.*

What if…

- We need to support multiple computational paradigms at the same time?
- We need to deploy transient experimental clusters?
- We need to deploy multiple development environment?
- We need to experiment new solutions?

Why Virtual Clusters?

- Virtualization is a consitent technology.
- Support for Multiple Computational Paradigms.
- Virtual Cluster makes the management of HPC environments flexible.
- The loss of performances can be acceptable (~5%).
- Support for hardware accelerator.
- Virtual Clusters can be saved for later use.

But…

- Virtual clusters operations can lead to scalability problems.
- Managing virtual clusters can be very difficult with traditional tools.
- Some users still want to run their code on traditional systems.

Build Smarter Management Tools:

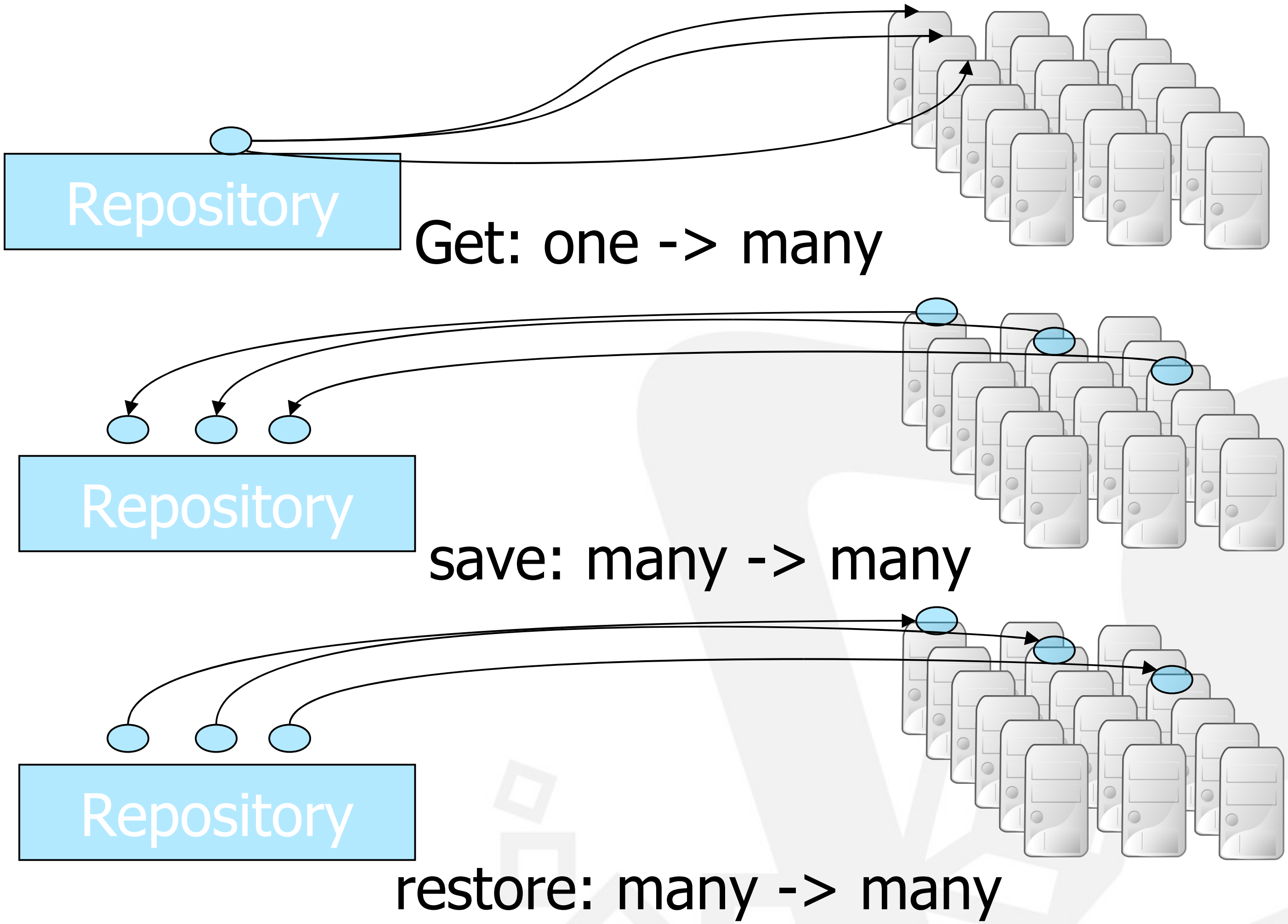
- Enable dynamic and flexible computational environments.
- Very different computational approaches can coexist on the same physical facility:
  - Map-reduce.
  - Standard parallel jobs.
  - *Virtual HPC Clusters.*

- Introduction
- Scalability Issues
- System architecture
- Conclusions & Future Works

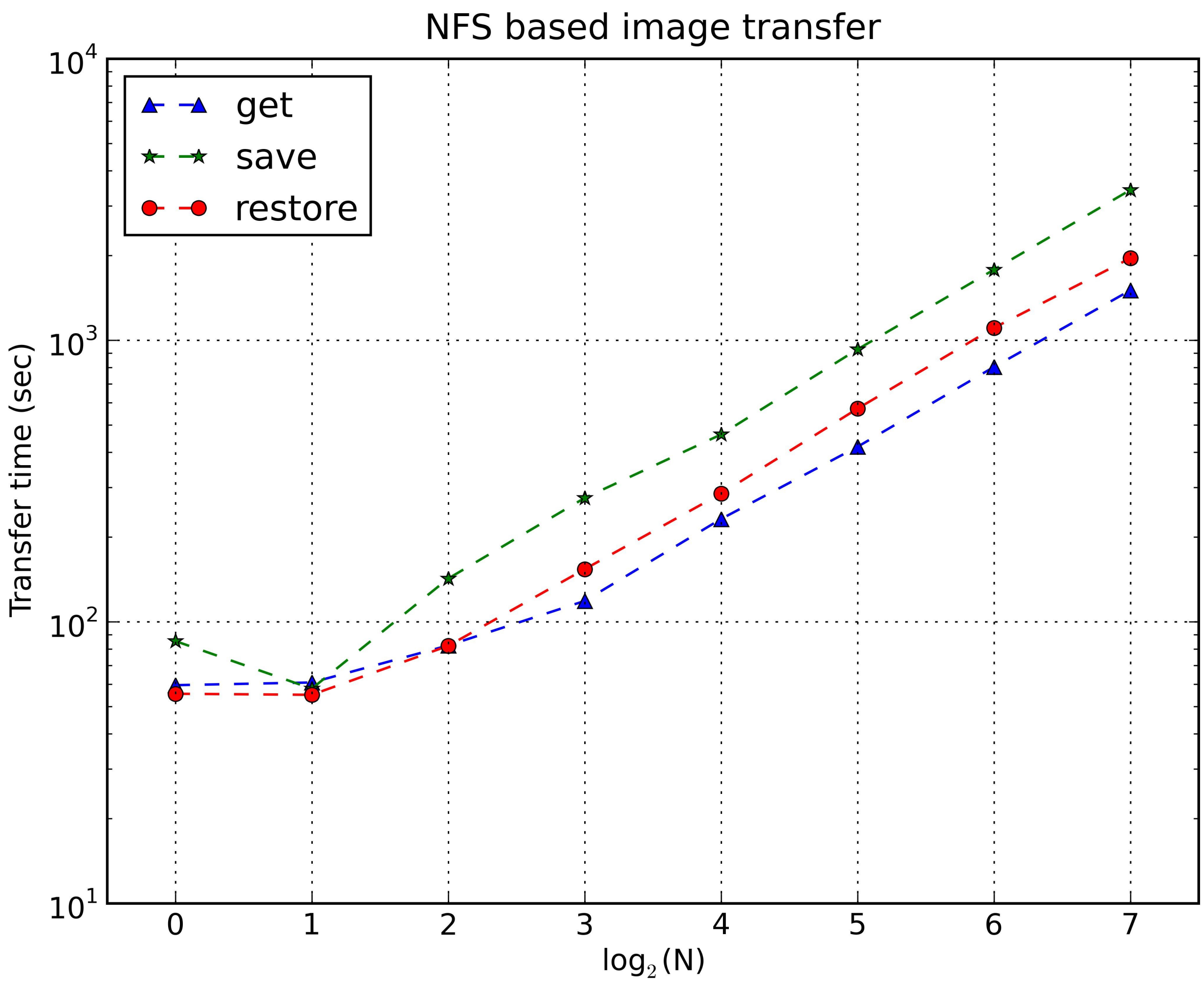
Virtual Clusters

- Virtual cluster are collections of virtual machines deployed and managed as a single entity (Foster et al. 2006).
- HPC virtual cluster are “atomic” objects
  - I.e., macro-computing task subdivided between the VC nodes.
- HPC virtual clusters are big objects
  - E.g., 128 nodes (3GB disk + 8GB memory) ~ 1.5TB.

Three Basic Operations on VM Images



NFS based image transfer

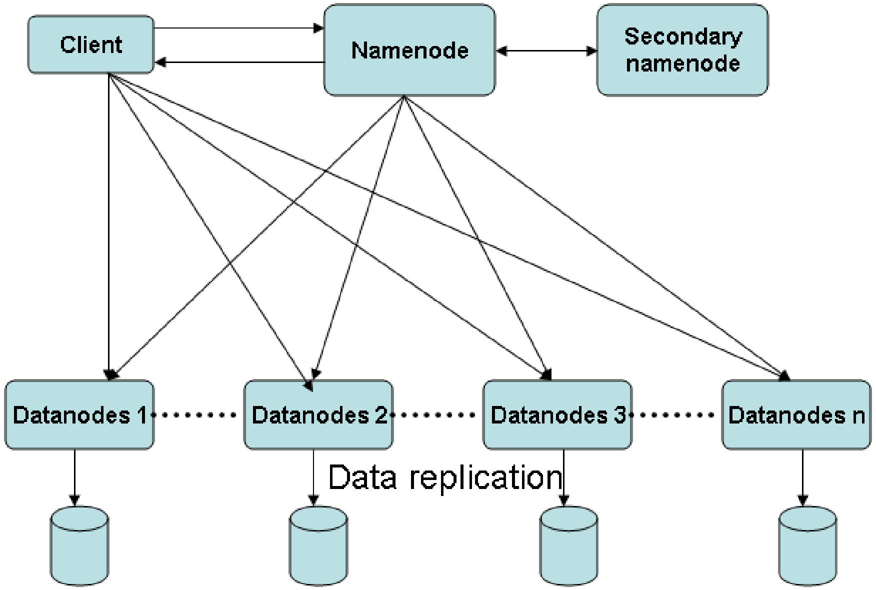


VCOperations

- VM Disk images are, as far as the repository is concerned, WORM (Write Once Read Many) objects.
- Get, save and restore are all “simple” I/O operations:
  - only one client writes and writes sequentially;
  - when a file is closed is “closed”, no appends needed;Suitable for applications that have large data sets.
- It appears that HDFS (KFS, GFS…) should be ok.

HDFS

- Distributed File System designed to run on commodity hardware.
- Suitable for applications that have large data sets.
- Highly Fault-Tolerant.

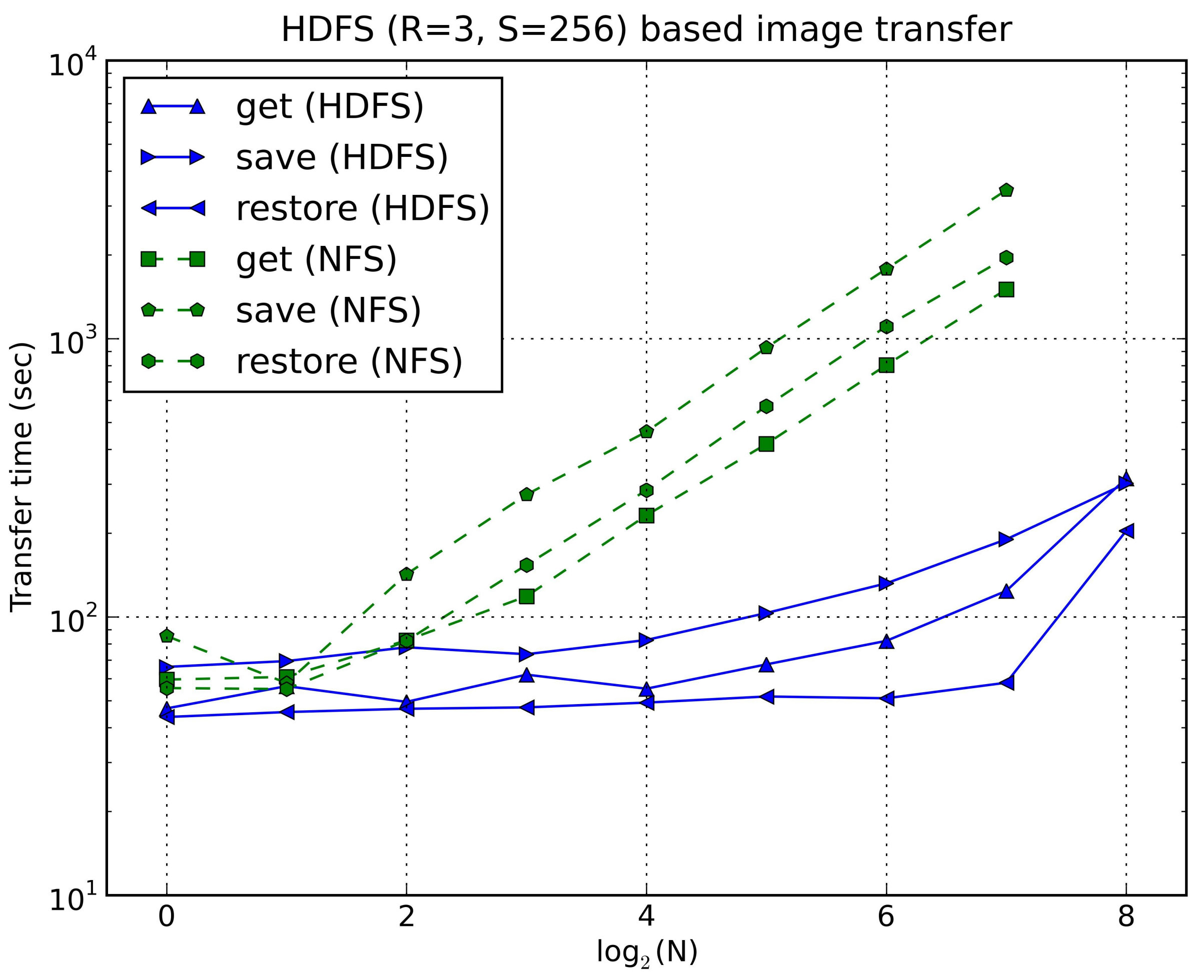


Measurement Procedure

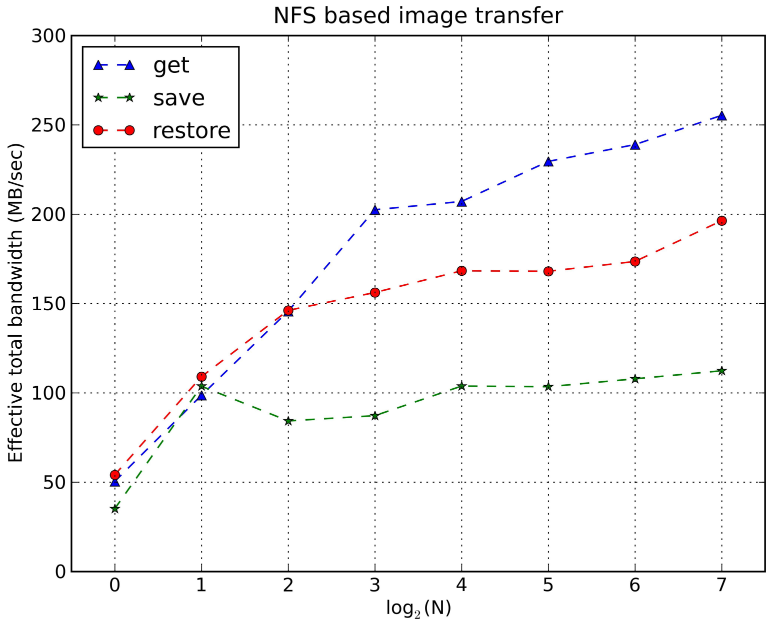
S := # of physical cluster nodes
N := # of virtual cluster nodes
R := block replication
Blocksize := 64MB

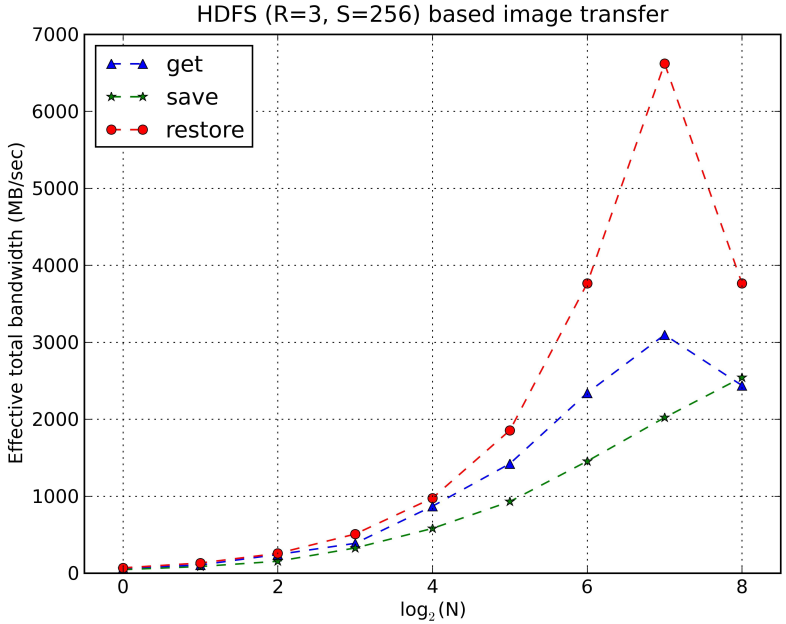
- Procedure
  - Allocate a cluster with S nodes and install HDFS
  - Save reference image in HDFS (from a node NOT in the cluster)
  - Randomly select groups of N=2,4,8,16,32,…,S nodes from the cluster
  - Use dsh for concurrent get, save and restore requests.

Transfer Time



Effective Bandwidth





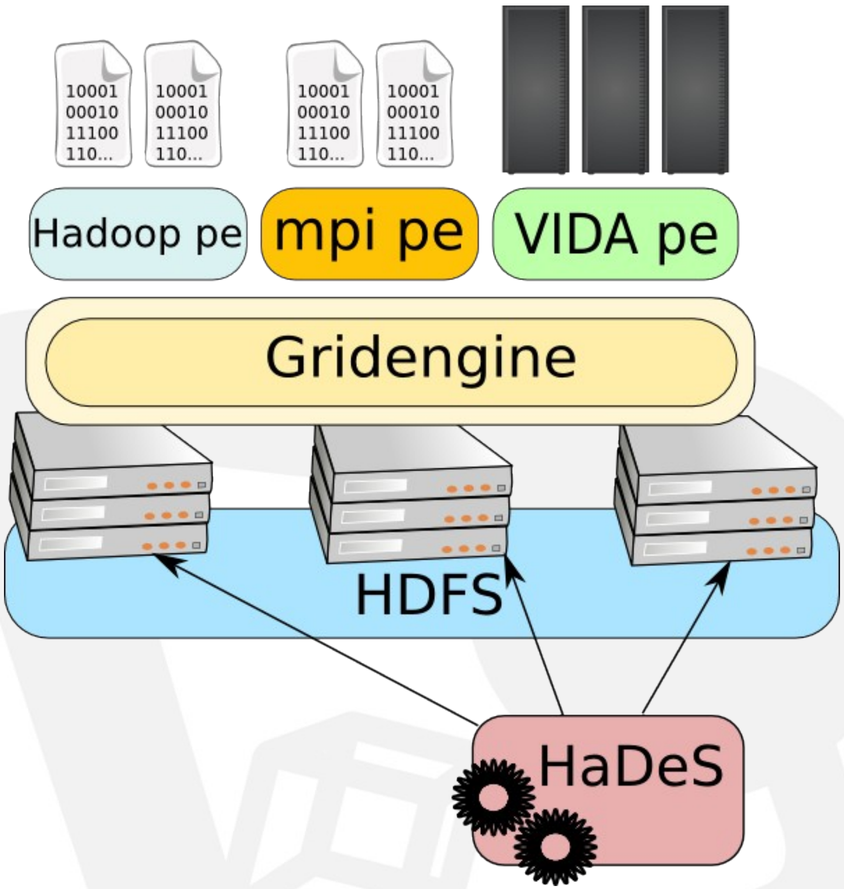
- Introduction
- Scalability Issues
- System architecture
- Conclusions & Future Works

Requirements

- Flexibility.
- Scalability.
- Support for Multiple Computational Paradigms.
- Encapsulation.
- Reliability and Security.
- High Performances.

Architecture

- VIDA:
  - Allocate the Virtual Clusters.
  - Manages all the Virtual Clusters operations.
- Gridengine
  - Allocate the physical resources.
  - Support different computational environment.
- HDFS:
  - A parallel filesystem.
- HaDeS:
  - A physical images deployment tool.



Gridengine

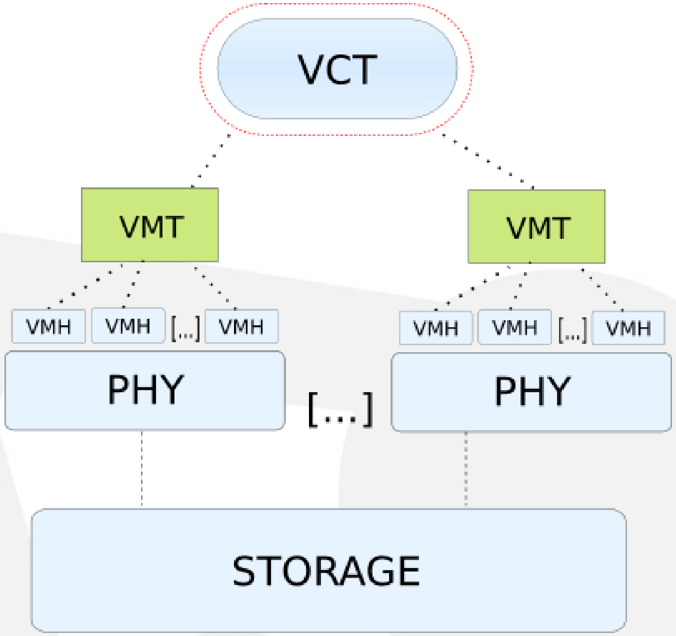
- An open source batch-queuing system.
- Supports advance reservation.
- Supports multiple computational paradigms.
- Integration with Hadoop.

VIDA: Why another tool?

- Traditional tools:
  - Virtual Machines Oriented.
  - Management operations are carried on using a polling approach.
  - Aren't very reliable.
- VIDA:
  - Virtual Cluster Oriented.
  - Management based on a heartbeat approach.
  - Very reliable.

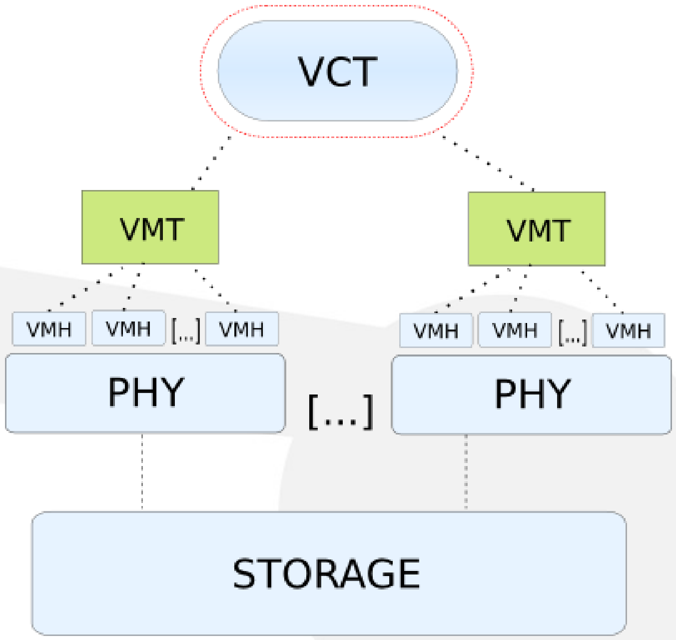
VIDA Architecture

- Virtual Cluster Tracker (VCT):
  - Manages all the clusters operations,
  - coordinates the creation of each single virtual machine,
  - collect and mantain all the status informations coming from the VMs on each node.

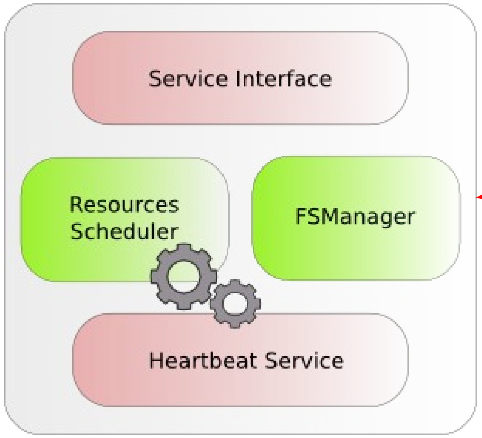


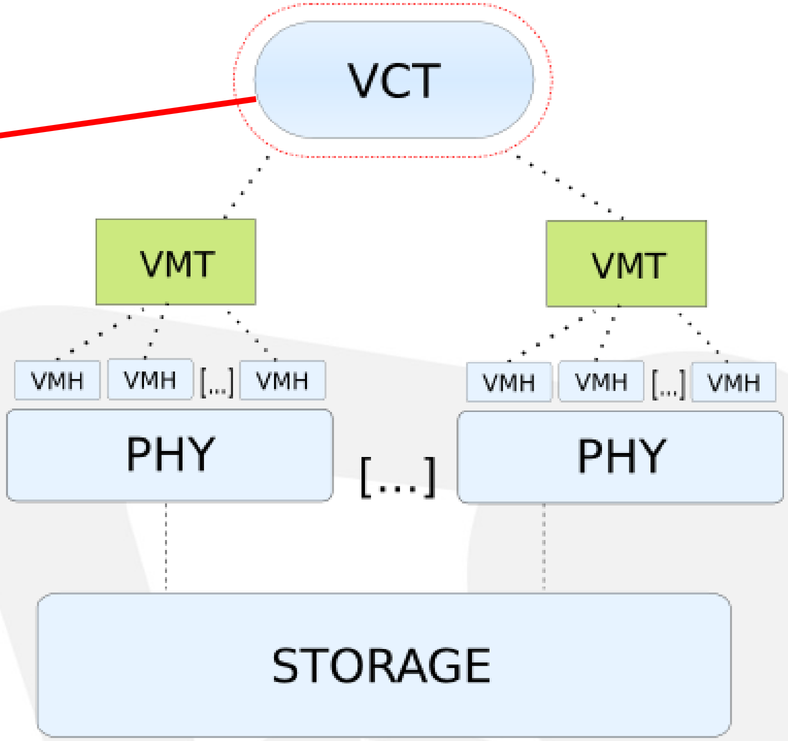
VIDA Architecture

- Virtual Machine Tracker (VMT):
  - Coordinates the operations on a specific node,
  - reports the status of the physical resources available on the host to the VCT,
  - creates and manages the virtual machines according to the directives received from the VCT.
- Virtual Machine Handler (VMH):
  - Control and administer a single virtual machine.



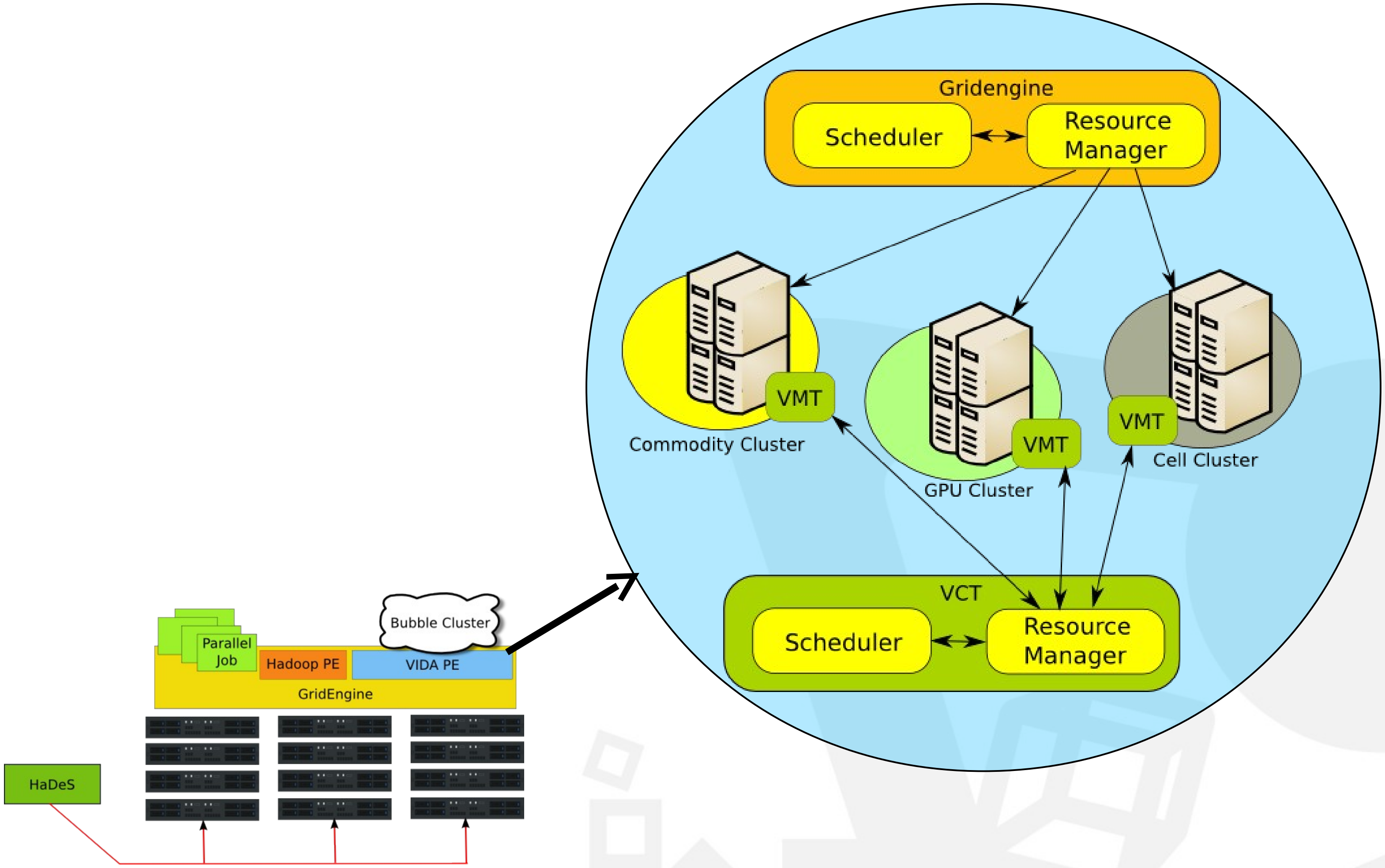
VCT



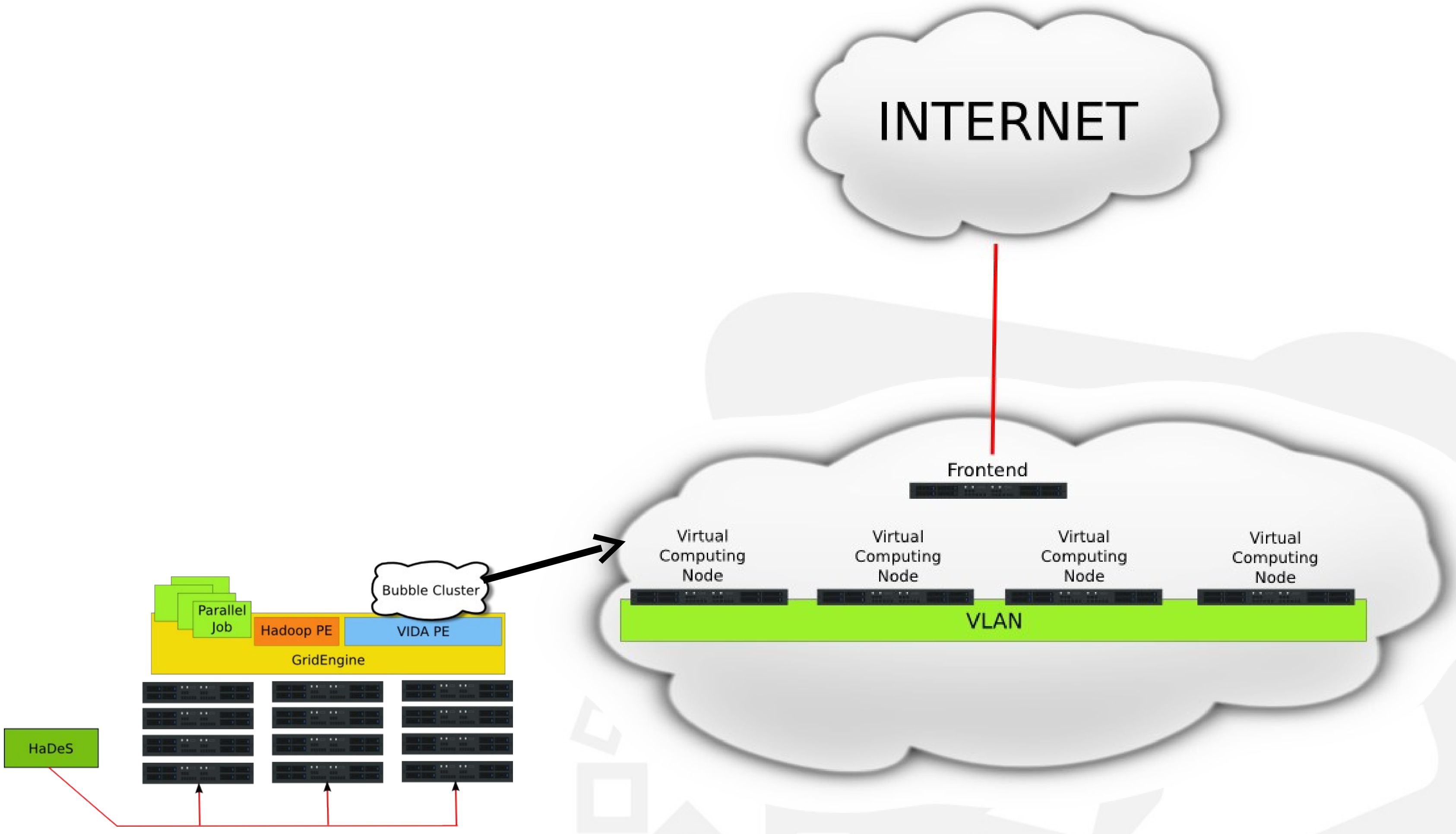


- Service Interface
- Heartbeat Service
- FSManager
- Resources Scheduler

VIDA-GE Integration

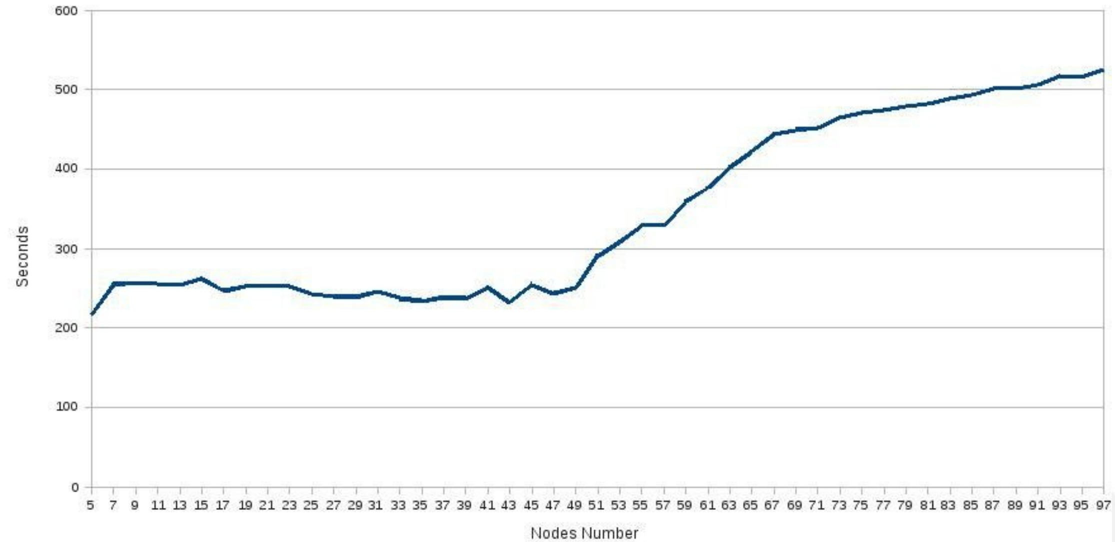


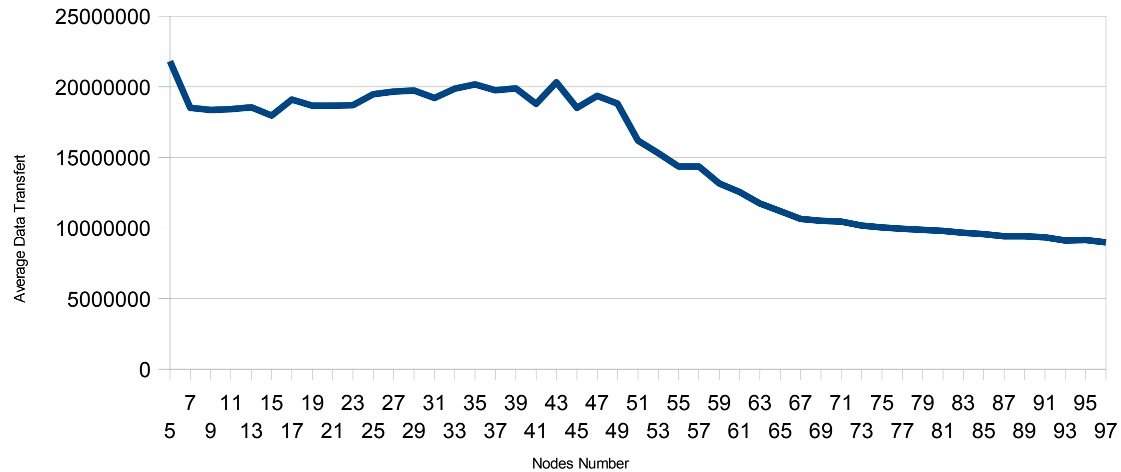
Virtual Bubble Cluster



VIDA Scalability

- Deploy Time vs Virtual Nodes Number.
- Average Data Transfer vs Virtual Nodes Number.
- Settings:
  - Core number: 132
  - Image size: 4.39 GB
  - Replication Factor: 3





- Introduction
- Virtualization and Cloud Computing
- Virtual Clusters
- System architecture
- Conclusions & Future Works

Conclusions

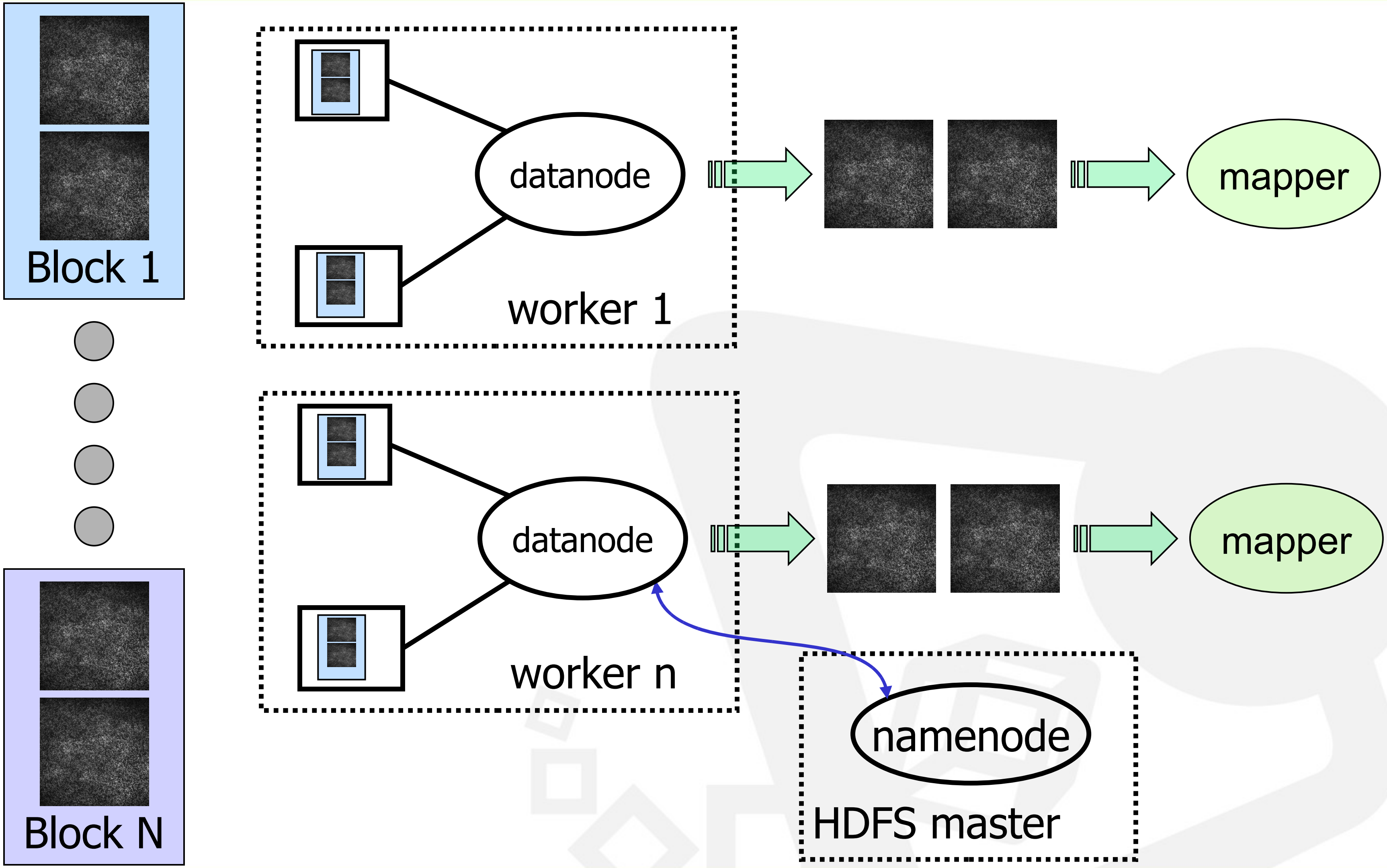
- Virtual Clusters simplify the management of HPC environments making them more flexible.
- Gridengine+VIDA = A very flexible architecture for the deployment and management of Virtual Clusters.
- Gridengine+VIDA+HDFS = A scalable architecture for the deployment and management of Virtual Clusters.

Future Works

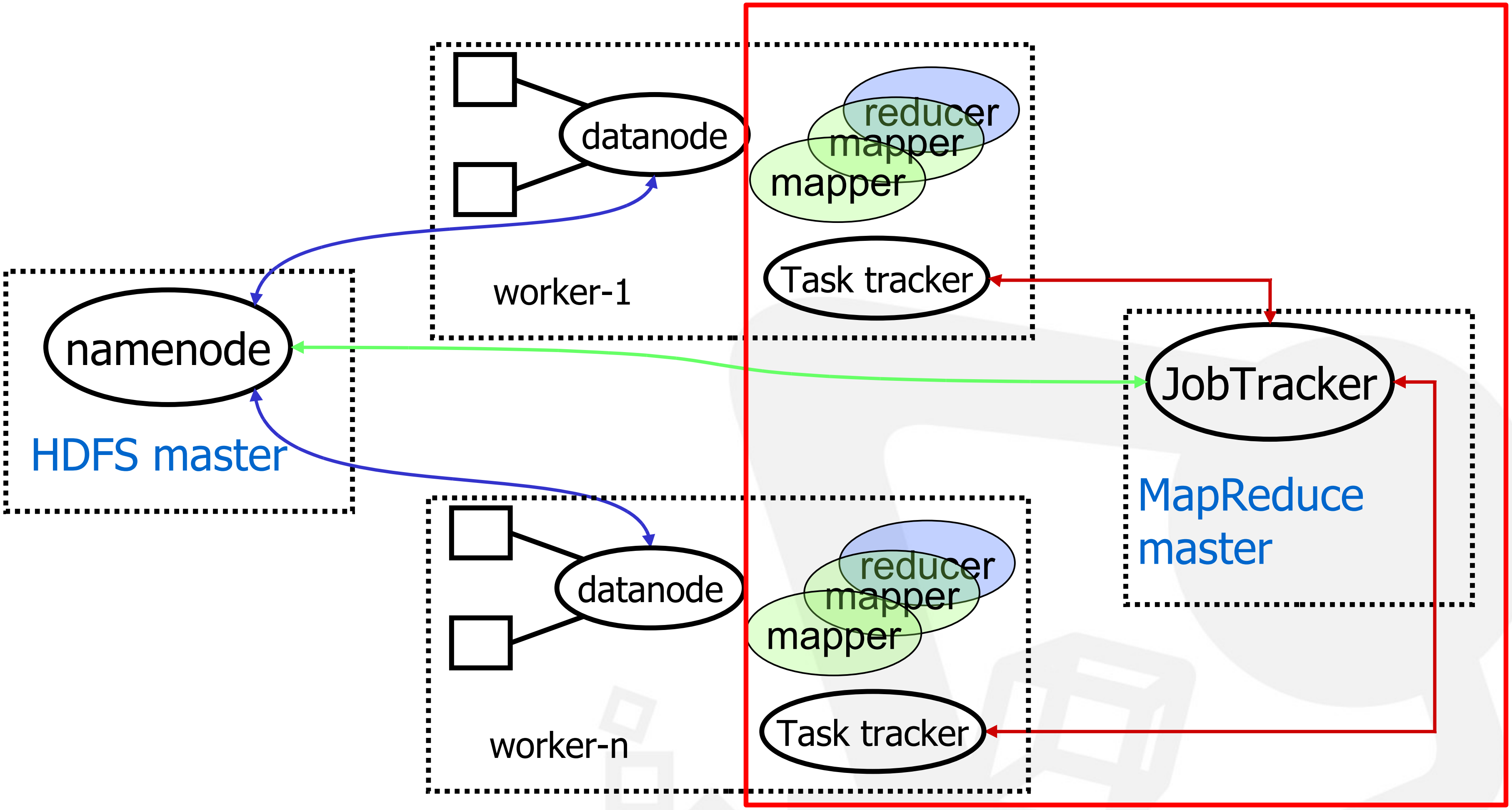
- Support for encrypted filesystems.
- VMs commissioning and decomissioning.
- Integration with the Haizea Scheduler.
- Release VIDA on Sourceforge.

THANK YOU!

WORM Data Need Specialized Filesystems

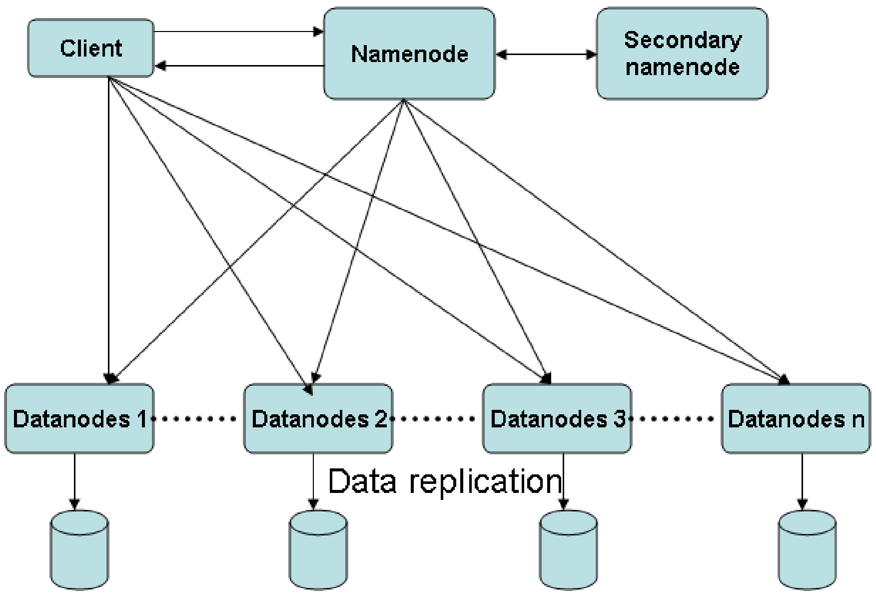


Virtual map-reduce cluster

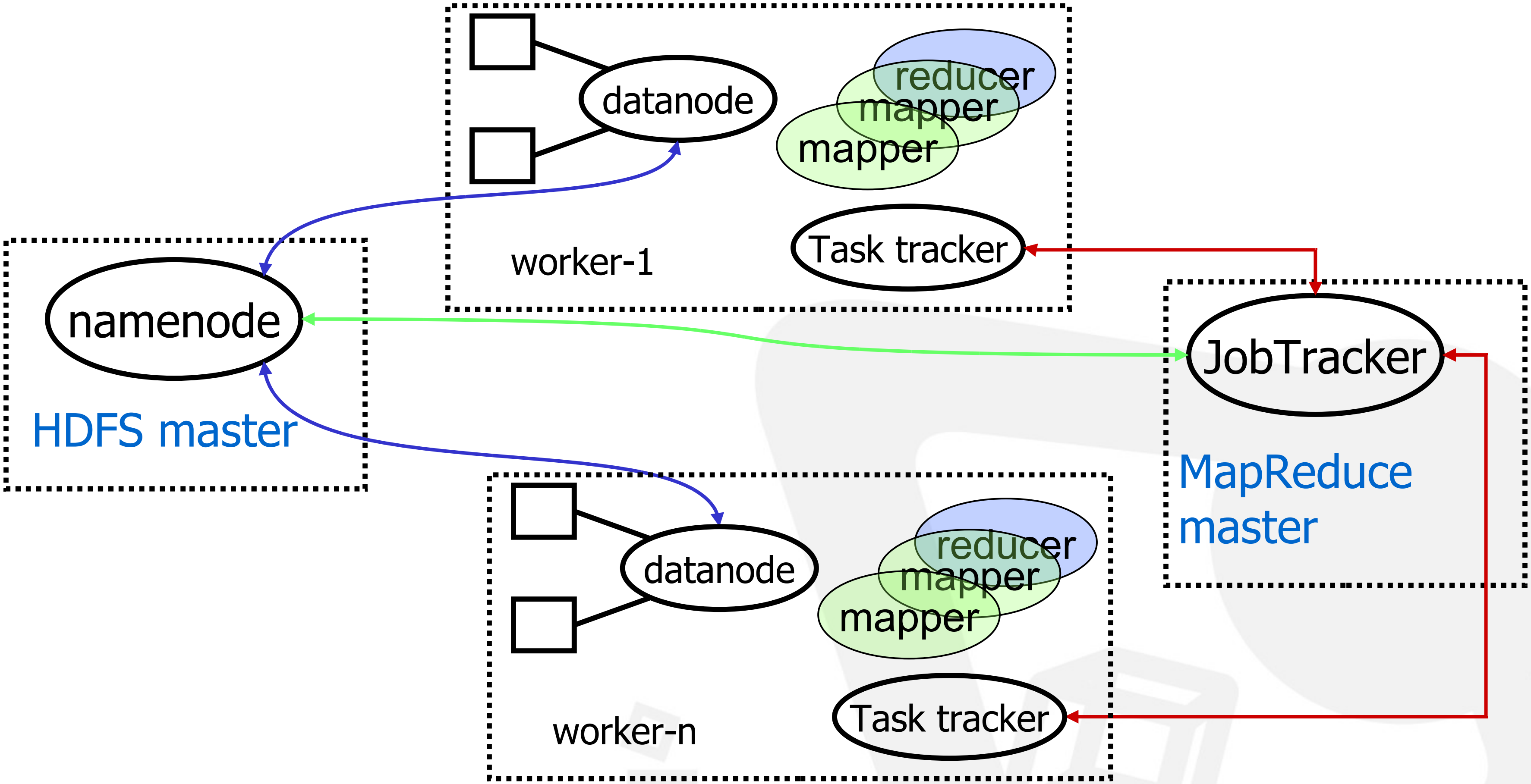


HDFS

- Distributed File System designed to run on commodity hardware.
- Suitable for applications that have large data sets.
- Highly Fault-Tolerant.

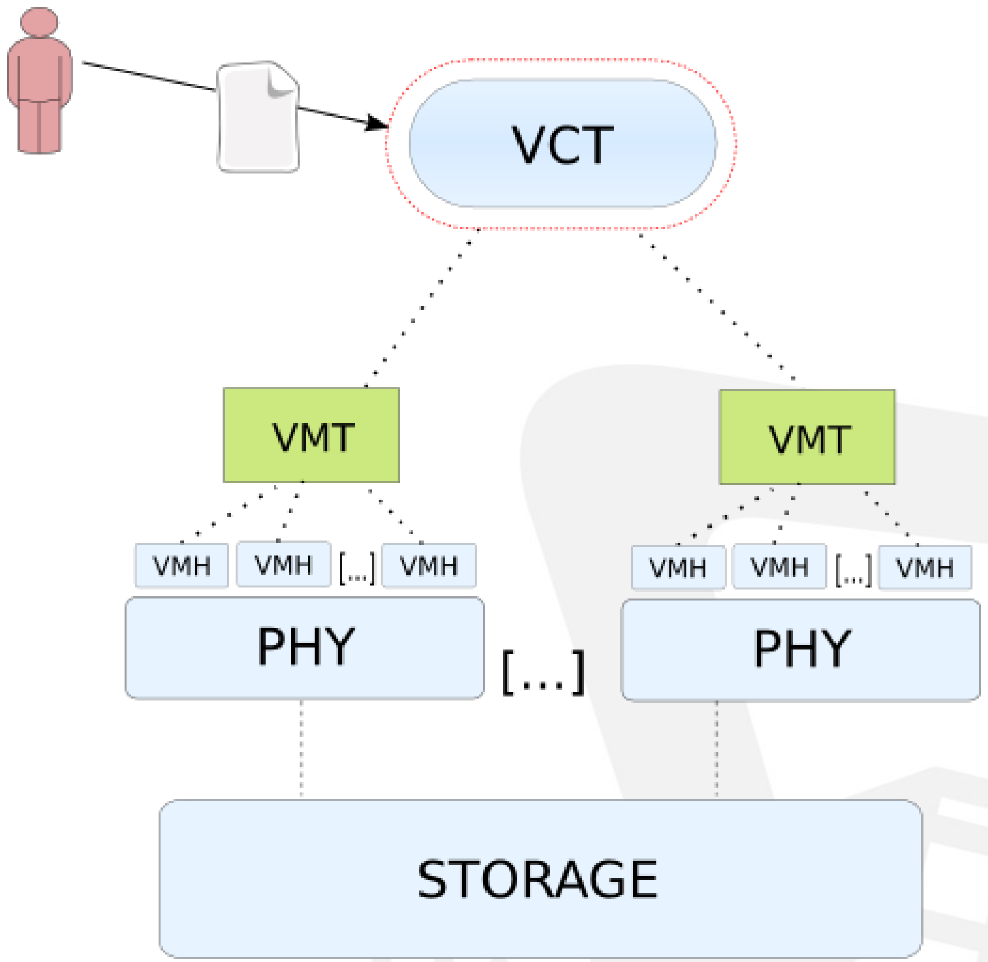


Hadoop

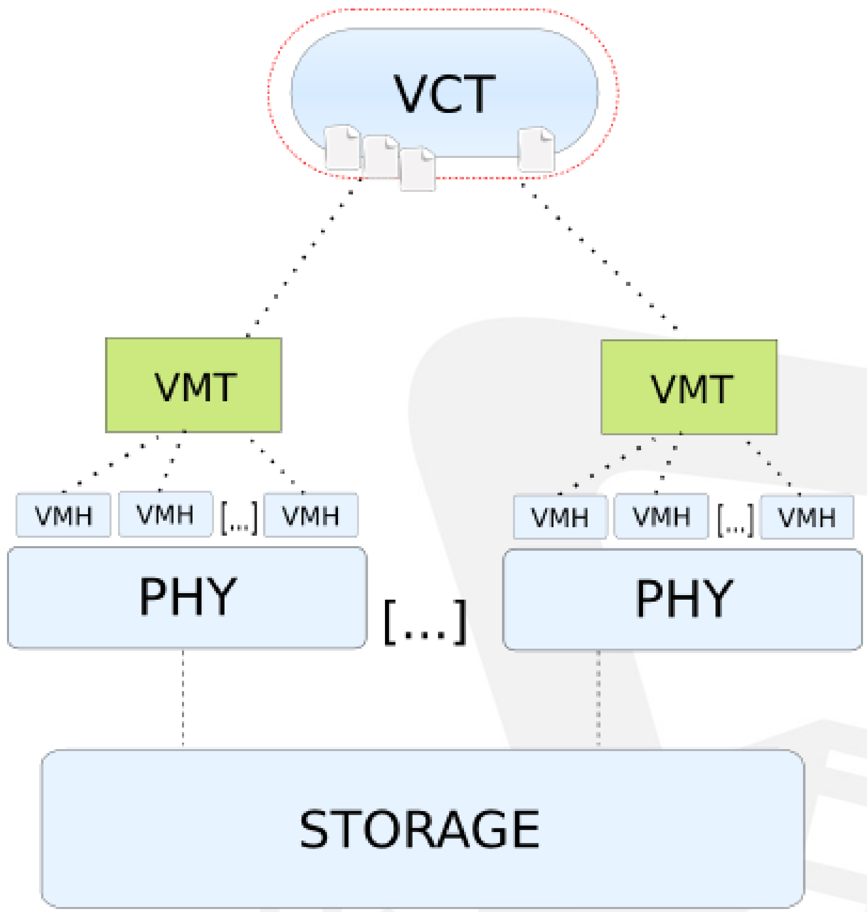


datanode
reducer
mapper
mapper
Task tracker
worker-1
namenode
HDFS master
JobTracker
MapReduce master
datanode
reducer
mapper
mapper
Task tracker
worker-n

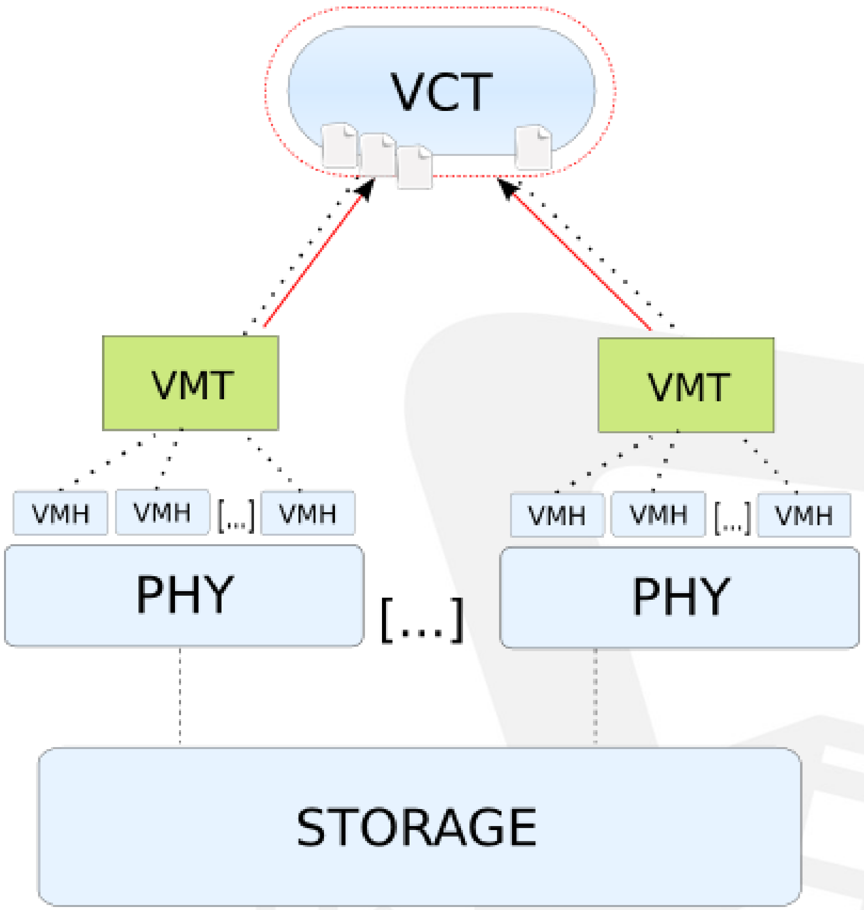
The Heartbeat Mechanism



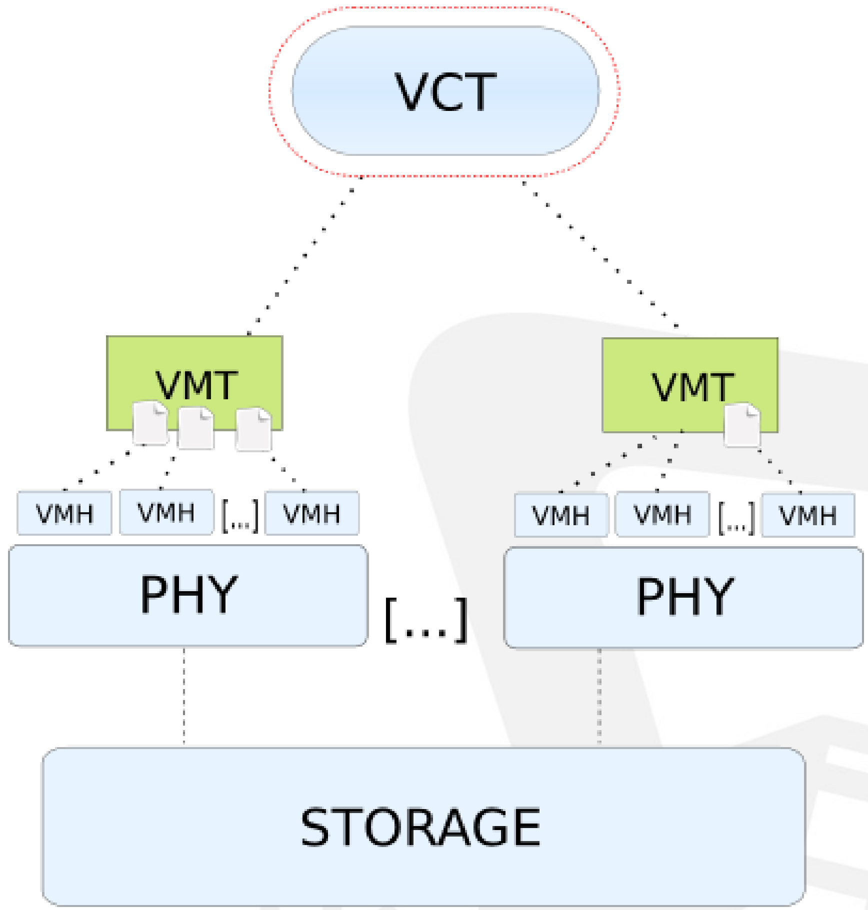
The Heartbeat Mechanism

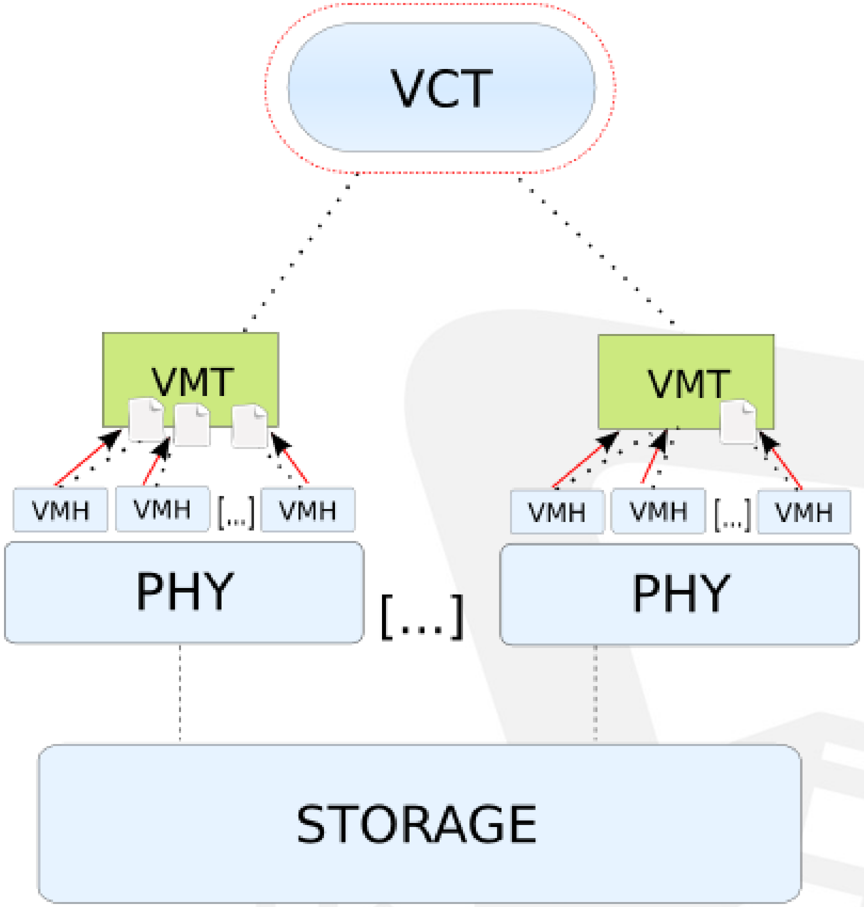


The Heartbeat Mechanism



The Heartbeat Mechanism




VCT
VMT
VMT
VMH
VMH
[...]
VMH
VMH
VMH
[...]
VMH
PHY
[...]
PHY
STORAGE

The Heartbeat Mechanism