# LLSuperCloud: Sharing HPC Systems for Diverse Rapid Prototyping

Albert Reuther, Jeremy Kepner, William Arcand, David Bestor, Bill Bergeron, Chansup Byun, Matthew Hubbell, Peter Michaleas, Julie Mullen, Andrew Prout, Antonio Rosa

IEEE-HPEC 2013

September 11, 2013

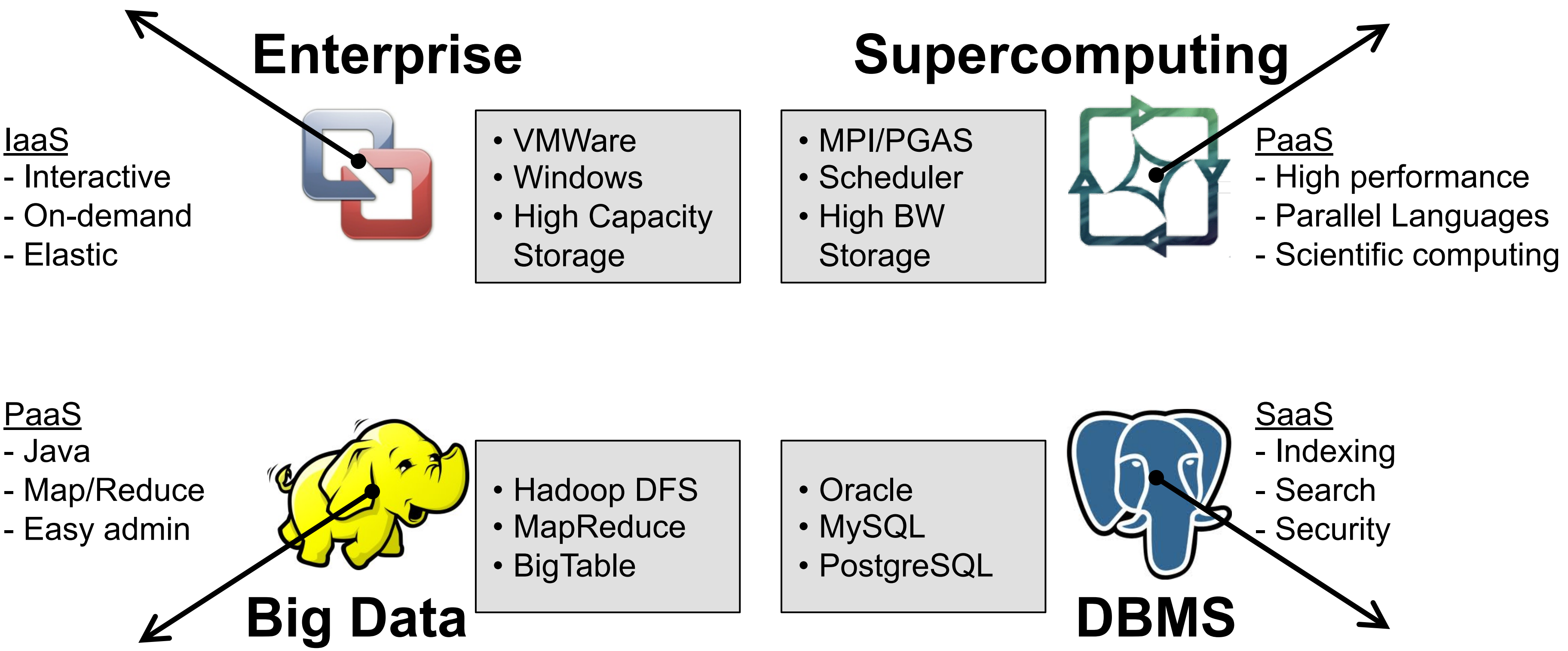
LINCOLN LABORATORY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

This work is sponsored by the Department of the Air Force under Air Force contract FA8721-05-C-0002. Opinions, interpretations, conclusions and recommendations are those of the author and are not necessarily endorsed by the United States Government.

# Outline

- **Introduction**
- **Technology Components**
- **Integration: Putting It All Together**
- **Launch Time Results**
- **Summary and Future Work**

# The Big Four Cloud Ecosystems



## Enterprise

IaaS
- Interactive
- On-demand
- Elastic

- VMWare
- Windows
- High Capacity Storage

## Supercomputing

- MPI/PGAS
- Scheduler
- High BW Storage

PaaS
- High performance
- Parallel Languages
- Scientific computing

## Big Data

PaaS
- Java
- Map/Reduce
- Easy admin

- Hadoop DFS
- MapReduce
- BigTable

## DBMS

- Oracle
- MySQL
- PostgreSQL

SaaS
- Indexing
- Search
- Security

- **Each ecosystem is at the center of a multi-$B market**
- **Pros/cons of each are numerous; diverging hardware/software**
- **Some missions can exist wholly in one ecosystem; some can't**

IaaS: Infrastructure as Service
PaaS: Platform as a Service
SaaS: Software as a Service

# The Big Four Cloud Ecosystems

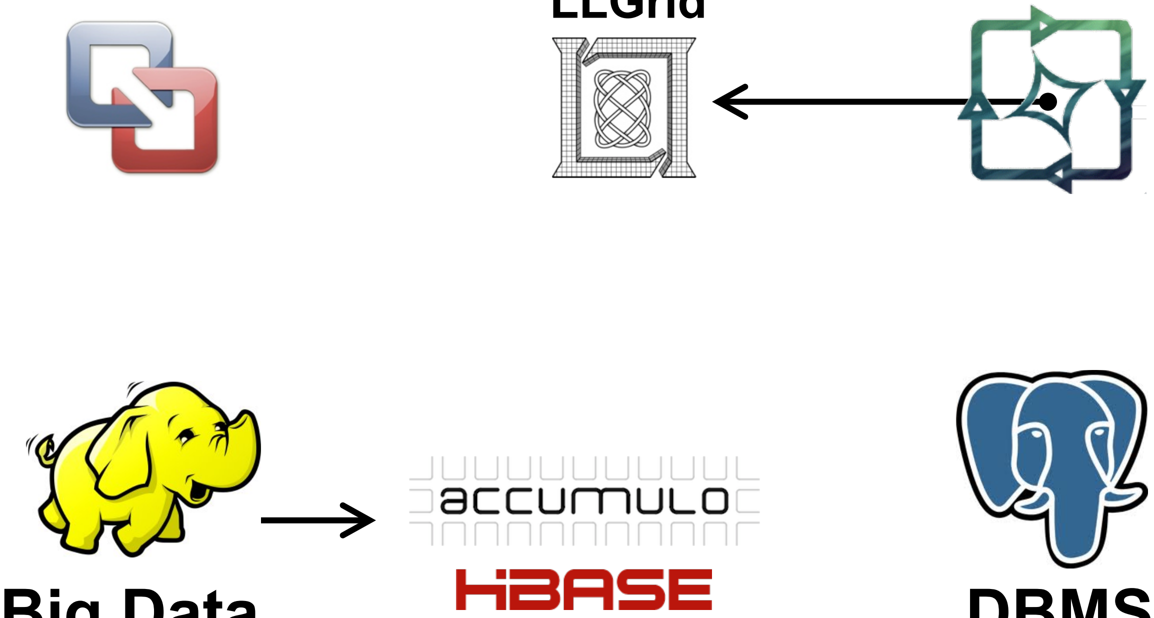
## Enterprise

IaaS
- Interactive
- On-demand
- Elastic

## Supercomputing

LLGrid

PaaS
- High performance
- Parallel Languages
- Scientific computing



PaaS
- Java
- Map/Reduce
- Easy admin

## Big Data

## DBMS

SaaS
- Indexing
- Search
- Security

- **LLGrid provides interactive, on-demand supercomputing**
- **Accumulo database provides high performance indexing, search, and authorizations within a Hadoop environment**

IaaS: Infrastructure as Service
PaaS: Platform as a Service
SaaS: Software as a Service

# The Big Four Cloud Ecosystems

Enterprise

IaaS
- Interactive
- On-demand
- Elastic

LLGrid

Supercomputing

PaaS
- High performance
- Parallel Languages
- Scientific computing

LLSuperCloud

PaaS
- Java
- Map/Reduce
- Easy admin

Big Data

LLMapReduce

accumulo

HBASE

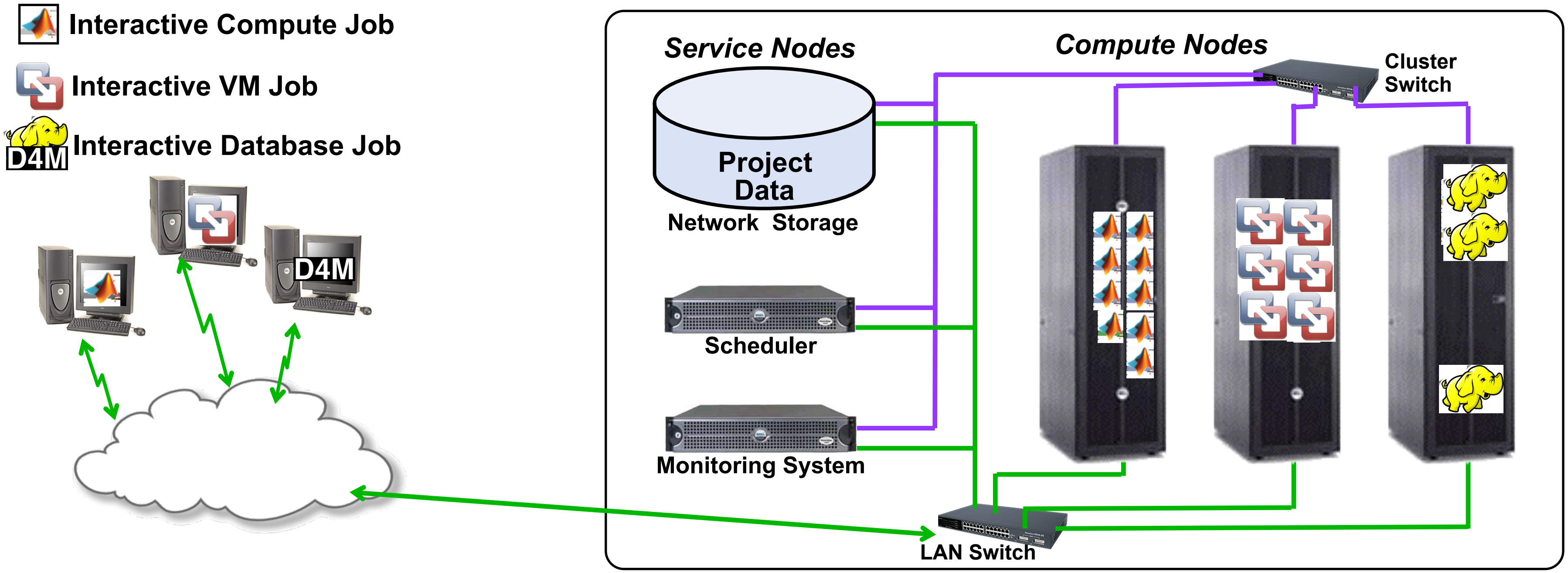
DBMS

SaaS
- Indexing
- Search
- Security

**LLSuperCloud enables a multi-ecosystem prototyping environment**

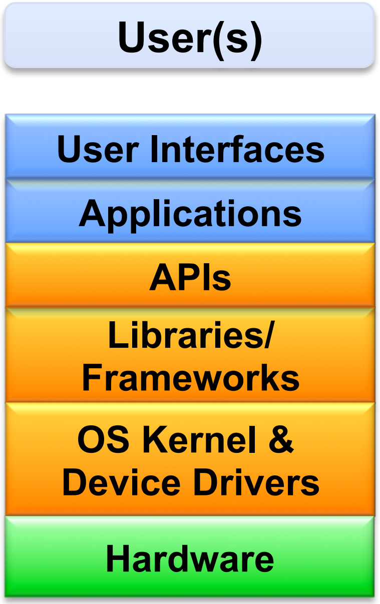
IaaS: Infrastructure as Service
PaaS: Platform as a Service
SaaS: Software as a Service

# LLSuperCloud



- LLSuperCloud allows traditional supercomputing, VMs and Hadoop/ Accumulo to dynamically share the same hardware
- Virtual Machines (VMs) give users sys admin control of their environment (e.g., OS, web services, build environment)
- Databases (DBs) give users low latency atomic access to large quantities of unstructured data with well defined interfaces-

# Operating System Basics


User(s)

User Interfaces

Applications

APIs

Libraries/ Frameworks

OS Kernel & Device Drivers

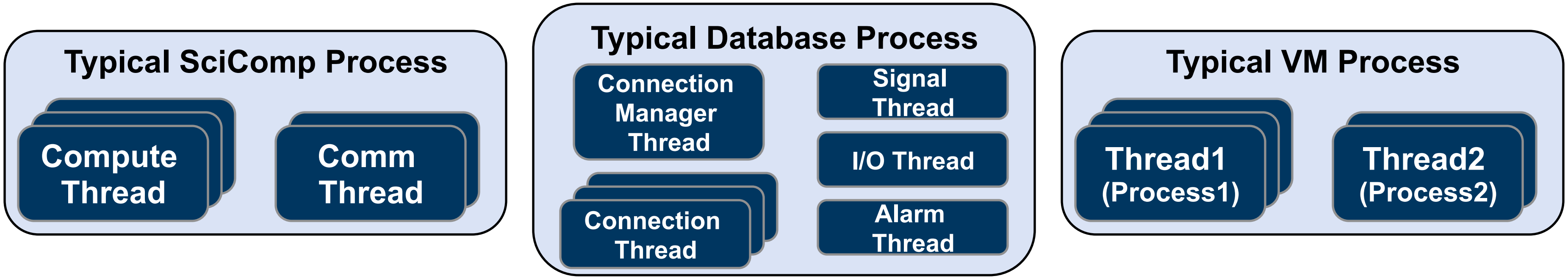
Hardware

- Applications and their User Interfaces – Programs that users run
- Application Programming Interface (API) – Rules and specifications for libraries and frameworks
- Libraries/Frameworks – Reusable software routines for building applications
- Operating System (OS) Kernel – Manager of computer hardware resources and of common services for application software
- Hardware – Physical components of the computer

- **Manages and controls shared hardware resources**
- **Provides common services to applications**
- **Abstracts hardware for users and applications**

Silbershatz, Galvin and Gagne, *Operating System Concepts*, Addison Wesley, 2011.

# Processes and Threads



| | Linux Process | Linux Thread (Lightweight Process) |
|---|---|---|
| Process State (waiting, running, etc.) | Own | Own state along with process state |
| Program Counter | Own | Own PC |
| CPU Registers | Own | Own registers |
| CPU Scheduling Info | Own | Shared |
| Associated Memory Pages (including Stack, Heap, etc.) | Own | Own stack, otherwise shared |
| I/O Files, Status | Own | Shared |
| Accounting Info | Own | Shared |

A. Silberschatz, P.B. Galvin, G. Gagne, *Operating System Concepts*, Ninth Ed., Wiley, 2012.

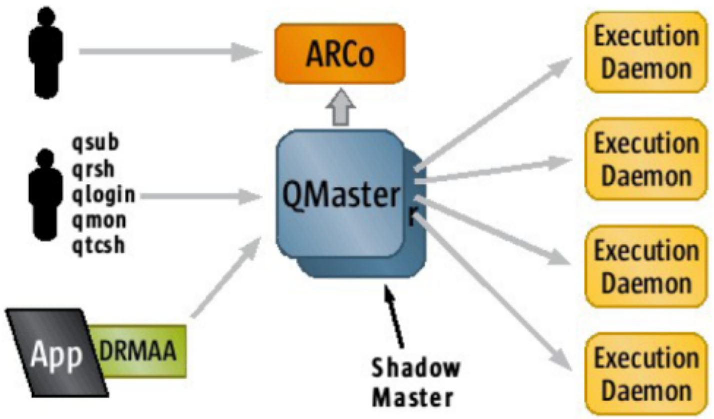
# Outline

- Introduction
- Technology Components
- Integration: Putting It All Together
- Launch Time Results
- Summary and Future Work

# Resource Manager / Scheduler

- **Resource Manager**
  - **Tracks compute resources**
    - **Resource capabilities**
    - **Status**
  - **Launches processes on compute nodes**
  - **Tracks processes**
    - **Assignments**
    - **Status**
    - **Execution state**
- **Scheduler**
  - **Takes input from resource manager**
  - **Matches job resource requirements with compute resources**

GRID ENGINE



From: Grid Engine Users Guide, Release 6.2 Update 7

**IBM Platform LSF**

HTCondor High Throughput Computing

GRID ENGINE

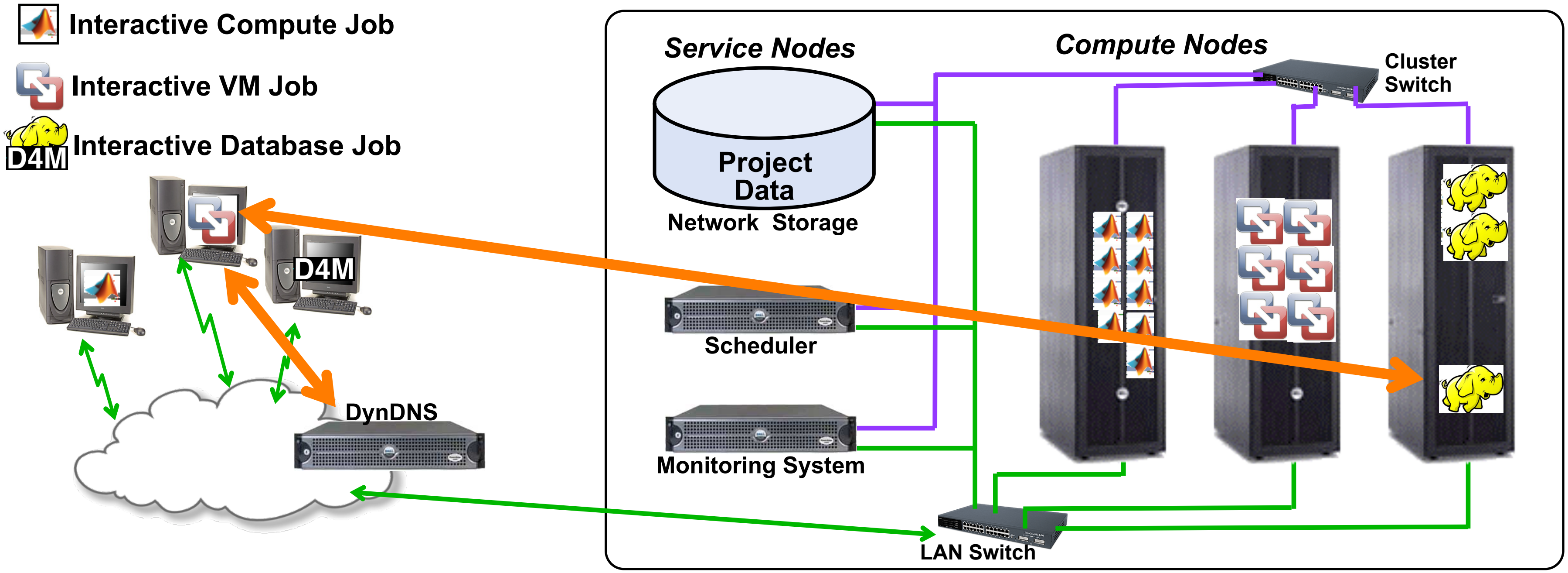
CLUSTER RESOURCES **Torque**

Adaptive COMPUTING **Maui & Moab**

PBS Professional®

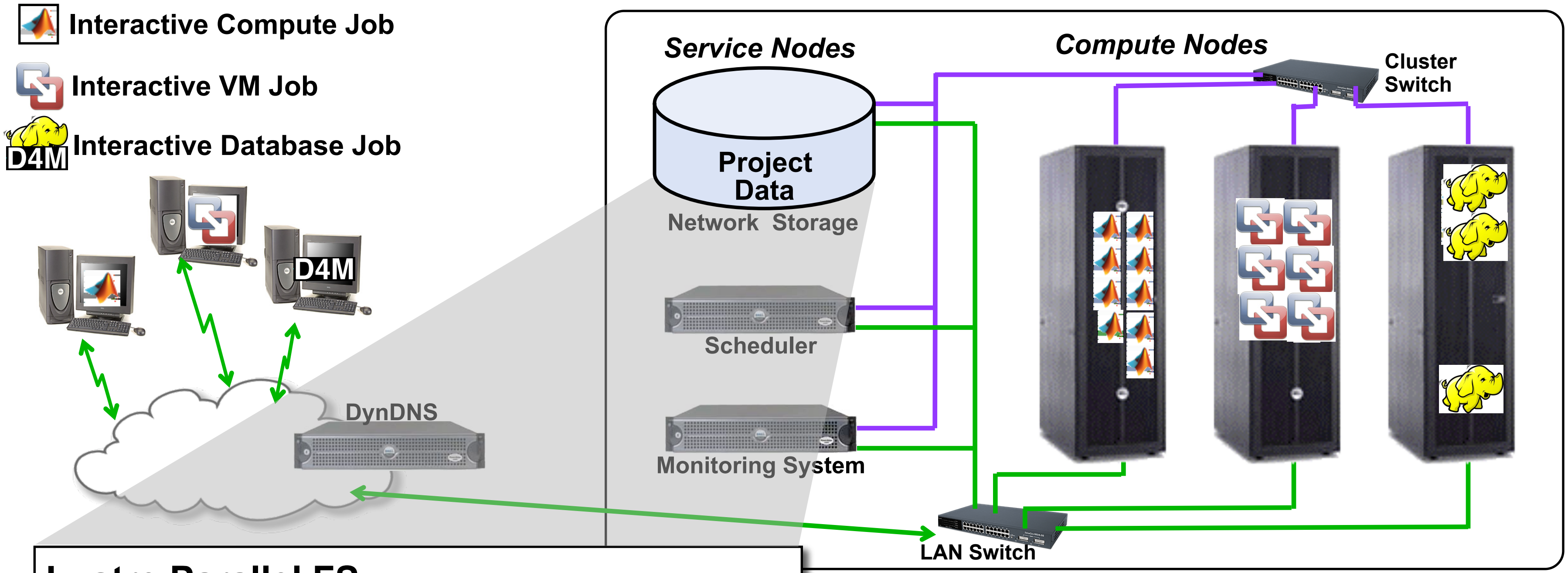
ARCo = Accounting and Reporting Console

# Dynamic Domain Name Service (DNS)



- Launching a service onto LLGrid places it on any one of the compute nodes
- Service must register it’s IP address to DynDNS server
- Clients then use URL name to abstract IP address

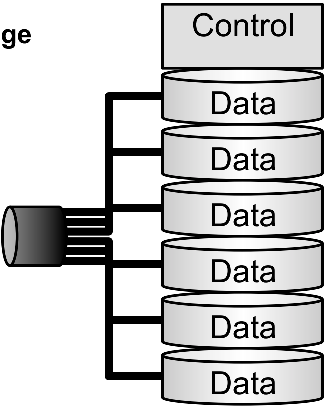
# Lustre Central File System



## Lustre Parallel FS

**High throughput, parallel data file storage**

- Data striped across many data servers
- Extremely scalable
- Have been using for several years
- Expanding our use
- TX-Green Pioneer: 1.2 PB

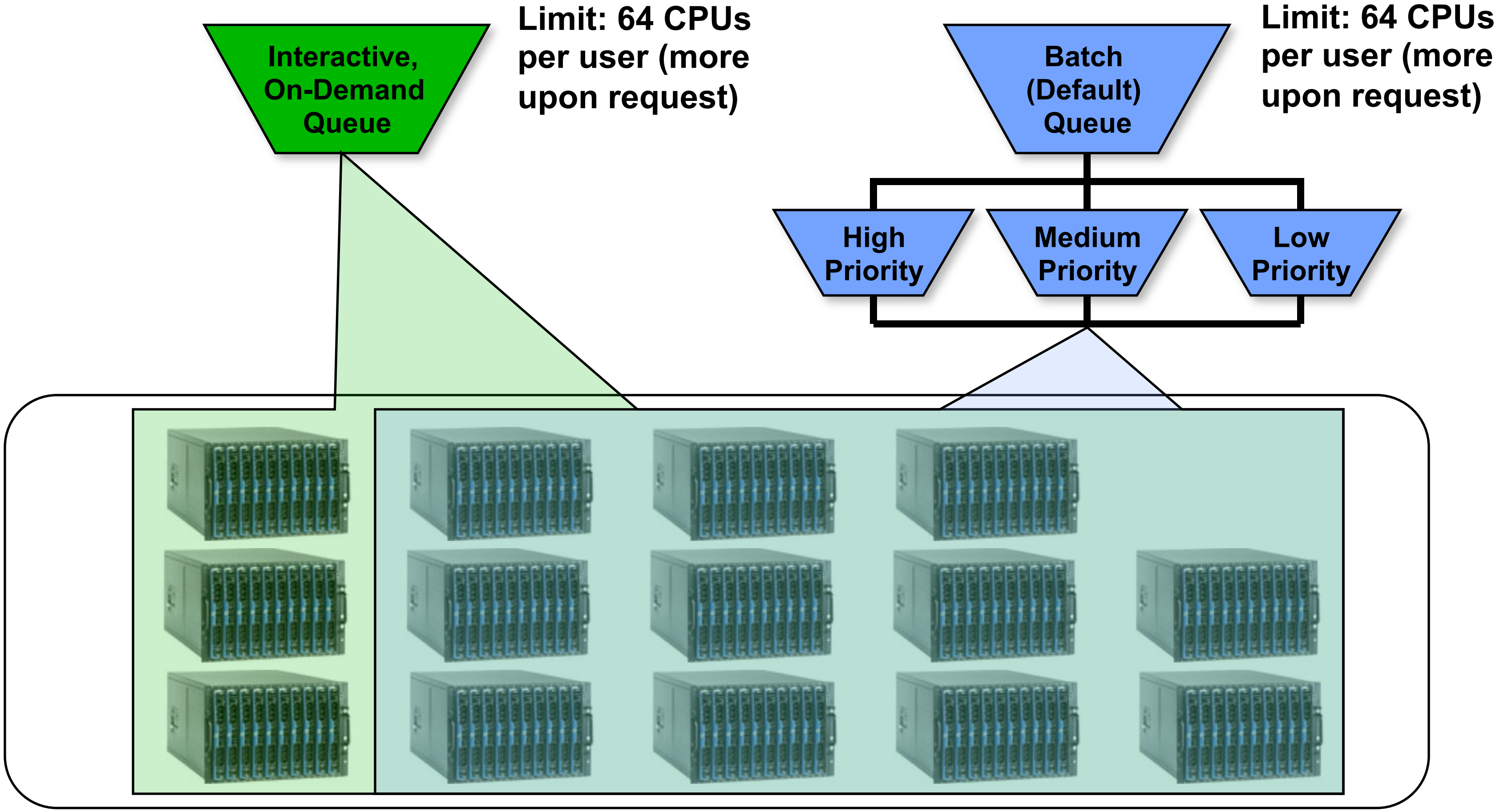


- **Lustre enables high bandwidth (large) file transfers**
- **Takes advantage of multiple data servers and network links**

# Outline

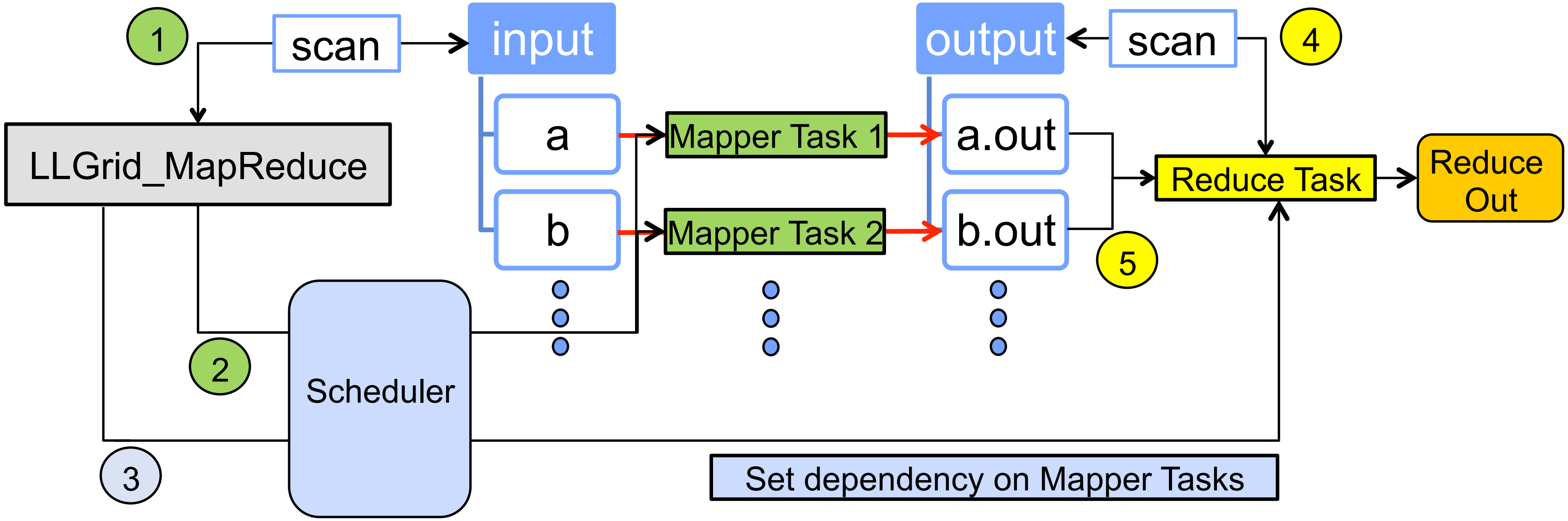
- Introduction
- Technology Components
- Integration: Putting It All Together
- Launch Time Results
- Summary and Future Work

# LLGrid User Queues for Interactive and Batch Jobs


Interactive, On-Demand Queue
Limit: 64 CPUs per user (more upon request)
Batch (Default) Queue
Limit: 64 CPUs per user (more upon request)
High Priority
Medium Priority
Low Priority

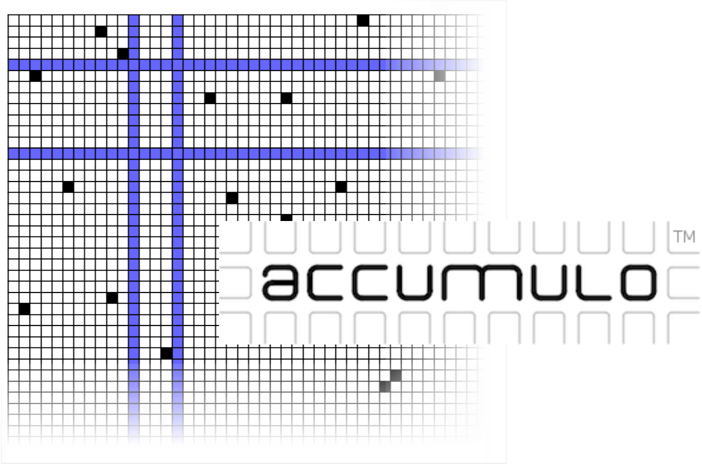
- Not using scheduler’s interactive features
- CPUs for interactive, on-demand jobs only
- CPU allotments will change when upgrading to larger system

# LLGrid_MapReduce Diagram



- **Launches job array with as many jobs as there are input files to batch queue**
- **Can execute jobs in any programming language (not just Java)**
- **Optional reduce task can compile results from mapper jobs**

# Accumulo (NoSQL) Database



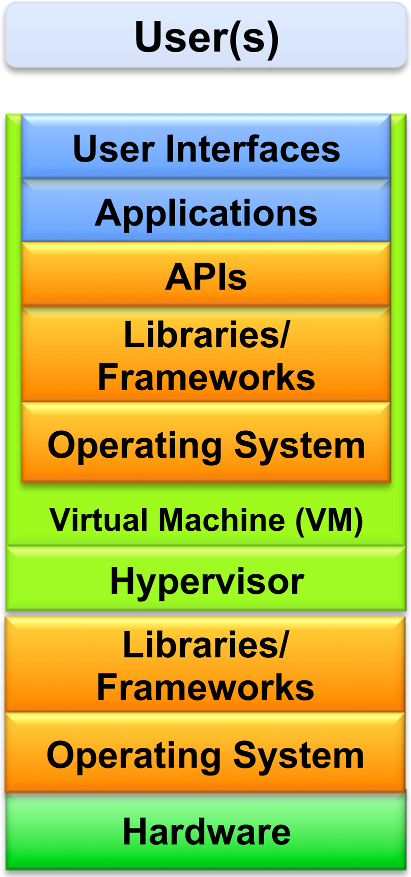
## Properties of Accumulo

- NOSQL-type database
- Utilizes row-column-value triples to store data
- Designed for fast ingest and queries on massive datasets
- Able to be deployed on distributed systems

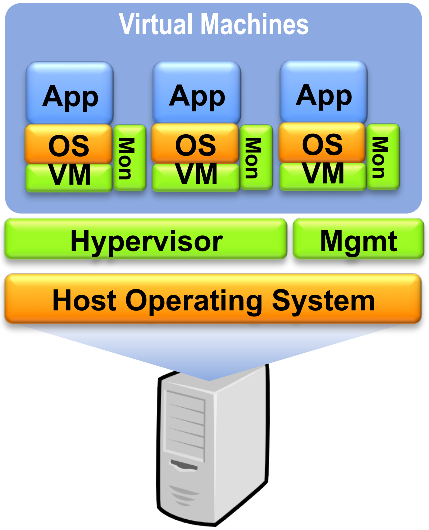
- Currently supports single-node Accumulo instance
- Data is pre-staged on node where Accumulo instance is launched
- Dynamic DNS name entry enables access of each instance
- On suspension, data is archived on central file system

| Scenario | Execution Time |
|---|---|
| Empty database startup | ~90 sec |
| Empty database stop | ~90 sec |
| 13.6 GB database startup | ~240 sec |
| 13.6 GB database stop | ~90 sec |
| 200 GB database startup | <10 min |

# Virtual Machine Jobs



- Only Type 2 (Host OS and hypervisor) supported
- Launched through scheduler

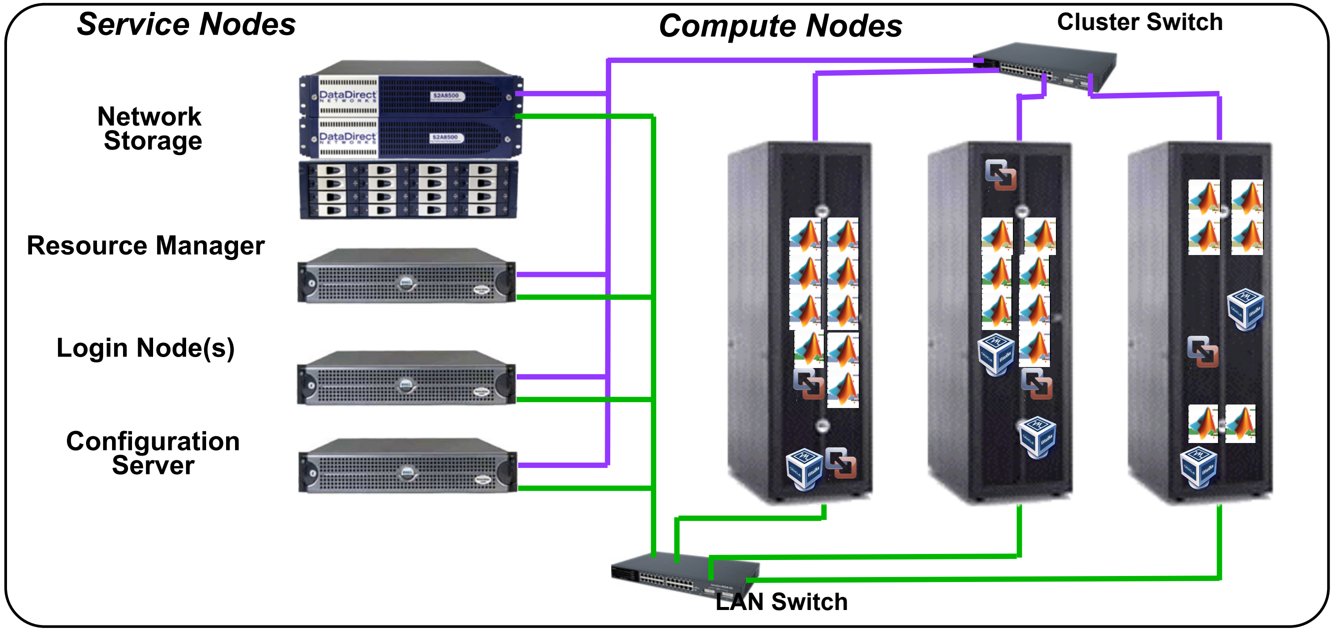


- To launch:
  - VM image cloned
  - VM imaged registered to compute node
  - Job execution written into init scripts
- To exit/terminate:
  - Shutdown script executed
  - VM image discarded

# Outline

- Introduction
- Technology Components
- Integration: Putting It All Together
- **Launch Time Results**
- Summary and Future Work

# Launch Time Experiment Setup

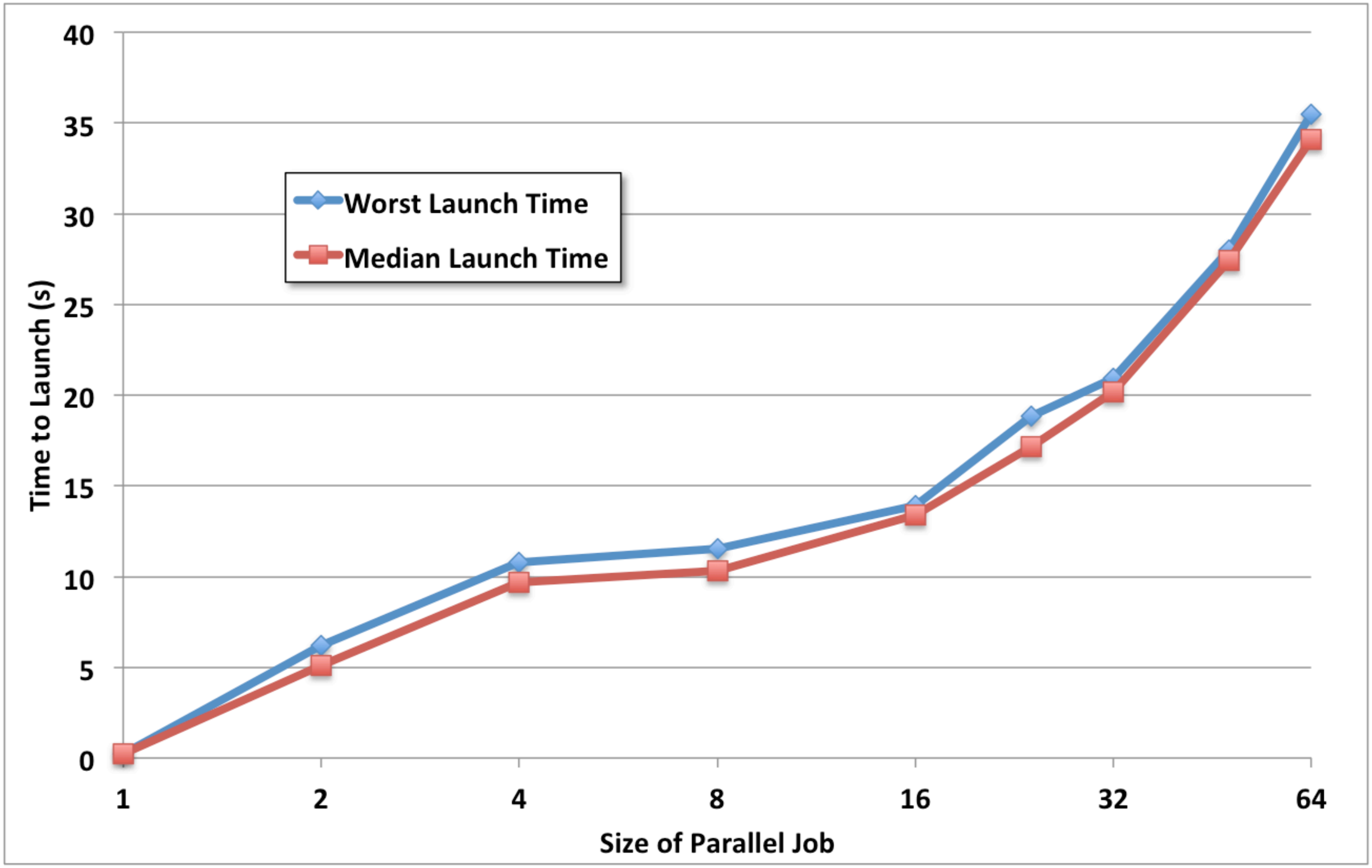


- Dell PowerEdge 1955 blades
- Dual-dual core 3.2 GHz Xeon CPU
- 8 GB RAM per blade
- 10GigE core network, 1GigE to blades
- DDN SFA 10K storage array
- Grid Engine ver. 6.2u5 scheduler
- VM images: Debian Linux 6.0.4 i386

- Eight dedicated nodes
- Compare optimized Virtual Box VMs and optimized VMWare images
- Varied jobslots launched and jobslot overloads
- Socket-based time logger

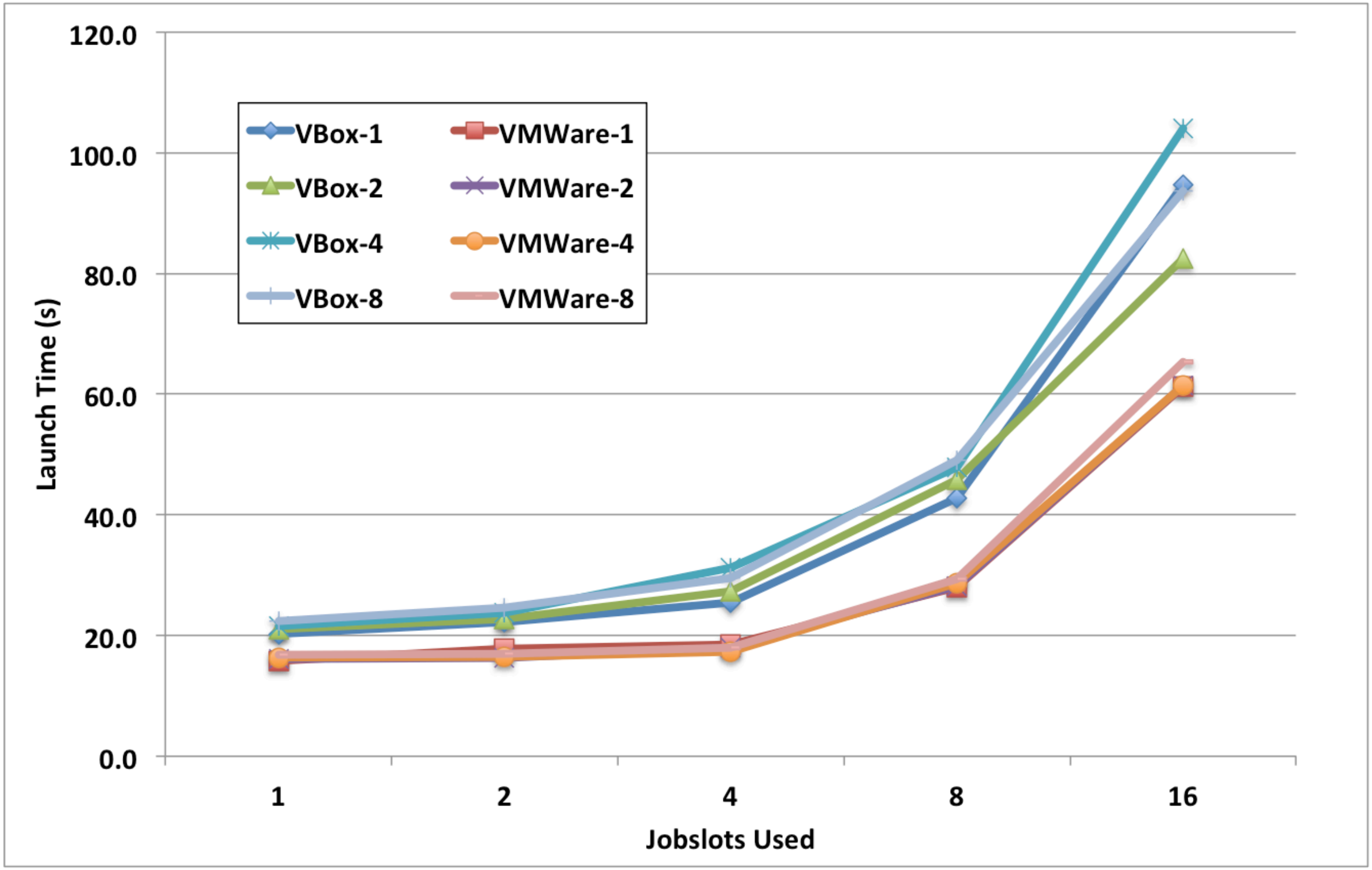
# pMatlab Launch Time Results

## Representative of Interactive, Batch, and LLMapReduce Jobs



- Launch times are tightly distributed
- Time overhead reasonable for interactive, on-demand launches

# VM Launch Time Results



- VMWare VMs launch markedly faster than Virtual Box
- Overloading jobslots does not impact launch time much

# Outline

- Introduction
- Technology Components
- Integration: Putting It All Together
- Launch Time Results
- Summary and Future Work

# Summary and Future Work

- Demonstrated flexible HPC prototyping capability for simultaneous HPC, cloud, database, and VM work
- Enabled by several necessary services: resource manager/ scheduler, DynDNS, Lustre central file system
- Modest launch times

- Future work
  - Add dynamic clustered/distributed database support
  - Add virtual network support
  - Add parallel virtual machine job launches for legacy MPI applications