Selection of Virtual Machines Based on Classification of MapReduce Jobs

Adam Pasqua Blaisse, Zachary Andrew Wagner, and Jie Wu

Department of Computer and Information Sciences, Temple University

Cloud Computing

- Large number of physical machines (PM)
- Strongly networked together
- Resources sold on an hourly basis as virtual machines (VM)
- Eucalyptus
- Amazon EC2

Motivation

- Find the minimal virtual machine that will run a Map Reduce job as fast as possible

Region: US East (N. Virginia)

	vCPU	ECU	Memory (GiB)	Instance Storage (GB)	Linux/UNIX Usage
General Purpose - Current Generation					
t2.micro	1	Variable	1	EBS Only	$0.013 per Hour
t2.small	1	Variable	2	EBS Only	$0.026 per Hour
t2.medium	2	Variable	4	EBS Only	$0.052 per Hour
m3.medium	1	3	3.75	1 x 4 SSD	$0.070 per Hour
m3.large	2	6.5	7.5	1 x 32 SSD	$0.140 per Hour
m3.xlarge	4	13	15	2 x 40 SSD	$0.280 per Hour
m3.2xlarge	8	26	30	2 x 80 SSD	$0.560 per Hour

Map Reduce

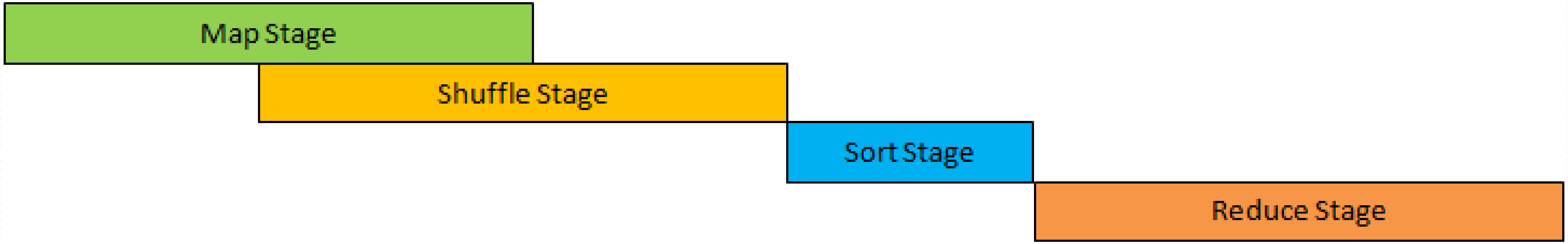
- Programming Paradigm for distributed computing
- Two phases
  - Map Phase
  - Reduce Phase
- Apache Hadoop
  - Open source implementation used

Hadoop Implementation of Map Reduce

- Map
  - Many small Map tasks
  - Each task takes a small chunk of data
  - Turn the data into Key value pair (i.e <the,1>)
  - Number of Map tasks varies based on input data size
  - When all Map task are finished data is Pasted to the Reduce Phase
- Reduce
  - Very few set number of Reduce tasks
  - Combine all the input key value pairs from the maps
  - Also takes care of shuffling data from Map Locations to Reduce Locations

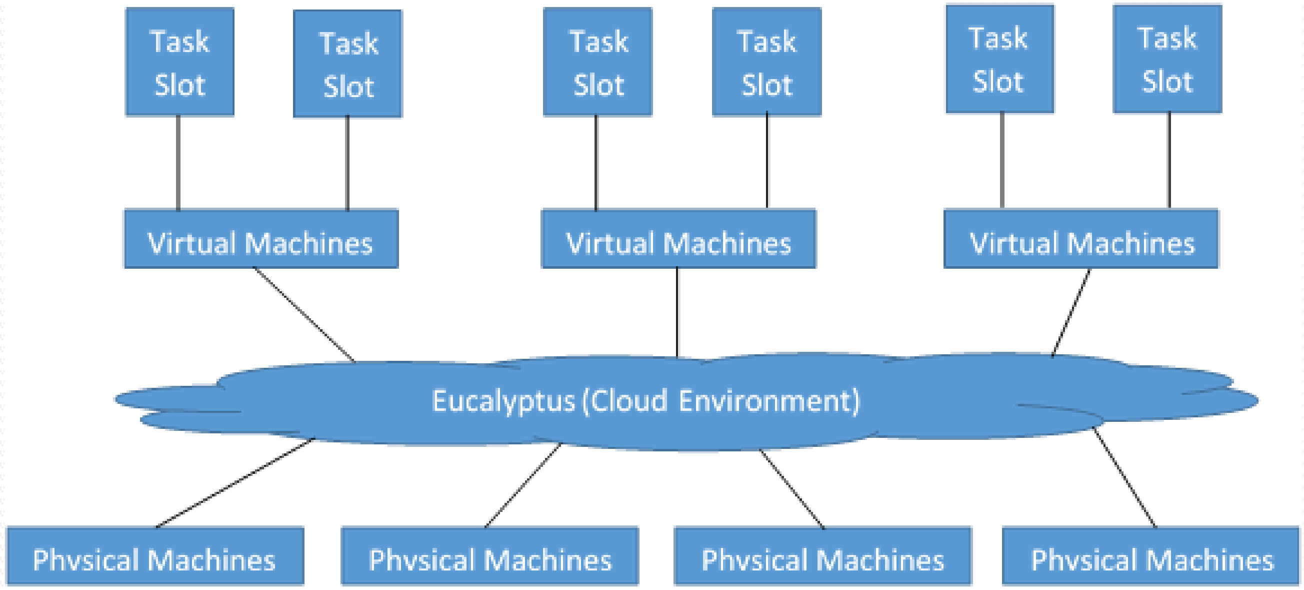
Hadoop Implementation of Map Reduce

- Reduce
  - All Mapping must finish before Reducing can start
  - Shuffling can start before Mapping ends



Issues when Used Together

- Some jobs run better on different configurations of virtual machines
- Different configurations of virtual machines have different costs
- Some jobs may need more CPU's while others may need I/O
- I.E Generating Data is I/O intense, and would be best run on Memory rich system



Our Approach

- Attempt to classify tasks into two types
  - CPU Bound Jobs
    - Jobs spent more time doing CPU work then I/O
    - Jobs need more CPUS's and less I/O
    - Smaller more numerous machines
  - I/O Bound Jobs
    - Jobs spent more time doing I/O work then CPU
    - Jobs need more I/O and less CPU
    - Less Larger Machines

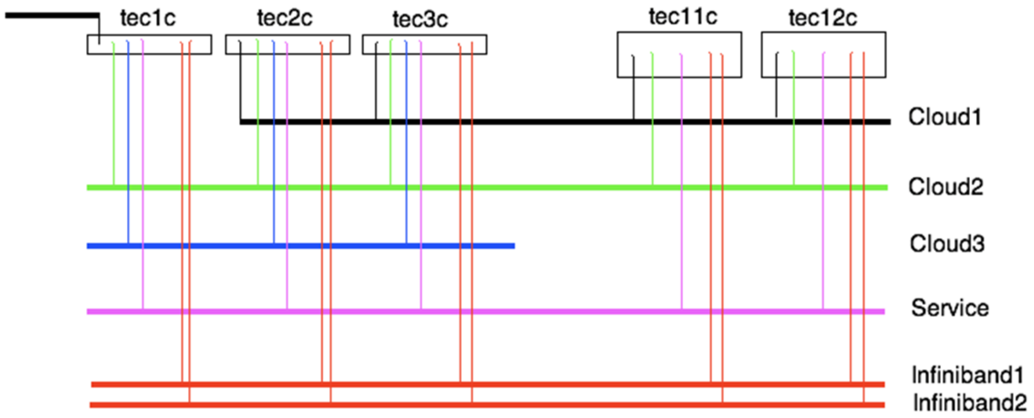
Mapping to machines

- If a job is classified as
  - CPU Bound Job
    - Many virtual machines
    - Little memory per virtual machine
  - I/O Bound Job
    - Fewer virtual machines
    - Each virtual machine has larger amounts of memory

Why?

- If a job is I/O bound
  - Would like to keep job running in memory rather then hit HDD
  - I/O more important then number of cores
- If a job is CPU bound
  - More important to have many cores running the maps
  - Less likely to hit HDD while running

TCloud (Virtual Cluster)



Cloud1 - GigE - All but tec1c connected - good for intra-VM communication

Cloud2 - GigE - All nodes connected - good for NAT access and network file system access

Cloud3 - GigE - All nodes 1U connected / No GPU systems connected - unsure what we can utilize other than intra-VM's limited to these nodes

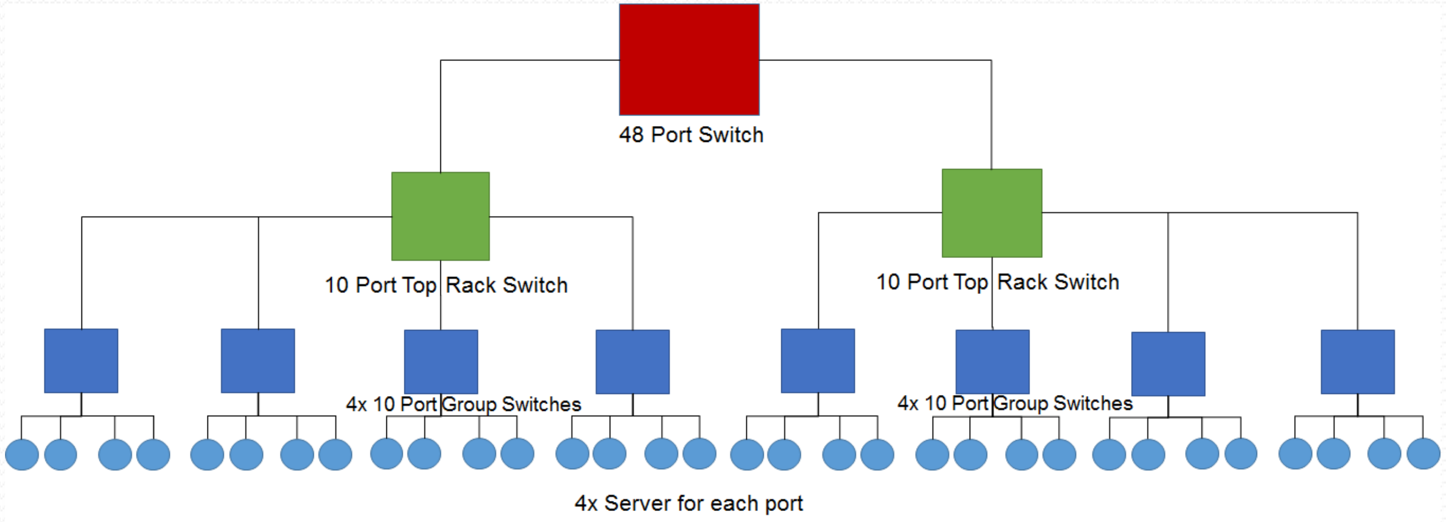
Service - GigE on 1U systems/ 100Mb on GPU systems - Nimbus cluster services and possibly cluster network file systems

Infiniband# - 40Gb - MPI access to Data storage for Nimbus

- Hardware
  - 12 Dell Power Edge R614 Servers
  - 96 conventional CPU Cores
  - 4-Way redundant 10 GB Ethernet
  - 2-Way redundant InfiniBand
- Software
  - Eucalyptus 3.3 (Amazon EC2 compatible)
- Public cloud used to create virtual machine clusters

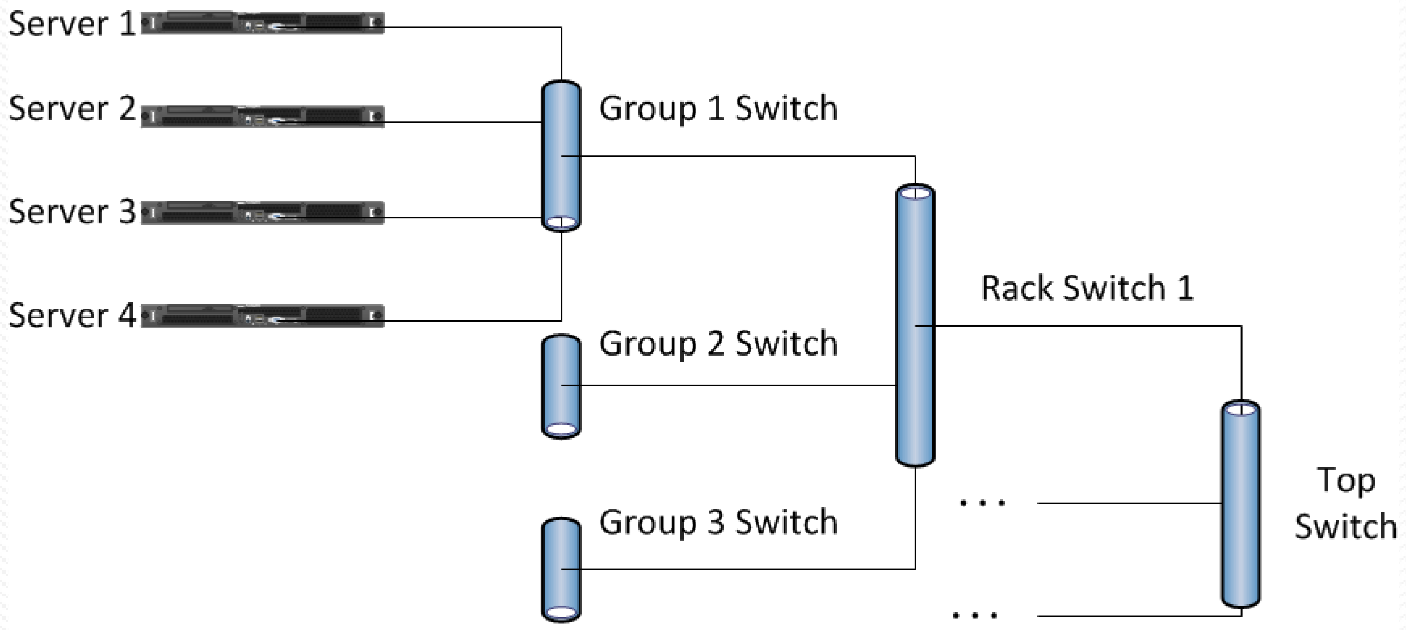
Net Cloud (Physical Cluster)

- Hardware
  - 32 Dell PowerEdge R210 servers
  - Each server has
    - 4 GB of RAM Memory
    - 500 GB HDD
- Software
  - Hadoop version 1.2.1
  - CentOS 6.6
- Physical machine cluster used for prediction



Net Cloud (continued)

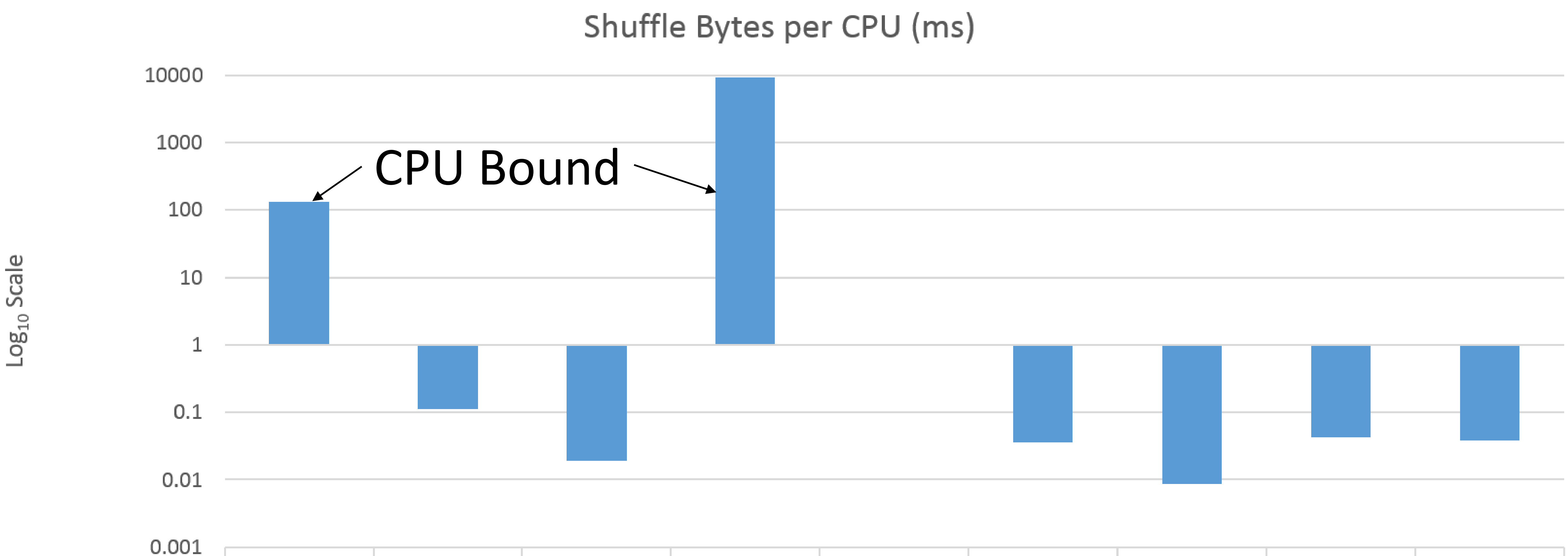
- Networking
  - Tree like structure
  - 4 machines to 1 group switch
  - 4 group switchs to 1 rack switch
  - 2 rack switches connected to 1 Top Switch



How to classify

- Metrics
  - Shuffle_bytes
  - CPU_time
- (Shuffle_bytes/CPU_time)
  - Take the average of the map tasks
  - If value is over 1, then job is I/O Bound
  - Else CPU Bound

Results from Physical Machine runs



	Word Count	Pi	Pentomino	TeraGen	TeraSort	Grep	MRBench	DFCIOTest Read	DFCIOTest Write
HDFS Bytes Written	133.4174082	0.109941888	0.019261066	9328.415925	0	0.036108882	0.008636364	0.042364532	0.038053097

Results on the Virtual Clusters

Job	I/O Bound System (S)	CPU Bound System (S)
Word Count	257.2338	235.2299
PI	473.3364	419.88242
Pentomino	408.1599	355.0055
TeraSort	603.9358	183.1389
TeraGen	89.2324	116.62483
Grep	217.8305	188.0857
MRBench	21.0116	18.6668
DFSCIOTest read	24.5882	19.5072
DFSCIOTest write	25.2971	20.2712

Conclusion

- Presented a method for selecting virtual machines
- Showed the intuition behind the selection process
- Tested the method on two test beds at Temple
- Future works
  - Finding a good constant to multiply by to for a cluster
  - Including more types of virtual machines
  - Including more metrics for prediction

Questions?

- Contact
  - Adam.blaisse@temple.edu
  - Astro.temple.edu/~tuc47904