HPC in Cloud

Presenter: Naresh K. Sehgal
Contributors: Billy Cox, John M. Acken, Sohum Sohoni

Agenda

- What is HPC?
- Problem Statement(s)
- Cloud Workload Characterization
- Translation from High Level
- Issues and dealing with them
- Next Steps and Summary

What is HPC?

Means Different Things to Different Customers

Lockheed-Martin F-22A Raptor, Mach 2
1 passenger up to 5,000 miles
Costs US$150 million

High Speed

RAF Typhoon, Mach 2
1 passenger up to 3,000 miles
cost $130 million

High speed at lesser cost

Boeing 737, Mach 0.74
200 passengers up to 3,000 nautical miles
Cost US$84.4 million

More load at less cost

Airbus A380, Mach 0.86 top speed
853 passengers up to 9,600 miles
Cost US$389.9 million

Delivers more load over larger distance

HPC in Cloud Problem Statement(s)

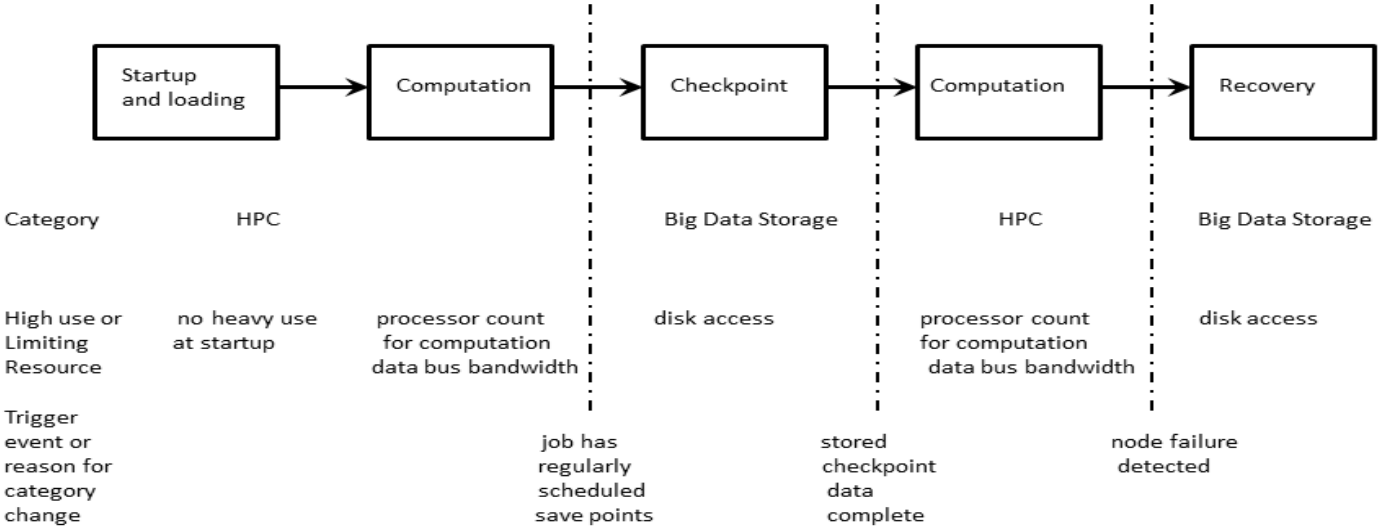
- No agreed upon definitions of Cloud workload categories
- Difficulties in matching Customers' requirements with available resources
- Problems in meeting SLA (Service-level agreements)
- Uncertain QoS (Quality of Service) for Cloudy Apps
- Resource usage planning for CSPs (Cloud Service Providers)

Computing Resource for WL categories

Workload Category	User View or Example Providers	Limiting Resources	Level of cloud relevance: "How cloud heavy is this category?"
Big Streaming Data	Netflix	Network Bandwidth	Heavy
Big Database Creation and Calculation	Google, US census	Persistent Storage, Computational Capability, Caching	Heavy
Big Database Search and Access	US census, Google, online shopping, online reservations	Persistent Storage, Network, Caching	Heavy
Big Data Storage	Rackspace, Softlayer Livedrive, Zip cloud Sugarsync, MyPC	Persistent Storage, Caching	Heavy
In Memory Database	Redis, SAP	Main Memory Size, Caching	Medium
Many Tiny Tasks (Ants)	Simple games, word or phrase translators, dictionary	Network, Many processors	Heavy
High Performance Computing (HPC)	Computer Aided Engineering, molecular modeling, genome analysis, and numerical modeling	Processor assignment and computational capability	Heavy
Highly Interactive Single Person	Terminal access, server administration	Network (latency)	Some
Highly Interactive Multi-Person Jobs	Collaborative online environment, e.g. Google Docs	Network (latency)	Medium
Single Computer Intensive Jobs	EDA tools (logic simulation, circuit simulation, board layout)	Computational capability	None
Private Local Tasks	Offline tasks	Persistent Storage	None
Slow Communication	E-mail, blog	Network	Some
Real-Time Local Tasks	Any Home Security System	Network	None
Location aware computing	Travel guidance	Local input hardware ports	added for security
Real-Time Geographically Dispersed	Remote machinery or vehicle control	Network	Light now, but future
Access Control	PayPal	Network	Some, light
Voice or video over IP	Skype, SIP	Network	Light if any

EDA Cloud Workload Characterization

- Some used a set of fixed traces[1]
  - Profiles of WLs based on machine utilization and wait times
- Others have evaluated HPC performance in Cloud[2]
  - Tradeoffs in migrating HPC workloads to public clouds
  - Evaluated the performance of a suite of benchmarks
  - Spectrum of applications for a typical HPC center
- Cloud EDA WL category may change with phases/steps of a HPC job



[1] Q. Zhang, J. L. Hellerstein, and R. Boutaba, "Characterizing task usage shapes in Google's compute clusters," *Proc. of Large-Scale Distributed Systems and Middleware (LADIS 2011),* 2011.

[2] K. R. Jackson, L. Ramakrishnan, K. Muriki, S. Canon, S. Cholia, J. Shalf, H. J. Wasserman, and N. J. Wright, "Performance analysis of high performance computing applications on the amazon web services cloud," in *Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on*, 2010, pp. 159-168.

Provider Options for VMs

Linux* cloud instances

AWS[1]

Type
Standard Small
Standard Medium
Standard Large
Standard Extra Large
2nd Gen Standard Extra Large
2nd Gen Standard Double Extra Large
Micro
High Memory Extra Large
High Memory Double Extra Large
High Memory Quadruple Extra Large
High CPU Medium
High CPU Extra Large
Cluster Quadruple Extra Large
Cluster Eight Extra Large
High Memory Cluster Eight Extra Large
High I/O Quadruple Extra Large
High Storage Eight Extra Large

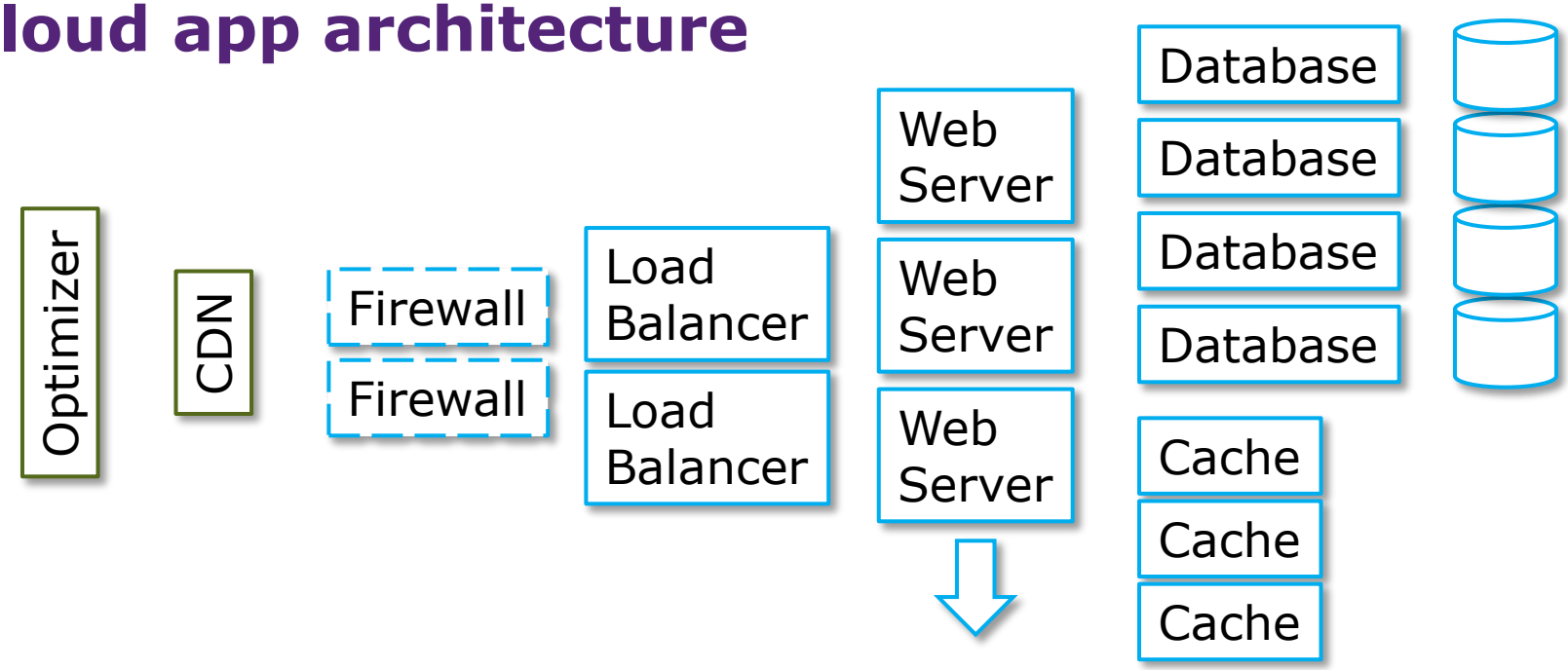
Rackspace[2]

RAM	vCPU	Disk	Public Network	Internal Network
512MB	1	20GB	20Mbps	40Mbps
1GB	1	40GB	30Mbps	60Mbps
2GB	2	80GB	60Mbps	120Mbps
4GB	2	160GB	100Mbps	200Mbps
8GB	4	320GB	150Mbps	300Mbps
15GB	6	620GB	200Mbps	400Mbps
30GB	8	1.2TB	300Mbps	600Mbps

1) From: http://aws.amazon.com/ec2/pricing/
2) From: http://www.rackspace.com/cloud/servers/pricing/

Anatomy of a 'Typical' Cloud App

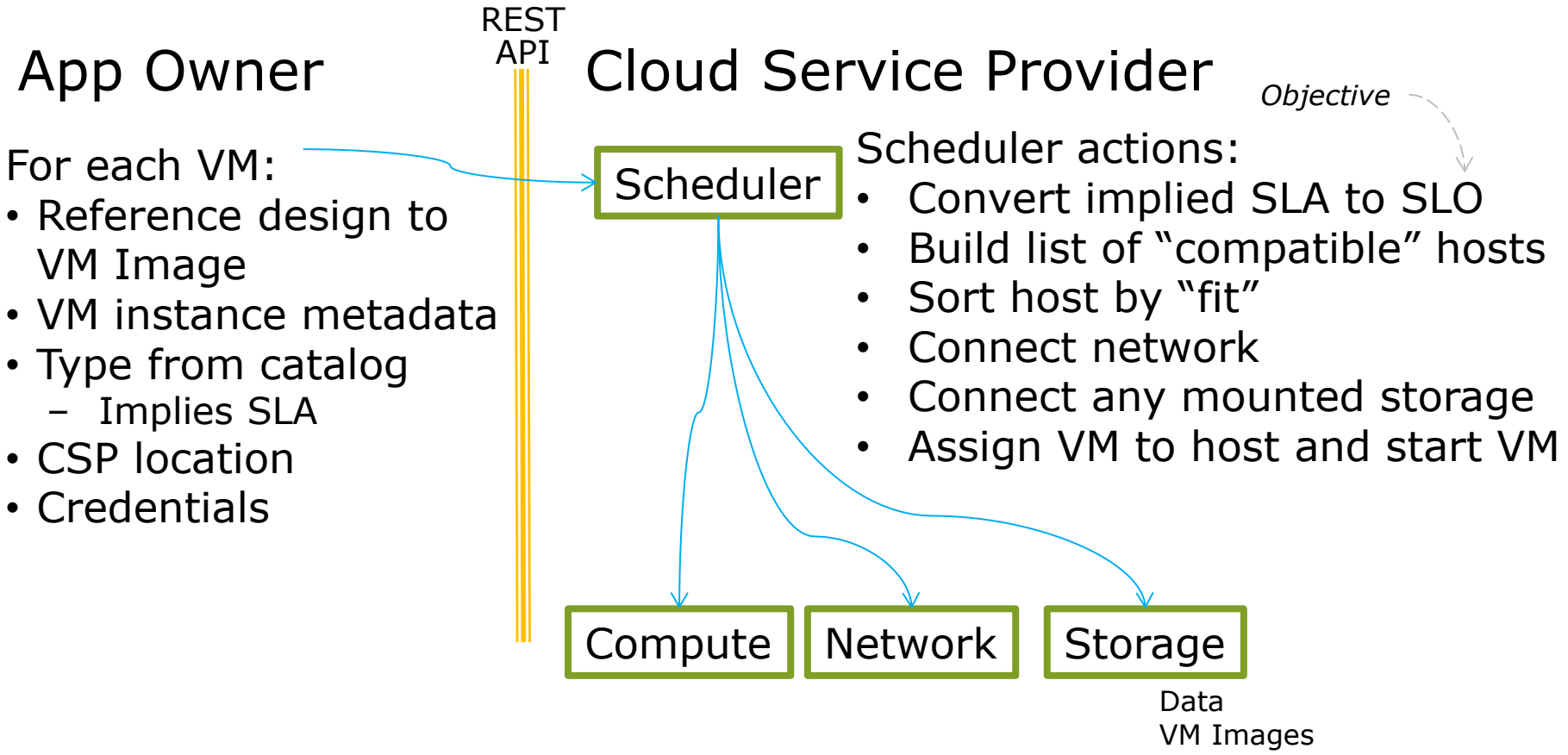
Cloud app architecture



- +Cache = optimize accesses to the (slow) database
- Scale out at all levels for availability and capacity
- Web server scales based on demand
- Database and cache size is 'fixed' at design time into shards

Next: Apply architecture to a provider

Cloud Service Provider: Actions to Host VM



"compatible" = able to run the image and has the capacity
"fit" = metric on prioritizing hosts (spare capacity, proximity, etc.)

Next: Convert the SLA into an SLO

SLA to SLO Translations Example

Service Level Objective

Example

SLA =

- 1000 MIPS of CPU instructions, 50Mb read/write/sec
- 1 GBPS of Network speed
- <3 ms response time, 90% of requests
- 3TB backend data, growing to 20TB
- 99.9999% up time (~1 hr/year)

Service Provider SLO Table

Name	Host			DC I/O
	CPU (G/s)	Mem	I/O	
Standard	1.000	1G	60M	30M
Medium	2.000	2G	100M	50M
Large	4.000	8G	200M	100M

SLA Request: **Standard**

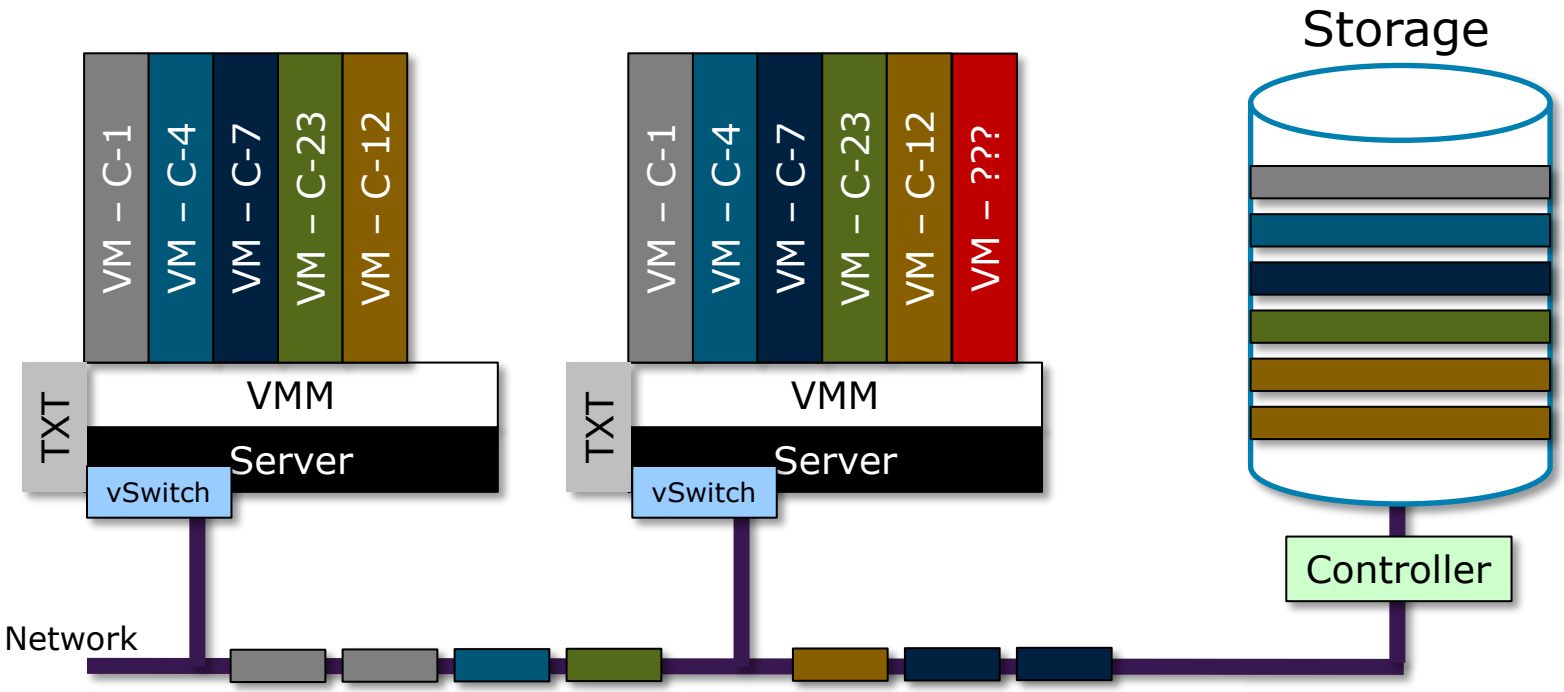
Example only

SLO result: **1.000/1G/60M/30M**

Translated values are used in VM placement to ensure SLA is met.

Issues: Multi-tenant

Multi-tenant infrastructures allow sharing of common resources



- Network traffic is multiplexed by the vSwitch
- Virtualization provides the compute isolation
- Intentional "bad guys" may be in the vm next to you
- Unintentional "noisy neighbor" may slow you down

Shared infrastructure exposes new challenges

Issues: Performance Variations

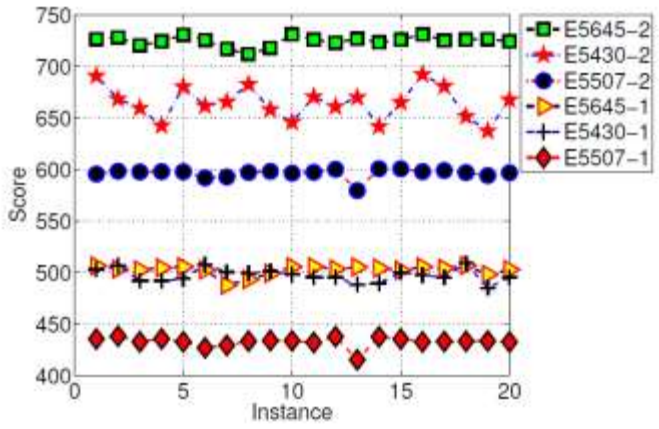


Figure 1: UnixBench score, one and two processes

Service	Server	Data Center	Test Date	Test Score	Test Date	Test Score
Amazon EC2	ec2-us-west.linux.c1.medium	CA, US	04/19/2010	3.42	05/14/2010	3.44
Amazon EC2	ec2-us-west.linux.c1.xlarge	CA, US	04/11/2010	6.48	03/19/2011	6.12
Amazon EC2	ec2-us-west.linux.m1.large	CA, US	04/19/2010	3.9	03/21/2011	3.37
Amazon EC2	ec2-us-west.linux.m1.small	CA, US	04/19/2010	1.17		
Amazon EC2	ec2-us-west.linux.m1.xlarge	CA, US	04/19/2010	4.91	03/23/2011	4.18
Amazon EC2	ec2-us-west.linux.m2.2xlarge	CA, US	04/19/2010	13.55	03/20/2011	12.99
Amazon EC2	ec2-us-west.linux.m2.4xlarge	CA, US	04/19/2010	26.32	03/21/2011	22.81

- Small variation within same processor type
- Larger variation based on processor for same instance request.

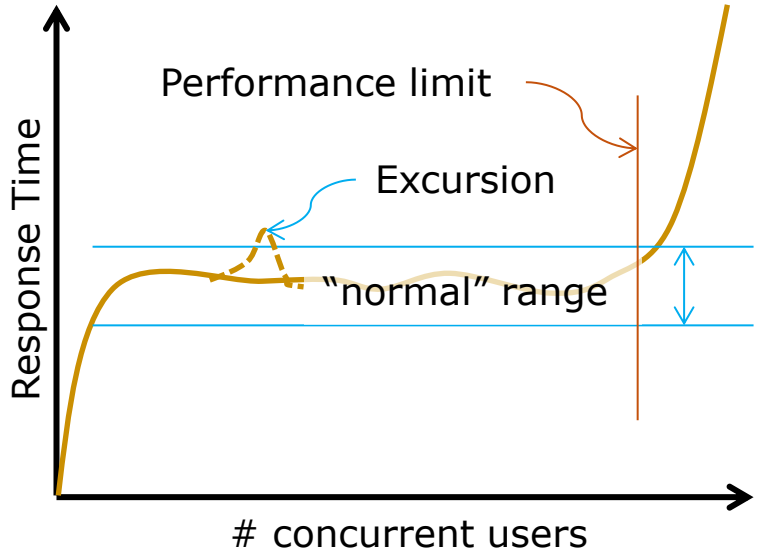
From: Exploiting Hardware Heterogeneity within the Same Instance Type of Amazon EC2

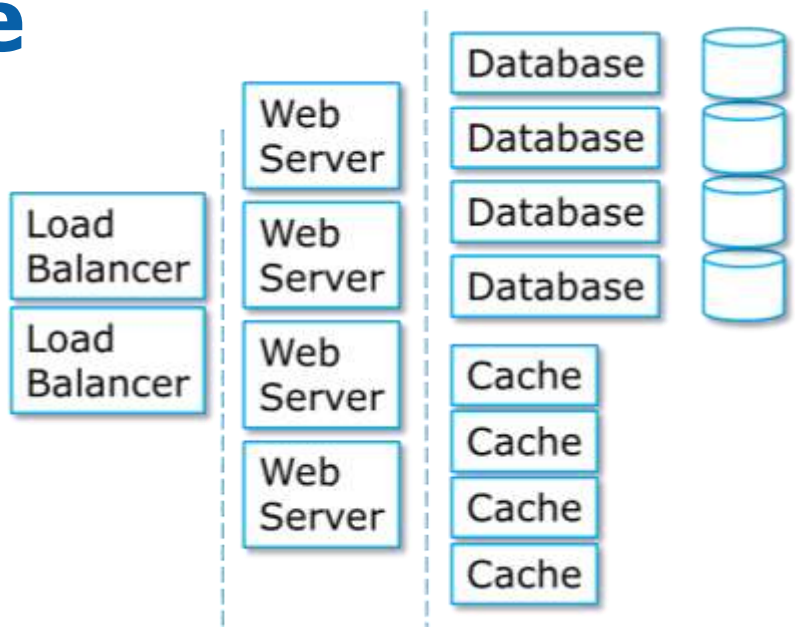
- 1% to 17% variation in performance (CPU) at same location for same instance type

From: Cloud Harmony, Mar 14 "Pi day", Aggregate CPU performance metric

Cost effectiveness of instances can vary significantly.

Issues: Performance





Assume: service has been operating within parameters

Question:

- What caused the excursion?

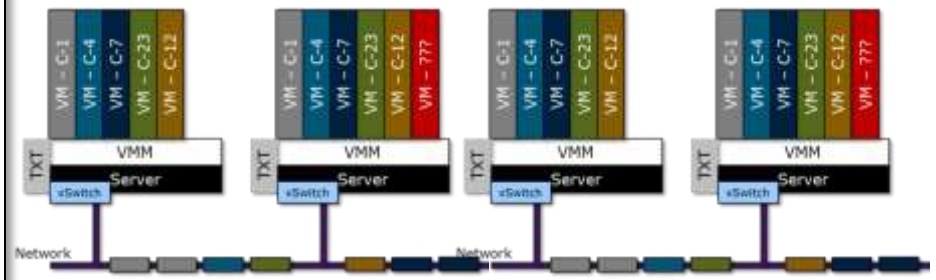
Consider:

- Shared exclusion locks
- JVM garbage collection
- Database contention

From the infrastructure (invisible)

- Overloaded compute
- Noisy neighbor VM
- Overloaded network

… is hosted on…



Dynamic Monitoring and Resource Allocation

- Proposed HPC Cloud Solution:
  - A cluster augmented with low level measurements
    - local measurements to determine resource contention
  - A robust aggregation framework
    - to determine the cluster level efficacy

- Research Questions:
  - What is the extent of resource contention in various shared cluster setting (e.g. HPC applications running in the cloud)?
  - Is there contention to warrant dynamic real-time monitoring?
  - How often should these measurements be taken?
    - sampling overhead, network bandwidth for data aggregation etc.
  - How these measurements drive smarter resource allocation?
    - how do low level metrics measurements relate to and indicate which resource is the limiting one?

Next Steps and Summary

- Use cloud design principles for HPC apps
  - Scale-out, assume failures, wide variations in loading
  - Local Excursions in response times
- Define SLAs and translate to SLOs
  - Measure, fail, analyze, adjust, repeat
- Find best platform/cost for your HPC Apps
  - Test for scalability, for failures
- Expect multitenant environments
  - Isolation, trust
- Optimize for performance and availability of EDA apps