

Cloud Based Testing & Capacity Planning (CloudPerf)

Joan A. Smith

Emory University Libraries

joan.smith@emory.edu

Frank Owen

Owenworks Inc.

frank@owenworks.biz

Full presentation materials and CloudPerf screencast are available at
<http://www.joanasmith.com/content/cloudperf>

Discussion Map

1. Cloud Strategies
2. Performance Testing
3. Challenges & Lessons Learned
4. Q & A

PART 1. CLOUD STRATEGIES

Motivation

- Once upon a time...
 - Grad-student implementation
 - Significant university digital scholarship website
 - Major PR event with NY Times, TV News, etc.
- Fairy tale to nightmare...
 - Fastest hardware on site was still inadequate
 - Max concurrent users < 15
 - Expected concurrent users: > 15,000
- Wake up, Neo...

Background

- Goals:
 - Deploy as economically as possible (don't overbuy hardware)
 - Tune application for performance before deployment (happy users)
- Traditional Approach Limitations:
 - Availability and cost of hardware (affordability as decision point)
 - Time in days or weeks to configure, test, reconfigure (rinse and repeat)
 - Blind investment in hardware solutions (fingers crossed)
 - High up-front costs and on-going costs (for test, admin, etc.)
- Cloud advantages
 - Quick to deploy, evaluate, adjust and redeploy
 - Rented cost of hardware is cheap
 - Easily tweak specific aspects of hardware and OS configurations
- Cloud disadvantages
 - High initial costs to create the standardized test environment
 - Not guaranteed hardware performance (can vary within type)
 - Only approximating target deployment environment

Definition: Cloud Service

- This is what we mean by cloud
 - Sold on demand (not a subscription)
 - Elastic (as much or as little at one time)
 - Hardware is managed by the provider
 - Rapid provisioning/deployment (near real-time for some)
- This is not a cloud
 - Monthly fee + access costs
 - Mainframe data center
 - Long deployment cycle

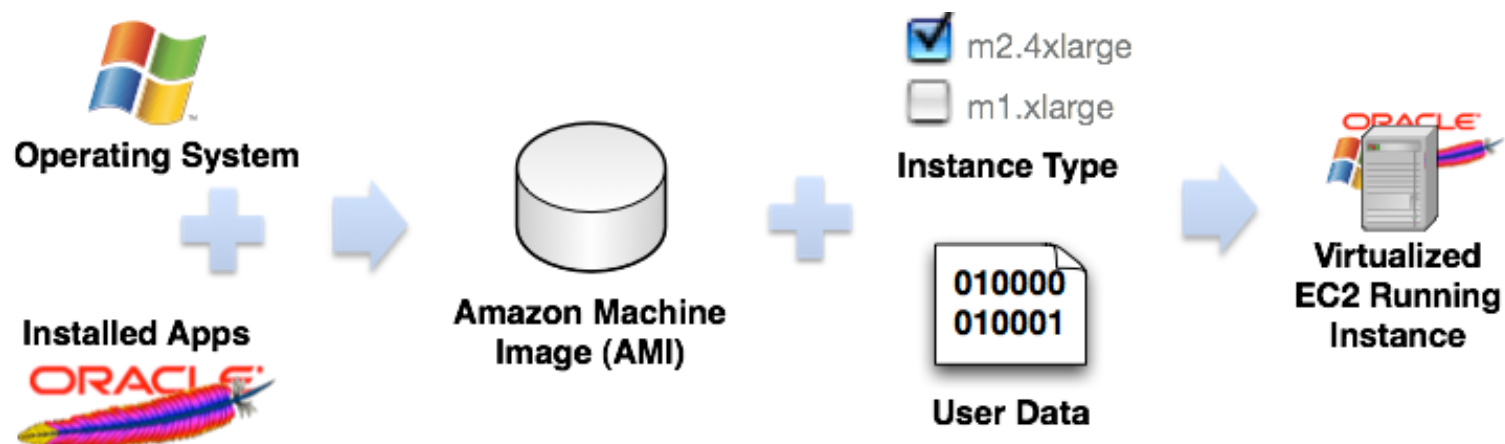
Types of clouds

- Software as a Service (SaaS)
 - Users log in to turnkey web application
 - Examples:
 - salesforce.com – Customer Relationship Management / Sales
 - freshbooks.com – Small business invoicing
 - soasta.com – Cloud based load testing
 - Google Docs – Document management
- Platform as a Service (PaaS)
 - Specific set of APIs for that one platform
 - Build custom application on top of APIs
 - Deploy apps to that platform
 - Example: Google App engine
 - Custom Java, Python APIs: Write your own everything
 - Auto-scales in cases of increased user demand
- Infrastructure as a Service (IaaS)
 - Configurable OS, Hardware
 - Example: Amazon
 - **Amazon EC2 - Compute cloud** ← CloudPerf uses EC2
 - Amazon S3 - Storage cloud

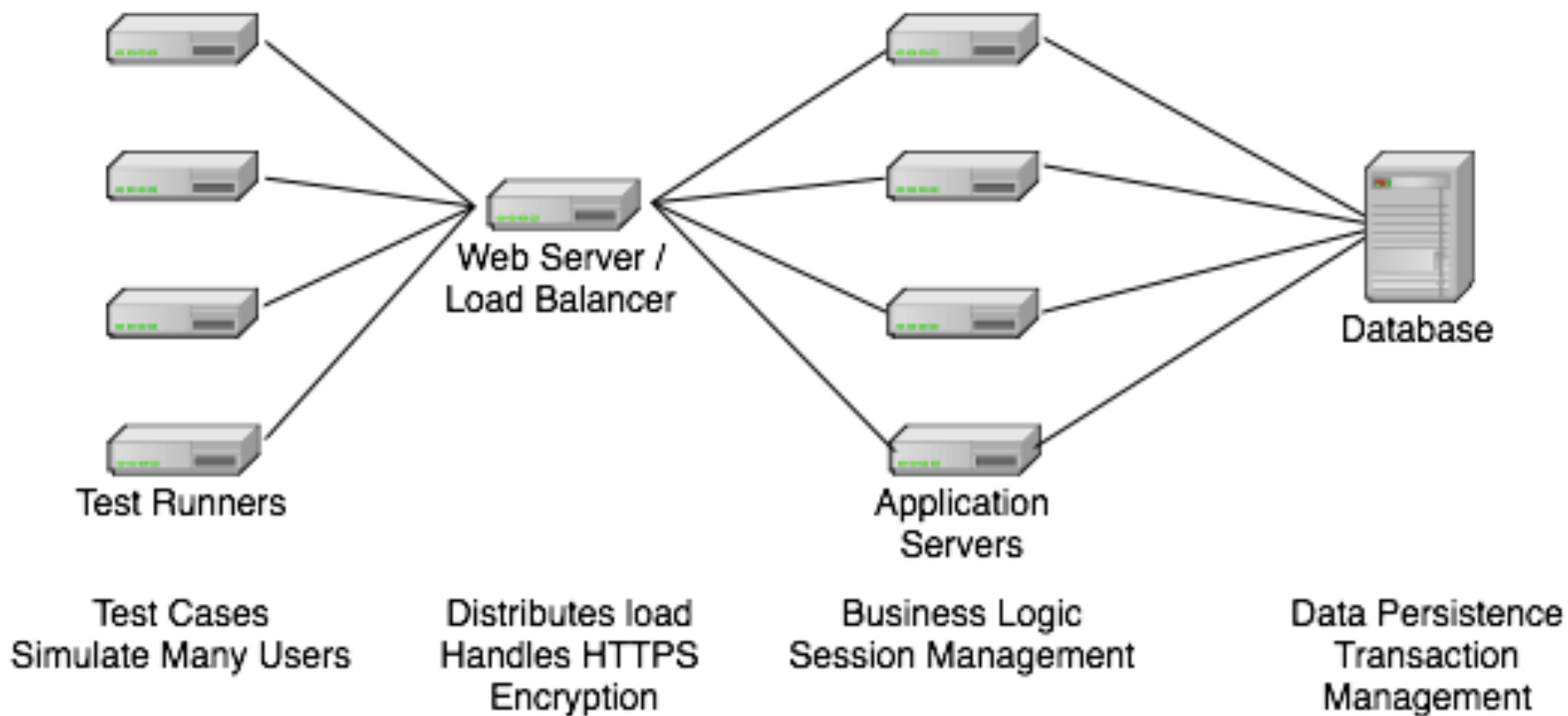
EC2: What it is & How it works

- EC2 = Elastic Compute Cloud
 - A virtual infrastructure
- Instance Based
 - Amazon Machine Image: Hard Drive image of an OS stored in Amazon Cloud (S3 or EBS)
 - Instance Type: Selectable Hardware
- Costing
 - Cost begins upon powering up
 - Instance Price per Hour: \$0.02-\$2.48
 - Additional \$\$ for network, storage usage
- Management
 - Web interface for administration
 - Programming APIs


Launching a Virtual Instance



Typical large app cluster



EC2 Manual Cluster Building

- One time AMI Setup:
 - Launch public EC2 image through Amazon Console
 - Install infrastructure software to the image
 - Oracle Database, Tomcat App Server, Apache HTTPD etc
 - Customize configuration files
 - Save your AMI as private AMI
 - Recurring Setup:
 - Launch n (10) AMIs, record network info
 - Reconfigure 4 app servers to talk to db
 - Reconfigure 1 load balancer to talk to apps
 - Reconfigure 4 test runners to talk to each other, load balancer
 - Fetch, install and configure latest software version (e.g., slave voyages)
- 
- Automate!

CloudPerf: Streamlined workflow

- New web-based management system
- Aims to minimize number of Machine Images we need to maintain
 - One windows image, One Linux image
 - All software installed on all images
 - Utilizes 16 KB User Data to instruct image what service to start
- Auto configures entire cluster for target web application
 - Peer discovery
 - Synchronization
 - Downloads and installs target software from repository

CloudPerf Cluster Configuration

Mozilla Firefox

Owenworks CloudPerf

[Instructions](#) [AMI Types Reference](#) [Update S3 Cache](#)

Software Releases

- emory-voyages-20101115_r39547.zip
- emory-voyages-20101112_r39528.zip
- emory-voyages-20101022_r39120.zip
- emory-voyages-20100922_r38602.zip
- emory-voyages-20101129_r39796.zip
- emory-voyages-20101124_r39769.zip
- emory-voyages-20101124_r39767.zip
- emory-voyages-20101118_r39662.zip

Database Exports

- none
- voyages-v1.exp.dmp.gz
- voyages-v2.exp.dmp.gz

AWS DataCenter

- ☐ us-east-1a
- ☐ us-east-1b
- ☐ us-east-1c
- ☒ us-east-1d

☒ **Database**

- MySQL Linux32
- Oracle Linux32
- Oracle 10g Linux64
- Oracle 11g Linux64
- Oracle Win64

☒ **App Server:**

- Tomcat 5 Linux32
- Tomcat 5 Linux64

☐ **Load Balancer:**

☐ **Test Runner:**

| | |
|---|---------------------------------------|
| <input checked="" type="radio"/> m1.large | <input type="radio"/> 1 GB |
| <input type="radio"/> m1.xlarge | <input type="radio"/> 2 GB |
| <input type="radio"/> c1.xlarge | <input type="radio"/> 4 GB |
| <input type="radio"/> m2.2xlarge | <input checked="" type="radio"/> 6 GB |
| <input type="radio"/> m2.4xlarge | <input type="radio"/> 8 GB |

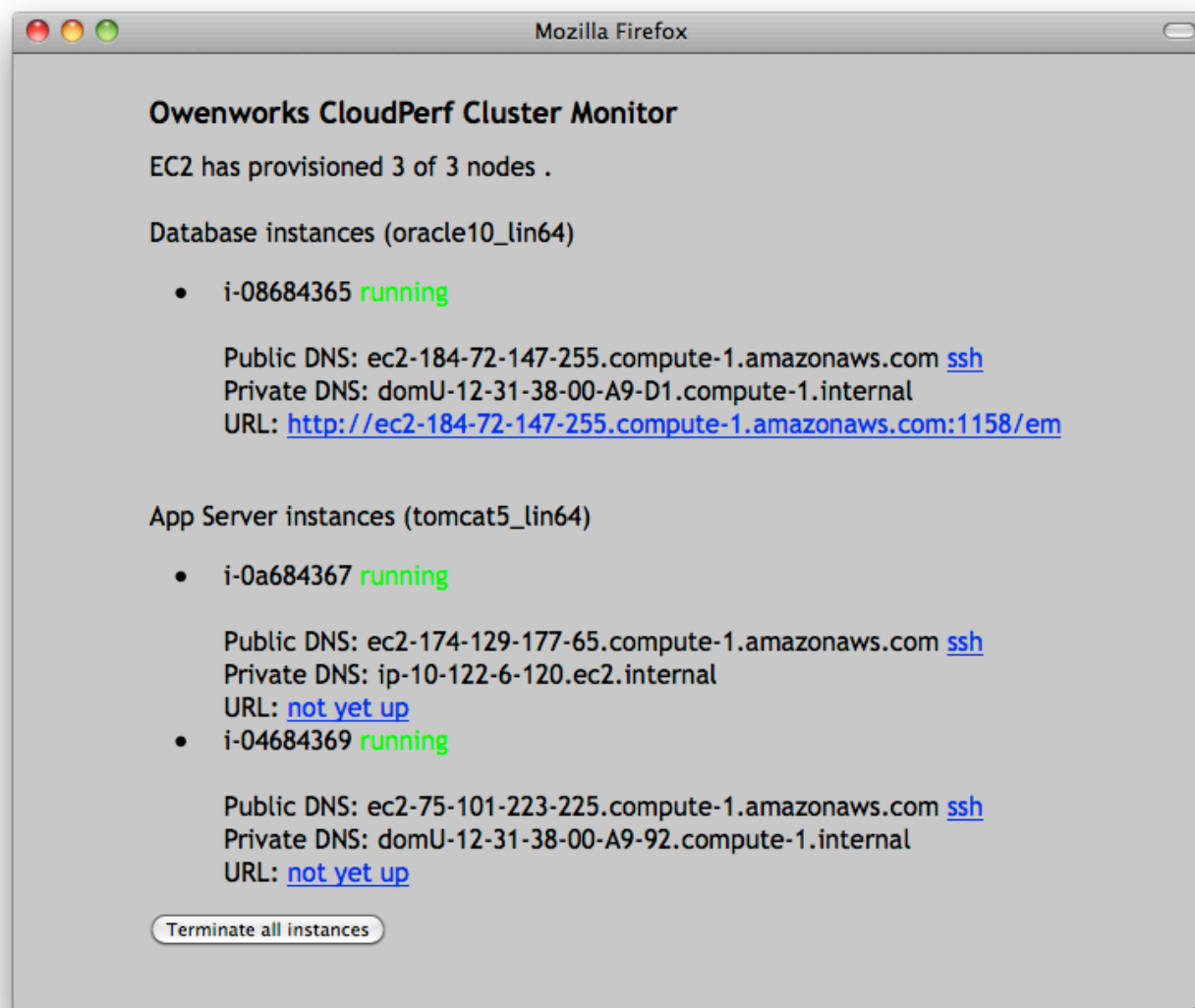
| | |
|--|---|
| <input type="radio"/> m1.large | <input checked="" type="radio"/> 512 MB |
| <input type="radio"/> m1.xlarge | <input type="radio"/> 2 GB |
| <input checked="" type="radio"/> c1.xlarge | <input type="radio"/> 4 GB |
| <input type="radio"/> m2.2xlarge | <input type="radio"/> 8 GB |
| <input type="radio"/> m2.4xlarge | <input type="radio"/> 16 GB |

☒ **Use Tomcat Native Library**

DB Conn: active idle wait

JVM Args:

CloudPerf Monitoring



Technical Challenges

- Software repository to fetch latest version
 - Your software changes: don't rebuild AMI every time
 - Amazon S3 works well as image storage
- Peer discovery and network configuration
 - Changes every launch: cannot be part of AMI
 - Private IPs protect your app during testing
- Synchronization
 - DB, then App Servers, then Load Balancer
 - Requires polling solution
- Instance types are approximate
- Infrastructure failures (network, heavy load etc)

Resolving startup dependencies

- We modeled startup after the Unix rc model
 - Runlevel 0: No dependencies
 - Runlevel 1: User-data is posted
 - Runlevel 2: All peer network addresses are known
- Timing is everything
 - Peer discovery is a synchronization problem

How does Amazon charge?

- Instances
 - \$0.02 to \$2.48 per hour, based on processing power and memory
- Network Data Transfer
 - Inbound: \$0.10 per GB
 - Outbound:
 - First GB Free
 - \$0.08 to \$0.15 per GB thereafter
- CloudWatch
 - \$0.015 per instance-hour
- EBS Storage
 - \$0.10 per GB/month
 - \$0.10 per million I/O requests
- S3 Storage
 - \$0.14 per GB for first 1 TB, less thereafter
 - Data In: \$0.10 per GB
 - Very small per request charges
- Prices vary by Data Center
 - East Coast, Europe, West Coast, etc.

Amazon Instance Types

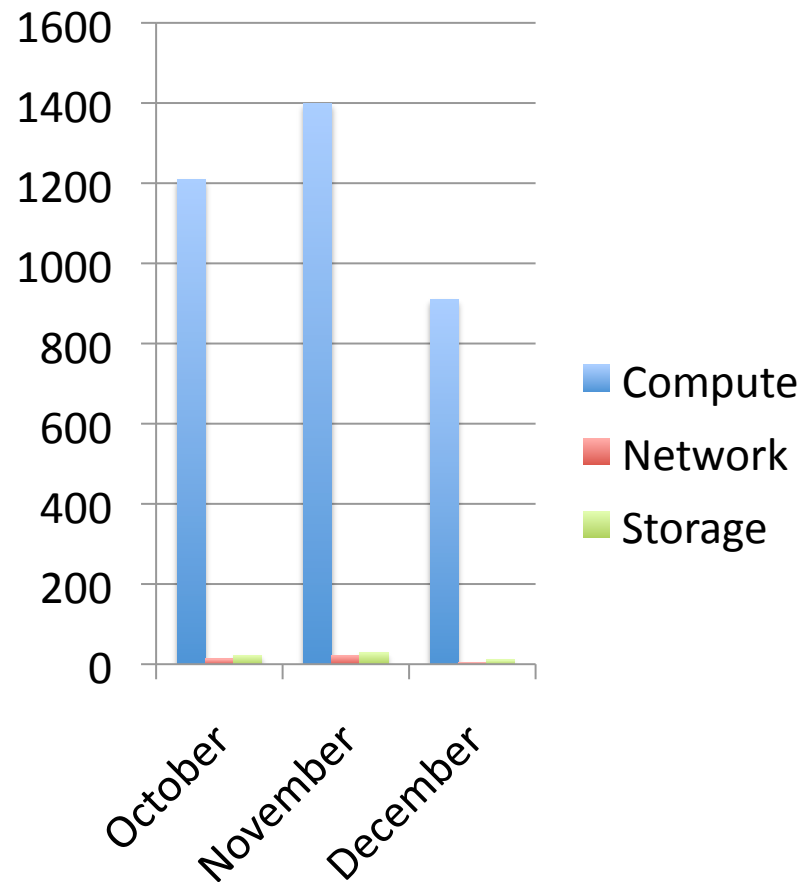
| Type | Memory | CPU CU | Disk GB | Linux | Windows | Note |
|-------------|--------|--------|---------|---------|---------|-----------------|
| t1.micro | 613 MB | 2 | | \$0.02 | \$0.03 | 750 hrs Free |
| m1.small | 1.7 GB | 1 | 160 | \$0.085 | \$0.12 | 32 bit only |
| m1.large | 7.5 GB | 4 | 850 | \$0.34 | \$0.48 | |
| m1.xlarge | 15 GB | 8 | 1690 | \$0.68 | \$0.96 | |
| m2.2xlarge | 34 GB | 13 | 850 | \$1.00 | \$1.24 | |
| m2.4xlarge | 68 GB | 26 | 1690 | \$2.00 | \$2.48 | |
| c1.medium | 1.7 GB | 5 | 350 | \$0.17 | \$0.29 | 32 bit only |
| c1.xlarge | 7 GB | 20 | 1690 | \$0.68 | \$1.16 | |
| cc1.4xlarge | 23 GB | 33.5 | 1690 | \$1.60 | N/A | Cluster Compute |
| cg1.4xlarge | 22 GB | 33.5 | 1690 | \$2.10 | N/A | Dual GPUs |

1 CU = One 2007 1.0-1.2 GHz Opteron or Xeon processor

1 CU = One 2006 1.7 GHz Xeon processor

Cost examples

- 1 full time performance tester at Owenworks
 - Mostly large 11-node clusters
 - Total bill for AWS: ~ \$1500/month
- One 2 node Linux cluster, 4 hours/day
 - m1.xlarge 15 GB RAM, 8 CU -> \$120 month



QA Costs for Cloud vs Traditional

- In a 'real' QA lab, it would take two knowledgeable individuals an entire work-week to load and configure 11 servers for a single test run
- With EC2, an entire cluster can be launched by an individual and be ready for testing in about 5 minutes.
- When the test is complete and the data gathered, the cluster can be terminated; if additional runs are needed, another cluster can be ready in minutes
- Multiple clusters can easily be configured to allow direct comparisons between software releases (valuable for troubleshooting)
- Using the cloud, the costs of purchasing and maintaining traditional QA lab equipment can be substantially reduced if not eliminated

PART 2: PERFORMANCE TESTING

Key Performance Testing Tasks

- Identify bounds
 - CPU? Disk I/O? RAM? Net Bandwidth?
 - Clarifies what to buy for deployment
- Identify hardware requirements
 - capacity planning
 - Which/how many CPUs
 - Optimum RAM, Disk, etc.
- Identify software bottlenecks
 - Many software defects only show up under load
 - Database concurrency, threading, cache sizes
- Identify failure modes
 - When the application is overloaded, how does it fail
 - Target responses for failure modes

Types of performance tests:

- Load testing
 - using expected customer workload
- Stress testing
 - exceeding expected load; what happens?
- Endurance testing
 - basically a longer-duration load test
- Capacity planning
 - Hardware needed to meet expected load

Test Planning Strategy

- Create Application Usage Models
 - Determine usage style (bursty, continuous)
 - Estimate expected load, users, transactions, reports, etc.
- Create Hardware Deployment Models
 - Capacity Tests
 - Performance Tests
- Write tests
 - Jmeter, Rational Performance Test, Silk, etc.
 - Validators are very important.
 - Most applications can generate error pages very efficiently
 - “404” is a successful page generated but not a successful outcome

Tools for EC2 Performance Testing

- Jmeter
 - Open source distributed performance test tool
- Amazon AWS management console
 - CloudWatch, if enabled, gives network stats, CPU usage, Memory usage
- ElasticFox Firefox plugin
 - Functionally equivalent to the EC2 portion of AWS management console
- S3Fox Firefox plugin
 - Functionally equivalent to the S3 portion of AWS management console
- Traditional OS-Specific Tools:
 - Unix: top, iostat, ganglia
 - Windows: Task Manager

JMeter Reporting

| Summary Report | | | | | | | | | |
|--|-----------|-----------|-------------------|-------|---------------------------------|------------------------------------|------------|----------|------------|
| Name: Summary. 76K DB m2.4xlg DB 32G; Four c1.xlg App 6G, m1.xlg Load, Ten m1.xlg JM 8GB 250 EMP/250 SUP | | | | | | | | | |
| Comments: default JVM ARGS | | | | | | | | | |
| Write results to file / Read from file | | | | | | | | | |
| Filename | | Browse... | Log/Display Only: | | <input type="checkbox"/> Errors | <input type="checkbox"/> Successes | Configure | | |
| Label | # Samples | Average | Min | Max | Std. Dev. | Error % | Throughput | KB/sec | Avg. Bytes |
| TiTo Employees:WebTA Start | 4000 | 1745 | 59 | 10451 | 1258.98 | 0.00% | 9.5/sec | 3184.84 | 345089.0 |
| TiTo Employees:Employee Login | 4000 | 775 | 52 | 7693 | 824.66 | 0.00% | 9.5/sec | 91.72 | 9922.0 |
| TiTo Employees:Click Timesheets | 4000 | 619 | 49 | 6690 | 472.41 | 0.02% | 9.5/sec | 411.41 | 44471.3 |
| TiTo Employees:Change Pay Period | 3999 | 476 | 53 | 6377 | 351.03 | 0.00% | 9.5/sec | 422.15 | 45532.4 |
| TiTo Employees:Select Day of Week | 39990 | 643 | 51 | 10633 | 622.31 | 0.00% | 93.2/sec | 4533.99 | 49796.8 |
| TiTo Employees:Enter Time, Click Save | 39990 | 842 | 61 | 12883 | 784.11 | 0.00% | 93.3/sec | 5188.39 | 56939.2 |
| TiTo Employees:Enter Time and Click Validate | 3999 | 1918 | 95 | 14375 | 1821.15 | 0.00% | 9.6/sec | 351.91 | 37607.0 |
| TiTo Employees:Click Affirm | 3999 | 953 | 81 | 11618 | 893.43 | 0.00% | 9.7/sec | 562.76 | 59617.2 |
| TiTo Employees:Click Logout | 3999 | 181 | 16 | 6227 | 226.12 | 0.00% | 9.7/sec | 83.07 | 8782.0 |
| TOTAL | 107976 | 797 | 16 | 14375 | 855.29 | 0.00% | 248.8/sec | 14565.81 | 59942.1 |

Amazon Management Console

The screenshot shows the AWS Management Console interface. The top navigation bar includes links for AWS, Products, Developers, Community, Support, and Account. The left sidebar contains a navigation menu with categories like EC2 Dashboard, INSTANCES, IMAGES, ELASTIC BLOCK STORE, and NETWORKING & SECURITY. The main content area is titled 'My Instances' and shows a table of EC2 instances. Below the table, there is a section for 'EC2 Instances: i-24e8dc49, i-b2efdbdf, i-8cefdbe1' with tabs for Description, Monitoring, and Tags. The Monitoring tab is active, displaying five graphs: Avg CPU Utilization (Percent), Avg Disk Reads (Bytes), Avg Disk Writes (Bytes), Max Network In (Bytes), and Max Network Out (Bytes). The graphs show data for the last hour, with a time range selector set to 'Last Hour' and a 'Refresh' button.

| Name | Instance | AMI ID | Root Device | Type | Status | Security Groups | Key Pair Name |
|-------------------------------------|------------|--------------|-------------|----------|---------|---------------------|---------------|
| <input type="checkbox"/> | i-b6efdbdb | ami-447b8f2d | ebs | m1.large | running | zgroup, database | jim.gray.a |
| <input type="checkbox"/> | i-b0efdbdd | ami-447b8f2d | ebs | m1.large | running | zgroup, app server | jim.gray.a |
| <input checked="" type="checkbox"/> | i-b2efdbdf | ami-447b8f2d | ebs | m1.large | running | zgroup, load balanc | jim.gray.a |
| <input checked="" type="checkbox"/> | i-8cefdbe1 | ami-447b8f2d | ebs | m1.large | running | zgroup, jmeter mas | jim.gray.a |
| <input checked="" type="checkbox"/> | i-24e8dc49 | ami-447b8f2d | ebs | m1.large | running | zgroup, database | jim.gray.a |
| <input type="checkbox"/> | i-26e8dc4b | ami-447b8f2d | ebs | m1.large | running | zgroup, app server | jim.gray.a |
| <input type="checkbox"/> | i-22e8dc4f | ami-447b8f2d | ebs | m1.large | running | zgroup, load balanc | jim.gray.a |

EC2 Instances: i-24e8dc49, i-b2efdbdf, i-8cefdbe1

Time Range: Last Hour Refresh

Graphs are for 3 instances with detailed monitoring enabled. Times are displayed in UTC.

Avg CPU Utilization (Percent)

Avg Disk Reads (Bytes)

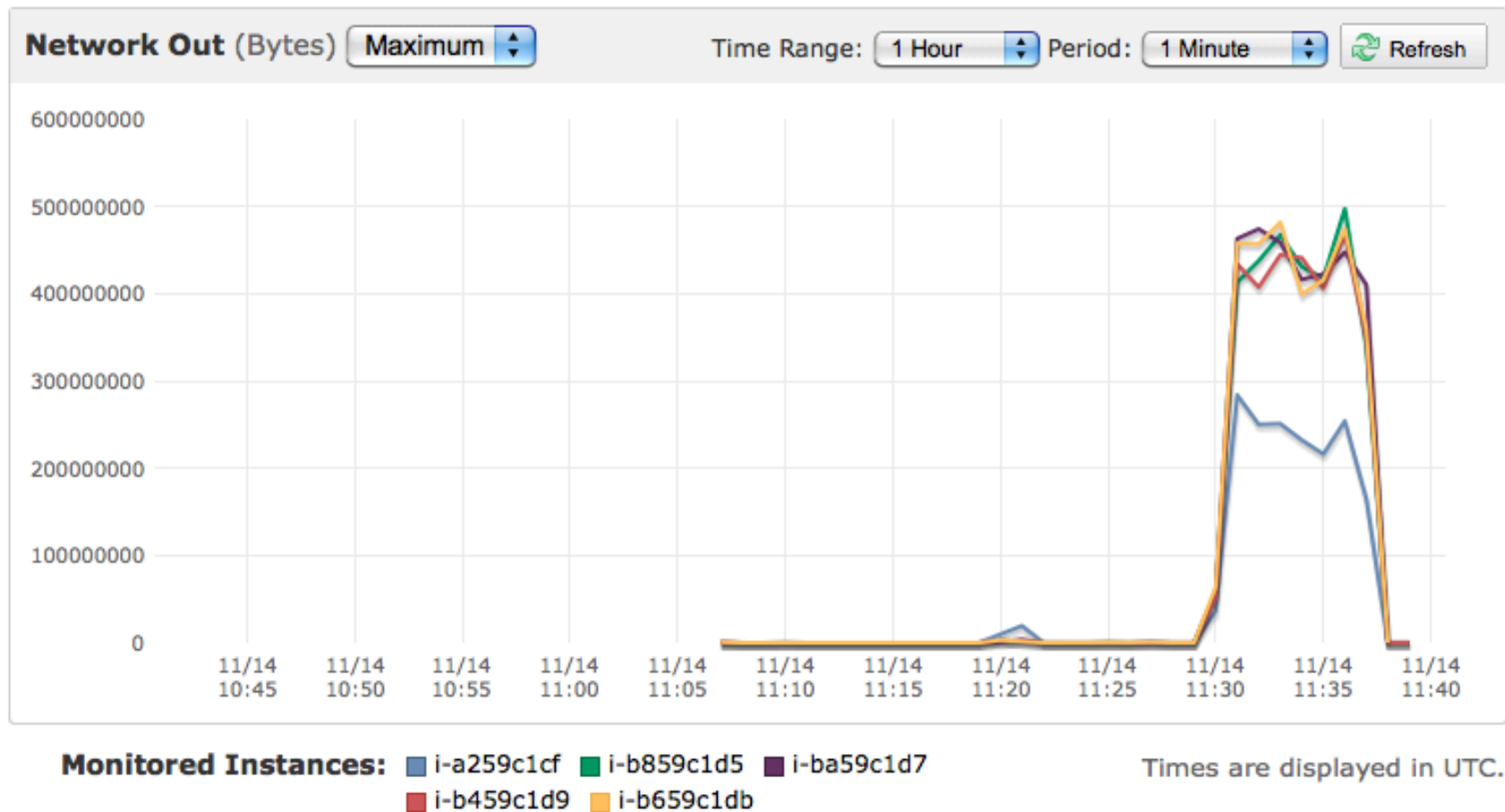
Avg Disk Writes (Bytes)

Max Network In (Bytes)

Max Network Out (Bytes)

© 2008 - 2010, Amazon Web Services LLC or its affiliates. All right reserved. | Feedback | Support | Privacy Policy | Terms of Use | An amazon.com company

Amazon CloudWatch: Graph Detail



Part 3

CHALLENGES & LESSONS LEARNED

Challenge: Analyzing Results

- Instance types are *approximate*
 - need multiple runs to even out results
- Mapping test results to real hardware is hard
 - How much CPU is a CU worth?
- EC2 network topology differs from deployment topology
- Disk I/O performance is poor on EC2
 - Few apps these days are disk bound
 - Disk bound apps not a good match for EC2 testing
- Lack of comparable historical data from similar sites

Challenge: Licensing Issues

- Not Mac compatible!!
 - Apple requires Server OS for virtualization
 - Apple license requires Mac Hardware
 - None of the cloud services use Mac Hardware
- Check the license
 - Some disallow running in virtualized environment
 - Some require separate license
 - Some provide an Amazon AMI, and charge for it by the hour (Windows OS)

Lessons Learned

- Keeping images in sync with scripts is key
 - Use scripts to modify images at runtime
- Minimize the number of images
 - Install all infrastructure software on every image
 - Pull target software versions from S3
 - Startup and configure based on user-data
- Results analysis requires skill & experience
- User models and historical usage data are important
- Separate software process problems from cloud problems
 - Automated testing can serve as a form of regression test

Final Thoughts: Cloud Value

- High up-front costs to script
 - Worthwhile if your apps share a common tech stack
 - Offset by not having to buy boat loads of hardware
- Value from:
 - Repeatability & rapid test turn-around
 - Hardware savings & reduced configuration time
 - Variety of hardware, allowing what-if scenarios
- Performance assurance
 - Right-sizing configuration
 - Right-sizing hardware

PART 4. Q & A

Full presentation materials and CloudPerf screencast are available at
<http://www.joanasmith.com/content/cloudperf>

THANK YOU!

ADDITIONAL DETAIL SLIDES

Cloud as Test Environment

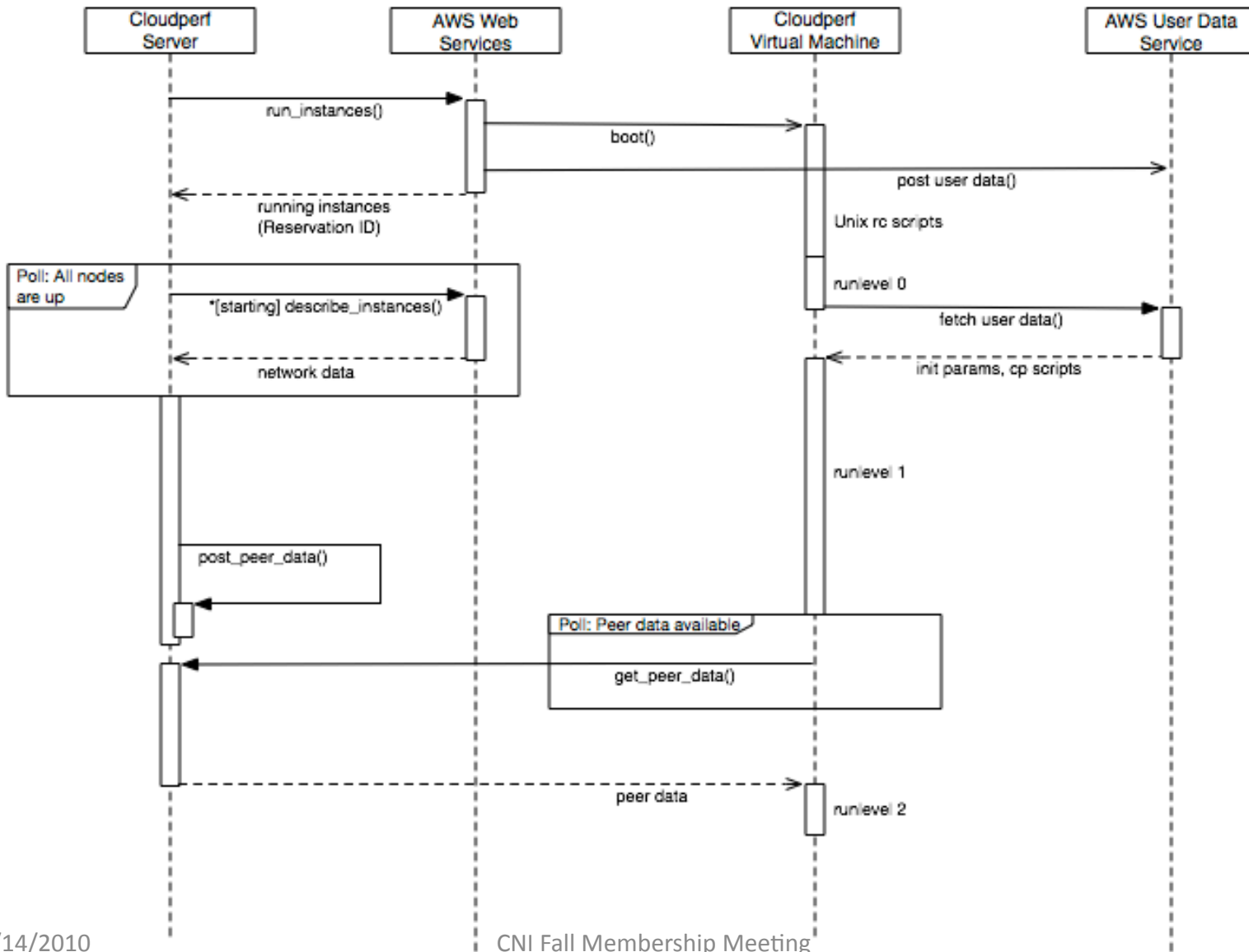
- Broad QA applicability
 - functional testing
 - regression testing
- Test plans vary by test type
 - but using the cloud provides the same sort of benefits
- Efficient, cost-effective, fast results

Cloudy?

- Term originates from network diagrams
 - “Cloud” didn’t matter
- Fuzzy concept
 - Overused by marketers
 - Calling it a cloud doesn’t make it a cloud
 - “Maximize buzzword synergy”

add pictures of (old) network clouds

CloudPerf Sequence Diagram



ElasticFox

The screenshot displays the ElasticFox application window. The top navigation bar includes tabs for 'Instances', 'Images', 'KeyPairs', 'Security Groups', 'Elastic IPs', 'Volumes and Snapshots' (which is selected), 'Bundle Tasks', 'Reserved Instances', 'Virtual Private Clouds', 'VPN Connections', and 'Availability Zones'. The 'Created Volumes' section shows a table with columns: VOL ID, Size (GB), SNAP ID, Availability Zone, Status, Local Creation Date, Instance ID, Device, Attachment Status, Attach Time, and Tag. Below this, the 'Saved Snapshots' section shows a table with columns: SNAP ID, VOL ID, Status, Local Start Time, Progress, Tag, Volume S..., Description, and Ow....

| VOL ID | Size (GB) | SNAP ID | Availability Zone | Status | Local Creation Date | Instance ID | Device | Attachment Status | Attach Time | Tag |
|--------------|-----------|---------------|-------------------|-----------|---------------------|-------------|-----------|-------------------|---------------------|-----|
| vol-3311da5a | 20 | snap-7214591b | us-east-1a | available | 2010-02-23 10:56:01 | | | | | |
| vol-36f2325f | 10 | | us-east-1a | available | 2010-01-22 19:27:14 | | | | | |
| vol-661ecb0e | 20 | snap-fb58ff91 | us-east-1b | in-use | 2010-12-11 20:20:00 | i-22e8dc4f | /dev/sda1 | attached | 2010-12-11 20:20... | |
| vol-6c1ecb04 | 20 | snap-fb58ff91 | us-east-1b | in-use | 2010-12-11 20:19:55 | i-26e8dc4b | /dev/sda1 | attached | 2010-12-11 20:19... | |
| vol-8533f8ec | 20 | snap-976f43fe | us-east-1a | available | 2010-02-23 13:47:08 | | | | | |
| vol-941facfc | 20 | snap-fb58ff91 | us-east-1b | in-use | 2010-12-11 20:19:54 | i-24e8dc49 | /dev/sda1 | attached | 2010-12-11 20:19... | |
| vol-961fcafe | 20 | snap-fb58ff91 | us-east-1b | in-use | 2010-12-11 20:19:54 | i-3ce8dc51 | /dev/sda1 | attached | 2010-12-11 20:19... | |
| vol-a01fcac8 | 20 | snap-fb58ff91 | us-east-1d | in-use | 2010-12-11 20:18:36 | i-8cefdbe1 | /dev/sda1 | attached | 2010-12-11 20:18... | |
| vol-a21fcaca | 20 | snap-fb58ff91 | us-east-1d | in-use | 2010-12-11 20:18:37 | i-b0efdbdd | /dev/sda1 | attached | 2010-12-11 20:18... | |
| vol-a41fracc | 20 | snap-fb58ff91 | us-east-1d | in-use | 2010-12-11 20:18:38 | i-h2efdhdf | /dev/sda1 | attached | 2010-12-11 20:18... | |

| SNAP ID | VOL ID | Status | Local Start Time | Progress | Tag | Volume S... | Description | Ow... |
|---------------|--------------|-----------|---------------------|----------|-----|-------------|---|--------|
| snap-fdefe097 | vol-291cd141 | completed | 2010-12-07 16:26:15 | 100% | | 1 | DemoGrid Chef partition 0.1 | 205... |
| snap-1155667b | vol-e95dae81 | completed | 2010-12-02 17:14:21 | 100% | | 30 | Test | 176... |
| snap-d14473bb | vol-adf60dc5 | completed | 2010-12-01 18:22:48 | 100% | | 10 | Created by CreateImage(i-a1054ecc) for ami... | 204... |
| snap-dd8aa2b7 | vol-41bb4029 | completed | 2010-11-30 16:45:30 | 100% | | 30 | Created by CreateImage(i-4159112c) for ami... | 847... |
| snap-f1b39a9b | vol-7d966e15 | completed | 2010-11-30 10:00:29 | 100% | | 100 | C. jejuni bowtie mappings 30.11.2010 BEAC... | 473... |
| snap-d93d12b3 | vol-ce8b94a7 | completed | 2010-11-29 19:38:32 | 100% | | 2 | nosql_summer_hackathon_working | 903... |
| snap-19cae873 | vol-edff0285 | completed | 2010-11-29 03:12:57 | 100% | | 20 | Lucier Genome Browser | 640... |
| snap-5392c839 | vol-9ebba8f7 | completed | 2010-11-27 02:05:37 | 100% | | 10 | 27112010_photobook | 162... |
| snap-55b6eb3f | vol-501f7639 | completed | 2010-11-25 08:31:11 | 100% | | 950 | Ensembl 59 MySQL data | 954... |
| snap-r35c01a9 | vol-d7639ebf | completed | 2010-11-25 05:43:15 | 100% | | 6 | | 977... |