

15-319 / 15-619

Cloud Computing

Recitation 5

September 29th & October 1st
2015

Overview

- Administrative Stuff
- Last Week's Reflection
 - Project 2.1, OLI Modules 5 & 6, Quiz 3
 - New concepts
- This week's schedule
 - Project 2.2, OLI Modules 7, 8 & 9 , Quiz 4
- Demo

Administrative Stuff

- Cloud TA's – “We are here to help you learn”
 - Office Hours: [Piazza](#) , [Calendar](#)
- Budget, Budget, Budget
 - Plan your approach, **Calculate** before you deploy
 - Use **Spot** Instances
 - **Tag** your instance (a spot instance is an instance)
 - Remember to **Terminate** your instances
 - Penalties add up and are real
 - **Monitor** AWS expenditures

Administrative Stuff (Contd.)

Popular Piazza questions

- **Q. Why is my RPS so low?**
 - Inbound and Outbound rules (enable all traffic on all ports)
 - Same subnet for LG and DCs
- **Q. Why is my Load Generator hanging?**
 - Check `http://<DNS of data center>/lookup/random` before feeding it to the load generator. (Both the instance and the application are running)
- **Q. Why did my ELB not perform well on the first attempt but did well on the second?**
 - Develop experience (and read) about ELB warm-up

Announcements

- 15-619 Students:
 - Form your 15619 Project teams [@5](#)
 - Up to 3 students per team
 - Frontend, databases, scripting, ...
- Monitor AWS expenses

Last Week's Reflection

Project 2.1 AWS APIs, Scalability & Elasticity

- Vertical scaling
 - Differences in RPS between t2.small, t2.medium and t2.large
- Horizontal scaling
 - Difference in RPS, going from 1x to 4xm3.mediums
- Provision and monitor AWS resource programmatically
- Initial experience with load balancing

- Quiz 3
 - Cloud Management (OLI Module 5)
 - Cloud Software Deployment Consideration (OLI Module 6)

New Concepts

- Horizontal and Vertical Scaling
 - What are vertical and horizontal scaling?
 - How does one differentiate between the two?
 - How does the scaling methodology affect throughput?
- Basic Load Balancing
 - Was there anything predictable about the load that the load generator was sending?

This Week



Project 2.1 AWS APIs, Scalability & Elasticity

- Vertically Scaling, Horizontally Scaling and Load Balancing.

- **Project 2.2 Elasticity, Failure and Cost**

- Able to horizontally scale out and scale in to maximize throughput, minimize cost and mitigate failure.

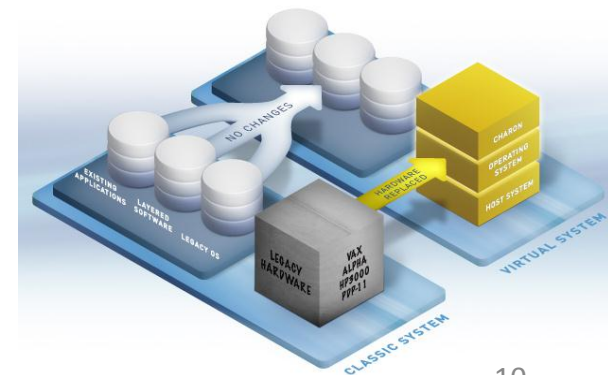
This Week's Schedule

- OLI Modules 7, 8 & 9
 - Virtualization
- Quiz 4
 - Due on Friday, Oct 2nd, 2015, 11:59PM ET
- Project 2.2
 - Due on Sunday, Oct 4th, 2015, 11:59 PM ET

OLI Module 7 - Virtualization

Introduction and Motivation

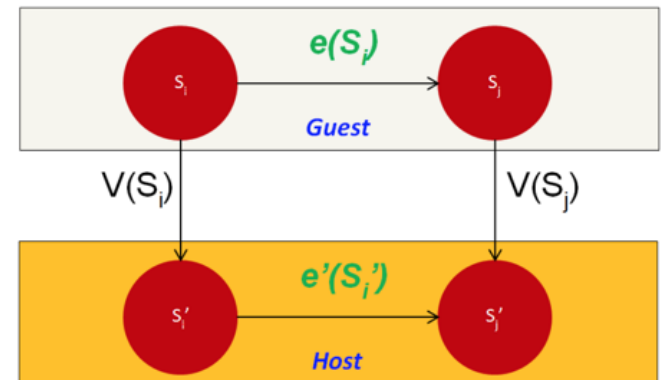
- Why Virtualization
 - Enabling the cloud computing system model
 - Elasticity
 - Resource sandboxing
 - Improved system utilization and reduce costs
 - Mixed OS environment
- Limitation of General – Purpose OS
- Resource Sharing
 - Time
 - Space



OLI Module 8

Virtualization

- What is Virtualization
 - Involves the construction of an isomorphism that maps a virtual guest system to a real (or physical) host system
 - Sequence of operations e modify guest state
 - Mapping function $V(S_i)$
- Virtual Machine Types
 - Process Virtual Machines
 - System Virtual Machines



OLI Module 9

Resource Virtualization - CPU

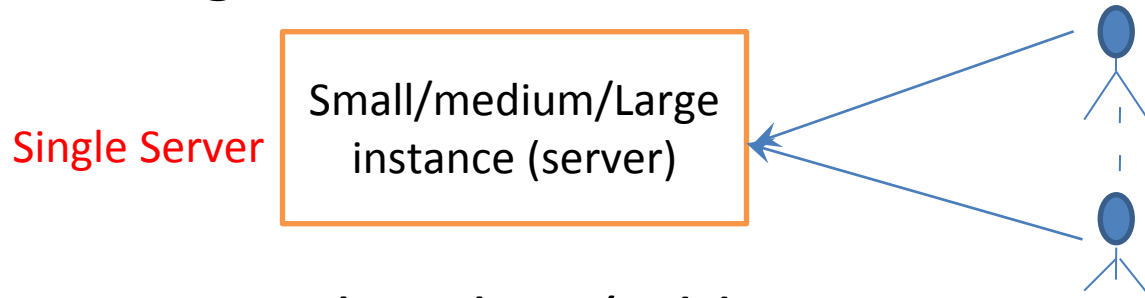
- Steps of CPU Virtualization
 - Multiplexing a physical CPU among virtual CPU's
 - Virtualizing the ISA (Instruction Set Architecture) of a CPU
- Code Patch, Full Virtualization and Para virtualization
- Emulation (Interpretation & Binary Translation)
- Virtual CPU

Quick Recap

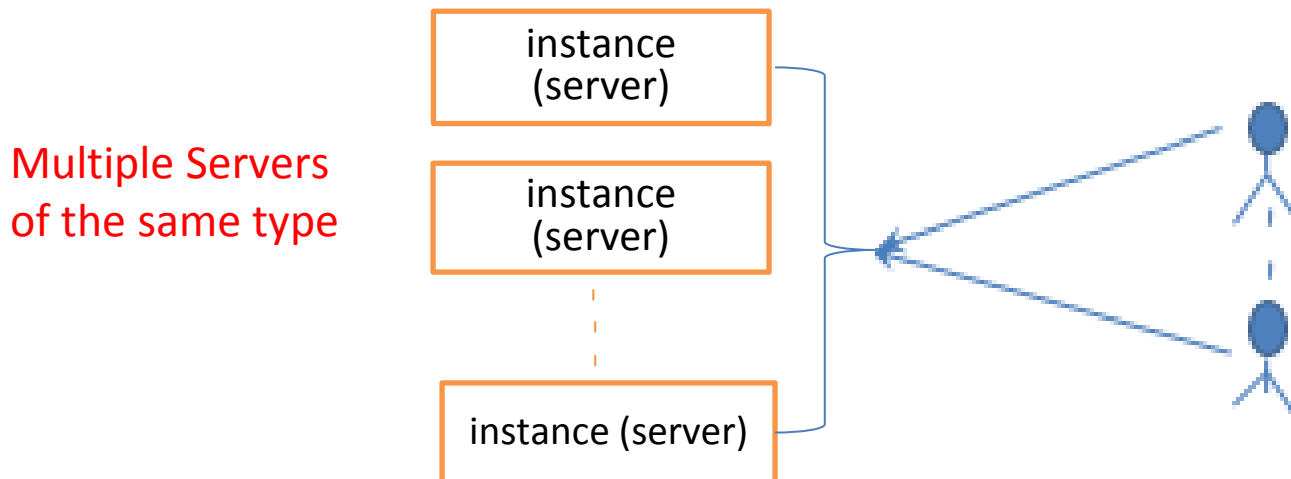
- When would you go for Vertical / Horizontal Scaling?
- How would you factor in cost, performance and failure into your decision?

Vertical Scaling vs. Horizontal Scaling

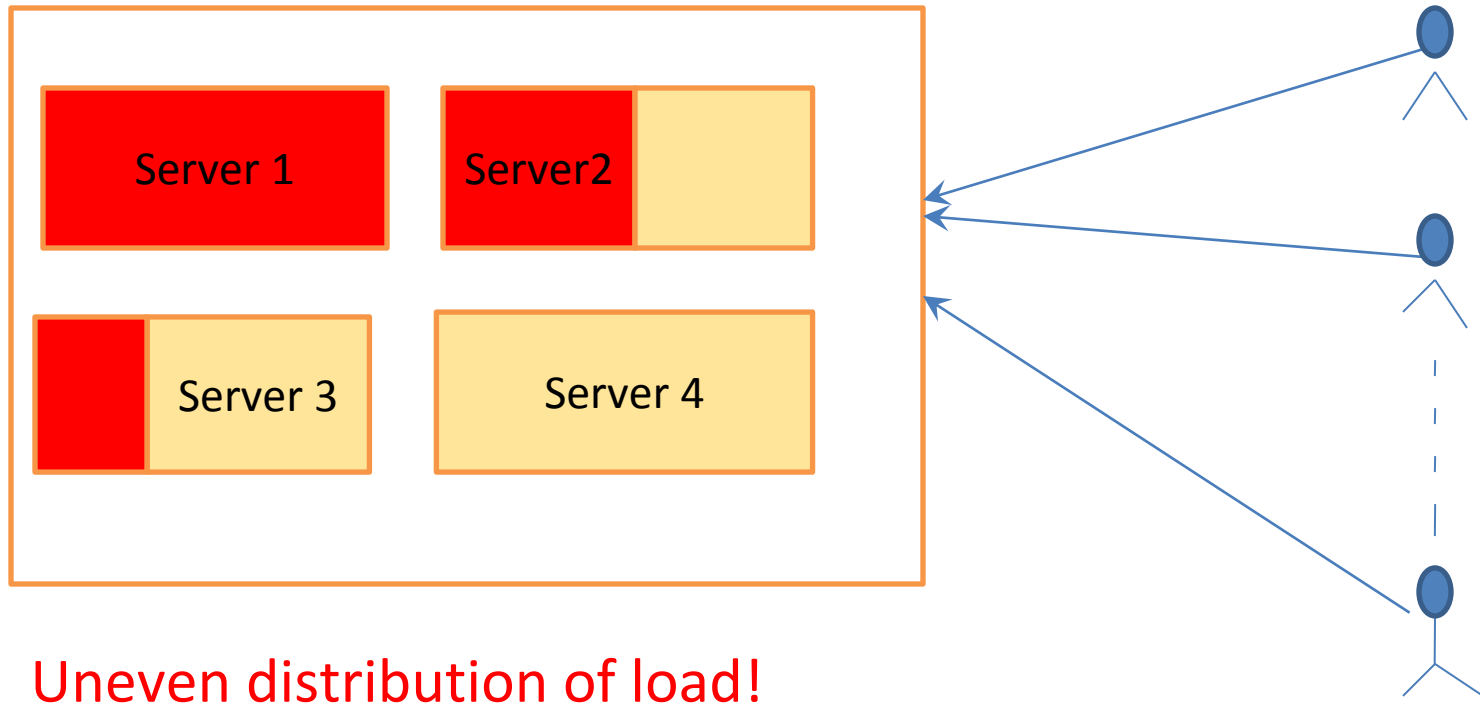
- Vertical Scaling Limitations
 - Can only increase the capacity to a limit
 - When scaling, need to transfer data, have to reboot





- Solution: Horizontal Scaling (add more resources)



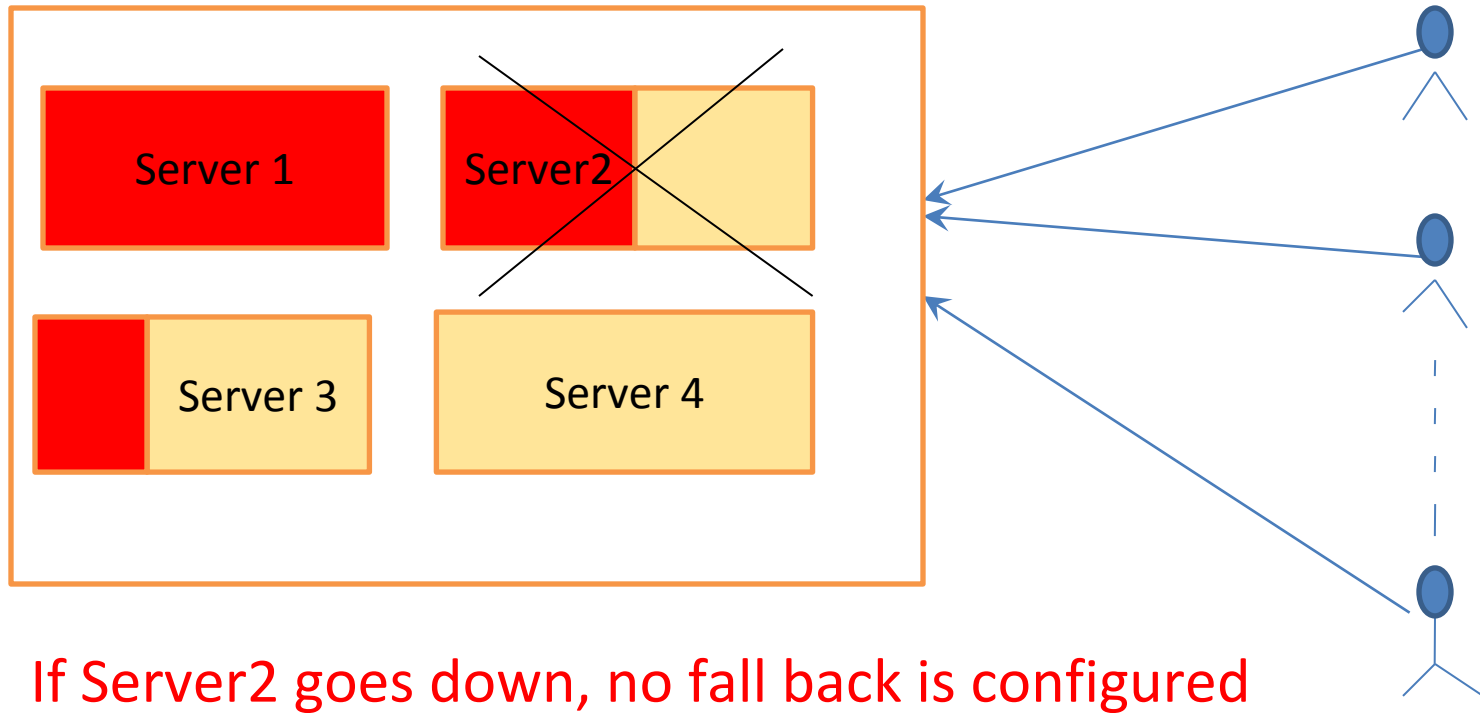
Load Balancing in Horizontal Scaling





Uneven distribution of load!

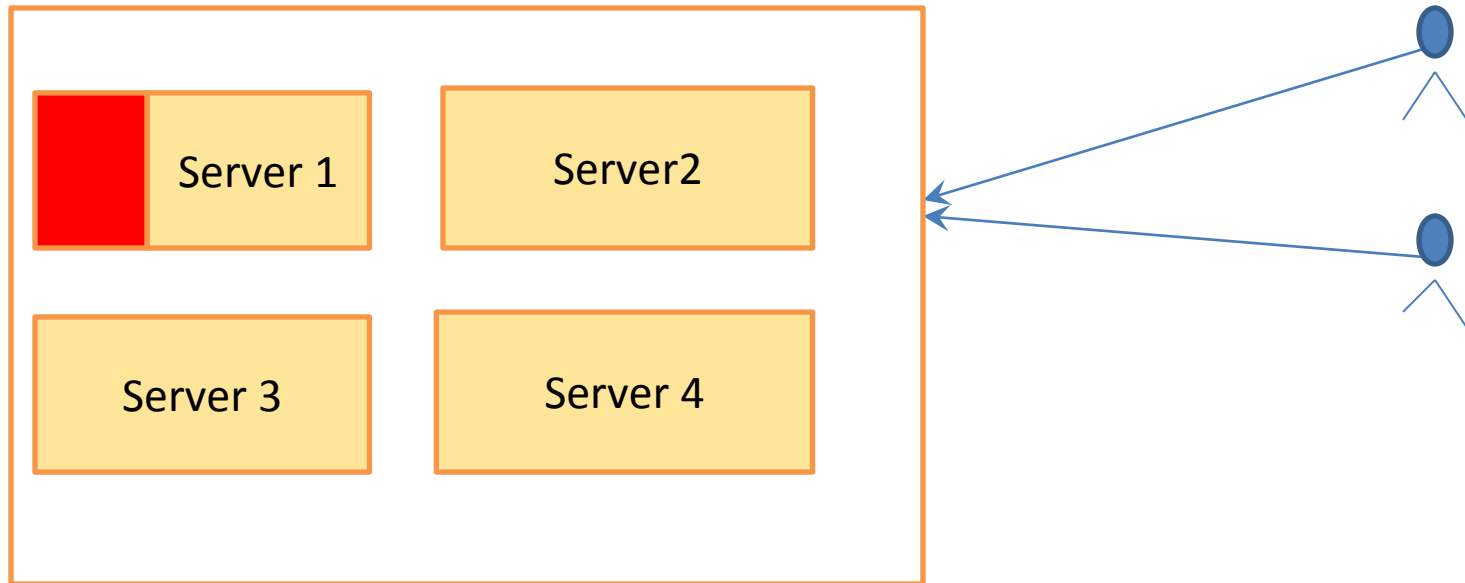
-  CPU utilization, memory utilization...
-  Available capacity

Health Check in Horizontal Scaling



-  CPU utilization, memory utilization...
-  Available capacity

Utilization in Horizontal Scaling



If load goes down, we need to change the number of servers



CPU utilization, memory utilization...

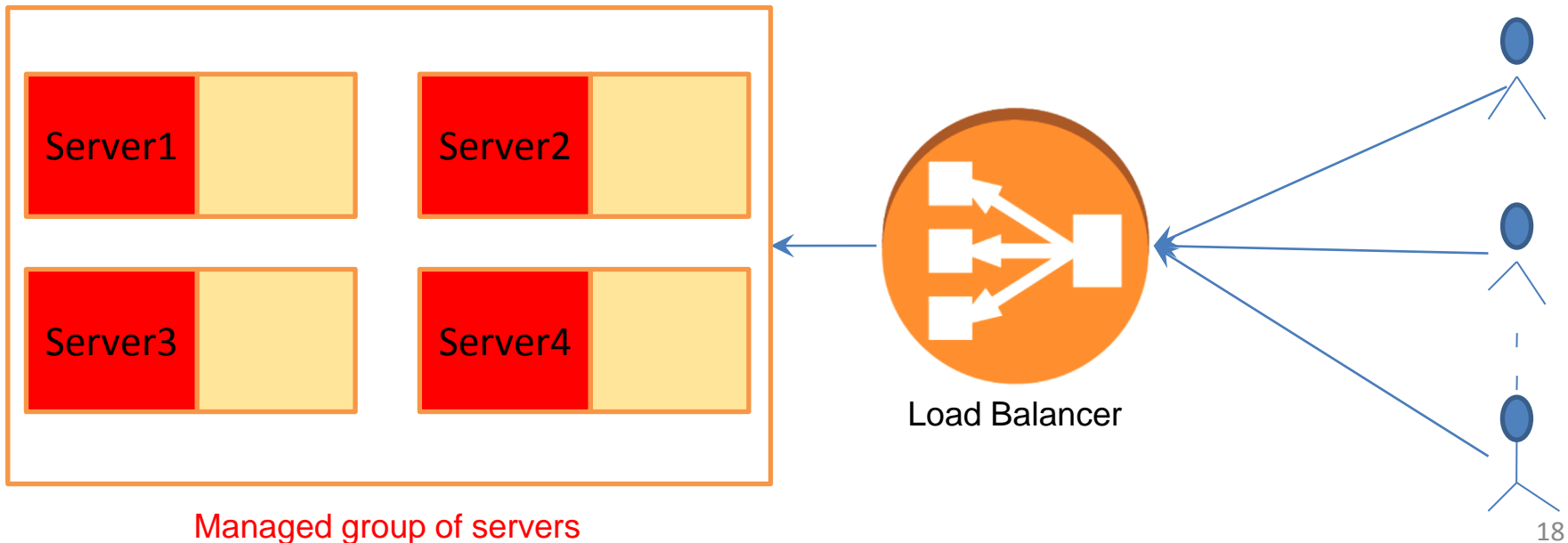


Available capacity

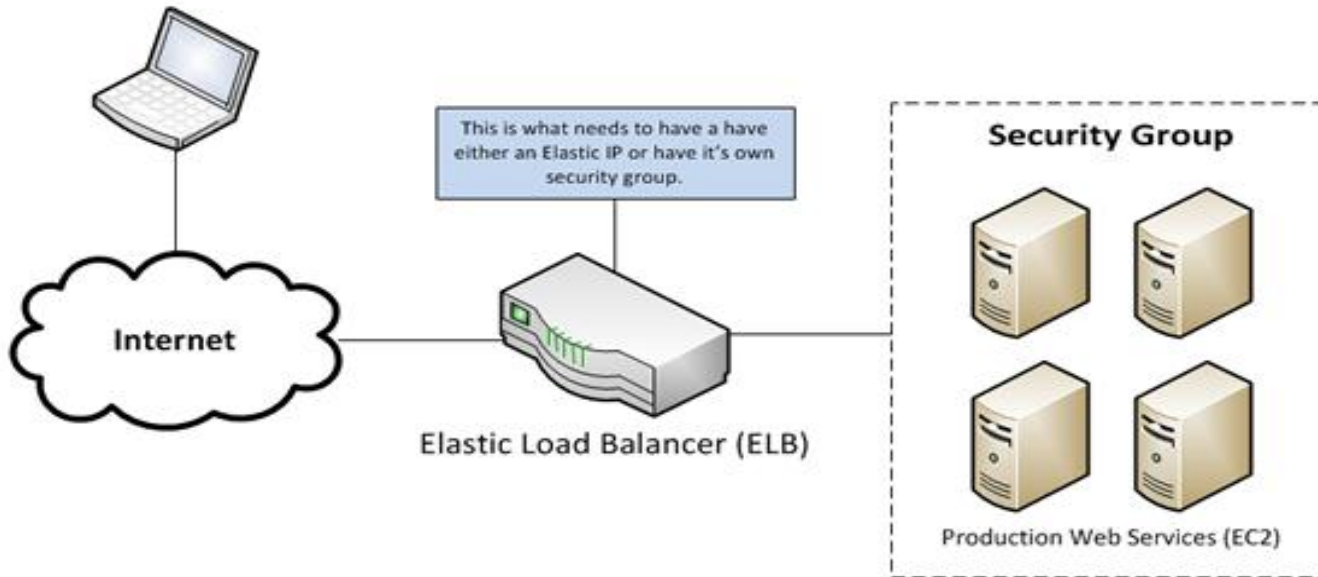
What You Need

- Make sure that workload is even on each server
- Do not assign load to servers that are down
- Increase/Remove servers according to changing load

How does AWS help solve these problems?



Load Balancer



- LB is a gateway that acts as a router interface and sends incoming requests to multiple Instances sitting behind it
- Distribute requests from clients to all servers equally

AWS ELB Features

- Distribute incoming to multiple Availability Zones
- Amazon EC2 instances' health
- Integration with an Auto Scaling Group (ASG)
- Handle varying load! (Well... Theoretically)

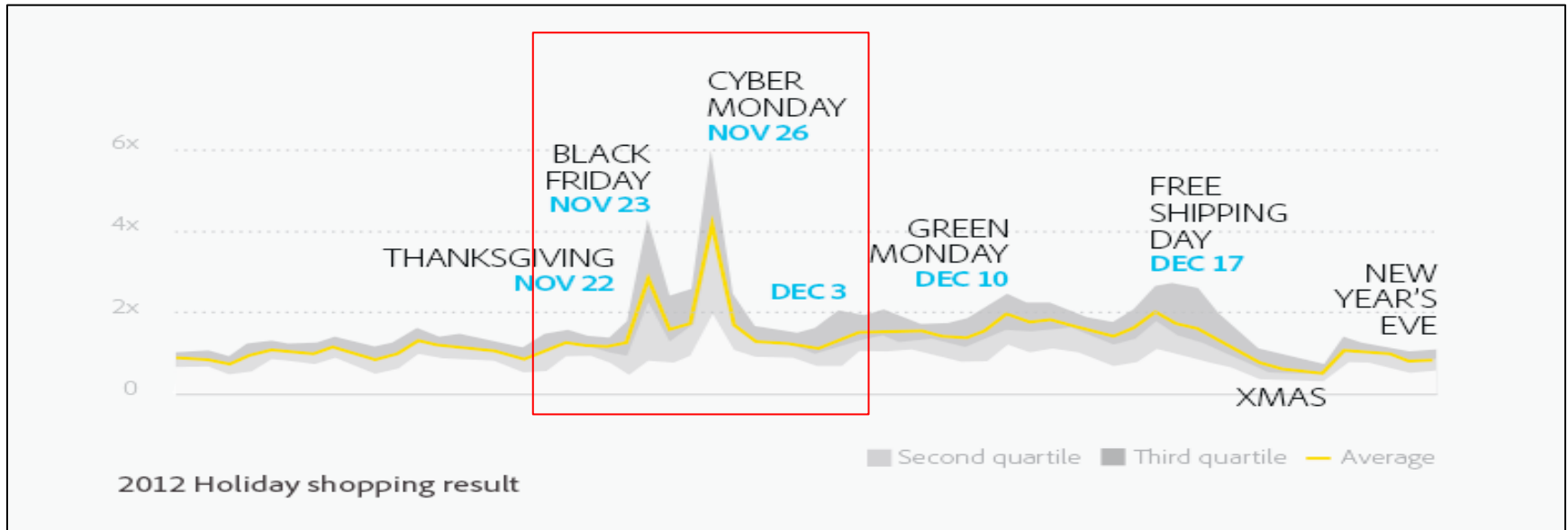
ELB Warmup

- ELB has a starting point for its initial capacity, and it will scale up or down based on traffic
- It struggles with high traffic spikes in shorter periods
- It is recommended that the load is increased at a rate of no more than 50 percent every five minutes (AWS Recommendation)

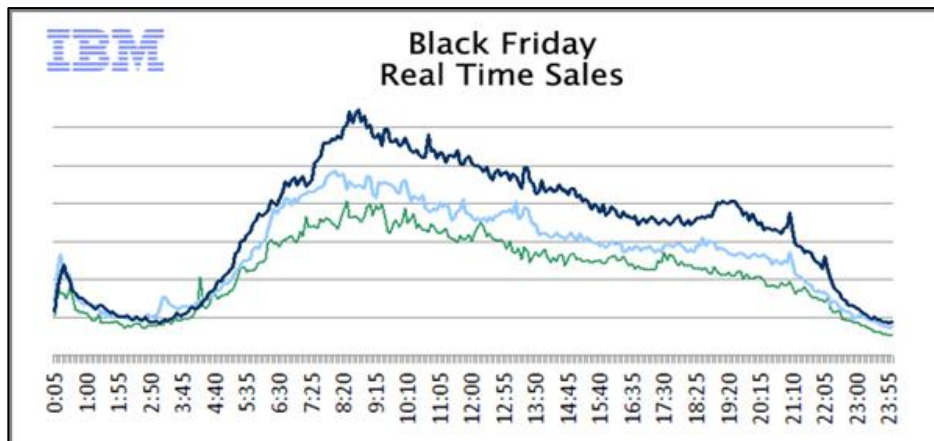
ELB Best Practices

<https://aws.amazon.com/articles/1636185810492479>

What is Auto Scaling?



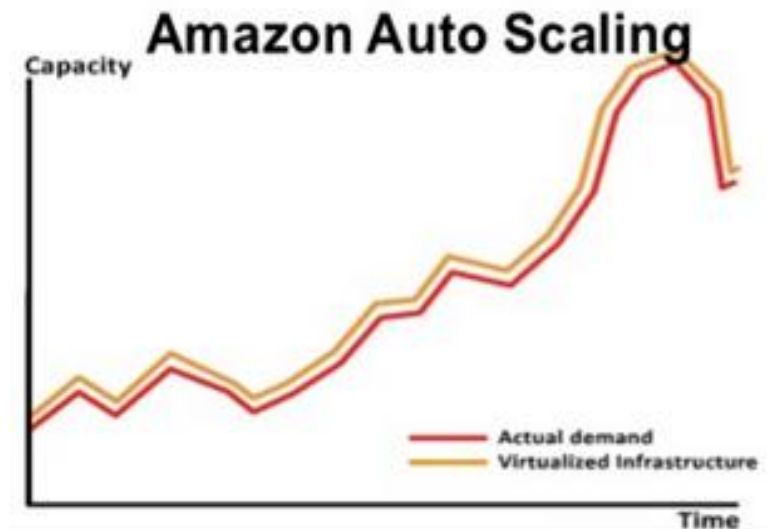
sapient.com



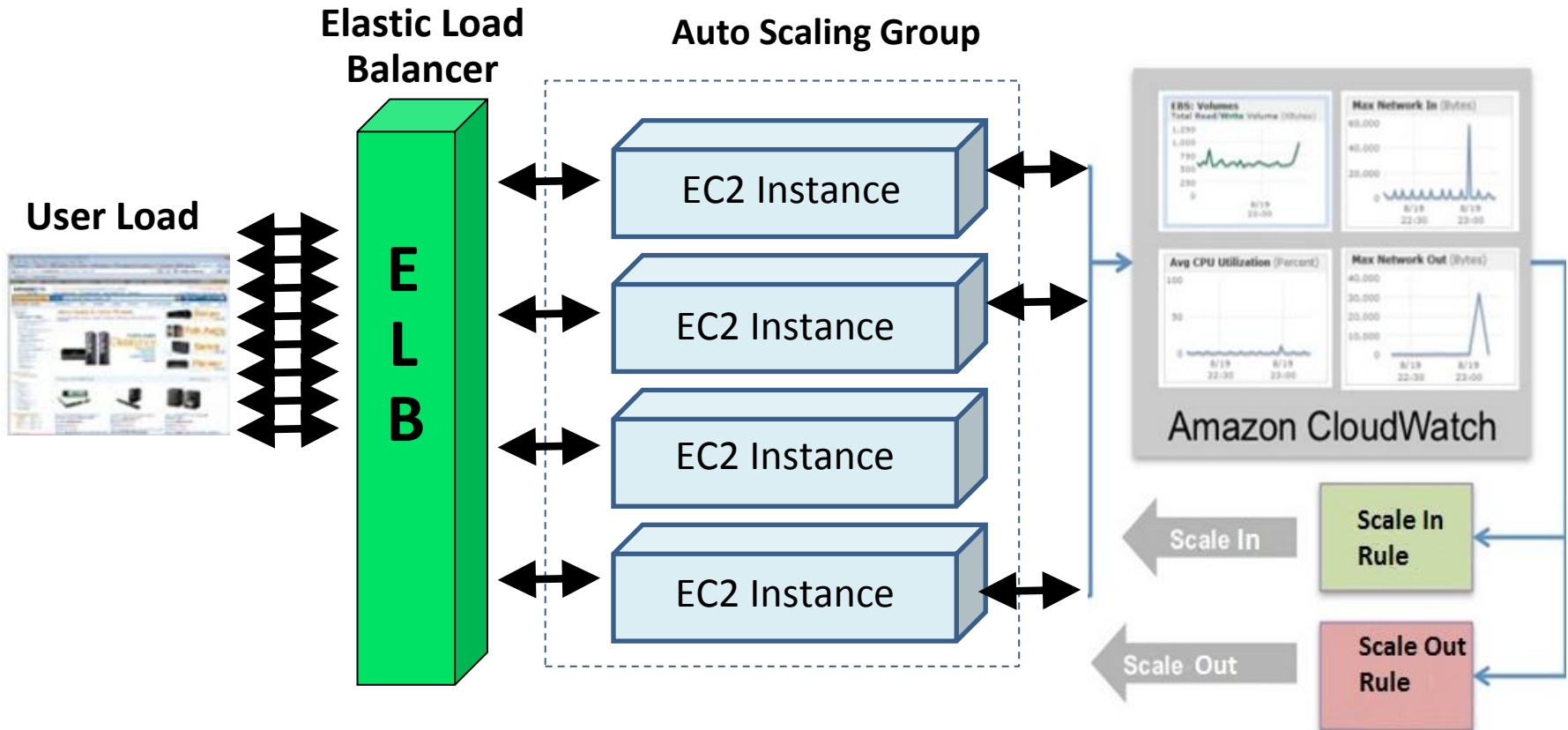
- Burst during the holiday season
- Losing customers with poor service
- Should the size vary with traffic?

Why Auto Scaling?

- Traditional Scaling:
 - Manually control the size
 - Under utilization or over provisioning of resources
 - Lose customers
- Auto Scaling:
 - Automatically adjust the size based on demand
 - Flexible capacities and scaling sizes
 - Save cost

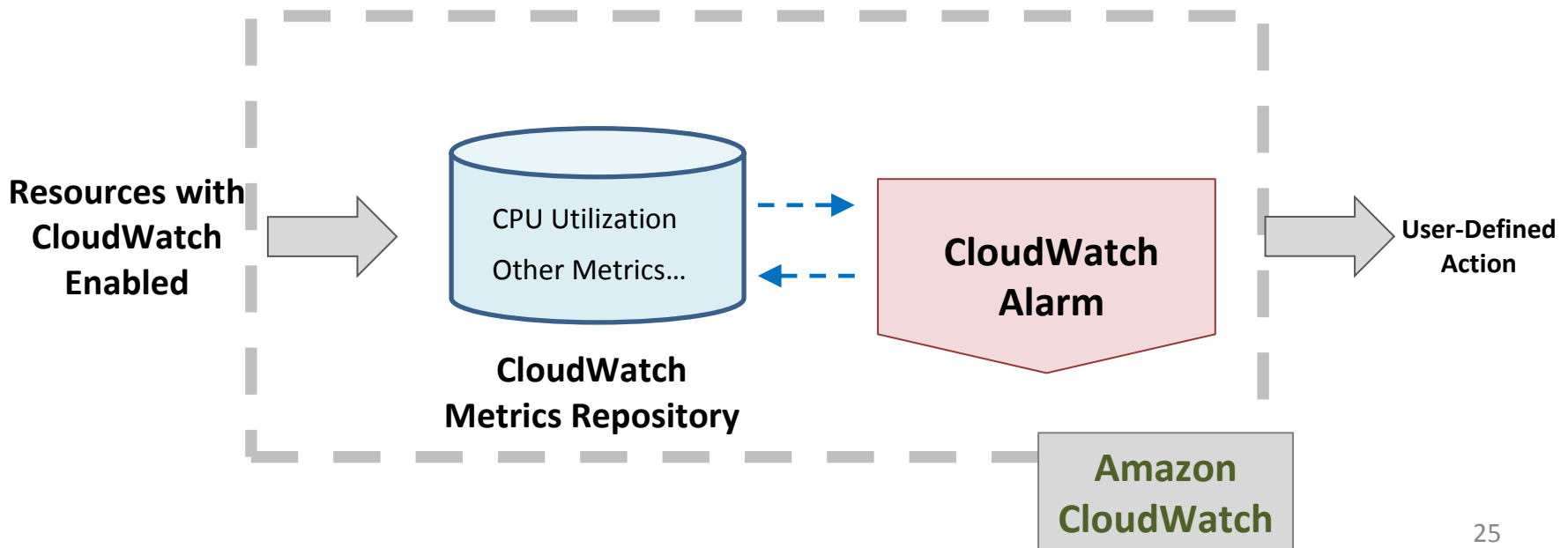


Amazon Auto Scaling Group



Amazon's CloudWatch Alarm

- Monitor CloudWatch metrics for some specified alarm conditions
- Take automated action when the condition is met



Case Study

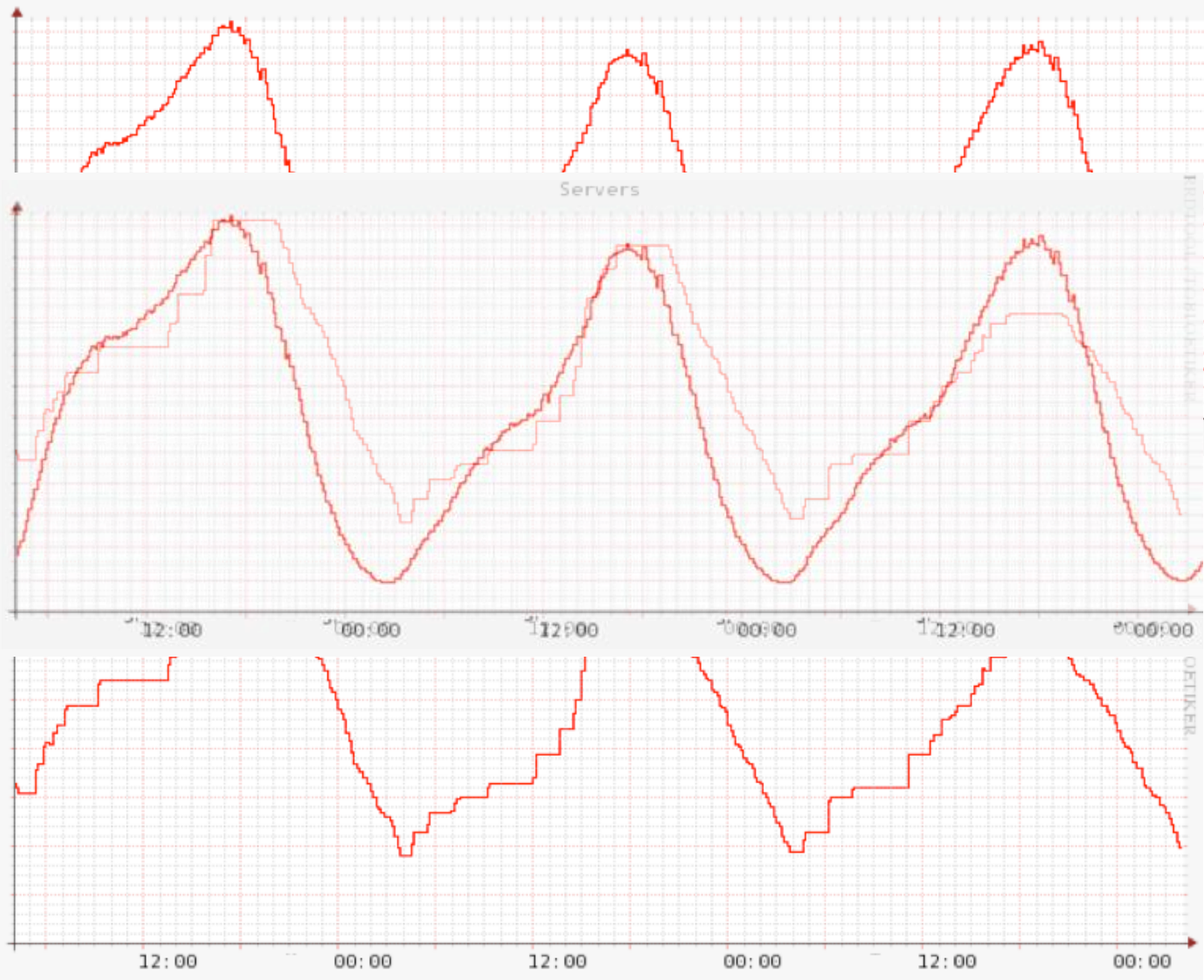
The Netflix logo, consisting of the word "NETFLIX" in white, bold, sans-serif capital letters, set against a red rectangular background.

- Netflix is one of the most popular provider of on-demand Internet streaming media
- Netflix has been using Amazon Auto Scaling Group for about 5 years.
- Data shows that use of ASG greatly improves the availability of Netflix services and provides an excellent means of optimizing cloud costs.
<http://techblog.netflix.com/2012/01/auto-scaling-in-amazon-cloud.html>

Case Study



Server Workload/Time

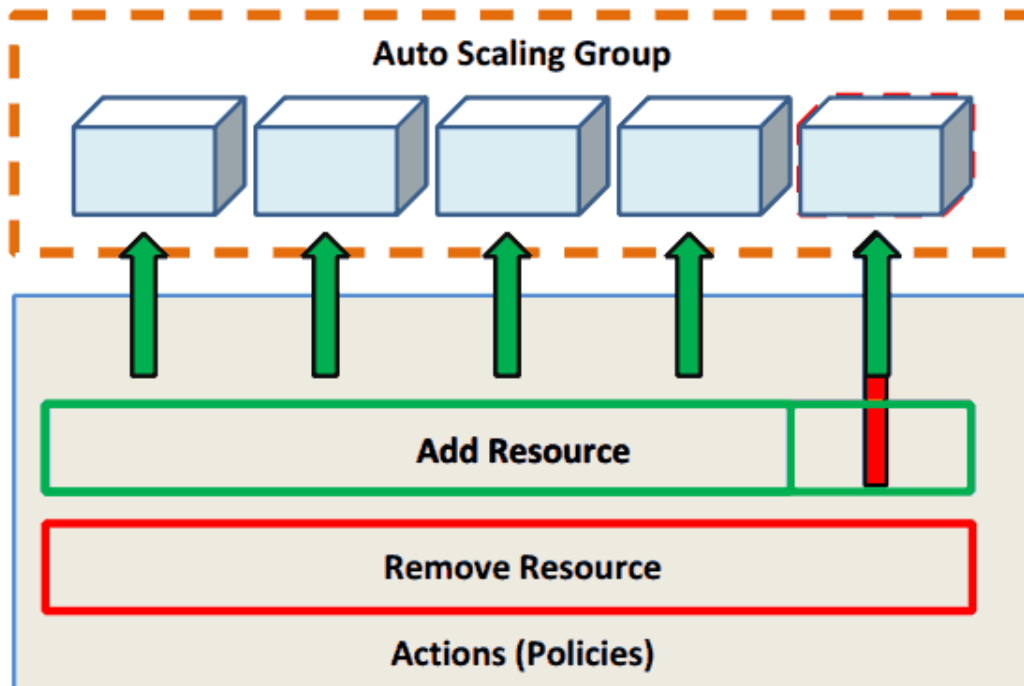


— Server Workload / Time

— Server Number / Time

P2.2 - Your Task

- Programmatically create an Elastic Load Balancer (ELB) and an Auto-Scaling Group (ASG) linked to ELB
- Test by submitting a URL request and observe changes
- Decide on Scale-Out and Scale-In policy
- Mitigate failure





Project 2.2 Suggestions

- Read project description more than once
- Think about the workflow before starting
- Look up API references
 - Read the overview first, then the details
 - Practice API calls
 - Search the Internet to debug
- Check every step carefully, debug with AWS console
- Don't forget to terminate your Auto Scaling Group and your Elastic Load Balancer after everything is done.

Resources

- Amazon's Auto Scaling Service
 - <http://aws.amazon.com/autoscaling/>
- Amazon's CloudWatch Alarm
 - <http://aws.amazon.com/cloudwatch/>
- Amazon's Scaling Developer
 - <http://aws.amazon.com/autoscaling/developer-resources/>

Upcoming Deadlines

- Project 2.2: Elasticity, Failure and Cost
 - Due: 10/04/15 11:59PM ET (Pittsburgh) 
- Unit 3: Virtualization Resource for the Cloud
 - Module 7: Introduction and Motivation
 - Module 8: Virtualization
 - Module 9: Resource Utilization – CPU
- Quiz 4
 - Due: 10/02/15 11:59PM ET (Pittsburgh) 

Project 2.2 Demo

Key Points to Remember

- Working with AWS APIs
 - Everything takes time to set up.
 - Request Resource -> Check Resource -> Use Resource
- Terminate EVERYTHING when DONE. (Unless otherwise specified)
- Budget Tip: Reuse existing ASG for subsequent tests.
 - The final test is 1 hour long.
 - Write code to reuse resources as much as you can.
- ELB needs warmup.
 - Keep a target in mind and warm-up ELB till you meet it before starting a test.
- Enable detail monitoring for < 5 min intervals.

```
[Minute 33]  
rps=851.88
```

```
[Minute 34]  
rps=0
```

```
[Minute 35]  
rps=0
```

```
[Minute 36]  
rps=0
```

```
[Minute 37]  
rps=0
```

```
[Minute 38]  
rps=0
```

```
[Minute 39]  
rps=0
```

```
[Minute 40]  
rps=-.01
```

```
[Minute 41]  
rps=0
```

```
[Minute 42]  
rps=66.66
```

Why is the RPS 0?

CloudWatch Monitoring Details



Sum HTTP 2XXs (Count)

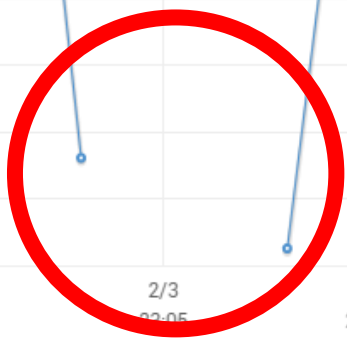
Statistic: Sum

Time Range: Last 3 Days

Period: 1 Minute



No Data?



Close

Questions?