

# HOW NETFLIX AUTOSCALES CI

Rahul Somasunderam



# WHAT DOES CI LOOK LIKE AT NETFLIX



# JENKINS @ NETFLIX

- 35 Jenkins controllers
- ~45k job definitions
- ~600k builds per week
- 650-1500 agents
- 1-100 executors per agent

# THE SPINNAKER VIEW

- 1 Application
- 35 stacks (Controller Clusters)
- 180 Agent Clusters
- 1+ ASG per cluster
- All workloads on **AWS**

# CLUSTERS AND ASGS

- A**W**S has Auto Scaling Groups
- Spinnaker calls them Server Groups
- <Application>-<Stack>-<Detail>-  
v<Version>
- jenkins-unstable-agent-highlander-  
v123

# HOW TO PLAN FOR CI INFRASTRUCTURE



# INFINITE RESOURCES

- **P**rovision capacity based on known maximum load
- Multiply by a safety factor for good measure
- Monitor and change the capacity as load increases

# INFINITE PATIENCE

- **P**lan capacity based on median load
- Builds will sit in queue for long times



# INSTANT RESOURCES

- You will get resources as soon as you request for them
- **W**orks well with Containerizable builds
- Not all builds can be containerized
- **D**oes not scale well with large numbers of short-lived builds

# AUTOSCALING

- Set up minimum and maximum capacity
- Scale based on some metric

# WHAT METRIC TO USE



# SYSTEM METRICS

C**PU**/Memory/**D**isk **I**O/Network throughput

- Natively supported by cloud providers and most metrics solutions

Scaling **P**olicies are supported by cloud providers

# SYSTEM METRICS

Not very useful for CI



# QUEUE DEPTH

Queue **D**epth seems adequately proportional.  
However, it is a trailing metric.

# AGENT UTILIZATION

**F**or each agent, find [idle, busy, offline] executors.

Sum these up by ASG.

Compute utilization as  $\frac{busy + offline}{busy + offline + idle}$

# MEASURING AGENT UTILIZATION



# AN AGENT'S ASG

**W**hen launching agents, use labels to specify the placement of the agent.



**Agent nflx-agent-unstable-i-0522989245ff3659d (Connect: `ssh -t i-0522989245ff3659d`)**

Mark this node temporarily offline

Agent is connected.

## Labels

asg:jenkins-unstable-bionic-v189 aws:test:us-east-1:jenkins-unstable-bionic-v189 **bionic** buildgroup:bionic carson.version:0.767.0 carson:true cloud:aws cluster:jenkins-unstable-bionic detail:bionic ec2.availZone:us-east-1e ec2.instanceType:m5d.xlarge ec2.region:us-east-1 env:test executors:4 iamRole:jenkinsInstanceProfile java.jvm:zulu8 java.runtime:1.8.0\_292-b10 nf.account:test **nf.app:jenkins** nflx.agent.build:569 os.arch:amd64 os.codename:bionic os.distribution:ubuntu os.name:linux os.release:18.04 stack:unstable us-east-1



# CAPTURING METRICS

**We** wrote a custom plugin that plays well with Atlas.  
You could write one for whatever your metrics capturing service is.

# AUTOSCALING



# HOW TO AUTOSCALE

A**W**S offers 2 ways to scale

- **T**arget **T**racking
- Step Scaling

# WHEN TO SCALE UP

Edit scaling policy



## Conditions

**Whenever**  of  [Search all metrics](#)



**is**

**for at least**  consecutive period(s) of



# HOW TO SCALE UP

## Actions

|   |    |                  |  |   |
|---|----|------------------|--|---|
| Add   | 20 | percent of group | when <code>jenkins.executorsUtilization</code> is between 0.65 and 0.8         |   |
| Add   | 40 | percent of group | when <code>jenkins.executorsUtilization</code> is greater than or equal to 0.8 |  |
| <div> Add step</div> |    |                  |  |   |

 [Documentation](#)

## Additional Settings

**Policy Name** jenkins-buildstest-bionic\_classic-v030-NFLX/EPIC-jenkins.executorsUtilization-GreaterThanThreshold-0.65-1-60-1620084562030



**Adjustment Step** Add instances in increments of at least 5 instance(s)

**Warmup** Instances need 600 seconds to warm up after each step



# WHEN TO SCALE DOWN

**Query**

[NAMED](#) [CUSTOM](#) [MANUAL](#)  [Help!](#)  [Atlas UI](#)

You can manually edit queries that are too complex for Custom Mode directly in Atlas Stack Language. If the edits result in an eligible query you can optionally switch to Custom Mode.

Query:

```
nf.app, jenkins, :eq,  
name, jenkins.executorsUtilization, :eq, :and,  
label, aws_.*.*v\d+, :re, :and,  
(, nf.app, nf.stack, label, ), :by,  
5m, :trend,  
0.25, :lt,  
15, :rolling-count,  
14, :gt,  
$(nf.app):$(nf.stack):$(label), :legend
```

Named Query

Compress

Auto-break

# HOW TO SCALE DOWN

| Controller  | ASG  | Exception | Idl | Tot | Rto | IC | TC | ZC | Count |
|-------------|--|-----------|-----|-----|-----|----|----|----|-------|
| jenkins/mce | test/us-east-1/jenkins-mce-bionic_classic-1-v020 |           | 19  | 20  | 6   | 6  | 6  | 6  | 6     |
|             | OK i-091aa9055f8dac251                           |           |     |     |     |    |    |    |       |
|             | OK i-08aeaf14573f2653d                           |           |     |     |     |    |    |    |       |
|             | OK i-04414343adb901c59                           |           |     |     |     |    |    |    |       |
|             | OK i-06a513fe9d989f10a                           |           |     |     |     |    |    |    |       |
|             | OK i-0f6e7eec07f0c3421                           |           |     |     |     |    |    |    |       |
|             | OK i-007fe724966b114bc                           |           |     |     |     |    |    |    |       |
|             | Terminate and shrink 6                           |           |     |     |     |    |    |    |       |



# RECAP



# WHAT WE LEARNT

- **T**his improved support experience
- **T**his improved the experience for spiky workloads

# THANK YOU!

[jobs.netflix.com](https://jobs.netflix.com)

