



LISA 2011

Practice and Experience Report

Scaling on EC2 in a fast-paced environment

Nicolas Brousse

nicolas@tubemogul.com

December 8th 2011

Introduction - About the speaker

- My name is Nicolas Brousse
- I previously worked for many industry leading company in France
 - From Web Hosting to Online Video services (Lycos, MultiMania, Kewego, MediaPlazza...)
 - Heavy traffic environment and large user databases
- I work as a Lead Operations Engineer at TubeMogul.com since 2008
- I help TubeMogul to scale its infrastructure
 - From 20 servers to +500 servers
 - Using 4 Amazon EC2 Regions + 1 Colo
 - Monitoring with Nagios over 6,000 actives services and 1,000 passives services
 - Collecting over 80,000 metrics with Ganglia
 - Managing over 300 TB of data in Hadoop HDFS
 - Billions HTTP queries a day
- Occasionally contribute to OpenSource projects
 - Ganglia (PHP and PERL module)
 - PHP Judy

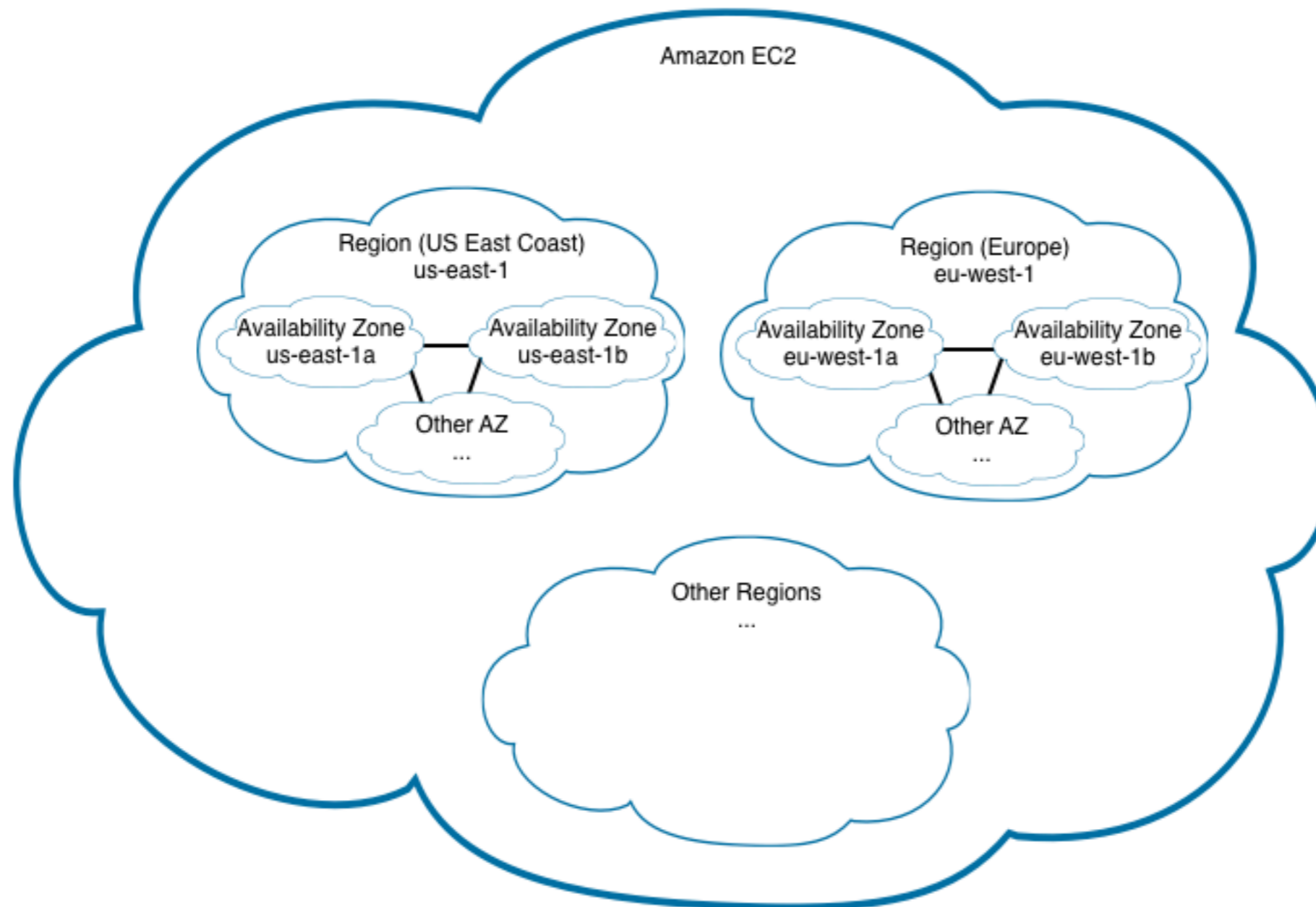
Introduction - About TubeMogul

- Created in November 2006 by John Hughes and Brett Wilson
- Formerly a video distribution and analytics platform
- Acquire Illuminex - a flash analytics firm - in October 2008
- New platform call PlayTime™ :
 - TubeMogul is a Video Marketing Company
 - Built for Branding
 - Integrate real-time media buying, ad serving, targeting, optimization and brand measurement

TubeMogul simplifies the delivery of video ads and maximizes the impact of every dollar spent by brand marketers

http://www.tubemogul.com/company/about_us

Amazon Cloud Environment



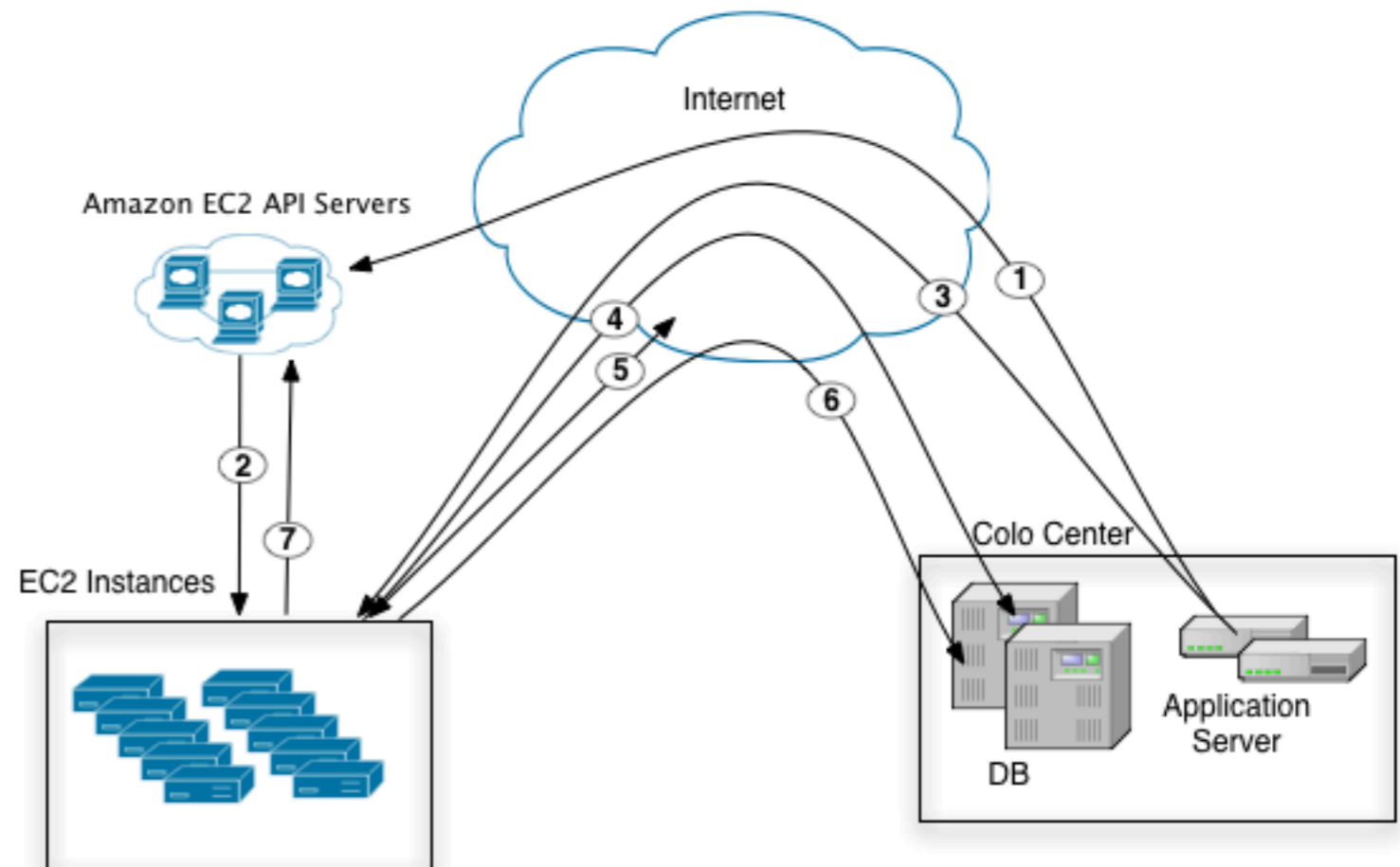
Amazon Cloud Environment

- We like it because....
 - We can quickly start new servers/clusters
 - We can quickly start new servers/clusters **in many regions**
 - US East (Virginia)
 - US West (North California)
 - Europe (Dublin)
 - Asia Pacific (Tokyo & Singapore)
 - We can use different type of instances (RAM, CPU, Disks, etc.)
 - It's easy to automate with EC2 API
 - It's easy to plug to a configuration management tool
- But...
 - It can be hard to troubleshoot some failures or network problems
 - Occasionally being notified of hardware failures after the facts
 - No Multicast (Though, possible with Amazon VPC)
 - Bandwidth cost between regions can get expensive

Auto scale a cloud application

Crawling partners' API to aggregate Analytics data:

1. Call AWS API to start instances
2. AWS Start instances
3. We push our code to the instances
4. Open SSH tunnel to DB
5. Crawl Partners API and aggregate the data
6. Push the data to our DB thru SSH Tunnel
7. Shutdown EC2 instance



First approach to manage “always on” instances

- Instances provisioning with a home made script called “Cerveza” in Tcl/Tk
 - Used to configure instance profile at boot
 - Let us run commands on multiple instances at once
- Using LDAP and SQLite to track hostname and instance profiles
 - DNS Bind plugged to LDAP
 - SQLite store EBS volume, Instances id, keypair, AMI, etc.
- Instance configuration with shell scripts from NFS mount
 - EBS Raid setup
 - Software deployment
 - Configuration changes

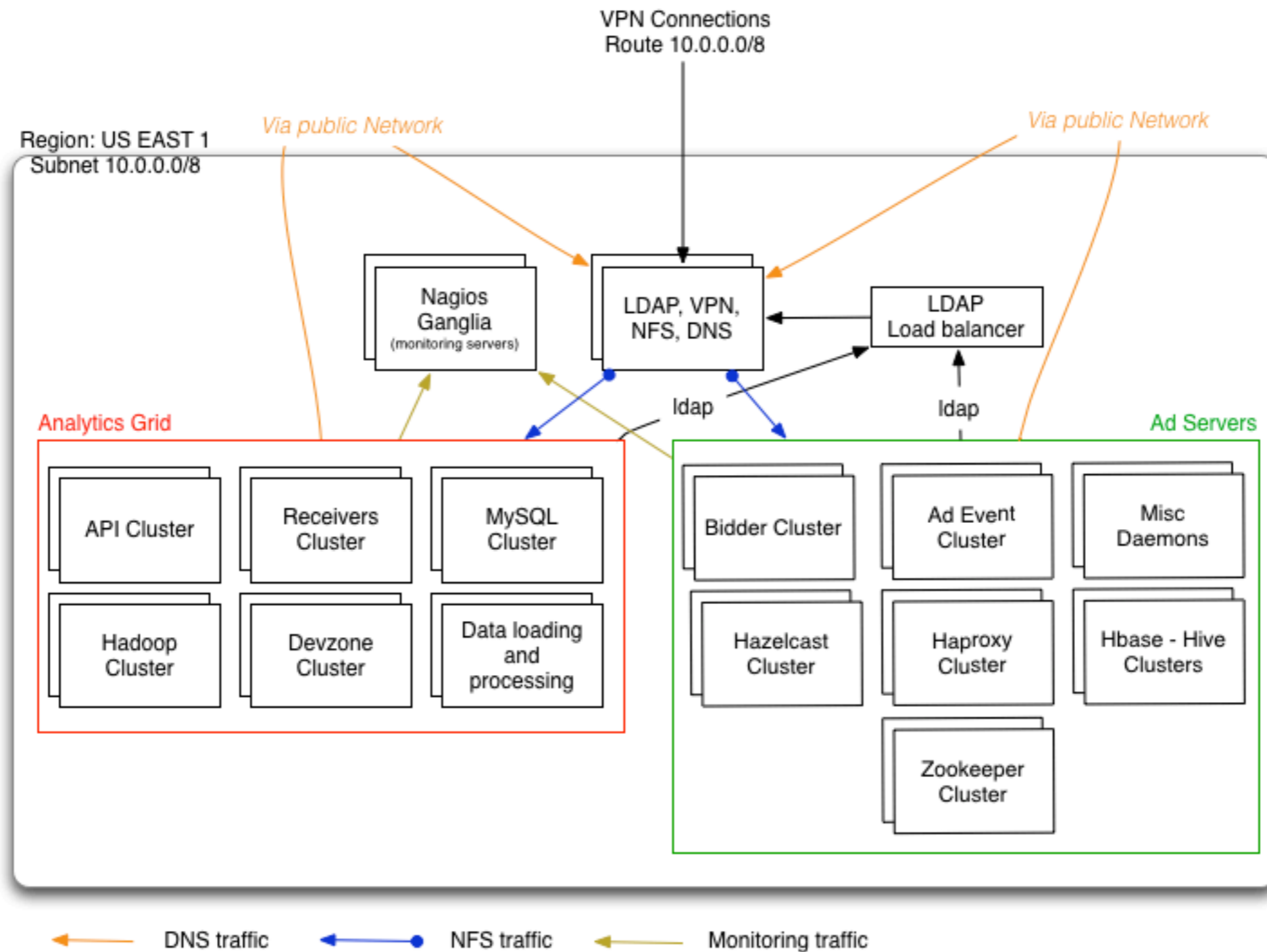
First approach to manage “always on” instances

- Access Control
 - Security Group for each Cluster type (Hadoop, MySQL, etc.)
 - SSH Access limited by VPN
 - SSH and VPN plugged to LDAP
- Easy way to identify instances
 - DNS plugged to LDAP
 - Each instance configured with human readable hostname
example: dev-mysql01, hadoop-namenode01, etc.
- Easy working process for devs
 - Users’ home directory using NFS auto-mount
- Automate Instances Monitoring
 - Trending with Ganglia, Alerting with Nagios
 - Using trending data for most Nagios checks
 - NSCA

Learning the hard way - the story

- SPOF in our infrastructure design lead us to a long outage
 - prevent us to login into servers for couple of days
 - couldn't start new instances
- 1. Main server with NFS/LDAP went down
 - corrupted EBS lead to fsck at boot time but no KVM on EC2...
 - starting new instance still get stuck on fsck at boot
 - end up using old AMI which we rebuild changing fstab
- 2. New instance took time to get ready to use because of old AMI
 - need to recover from ldap backup first
 - lost lot of configuration setup, slowing down our ability to manage instances
 - need to re-install and configure our instance management tool "Cerveza"
- 3. /etc/resolv.conf need to be changed on every instances
 - ssh backdoor didn't work all the time because of LDAP/SSH/NFS timeout
- 4. Security Group rules with private IP made the recovery process more complex

Learning the hard way - quick fixes

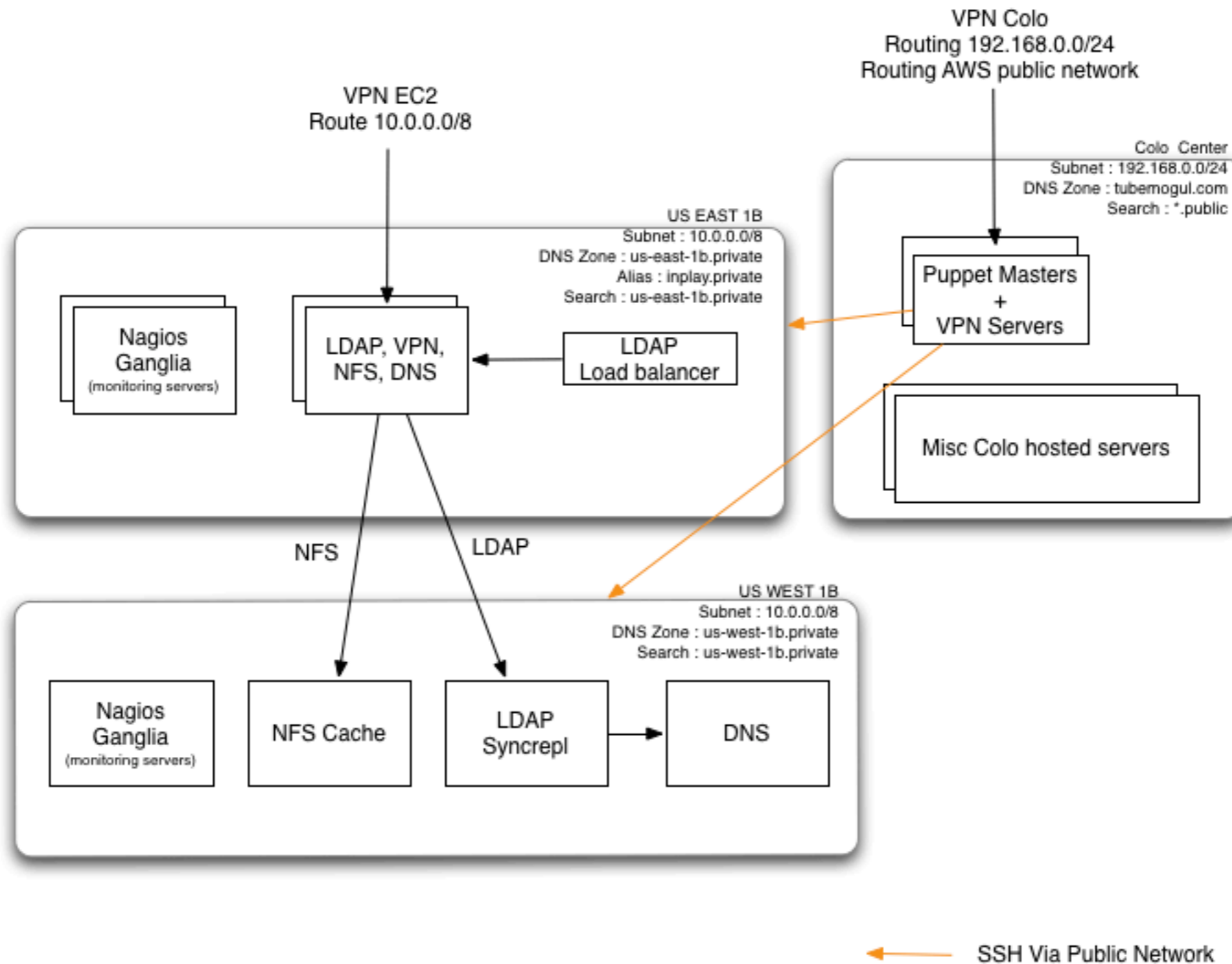


Going worldwide

- Need to handle multiple EC2 region for improved response time
 - use of gateway server in each Availability Zone
 - DNS Caching, LDAP Syncrepl, NFS with FS-Cache
- “Cerveza” rewrite in Python
 - Replace SQLite by SimpleDB
 - can be run from our laptop, need to run on a specific server
 - handle full provisioning (start/stop/reboot) instances
 - easily define instances profiles with YAML
 - use EC2 tags to identify instances and EBS volumes
- Stop building our own AMI
 - use of public ubuntu AMI server to reduce maintenance and support burden
 - use of cloud-init to easy start and preconfigure new instances
 - use of Puppet as our configuration management tool

```
--- !InstanceProfile
name: UbuntuGeneric32Bit
desc: Ubuntu Generic instance profile without EBS
Volumes
aws: !InstanceAws
  ami: { us-east-1: ami-a6f504cf, us-west-1:
ami-957e2ed0, ap-southeast-1: ami-7c423c2e, ap-
northeast-1: ami-3a0fa43b, eu-west-1: ami-339ca947 }
  security_group: devzone
  key_pair: tm-devzone
  type: c1.medium
  elastic_ip: false
volumes: [ ]
startup_scripts: [ ]
shutdown_scripts: [ shutdown ]
user_data: [ cloud-config-base.txt, setup-hostname.sh,
root-login.sh, cloud-config-puppet.txt ]
check_ec2_kernel: 2.6.35-28-virtual
```

Going worldwide



Lesson Learned

- Cloud environment doesn't mean you should by-pass basic infrastructure management rules
- Make sure the evolution of your infrastructure doesn't introduce SPOF
- Keep it simple, stupid
- Don't forget about backup and recovery process
- Use a configuration management tool early to prevent headache later

Thank You...

TubeMogul is Hiring !

<http://www.tubemogul.com/company/careers>
jobs@tubemogul.com

Follow us on Twitter



@tubemogul



@orieg