CSE 803 Introduction to Distributed Computing

[redacted]

Naming & Cloud Computing

① Basic Concepts

② Naming Services

③ Attribute-based Naming (aka Directory Services)

④ Distributed hash tables

What is Naming?

Systems manage a wide collection of entities of different kinds. They are identified by different kinds of names:

➜Files (/boot/vmlinuz), Processes (1, 14293), Users (chak, ikuz, cs9243), Hosts (weill, facebook.com), . . .

Examples of naming in distributed systems? What's the difficulty?

BASIC CONCEPTS

Name:

➜**String of bits or characters**

➜**Refers to an entity**

Entity:

➜**Resource, process, user, etc.**

➜**Operations performed on entities at access points**

Address:

➜**Access point named by an address**

➜**Entity address = address of entity's access point**

➜**Multiple access points per entity**

➜**Entity's access points may change**

BASIC CONCEPTS

➜ **Name that *uniquely* identifies entity**

➜ **Properties:**

① **Refers to at most one entity**

② **Entity referred to by at most one identifier**

③ **Always refers to same entity (i.e. no reuse)**

➜ **Allows easy comparison of references**

System-Oriented vs Human-Oriented Names

System-Oriented Names:

➜Represented in machine readable form (32 or 64 bit strings)

➜Structured or unstructured

V Easy to store, manipulate, compare

X Not easy to remember, hard for humans to use

➜ Example: inode (0x00245dad)

Human-Oriented Names:

➜ Variable length character strings

➜Usually structured

➜Often many human-oriented names map onto a single system-oriented name

V Easy to remember and distinguish between

X Hard for machine to process

➜Example: URL (http://www.cse.unsw.edu.au/~cs9243/lectures)

Name Spaces

Container for a set of related names

Structure options:

- **Flat (only leaf nodes)**
- **Hierarchical (Strictly hierarchical, DAG, Multiple root nodes)**
- **Tag-based**

Path Names (in hierarchies):

- **Sequence of edge labels**
- **Absolute: if first node in path name is a root node**
- **Relative: otherwise**

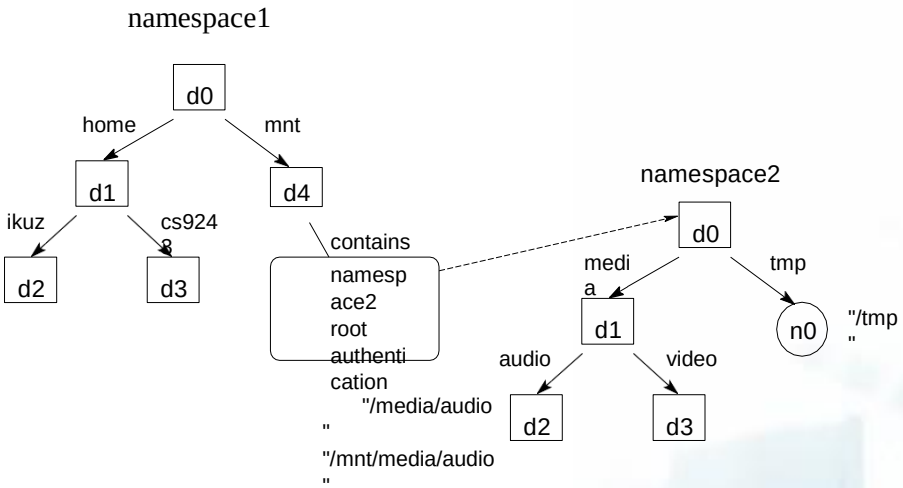
Aliasing:

- **Alias: another name for an entity**
- **Hard link: two or more paths to an entity in the graph**
- **Soft link: leaf node stores a (absolute) path name to another node**

NAME SPACES

➜ **Mounting**

- **Directory node stores info about a directory node in other name space**
- **Need: protocol, server, path name, authentication and authorisation info, keys for secure communication, etc.**



➜ Combining name spaces

- http://www.cse.unsw.edu.au/~cs9243/naming-slides.pdf
- Name Spaces: Protocol, DNS, File System

NAMING SERVICES

A naming service provides a name space

Name Server:

- **Naming service implemented by name servers**
- **Implements naming service operations**

Operations:

- Lookup: **resolve a path name, or element of a path name**
- Add: **add a directory or leaf node**
- Remove: **remove a subtree or leaf node**
- Modify: **modify the contents of a directory or leaf node**

Client:

- **Invokes naming service operations**

Centralised vs Distributed Naming Service

NAME RESOLUTION

The process of looking up a name

Resolution:

- ➜ **Mapping a name onto the node referred to by the name**
- ➜ **Interested in the data stored by the node**

Path Name Resolution:

- ➜ **Starts at a begin node (first element of the path name)**

•Root node for absolute name

•Directory node for relative name

- ➜ **Ends with data from (or a reference to) the last node (last element of path name)**

Resolver:

- ➜ **Does name resolution on behalf of client**
- ➜ **In client process, in client's kernel, process on client's machine**

PARTITIONING

Split name space over multiple servers

Structured Partitioning:

➜ split name space according to graph structure

➜ Name resolution can use zone hints to quickly find appropriate server

V Improved lookup performance due to knowledge of structure

X Rigid structure

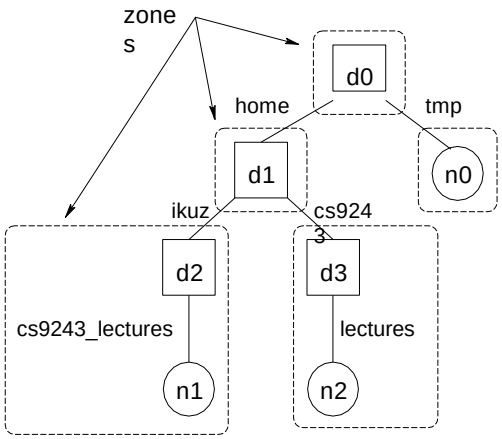
Structure-free Partitioning:

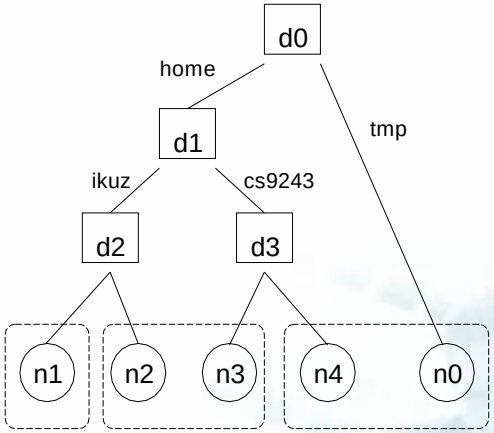
➜ content placed on servers independent of name space

V Flexible

X Decreased lookup performance, increased load on root

PARTITIONING





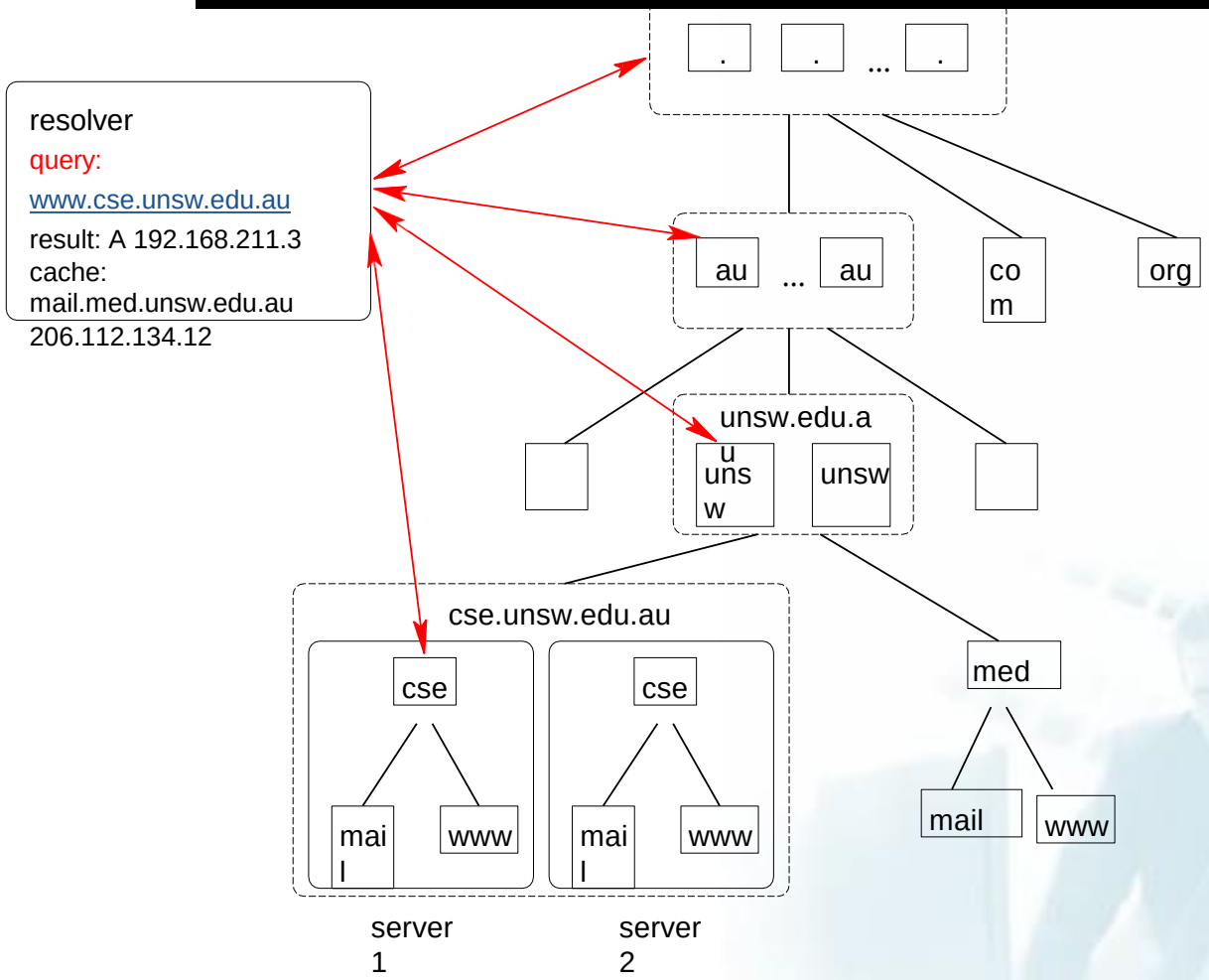
DNS (Domain Name System)

Structure:

- ➜ **Hierarchical structure (tree)**
- ➜ **Top-level domains (TLD) (.com, .org, .net, .au, .nl, ...)**
- ➜ ***Zone*: a (group of) directory node**
- ➜ ***Resource records*: contents of a node**
- ➜ ***Domain*: a subtree of the global tree**
- ➜ ***Domain name*: an absolute path name**

Type of record	Associated entity	Description
SOA	Zone	Holds information on the represented zone
A	Host	Contains an IP address of the host this node represents
MX	Domain	Refers to a mail server to handle mail addressed to this node
SRV	Domain	Refers to a server handling a specific service
NS	Zone	Refers to a name server that implements the represented zone
CNAME	Node	Symbolic link with the primary name of the represented node
PTR	Host	Contains the canonical name of a host
HINFO	Host	Holds information on the host this node represents
TXT	Any kind	Contains any entity-specific information considered useful

DNS (DOMAIN NAME SYSTEM)



DNS (DOMAIN NAME SYSTEM)

➜ Each zone implemented by a name server

Replication:

➜ Each zone replicated on at least two servers

➜ Updates performed on *primary*

➜ Contents transferred to *secondary* using *zone transfer*

➜ Higher levels have many more replicas (13 root servers: A-M.root-servers.net. Actually 386 replicas using anycast)

Caching:

➜ Servers cache results of queries

➜ Original entries have time-to-live field (TTL)

➜ Cached data is non-authoritative, provided until TTL expires

Name Resolution:

➜ Query sent to local server

➜ If cannot resolve locally then sent to root

➜ Resolved recursively or iteratively

Cloud Computing

① What is Cloud Computing?

② X as a Service

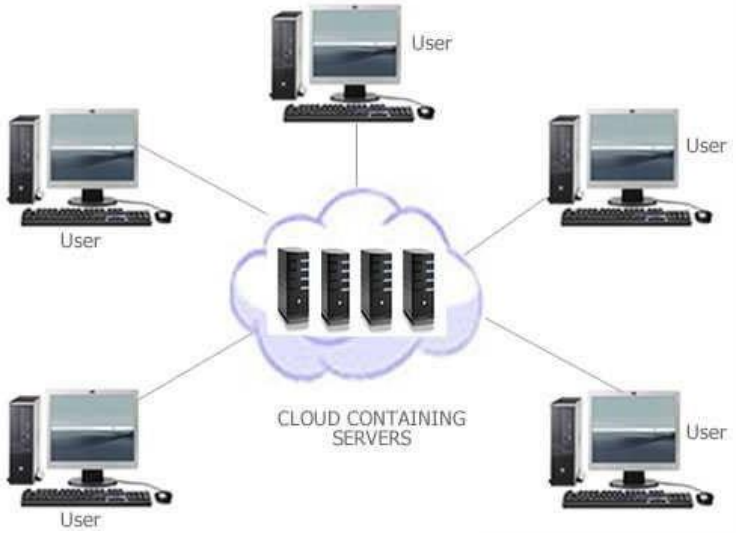
③ Key Challenges

④ Developing for the Cloud

What is Cloud Computing?

A style of computing in which dynamically scalable and often virtualized resources are provided as a service over the Internet. [Wikipedia]

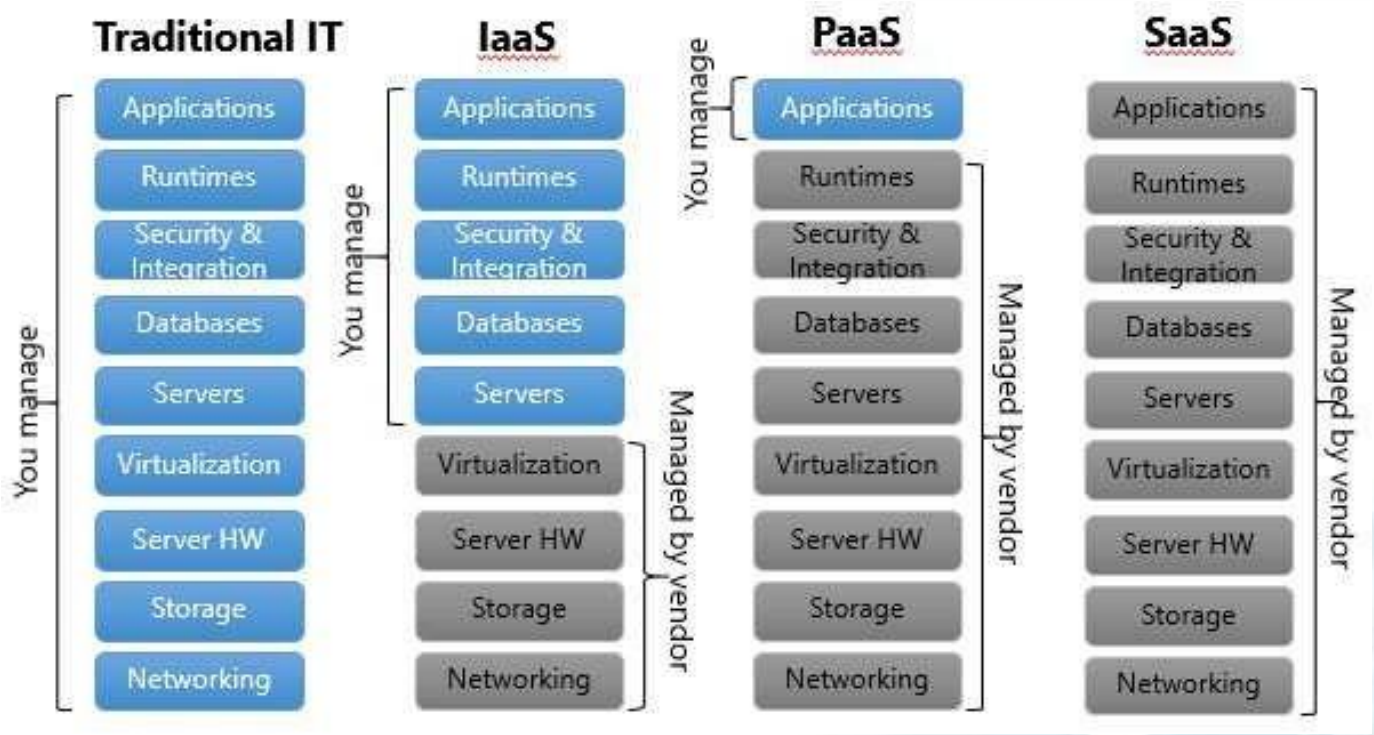
WHAT IS CLOUD COMPUTING?



Why is it called *Cloud*?

- ➜ **services provided on virtualised resources**
- ➜ **virtual machines spawned on demand**
- ➜ **location of services no longer certain**
- ➜ **similar to *network cloud***

WHAT IS CLOUD COMPUTING?



http://www.mazikglobal.com/blog/cloud-computing-stack-saas-paas-iaas/

WHAT IS CLOUD COMPUTING?

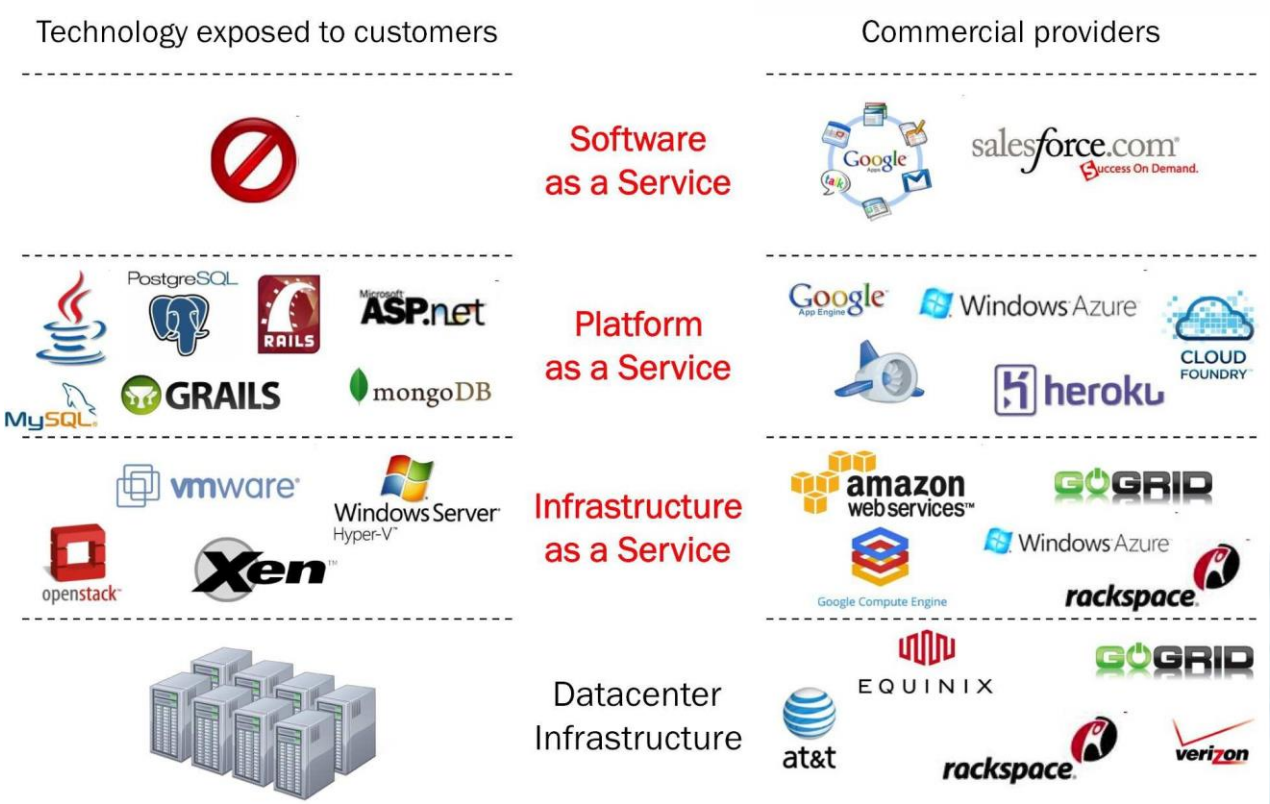


Figure from Hiroshi Wada

KEY CHARACTERISTICS OF CLOUD COMPUTING

KEY CHARACTERISTICS OF CLOUD COMPUTING

SP 800-145. The NIST Definition of Cloud Computing:

①On-demand, self-service

- get resources (CPU, storage, bandwidth etc),
- automated: as needed, right now!

②Network access

- services accessible over the network, standard protocols

③Pooled resources

- provider: multi-tenant pool of resources
- dynamically assigned and reassigned per customer demand

④Elasticity

- Scalability: rapidly adjust resource usage as needed

⑤Measured service

- monitor resource usage
- billing for resources used

BENEFITS

Flexibility:

➜Flexible provisioning

➜Add machines on demand

➜Add storage on demand

Effort:

➜Low barrier to entry

➜Initial effort: no need to spec and set up physical infrastructure

➜Continuing effort: no need to maintain physical infrastructure

Speed:

➜ Most cloud computing services are provided self service and on demand, so even vast amounts of computing resources can be provisioned in minutes, typically with just a few mouse clicks, giving businesses a lot of flexibility and taking the pressure off capacity planning

BENEFITS

Cost:

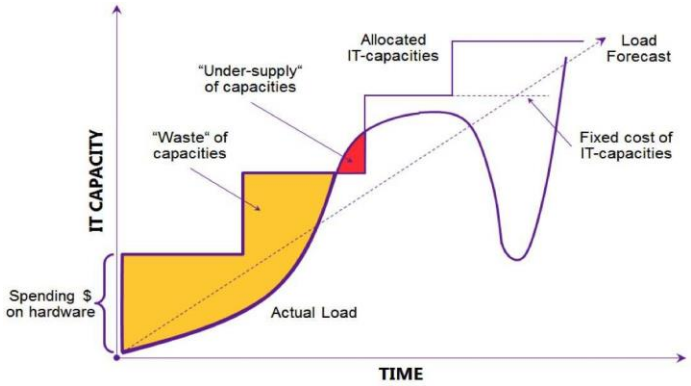
➜ **Cloud computing eliminates the capital expense of buying hardware and software**

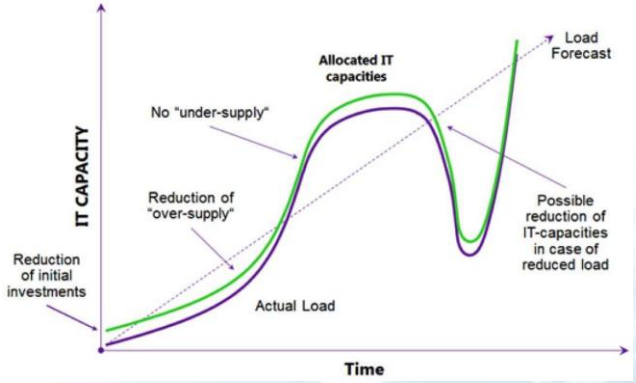
➜ **setting up and running on-site datacenters—the racks of servers, the round-the-clock electricity for power and cooling, and the IT experts for managing the infrastructure. It adds up fast.**

➜ **Low initial capital expenditure**

➜ **Avoid costs of over-provisioning for scalability**

➜ **Pay for what you use**





in "Developing and Extending Applications for Windows Azure with Visual Studio"

BENEFITS

Global scale

➜ **The benefits of cloud computing services include the ability to scale elastically. In cloud speak, that means delivering the right amount of IT resources—for example, more or less computing power, storage, bandwidth—right when they're needed, and from the right geographic location.**

Security

➜ **Many cloud providers offer a broad set of policies, technologies, and controls that strengthen your security posture overall, helping protect your data, apps, and infrastructure from potential threats.**

- ➜ **Redundancy**
- ➜ **Trust reliability of provider**
- ➜ **Data backups**
- ➜ ***What happens when provider goes down?***
- ➜ ***What about Security? Privacy?***

BENEFITS

Performance:

➜ The biggest cloud computing services run on a worldwide network of secure datacenters, which are regularly upgraded to the latest generation of fast and efficient computing hardware. This offers several benefits over a single corporate datacenter, including reduced network latency for applications and greater economies of scale.

Reliability

➜ Cloud computing makes data backup, disaster recovery, and business continuity easier and less expensive because data can be mirrored at multiple redundant sites on the cloud provider's network.

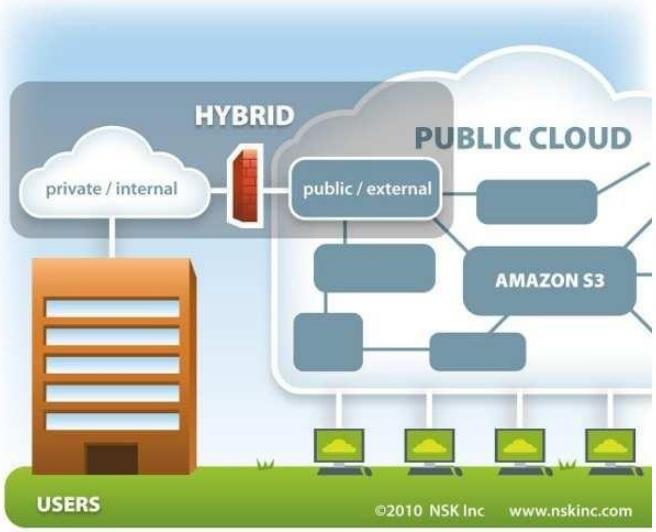
Productivity

➜ On-site datacenters typically require a lot of "racking and stacking"—hardware setup, software patching, and other time-consuming IT management chores. Cloud computing removes the need for many of these tasks, so IT teams can spend time on achieving more important business goals.

Speed

➜ Most cloud computing services are provided self service and on demand, so even vast amounts of computing resources can be provisioned in minutes, typically with just a few mouse clicks, giving businesses a lot of flexibility and taking the pressure off capacity planning.

Types of cloud computing



Public: open services available to everyone

Private: owned, operated, and available to specific organisation Is this still cloud computing?

Hybrid: system uses some private cloud services and some public cloud services.
http://blog.nskinc.com/IT-Services- Boston/bid/32590/Private-Cloud-or-Public- Cloud

Types of cloud services

Most cloud computing services fall into four broad categories: infrastructure as a service (IaaS), platform as a service (PaaS), serverless, and software as a service (SaaS). These are sometimes called the cloud computing "stack" because they build on top of one another. Knowing what they are and how they're different makes it easier to accomplish your business goals.

Infrastructure as a service (IaaS)

The most basic category of cloud computing services. With IaaS, you rent IT infrastructure—servers and virtual machines (VMs), storage, networks, operating systems—from a cloud provider on a pay-as-you-go basis.

Platform as a service (PaaS)

Platform as a service refers to cloud computing services that supply an on-demand environment for developing, testing, delivering, and managing software applications. PaaS is designed to make it easier for developers to quickly create web or mobile apps, without worrying about setting up or managing the underlying infrastructure of servers, storage, network, and databases needed for development.

Serverless computing

Overlapping with PaaS, serverless computing focuses on building app functionality without spending time continually managing the servers and infrastructure required to do so. The cloud provider handles the setup, capacity planning, and server management for you. Serverless architectures are highly scalable and event-driven, only using resources when a specific function or trigger occurs.

Types of cloud services

Software as a service (SaaS)
Software as a service is a method for delivering software applications over the Internet, on demand and typically on a subscription basis. With SaaS, cloud providers host and manage the software application and underlying infrastructure, and handle any maintenance, like software upgrades and security patching. Users connect to the application over the Internet, usually with a web browser on their phone, tablet, or PC.

INFRASTRUCTURE AS A SERVICE: IAAS

Service provider provides:

- ➜Server and network hardware
- ➜Virtual machines
- ➜IP addresses
- ➜Services to manage VMs (create, start, stop, migrate)
- ➜Optional: storage, database, synchronization, communication

Client provides:

- ➜OS and OS environment
- ➜Web server, DBMS, etc.
- ➜Middleware
- ➜Application software

INFRASTRUCTURE AS A SERVICE: IAAS

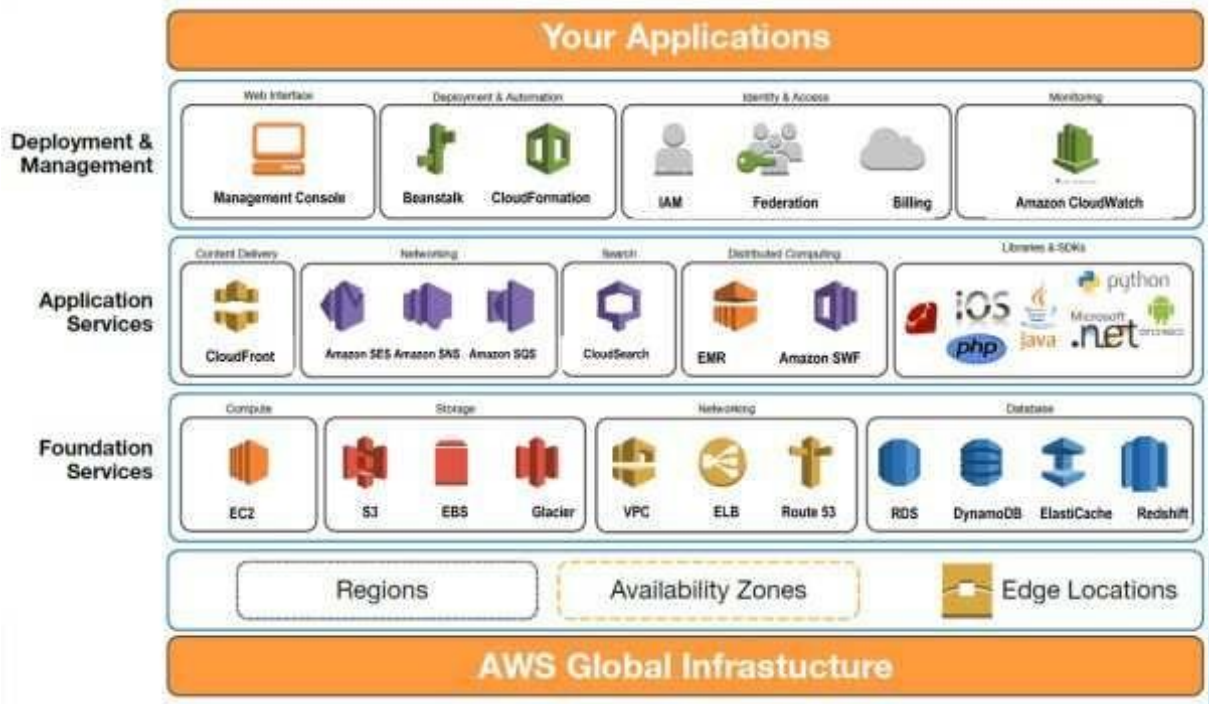
Challenges

- **Transparency (naming, redirection)**
- **Scalability: replication and load balancing decisions**
- **Synchronization and coordination**
- **Security**
- **Fault tolerance**
- **Software maintenance and sys admin**

Challenges – Provider:

- **Hardware provisioning and maintenance**
- **Load management**
- **IP address management, DNS management**
- **Infrastructure fault tolerance**
- **Monitoring, logging, billing**
- **Storage**

EXAMPLE 1: AMAZON WEB SERVICES (AWS)



- **Elastic Compute Cloud (EC2)**
- **Simple Storage Solution (S3)**
- **Simple DB**
- **Simple Queue Service**

http://vmtoday.com/2013/07/introduction-to-amazon-web-services-aws/

EXAMPLE 1: AMAZON WEB SERVICES (AWS)

➜ **Instances: virtual cores, memory, storage**

- **instance types (cpu,memory,net, storage options):**
- **t, m, c, p, g, x, r, i, d**
- **micro, small, medium, large, xlarge, ...**

➜ **Cost:**

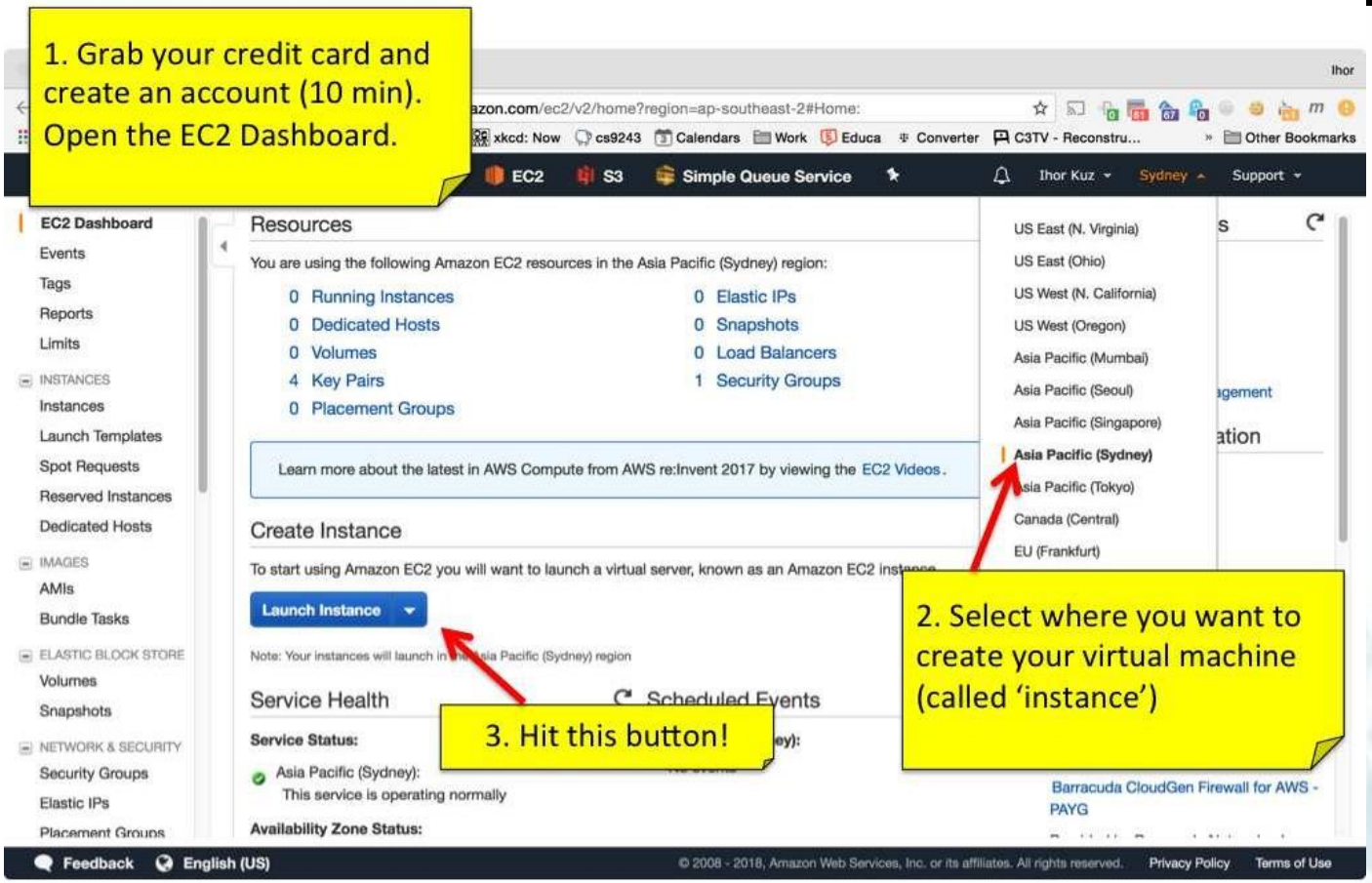
- **free tier: limited instances, free CPU hours**
- **on-demand: $0.007 - $39 per hour**
- **reserved: 1-3 years, discounted, fixed cost**

➜ **Launch Amazon Machine Image (AMI) on instances**

➜ **Preconfigured or custom images**

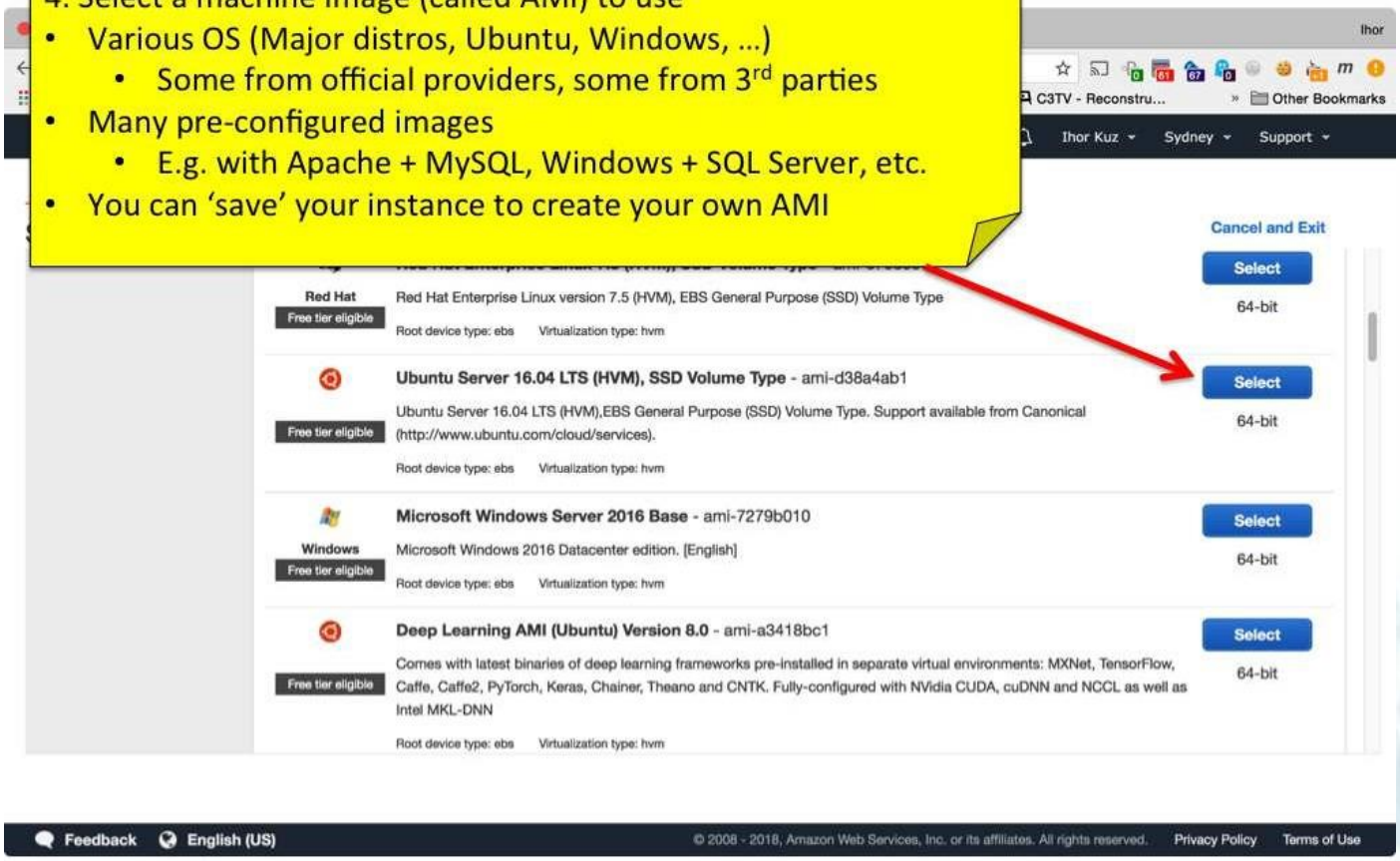
USING EC2

USING EC2


1. Grab your credit card and create an account (10 min). Open the EC2 Dashboard.
EC2
S3
Simple Queue Service
Ihor Kuz
Sydney
Support
EC2 Dashboard
Events
Tags
Reports
Limits
INSTANCES
Instances
Launch Templates
Spot Requests
Reserved Instances
Dedicated Hosts
IMAGES
AMIs
Bundle Tasks
ELASTIC BLOCK STORE
Volumes
Snapshots
NETWORK & SECURITY
Security Groups
Elastic IPs
Placement Groups
Resources
You are using the following Amazon EC2 resources in the Asia Pacific (Sydney) region:
0 Running Instances
0 Dedicated Hosts
0 Volumes
4 Key Pairs
0 Placement Groups
0 Elastic IPs
0 Snapshots
0 Load Balancers
1 Security Groups
Learn more about the latest in AWS Compute from AWS re:Invent 2017 by viewing the EC2 Videos.
Create Instance
To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.
Launch Instance
Note: Your instances will launch in the Asia Pacific (Sydney) region
Service Health
Scheduled Events
Service Status:
Asia Pacific (Sydney):
This service is operating normally
Availability Zone Status:
US East (N. Virginia)
US East (Ohio)
US West (N. California)
US West (Oregon)
Asia Pacific (Mumbai)
Asia Pacific (Seoul)
Asia Pacific (Singapore)
Asia Pacific (Sydney)
Asia Pacific (Tokyo)
Canada (Central)
EU (Frankfurt)
2. Select where you want to create your virtual machine (called 'instance')
3. Hit this button!
Barracuda CloudGen Firewall for AWS - PAYG
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

USING EC2

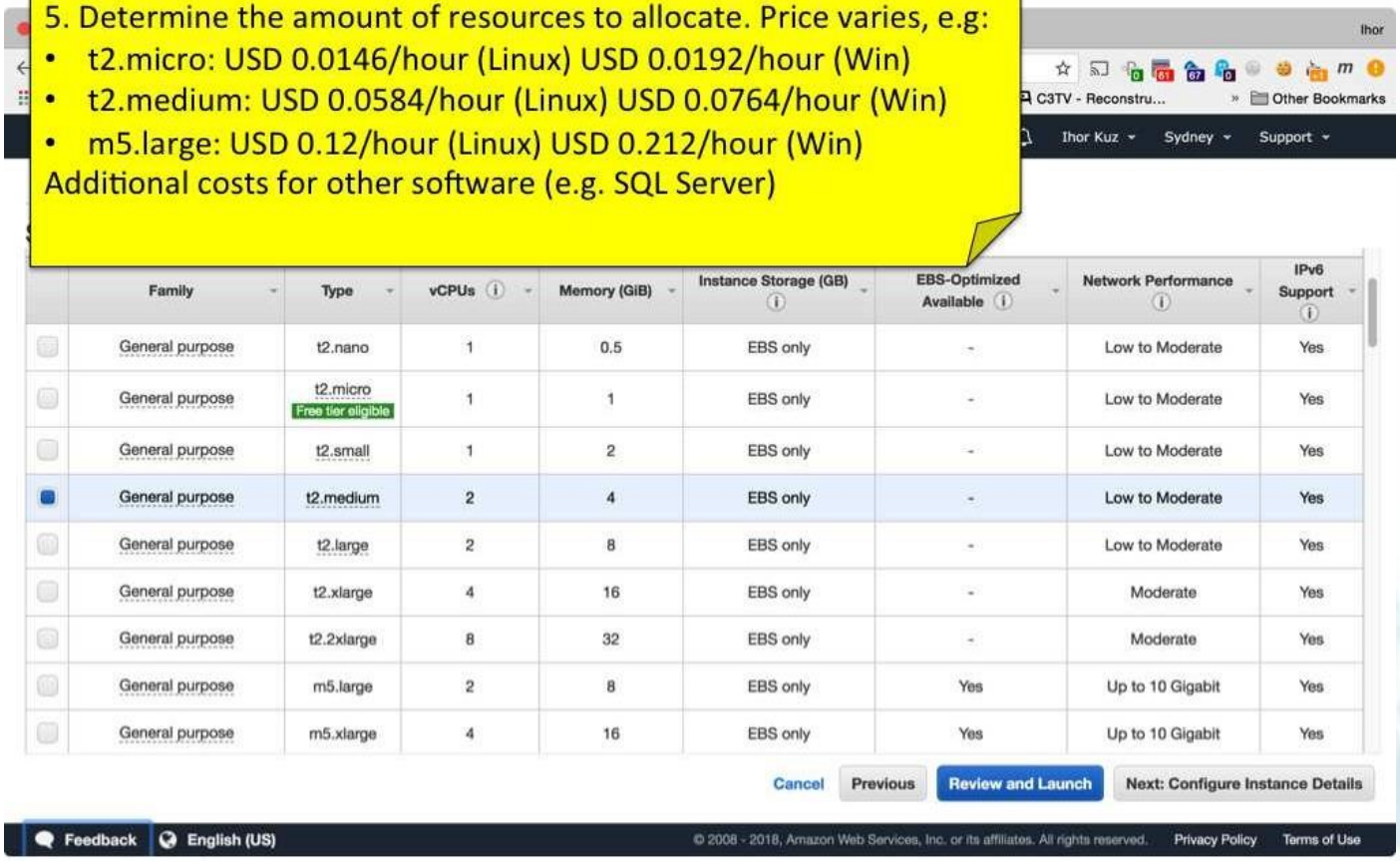
4. Select a machine image (called AMI) to use
- Various OS (Major distros, Ubuntu, Windows, …)
  - Some from official providers, some from 3rd parties
- Many pre-configured images
  - E.g. with Apache + MySQL, Windows + SQL Server, etc.
- You can 'save' your instance to create your own AMI



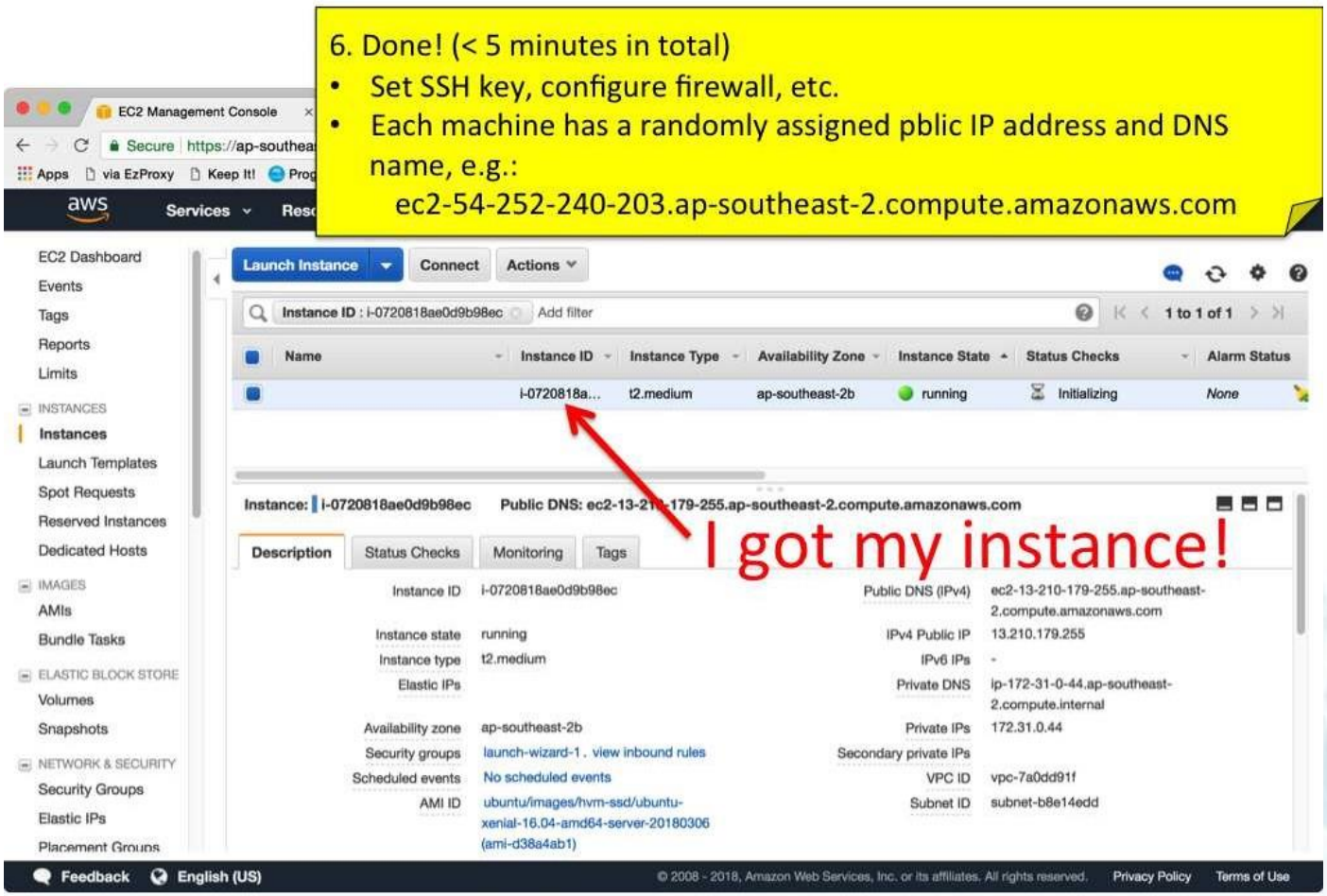
USING EC2

5. Determine the amount of resources to allocate. Price varies, e.g:
- t2.micro: USD 0.0146/hour (Linux) USD 0.0192/hour (Win)
- t2.medium: USD 0.0584/hour (Linux) USD 0.0764/hour (Win)
- m5.large: USD 0.12/hour (Linux) USD 0.212/hour (Win)

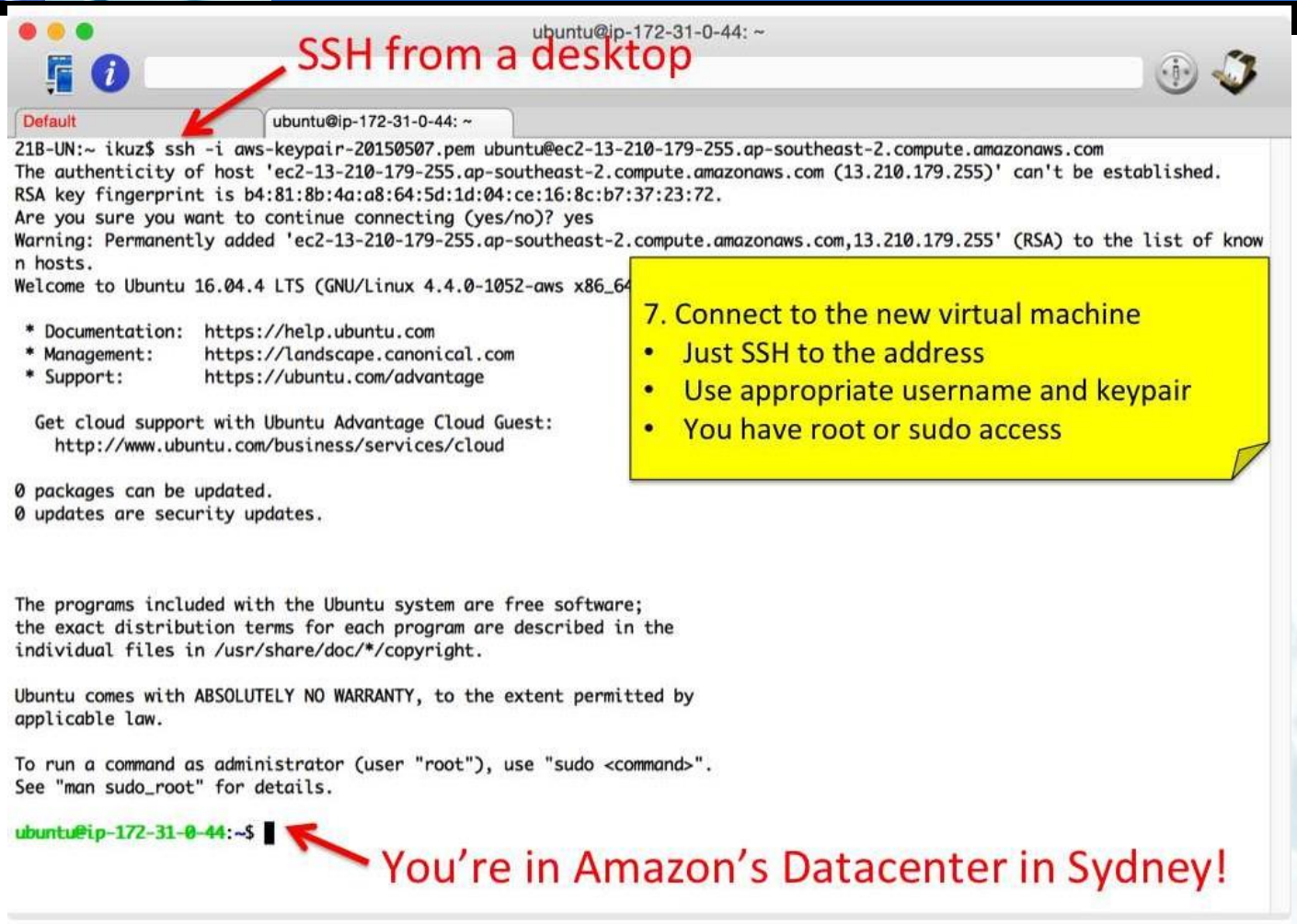
Additional costs for other software (e.g. SQL Server)



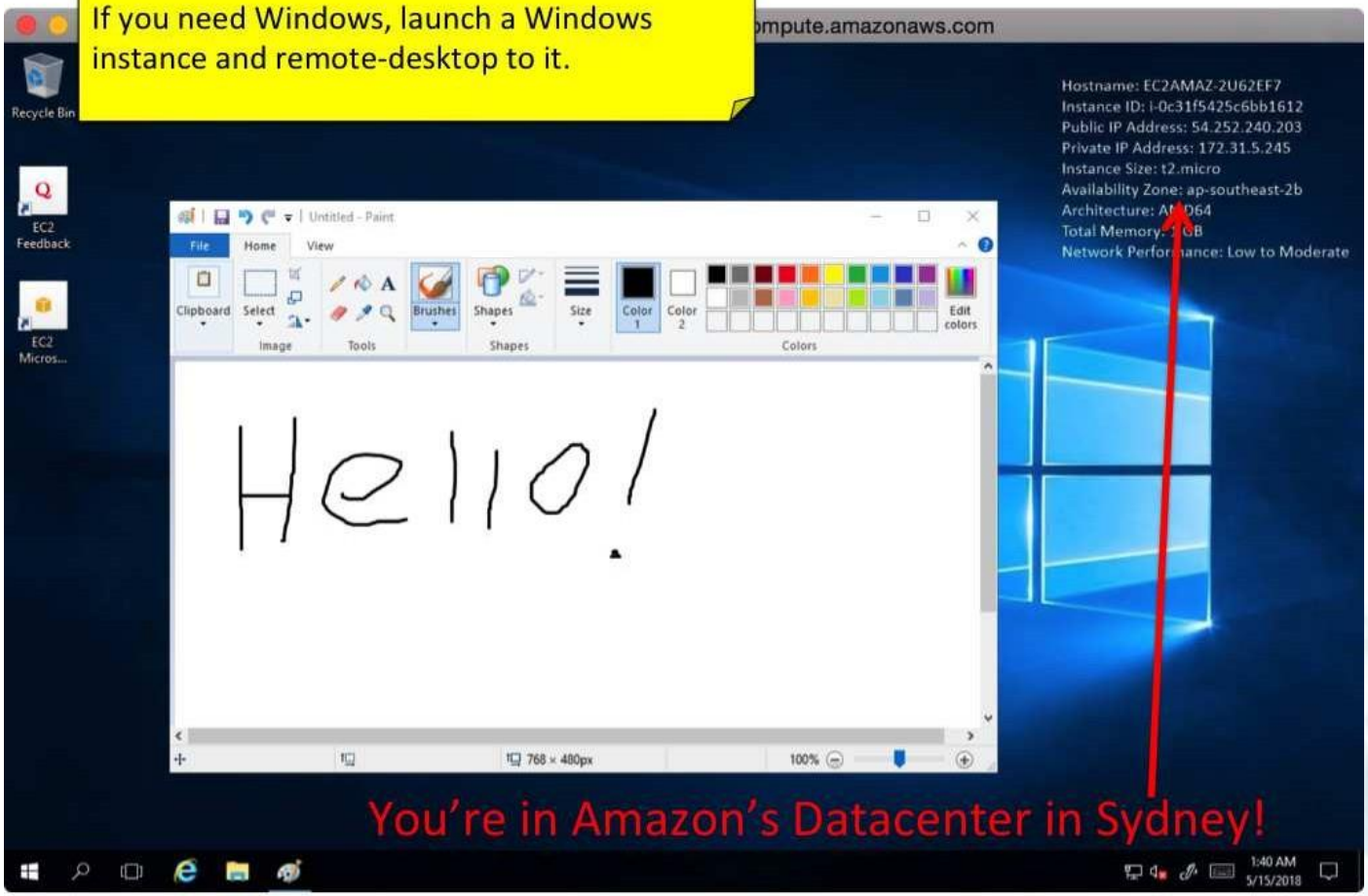
USING EC2


6. Done! (< 5 minutes in total)
• Set SSH key, configure firewall, etc.
• Each machine has a randomly assigned pblic IP address and DNS name, e.g.:
ec2-54-252-240-203.ap-southeast-2.compute.amazonaws.com
EC2 Management Console
Launch Instance
Connect
Actions
Instance ID : i-0720818ae0d9b98ec
Add filter
1 to 1 of 1
Name
Instance ID
Instance Type
Availability Zone
Instance State
Status Checks
Alarm Status
i-0720818a...
t2.medium
ap-southeast-2b
running
Initializing
None
Instance: i-0720818ae0d9b98ec
Public DNS: ec2-13-210-179-255.ap-southeast-2.compute.amazonaws.com
I got my instance!
Description
Status Checks
Monitoring
Tags
Instance ID i-0720818ae0d9b98ec
Instance state running
Instance type t2.medium
Elastic IPs
Availability zone ap-southeast-2b
Security groups launch-wizard-1. view inbound rules
Scheduled events No scheduled events
AMI ID ubuntu/images/hvm-ssd/ubuntu-xenial-16.04-amd64-server-20180306 (ami-d38a4ab1)
Public DNS (IPv4) ec2-13-210-179-255.ap-southeast-2.compute.amazonaws.com
IPv4 Public IP 13.210.179.255
IPv6 IPs -
Private DNS ip-172-31-0-44.ap-southeast-2.compute.internal
Private IPs 172.31.0.44
Secondary private IPs
VPC ID vpc-7a0dd91f
Subnet ID subnet-b8e14edd
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

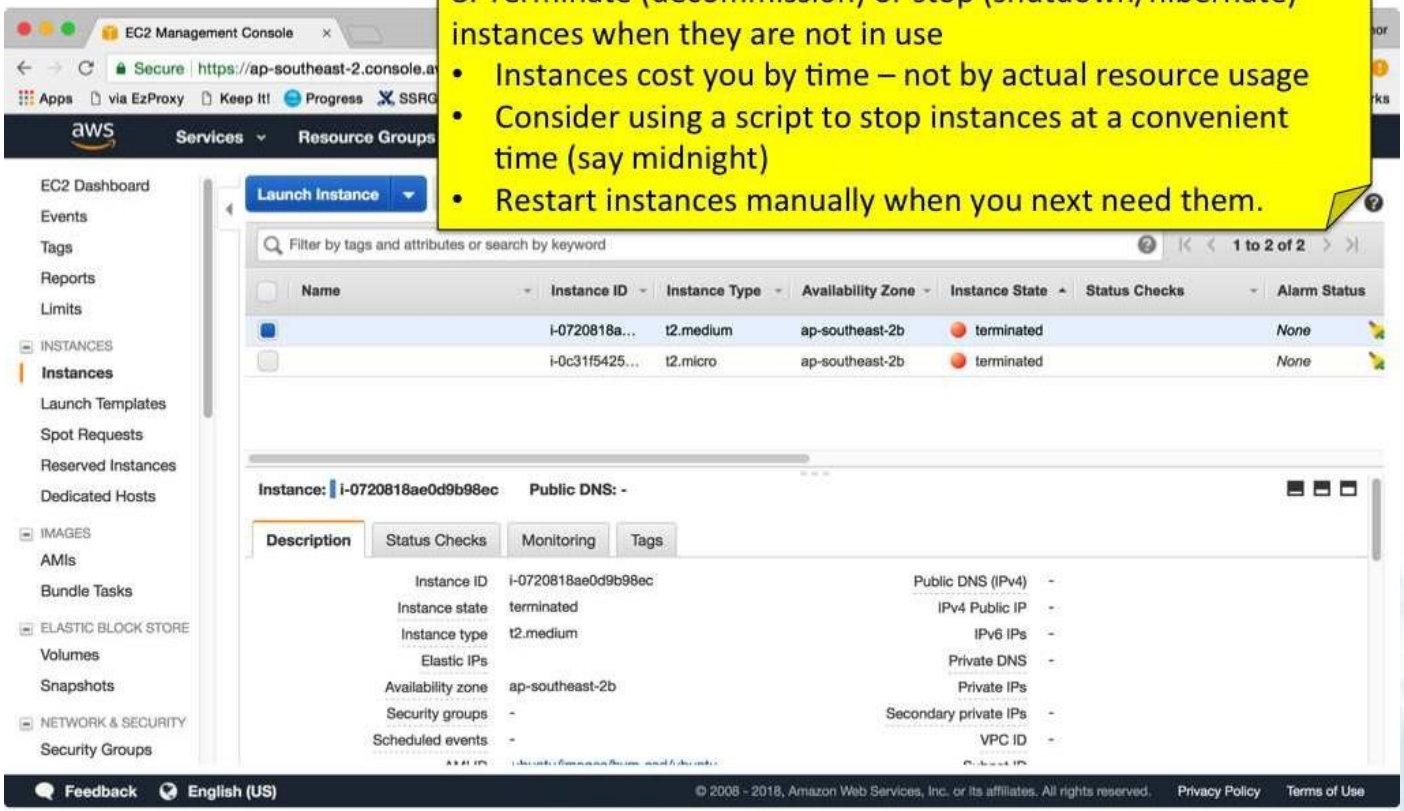
USING EC2


ubuntu@ip-172-31-0-44: ~
SSH from a desktop
Default
ubuntu@ip-172-31-0-44: ~
21B-UN:~ ikuz$ ssh -i aws-keypair-20150507.pem ubuntu@ec2-13-210-179-255.ap-southeast-2.compute.amazonaws.com
The authenticity of host 'ec2-13-210-179-255.ap-southeast-2.compute.amazonaws.com (13.210.179.255)' can't be established.
RSA key fingerprint is b4:81:8b:4a:a8:64:5d:1d:04:ce:16:8c:b7:37:23:72.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-13-210-179-255.ap-southeast-2.compute.amazonaws.com,13.210.179.255' (RSA) to the list of know
n hosts.
Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.4.0-1052-aws x86_64
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage
Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud
0 packages can be updated.
0 updates are security updates.
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
ubuntu@ip-172-31-0-44:~$
7. Connect to the new virtual machine
• Just SSH to the address
• Use appropriate username and keypair
• You have root or sudo access
You're in Amazon's Datacenter in Sydney!

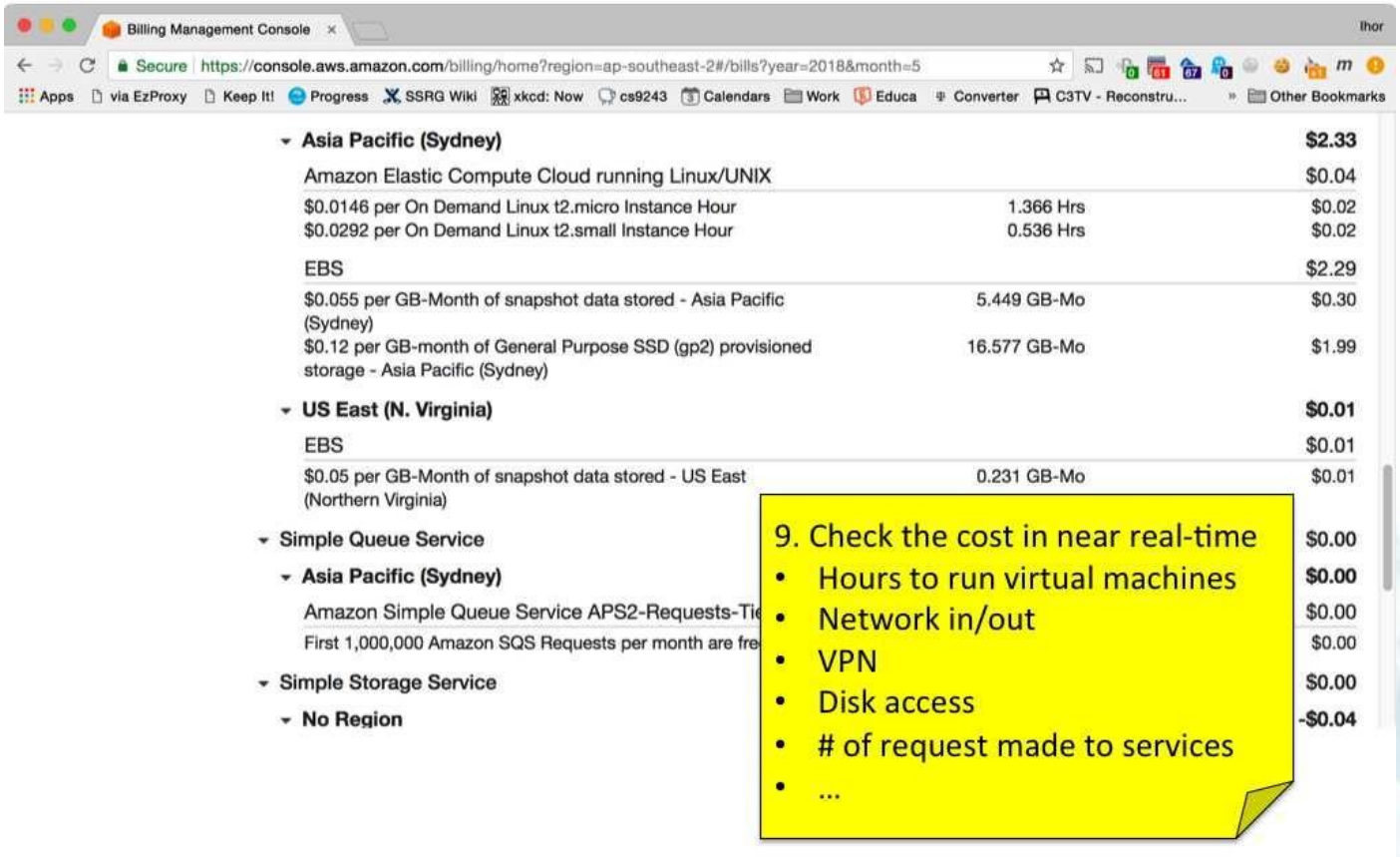
USING EC2


If you need Windows, launch a Windows instance and remote-desktop to it.
Hostname: EC2AMAZ-2U62EF7
Instance ID: i-0c31f5425c6bb1612
Public IP Address: 54.252.240.203
Private IP Address: 172.31.5.245
Instance Size: t2.micro
Availability Zone: ap-southeast-2b
Network Performance: Low to Moderate
Hello!
You're in Amazon's Datacenter in Sydney!

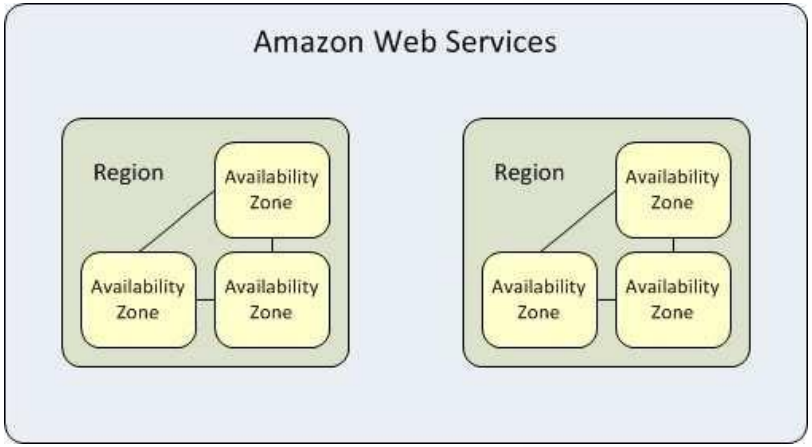
USING EC2


8. Terminate (decommission) or stop (shutdown/hibernate) instances when they are not in use
• Instances cost you by time – not by actual resource usage
• Consider using a script to stop instances at a convenient time (say midnight)
• Restart instances manually when you next need them.
EC2 Management Console
Launch Instance
Filter by tags and attributes or search by keyword
1 to 2 of 2
Name
Instance ID
Instance Type
Availability Zone
Instance State
Status Checks
Alarm Status
i-0720818a...
t2.medium
ap-southeast-2b
terminated
None
i-0c31f5425...
t2.micro
ap-southeast-2b
terminated
None
Instance: i-0720818ae0d9b98ec
Public DNS: -
Description
Status Checks
Monitoring
Tags
Instance ID
i-0720818ae0d9b98ec
Instance state
terminated
Instance type
t2.medium
Elastic IPs
Availability zone
ap-southeast-2b
Security groups
Scheduled events
Public DNS (IPv4)
IPv4 Public IP
IPv6 IPs
Private DNS
Private IPs
Secondary private IPs
VPC ID
EC2 Dashboard
Events
Tags
Reports
Limits
INSTANCES
Instances
Launch Templates
Spot Requests
Reserved Instances
Dedicated Hosts
IMAGES
AMIs
Bundle Tasks
ELASTIC BLOCK STORE
Volumes
Snapshots
NETWORK & SECURITY
Security Groups
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

USING EC2


Billing Management Console
Secure https://console.aws.amazon.com/billing/home?region=ap-southeast-2#/bills?year=2018&month=5
Apps via EzProxy Keep It! Progress SSRG Wiki xkcd: Now cs9243 Calendars Work Educa Converter C3TV - Reconstru... Other Bookmarks
Asia Pacific (Sydney) $2.33
Amazon Elastic Compute Cloud running Linux/UNIX $0.04
$0.0146 per On Demand Linux t2.micro Instance Hour 1.366 Hrs $0.02
$0.0292 per On Demand Linux t2.small Instance Hour 0.536 Hrs $0.02
EBS $2.29
$0.055 per GB-Month of snapshot data stored - Asia Pacific (Sydney) 5.449 GB-Mo $0.30
$0.12 per GB-month of General Purpose SSD (gp2) provisioned storage - Asia Pacific (Sydney) 16.577 GB-Mo $1.99
US East (N. Virginia) $0.01
EBS $0.01
$0.05 per GB-Month of snapshot data stored - US East (Northern Virginia) 0.231 GB-Mo $0.01
Simple Queue Service $0.00
Asia Pacific (Sydney) $0.00
Amazon Simple Queue Service APS2-Requests-Tie $0.00
First 1,000,000 Amazon SQS Requests per month are fre $0.00
Simple Storage Service $0.00
No Region -$0.04
9. Check the cost in near real-time
• Hours to run virtual machines
• Network in/out
• VPN
• Disk access
• # of request made to services
• ...

RELIABILITY



http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-regions-availability-zones.html

Regions and Availability Zones:

- ➜ 99.95% availability per service region
- ➜ Regions: geographically dispersed, independent
- ➜ Availability zones: contained in Regions
- ➜ Availability zones: isolated from failures in other zones, but connected

RELIABILITY

- ➜ IP address associated with account
- ➜ Dynamic remapping to specific instances

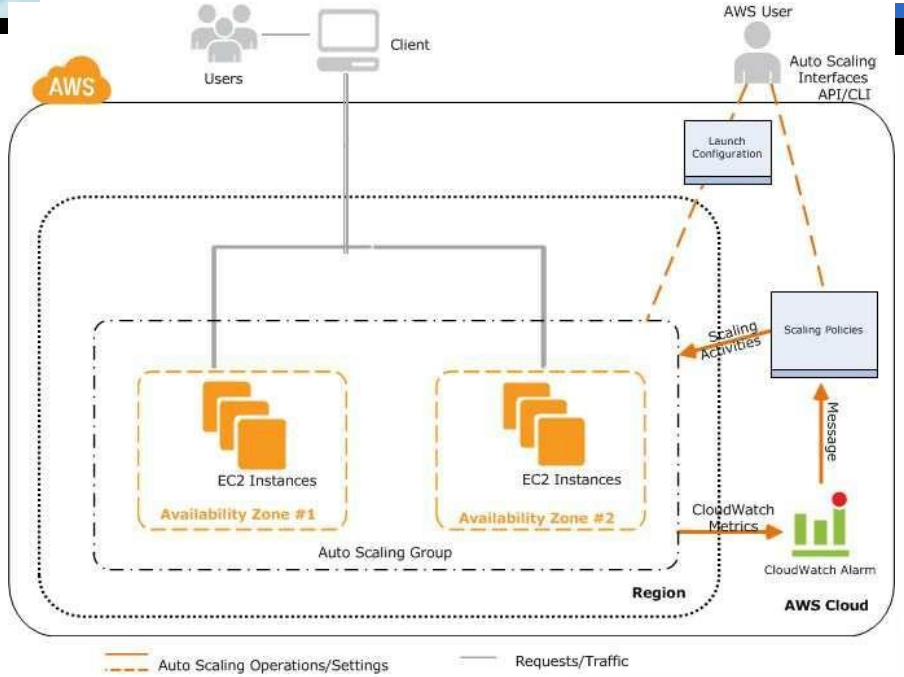
•instance has *private IP address* and *public IP address*

•*Elastic IP* can be mapped (and re-mapped) to private IP

Elastic Load Balancing:

- ➜ Distributes traffic across instances
- ➜ Monitors 'health' of instances: customisable
- ➜ Routes to healthy instances

RELIABILITY



Auto Scaling:

➜ Automatically start or stop new instances

➜ User-defined conditions

•manual (minimum group size), schedule

•instance health, CloudWatch input

https://docs.aws.amazon.com/autoscaling/ec2/userguide/what-is-amazon-ec2-auto- scaling.html

RELIABILITY

➜ **Infrastructure Security**

- **Data centre physical security**
- **Software and hardware maintenance**
- **Monitoring and Testing (automatic and manual)**

➜ **Application Security**

- **API access control (access keys)**
- **Firewall settings for instances (security groups)**
- **Virtual Private Cloud (VPC): private or public subnetworks**
- **Encrypted storage support**
- **Logging**

STORAGE

Elastic Block Store:

➜Network Attached Storage (NAS) (servers with disks)

➜Block level storage volumes

➜Mounted as block device (e.g. disk) on an instance

➜Physical Servers and Disks shared by customers (no caching, competing for disk and net IO)

➜Replicated in Availability zone

➜Cost: per GB/per month

STORAGE

➜ **Buckets: store objects**

•**Can be placed in specific regions**

➜ **Objects: data and metadata**

•**metadata: key-value pairs describing the object**

•**identified by key (unique within a bucket)**

•**versioned**

➜ **Consistency:**

•**highly replicated**

•**eventual consistency, no locking**

•**atomic object update**

➜ **Access control**

STORAGE

- ➜ Point in time copy of EBS volume
- ➜ Stored in S3
- ➜ Differential
- ➜ Can be used to bootstrap image

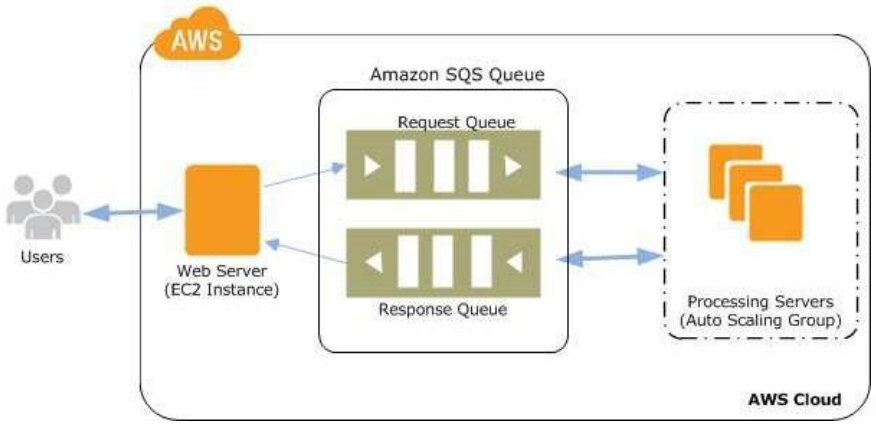
Simple Database Service (SimpleDB):

- ➜ Non-relational database: key-value
- ➜ Partitioned into *domains*
- ➜ Consistency

•highly replicated
•eventual consistency

- ➜ Typical uses: logging, indexing S3 data
- ➜ Erlang!
- ➜ Replaced by DynamoDB

COMMUNICATION



Simple Queue Service (SQS):

➜**Message-queue oriented communication service**

➜**Persistent, asynchronous messaging**

➜**At-least once delivery guarantee**

➜**No ordering guarantee**

➜**Access control**

https://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/

PLATFORM AS A SERVICE

Service provider provides:

- ➜**Hardware infrastructure**
- ➜**OS and platform software (middleware)**
- ➜**Distributed storage management**
- ➜**Load balancing, replication, migration**
- ➜**Management and Monitoring services**

Client provides:

- ➜**Application**

Challenges

- Learn new API and environment
- Follow API
- Optimise to limits of API and platform
- Security for own app

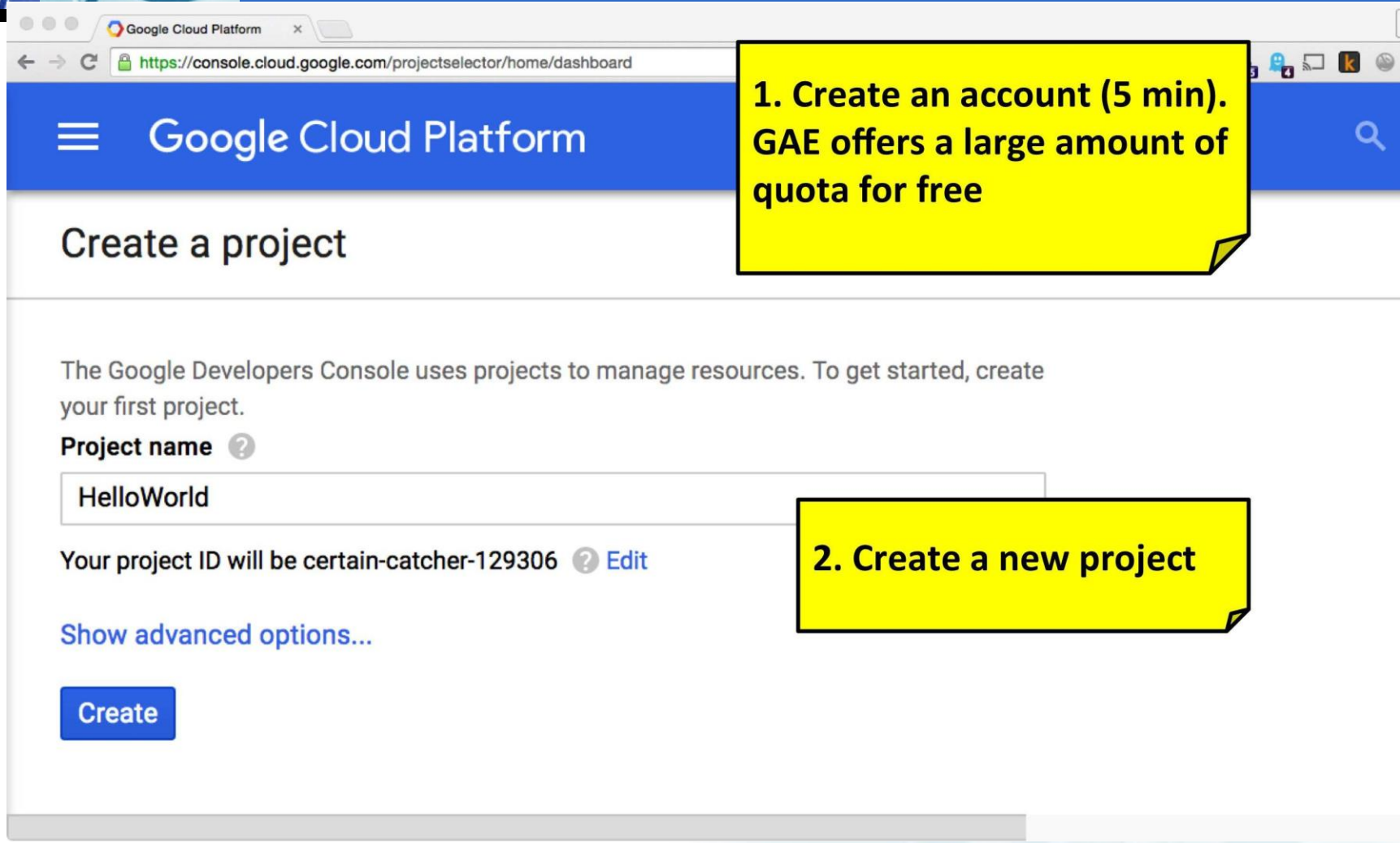
Challenges – Provider:

- Transparency (naming, redirection)
- Scalability: replication and load balancing decisions
- Synchronisation and coordination
- Security
- Fault tolerance
- Monitoring
- Software maintenance and sys admin

EXAMPLE 2: APP ENGINE

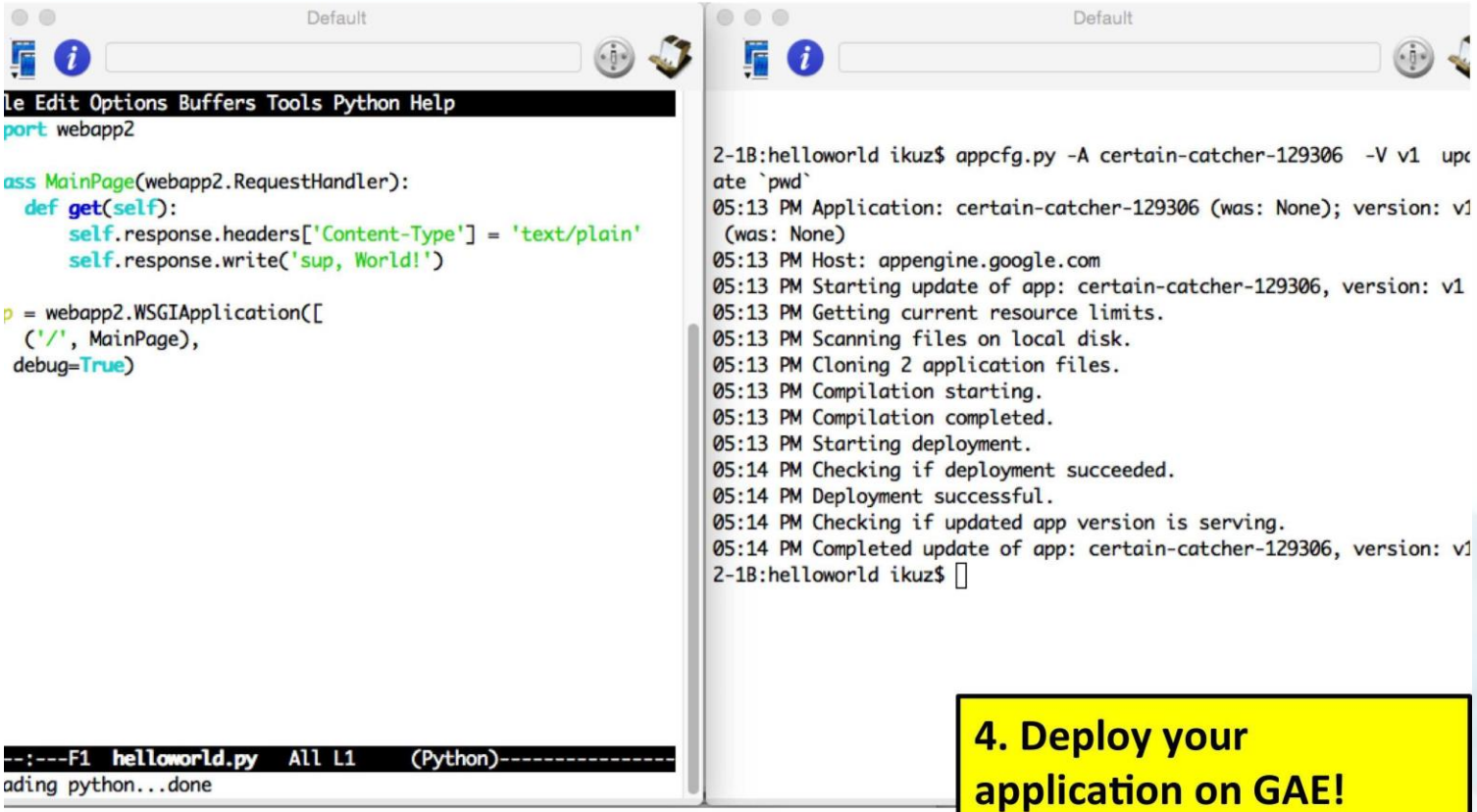
- Various development languages (Python, Java, PHP, Go)
- ... and runtime environments
- Storage based on Big Table
- Optimisation via Memcache
- Lots of APIs
- Per use billing
- Transparent scaling

EXAMPLE 2: APP ENGINE



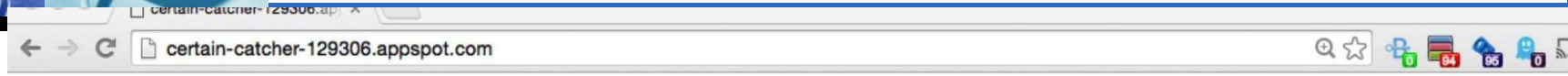
EXAMPLE 2: APP ENGINE

3. Write an application using GAE's framework



4. Deploy your application on GAE!

EXAMPLE 2: APP ENGINE



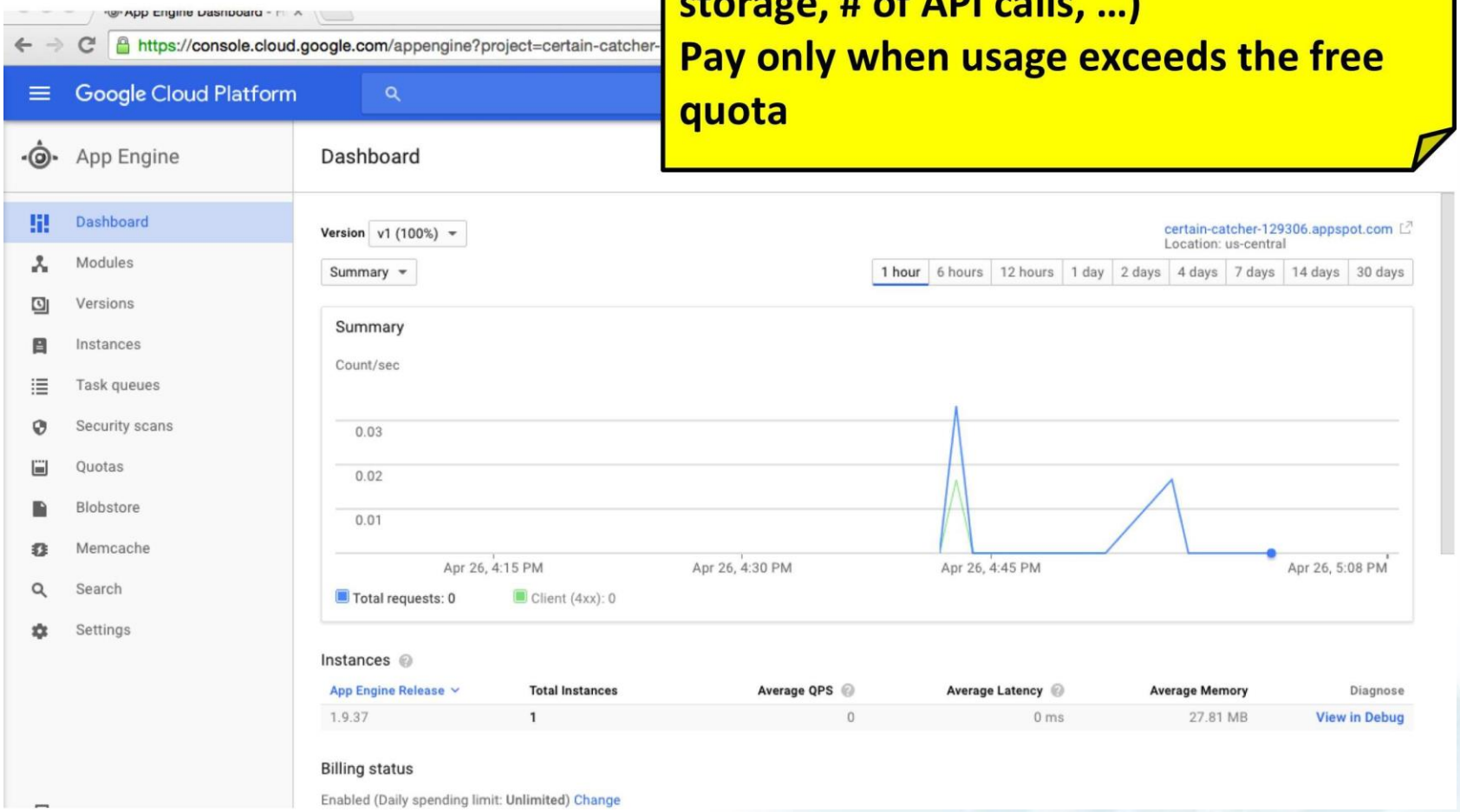
sup, World!

5. Running application.

Scale up/down, load balancing, replication, database management, … many services are provided by GAE.

EXAMPLE 2: APP ENGINE

6. Check your resource usage (CPU, storage, # of API calls, …)
Pay only when usage exceeds the free quota



SOFTWARE AS A SERVICE

Service provider provides:

- ➜Hardware infrastructure
- ➜OS and platform software (middleware)
- ➜Distributed storage management
- ➜Load balancing, replication, migration
- ➜Management and Monitoring services
- ➜Application

Client provides:

- ➜Data

Challenge

- ➜ Learn new application
- ➜ Deal with potential restrictions

•Web interface, restricted functionality

•No offline access, no local storage

Challenges – Provider:

- ➜ Transparency (naming, redirection)
- ➜ Scalability: replication and load balancing decisions
- ➜ Synchronisation and coordination
- ➜ Security
- ➜ Fault tolerance
- ➜ Monitoring
- ➜ Software maintenance and sys admin
- ➜ Application development and maintenance

KEY CHALLENGES OF CLOUD COMPUTING

Scalability:

➜Datacentre vs Global

➜Partitioning

•Services and Data

➜Replication

Consistency:

➜Dealing with consequences of CAP Theorem

➜Dealing with un-usability of eventual consistency

KEY CHALLENGES OF CLOUD COMPUTING

➜ **SLA (Service Level Agreement): guarantees given by provider**

- **How reliable are the guarantees?**
- **What is the consequence if they aren't met?**

➜ **Redundancy and Replication**

- **within same provider (e.g. Availability Zones, Regions, etc.)**
- **migration across providers**

➜ **Geographically distributed architecture**

KEY CHALLENGES OF CLOUD COMPUTING

➜ **Design for failure: Chaos Monkey**

- **test how well system deals with failure**
- **regularly and randomly kill system services**

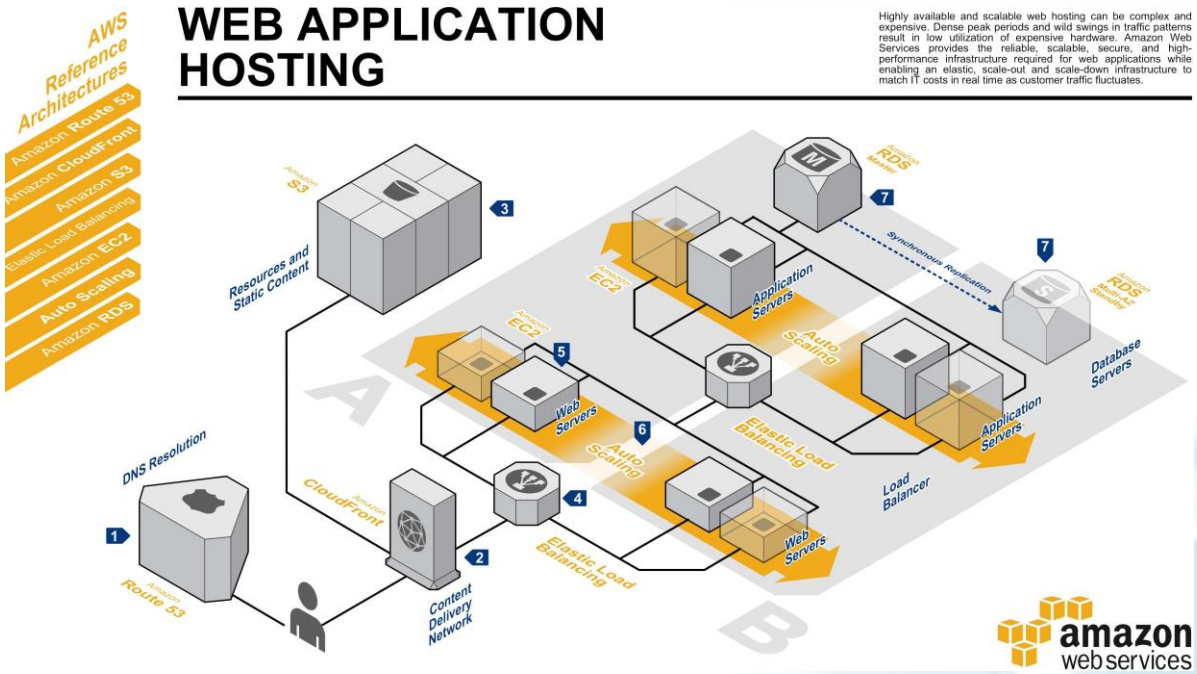
Security and Privacy:

- ➜ **External threats**
- **Denial of Service**
- **Infrastructure or platform service compromise**
- **SaaS compromise: data theft**
- ➜ **Co-located threats: other customers**
- **Isolation: but, covert channels, bugs in isolation**
- ➜ **Privacy: data collected by providers**
- **IaaS and PaaS providers: encryption only helps a bit**
- **SaaS providers: at mercy of service provider**
- **Governments and others: where is your data stored or processed?**

Which laws apply?

DEVELOPING FOR THE CLOUD

Examples from Amazon:



http://aws.amazon.com/architecture/