INGI2145 Cloud Computing Lab 1

Nicolas Laurent

Lecture slides adapted from Upenn NETS212 by A. Haeberlen, Z. Ives
Reproduced with permission

The INGI2145 Virtual Machine

- The course virtual machine has the necessary software for the homework assignments
  - Linux, Eclipse IDE, Hadoop, Amazon CLI, …
- We now demonstrate how to provision the VM in VirtualBox through Vagrant and Puppet
  - A copy of the VM is also available on a USB stick
  - If you run into problems, please post a message to the discussion group
    - Please avoid sending questions via email - if you post your question on Piazza, everyone will be able to see the answer!

Plan for today

- Provision INGI2145 Virtual Machines
- Introduction to Amazon Web Services (AWS)
  - Elastic Compute Cloud (EC2) NEXT
  - Elastic Block Storage (EBS)
  - Other services: Mechanical Turk, CloudFront, ...
  - Next time: S3 and DynamoDB
- Hands on session

Insert your favorite cloud here

Why Amazon AWS and not □?

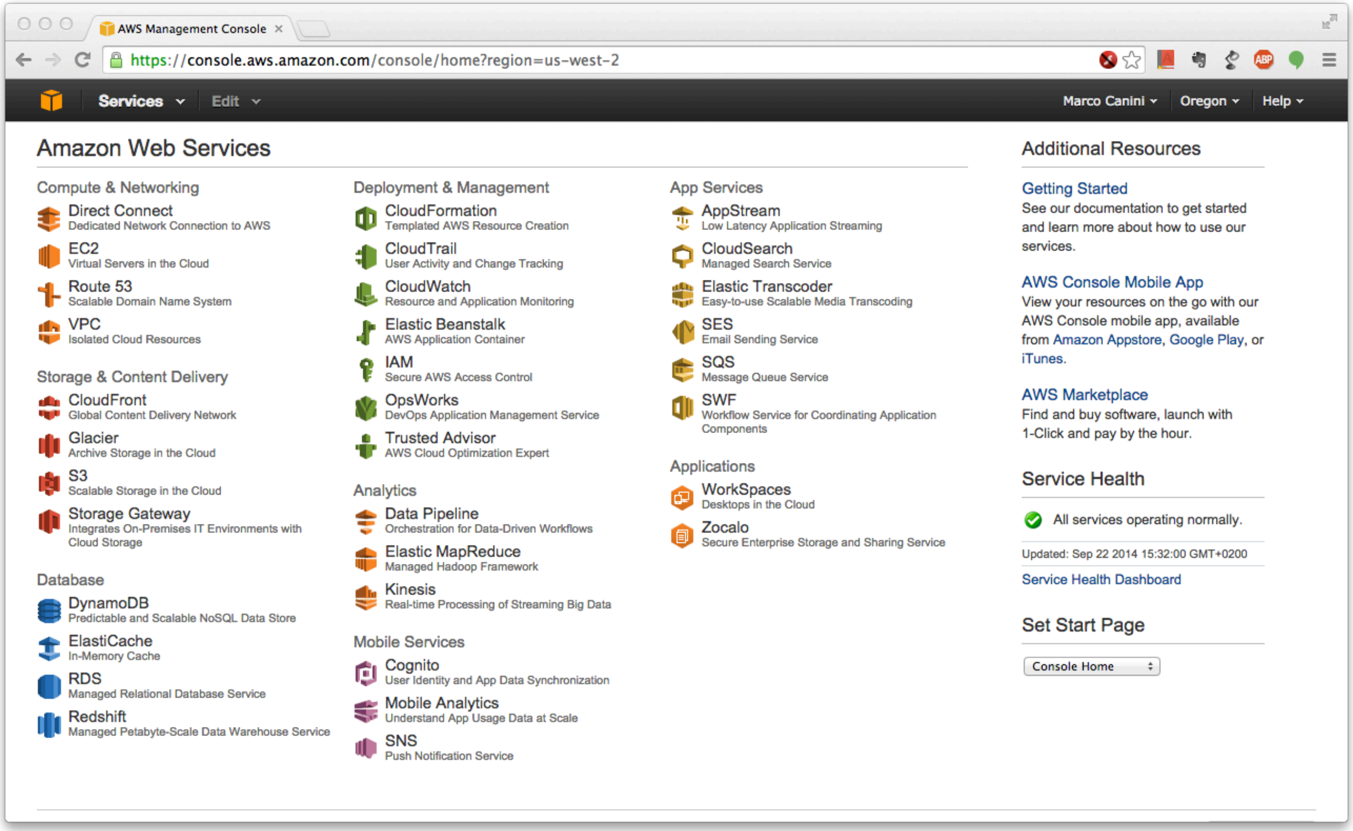
- Amazon is only one of several cloud providers
  - Others include Microsoft Azure, Google App Engine, ...
- But there is no common standard (yet)
  - App Engine is PaaS and supports Java/JVM or Python
  - Azure is PaaS and supports .NET/CLR
  - AWS is PaaS/IaaS and supports IA-32 virtual machines
- So we had to pick one specific provider
  - Amazon AWS is going to be used for the rest of this class
  - Full disclosure: Amazon's only involvement is providing free EC2 access for this class

What is Amazon AWS?

- Amazon Web Services (AWS) provides a number of different services, including:
  - Amazon Elastic Compute Cloud (EC2)
    Virtual machines for running custom software
  - Amazon Simple Storage Service (S3)
    Simple key-value store, accessible as a web service
  - Amazon DynamoDB
    Simple distributed database
  - Amazon Elastic MapReduce
    Scalable MapReduce computation

    Used for the projects
  - Amazon Mechanical Turk (MTurk)
    A 'marketplace for work'
  - Amazon CloudFront
    Content delivery network
  - ...

The AWS management console


AWS Management Console
https://console.aws.amazon.com/console/home?region=us-west-2
Services
Edit
Marco Canini
Oregon
Help

Amazon Web Services

Compute & Networking

- Direct Connect – Dedicated Network Connection to AWS
- EC2 – Virtual Servers in the Cloud
- Route 53 – Scalable Domain Name System
- VPC – Isolated Cloud Resources

Storage & Content Delivery

- CloudFront – Global Content Delivery Network
- Glacier – Archive Storage in the Cloud
- S3 – Scalable Storage in the Cloud
- Storage Gateway – Integrates On-Premises IT Environments with Cloud Storage

Database

- DynamoDB – Predictable and Scalable NoSQL Data Store
- ElastiCache – In-Memory Cache
- RDS – Managed Relational Database Service
- Redshift – Managed Petabyte-Scale Data Warehouse Service

Deployment & Management

- CloudFormation – Templated AWS Resource Creation
- CloudTrail – User Activity and Change Tracking
- CloudWatch – Resource and Application Monitoring
- Elastic Beanstalk – AWS Application Container
- IAM – Secure AWS Access Control
- OpsWorks – DevOps Application Management Service
- Trusted Advisor – AWS Cloud Optimization Expert

Analytics

- Data Pipeline – Orchestration for Data-Driven Workflows
- Elastic MapReduce – Managed Hadoop Framework
- Kinesis – Real-time Processing of Streaming Big Data

Mobile Services

- Cognito – User Identity and App Data Synchronization
- Mobile Analytics – Understand App Usage Data at Scale
- SNS – Push Notification Service

App Services

- AppStream – Low Latency Application Streaming
- CloudSearch – Managed Search Service
- Elastic Transcoder – Easy-to-use Scalable Media Transcoding
- SES – Email Sending Service
- SQS – Message Queue Service
- SWF – Workflow Service for Coordinating Application Components

Applications

- WorkSpaces – Desktops in the Cloud
- Zocalo – Secure Enterprise Storage and Sharing Service

Additional Resources

Getting Started
See our documentation to get started and learn more about how to use our services.

AWS Console Mobile App
View your resources on the go with our AWS Console mobile app, available from Amazon Appstore, Google Play, or iTunes.

AWS Marketplace
Find and buy software, launch with 1-Click and pay by the hour.

Service Health

✓ All services operating normally.

Updated: Sep 22 2014 15:32:00 GMT+0200

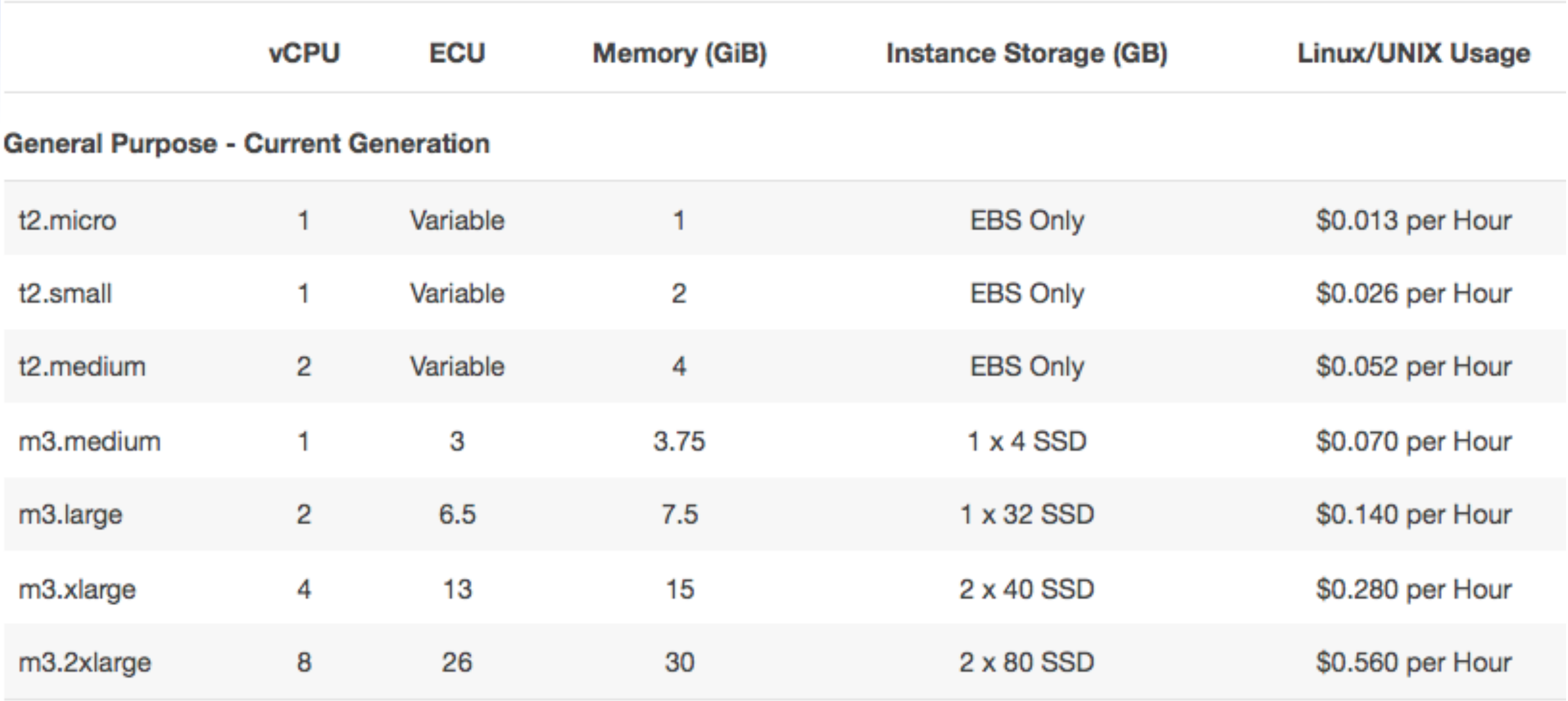
Service Health Dashboard

Set Start Page

Console Home

- Used to control many AWS services:
  - For example, start/stop EC2 instances, create S3 buckets...

What is Amazon EC2?



	vCPU	ECU	Memory (GiB)	Instance Storage (GB)	Linux/UNIX Usage
General Purpose - Current Generation					
t2.micro	1	Variable	1	EBS Only	$0.013 per Hour
t2.small	1	Variable	2	EBS Only	$0.026 per Hour
t2.medium	2	Variable	4	EBS Only	$0.052 per Hour
m3.medium	1	3	3.75	1 x 4 SSD	$0.070 per Hour
m3.large	2	6.5	7.5	1 x 32 SSD	$0.140 per Hour
m3.xlarge	4	13	15	2 x 40 SSD	$0.280 per Hour
m3.2xlarge	8	26	30	2 x 80 SSD	$0.560 per Hour

http://aws.amazon.com/ec2/#pricing (09/2014)

- Infrastructure-as-a-Service (IaaS)
  - You can rent various types of virtual machines by the hour
  - In your VMs, you can run your own (Linux/Windows) programs
    - Examples: Web server, search engine, movie renderer, ...

Example: REST

Invoked method

Parameters

Credentials

```
https://ec2.amazonaws.com/?Action=DescribeKeyPairs
&KeyName.1=my-key-pair
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAM …
&X-Amz-Date=20130813T150206Z
&X-Amz-SignedHeaders=content-type%3host%3x-amz-date
&X-Amz-Signature=ced6826de92d2bdeed8f8 …
```

Sample request

```
<DescribeKeyPairsResponse xmlns="http://ec2.amazonaws.com/doc/2014-06-15/">
<requestId>59dbff89-35bd-4eac-99ed-be587EXAMPLE</requestId>
<keySet>
   <item>
      <keyName>my-key-pair</keyName>
      <keyFingerprint>1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f</keyFingerprint>
   </item>
</keySet>
</DescribeKeyPairsResponse>
```

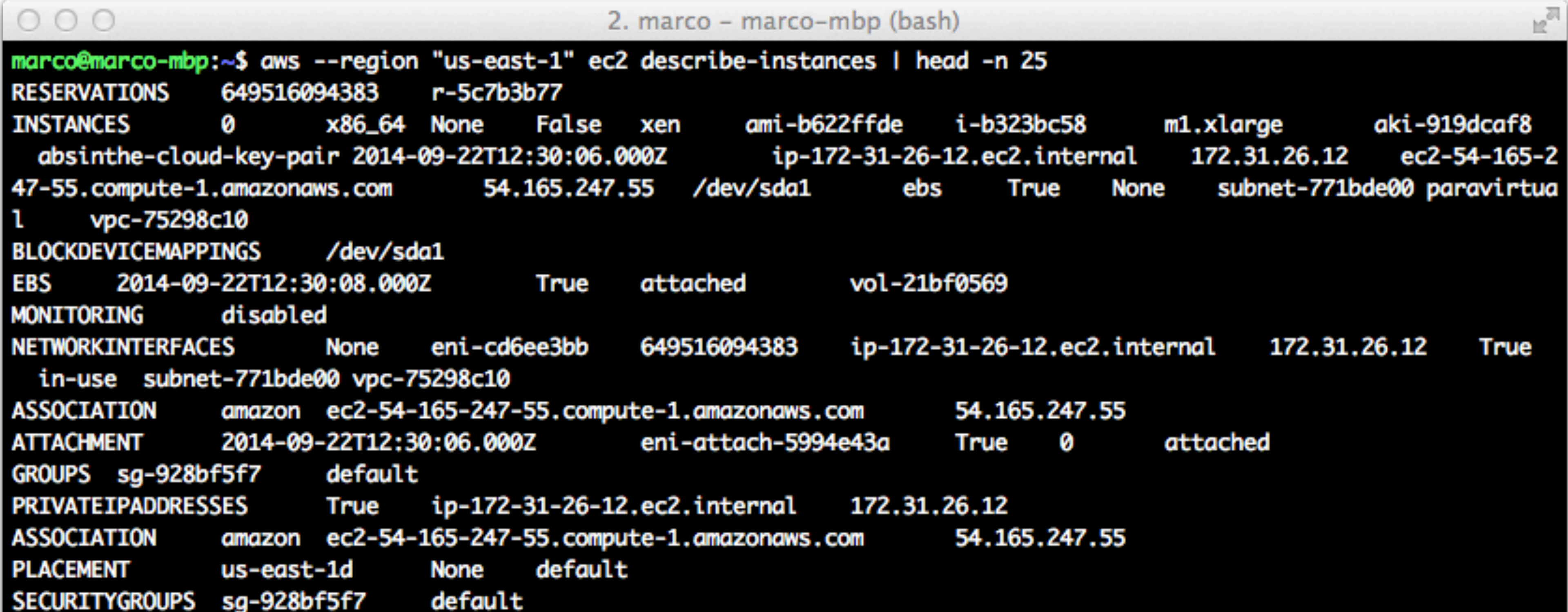
Response elements

Sample response

Source: http://awsdocs.s3.amazonaws.com/SDB/latest/sdb-dg.pdf

Accessing AWS via CLI

- Example of access via Command Line Interface
  - http://aws.amazon.com/cli/



```
marco@marco-mbp:~$ aws --region "us-east-1" ec2 describe-instances | head -n 25
RESERVATIONS    649516094383    r-5c7b3b77
INSTANCES       0       x86_64  None    False   xen     ami-b622ffde    i-b323bc58      m1.xlarge       aki-919dcaf8
  absinthe-cloud-key-pair 2014-09-22T12:30:06.000Z        ip-172-31-26-12.ec2.internal    172.31.26.12    ec2-54-165-2
47-55.compute-1.amazonaws.com       54.165.247.55   /dev/sda1       ebs     True    None    subnet-771bde00 paravirtua
l      vpc-75298c10
BLOCKDEVICEMAPPINGS     /dev/sda1
EBS     2014-09-22T12:30:08.000Z        True    attached        vol-21bf0569
MONITORING      disabled
NETWORKINTERFACES       None    eni-cd6ee3bb    649516094383    ip-172-31-26-12.ec2.internal    172.31.26.12    True
  in-use  subnet-771bde00 vpc-75298c10
ASSOCIATION     amazon  ec2-54-165-247-55.compute-1.amazonaws.com       54.165.247.55
ATTACHMENT      2014-09-22T12:30:06.000Z        eni-attach-5994e43a     True    0       attached
GROUPS  sg-928bf5f7     default
PRIVATEIPADDRESSES      True    ip-172-31-26-12.ec2.internal    172.31.26.12
ASSOCIATION     amazon  ec2-54-165-247-55.compute-1.amazonaws.com       54.165.247.55
PLACEMENT       us-east-1d      None    default
SECURITYGROUPS  sg-928bf5f7     default
```

Oh no - where has my data gone?

- EC2 instances do not have persistent storage
  - Data survives stops & reboots, but not termination

If you store data on the virtual hard disk of your instance
and the instance fails or you terminate it,
your data WILL be lost!

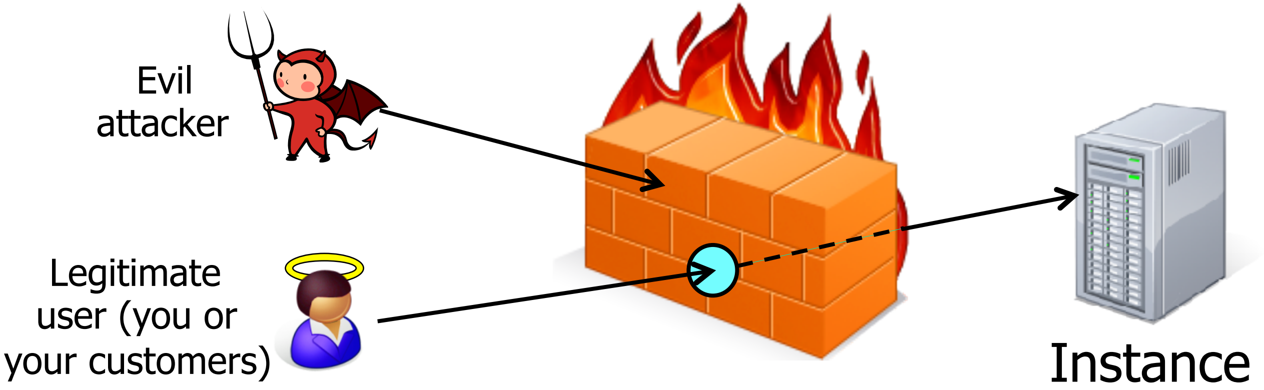
- So where should I put persistent data?
  - Elastic Block Store (EBS) - in a few slides
  - Ideally, use an AMI with an EBS root (Amzon's default AMI has this property)

Amazon Machine Images

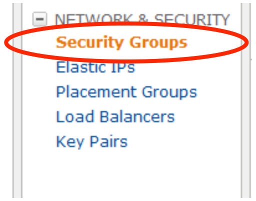
- When I launch an instance, what software will be installed on it?
  - Software is taken from an Amazon Machine Image (AMI)
  - Selected when you launch an instance
  - Essentially a file system that contains the operating system, applications, and potentially other data
  - Lives in S3

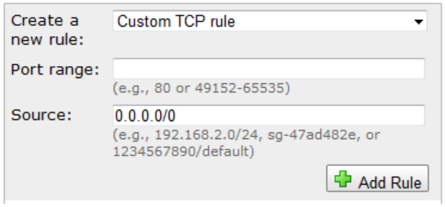
- How do I get an AMI?
  - Amazon provides several generic ones, e.g., Amazon Linux, Fedora Core, Windows Server, ...
  - You can make your own
    - You can even run your own custom kernel (with some restrictions)

Security Groups



- Basically, a set of firewall rules
  - Can be applied to groups of EC2 instances
  - Each rule specifies a protocol, port numbers, etc...
  - Only traffic matching one of the rules is allowed through
- Sometimes need to explicitly open ports



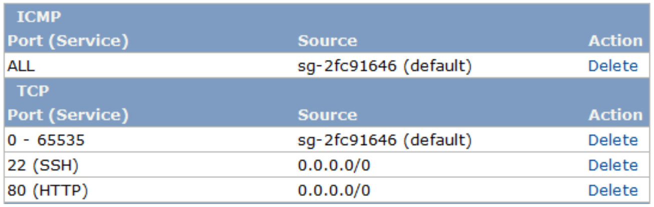


Create a new rule: Custom TCP rule

Port range:
(e.g., 80 or 49152-65535)

Source: 0.0.0.0/0
(e.g., 192.168.2.0/24, sg-47ad482e, or 1234567890/default)

Add Rule

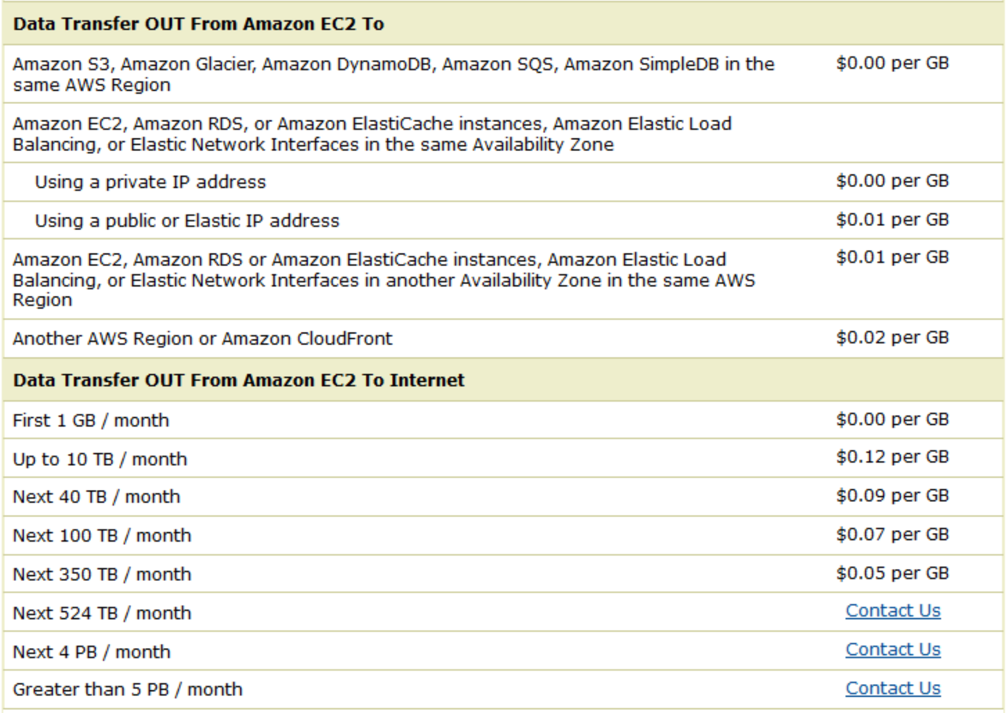


ICMP		
Port (Service)	Source	Action
ALL	sg-2fc91646 (default)	Delete
TCP		
Port (Service)	Source	Action
0 - 65535	sg-2fc91646 (default)	Delete
22 (SSH)	0.0.0.0/0	Delete
80 (HTTP)	0.0.0.0/0	Delete

Regions and Availability Zones

- Where exactly does my instance run?
  - No easy way to find out - Amazon does not say
- Instances can be assigned to regions
  - Currently 9 availble: US East (Northern Virginia), US West (Northern California), US West (Oregon), EU (Ireland), Asia/Pacific (Singapore), Asia/Pacific (Sydney), Asia/Pacific (Tokyo), South America (Sao Paulo), AWS GovCloud
  - Important, e.g., for reducing latency to customers
- Instances can be assigned to availability zones
  - Purpose: Avoid correlated fault
  - Several availability zones within each region

Network pricing

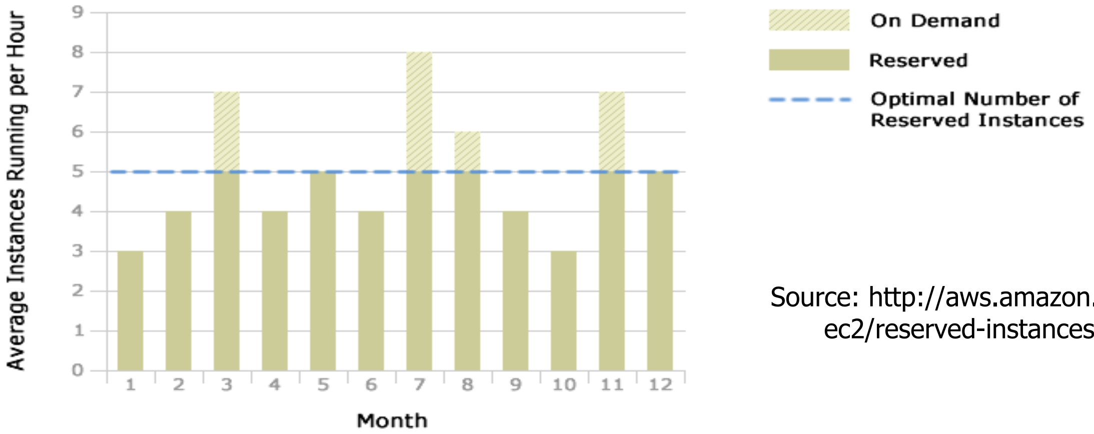


Data Transfer OUT From Amazon EC2 To	
Amazon S3, Amazon Glacier, Amazon DynamoDB, Amazon SQS, Amazon SimpleDB in the same AWS Region	$0.00 per GB
Amazon EC2, Amazon RDS, or Amazon ElastiCache instances, Amazon Elastic Load Balancing, or Elastic Network Interfaces in the same Availability Zone	
Using a private IP address	$0.00 per GB
Using a public or Elastic IP address	$0.01 per GB
Amazon EC2, Amazon RDS or Amazon ElastiCache instances, Amazon Elastic Load Balancing, or Elastic Network Interfaces in another Availability Zone in the same AWS Region	$0.01 per GB
Another AWS Region or Amazon CloudFront	$0.02 per GB
Data Transfer OUT From Amazon EC2 To Internet	
First 1 GB / month	$0.00 per GB
Up to 10 TB / month	$0.12 per GB
Next 40 TB / month	$0.09 per GB
Next 100 TB / month	$0.07 per GB
Next 350 TB / month	$0.05 per GB
Next 524 TB / month	Contact Us
Next 4 PB / month	Contact Us
Greater than 5 PB / month	Contact Us

http://aws.amazon.com/ec2/#pricing (9/11/2013)

- AWS does charge for network traffic
  - Price depends on source and destination of traffic
  - Free within EC2 and other AWS svcs in same region (e.g., S3)
  - Remember: ISPs are typically charged for upstream traffic

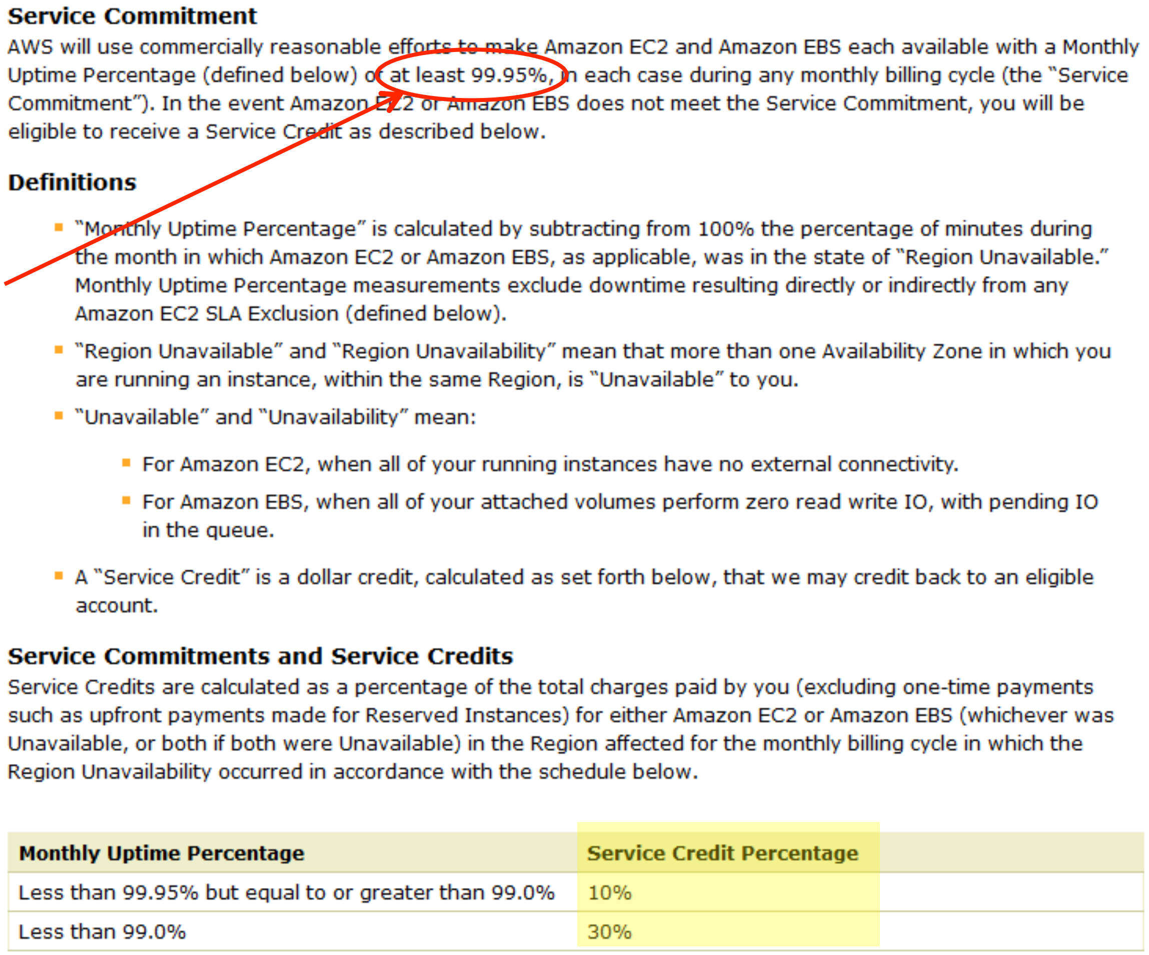
Instance types



Source: http://aws.amazon.com/ec2/reserved-instances/

- So far: On-demand instances
- Also available: Reserved instances
  - One-time reservation fee to purchase for 1 or 3 years
  - Usage still billed by the hour, but at a considerable discount
- Also available: Spot instances
  - Spot market: Can bid for available capacity
  - Instance continues until terminated or price rises above bid

Service Level Agreement



Service Commitment

AWS will use commercially reasonable efforts to make Amazon EC2 and Amazon EBS each available with a Monthly Uptime Percentage (defined below) of at least 99.95%, in each case during any monthly billing cycle (the "Service Commitment"). In the event Amazon EC2 or Amazon EBS does not meet the Service Commitment, you will be eligible to receive a Service Credit as described below.

4.38h downtime per year allowed

Definitions

- "Monthly Uptime Percentage" is calculated by subtracting from 100% the percentage of minutes during the month in which Amazon EC2 or Amazon EBS, as applicable, was in the state of "Region Unavailable." Monthly Uptime Percentage measurements exclude downtime resulting directly or indirectly from any Amazon EC2 SLA Exclusion (defined below).
- "Region Unavailable" and "Region Unavailability" mean that more than one Availability Zone in which you are running an instance, within the same Region, is "Unavailable" to you.
- "Unavailable" and "Unavailability" mean:
  - For Amazon EC2, when all of your running instances have no external connectivity.
  - For Amazon EBS, when all of your attached volumes perform zero read write IO, with pending IO in the queue.
- A "Service Credit" is a dollar credit, calculated as set forth below, that we may credit back to an eligible account.

Service Commitments and Service Credits

Service Credits are calculated as a percentage of the total charges paid by you (excluding one-time payments such as upfront payments made for Reserved Instances) for either Amazon EC2 or Amazon EBS (whichever was Unavailable, or both if both were Unavailable) in the Region affected for the monthly billing cycle in which the Region Unavailability occurred in accordance with the schedule below.

Monthly Uptime Percentage	Service Credit Percentage
Less than 99.95% but equal to or greater than 99.0%	10%
Less than 99.0%	30%

http://aws.amazon.com/ec2-sla/ (1 June 2013; excerpt)

Recap: EC2

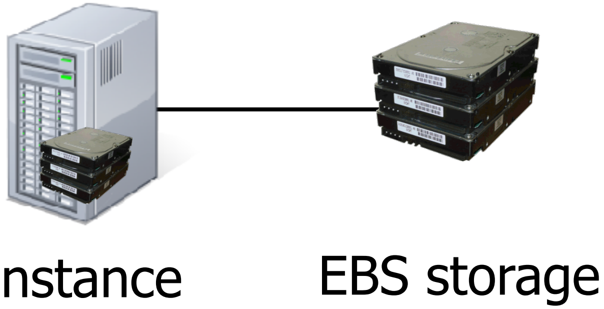
- What EC2 is:
  - IaaS service - you can rent virtual machines
  - Various types: Very small to very powerful

- How to use EC2:
  - Ephemeral state - local data is lost when instance terminates
  - AMIs - used to initialize an instance (OS, applications, ...)
  - Security groups - "firewalls" for your instances
  - Regions and availability zones
  - On-demand/reserved/spot instances
  - Service level agreement (SLA)

Plan for today

- Provision INGI2145 Virtual Machines
- Introduce one specific commercial cloud
  - Amazon Web Services (AWS)
  - Elastic Compute Cloud (EC2)
  - Elastic Block Storage (EBS) NEXT
  - Other services: Mechanical Turk, CloudFront, ...
  - Next time: S3 and DynamoDB
- Hands on session

What is Elastic Block Store (EBS)?



- Persistent storage
  - Unlike the local instance store, data stored in EBS is not lost when an instance fails or is terminated
- Should I use the instance store or EBS?
  - Typically, instance store is used for temporary data

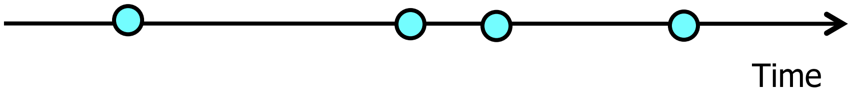
Volumes

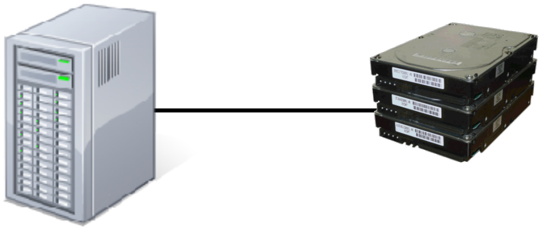
- EBS storage is allocated in volumes
  - A volume is a 'virtual disk' (size: 1GB - 1TB)
  - Basically, a raw block device
  - Can be attached to an instance (but only one at a time)
  - A single instance can access multiple volumes
- Placed in specific availability zones
  - Why is this useful?
  - Be sure to place it near instances (otherwise can't attach)
- Replicated across multiple servers
  - Data is not lost if a single server fails
  - Amazon: Annual failure rate is 0.1-0.5% for a 20GB volume

EC2 instances with EBS roots

- EC2 instances can have an EBS volume as their root device ("EBS boot")
  - Result: Instance data persists independently from the lifetime of the instance
  - You can stop and restart the instance, similar to suspending and resuming a laptop
    - You won't be charged for the instance while it is stopped (only for EBS)
  - You can enable termination protection for the instance
    - Blocks attempts to terminate the instance (e.g., by accident) until termination protection is disabled again
- Alternative: Use instance store as the root
  - You can still store temporary data on it, but it will disappear when you terminate the instance
  - You can still create and mount EBS volumes explicitly

Snapshots





- You can create a snapshot of a volume
  - Copy of data in the volume at the time snapshot was made
  - Only the first snapshot makes a full copy; subsequent snapshots are incremental
- What are snapshots good for?
  - Sharing data with others
    - Access control list (specific account numbers) or public access
  - Instantiate new volumes
  - Point-in-time backups

Pricing

- You pay for...
  - Storage space: $0.10 per allocated GB per month
  - I/O requests: $0.10 per million I/O requests
- Charge is only for actual storage used
  - Empty space does not count

Recap: Elastic Block Store (EBS)

- What EBS is:
  - Basically a virtual hard disk; can be attached to EC2 instances
  - Persistent - state survives termination of EC2 instance

- How to use EBS:
  - Allocate volume - empty or initialized with a snapshot
  - Attach it to EC2 instance and mount it there
  - Can create snapshots for data sharing, backup

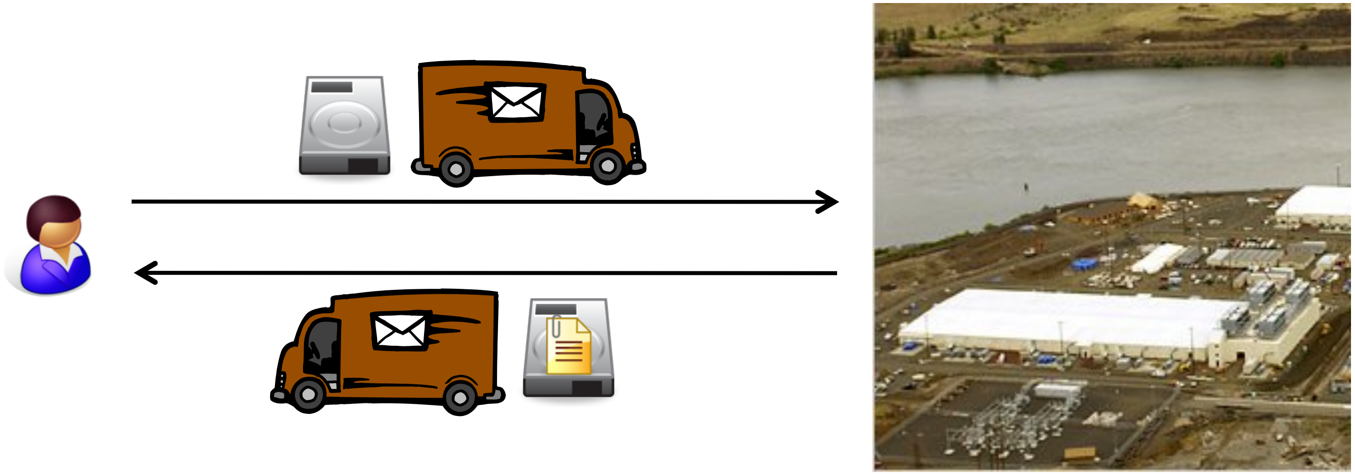
Plan for today

- Provision INGI2145 Virtual Machines
- Introduce one specific commercial cloud
  - Amazon Web Services (AWS)
  - Elastic Compute Cloud (EC2)
  - Elastic Block Storage (EBS)
  - Other services: Mechanical Turk, CloudFront, ... NEXT
  - Next time: S3 and SimpleDB
- Hands on session

AWS Import/Export

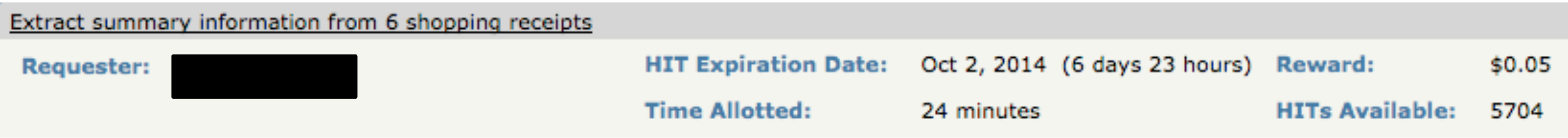
Method	Time
Internet (20Mbps)	45 days
FedEx	1 day

Time to transfer 10TB [AF10]



- Import/export large amounts of data to/from S3 buckets via physical storage device
  - Mail an actual hard disk to Amazon (power adapter, cables!)
  - Signature file for authentication
  - Discussion: Is this the Right Way to be shipping data, or should we rather be using a network?

Mechanical Turk (MTurk)

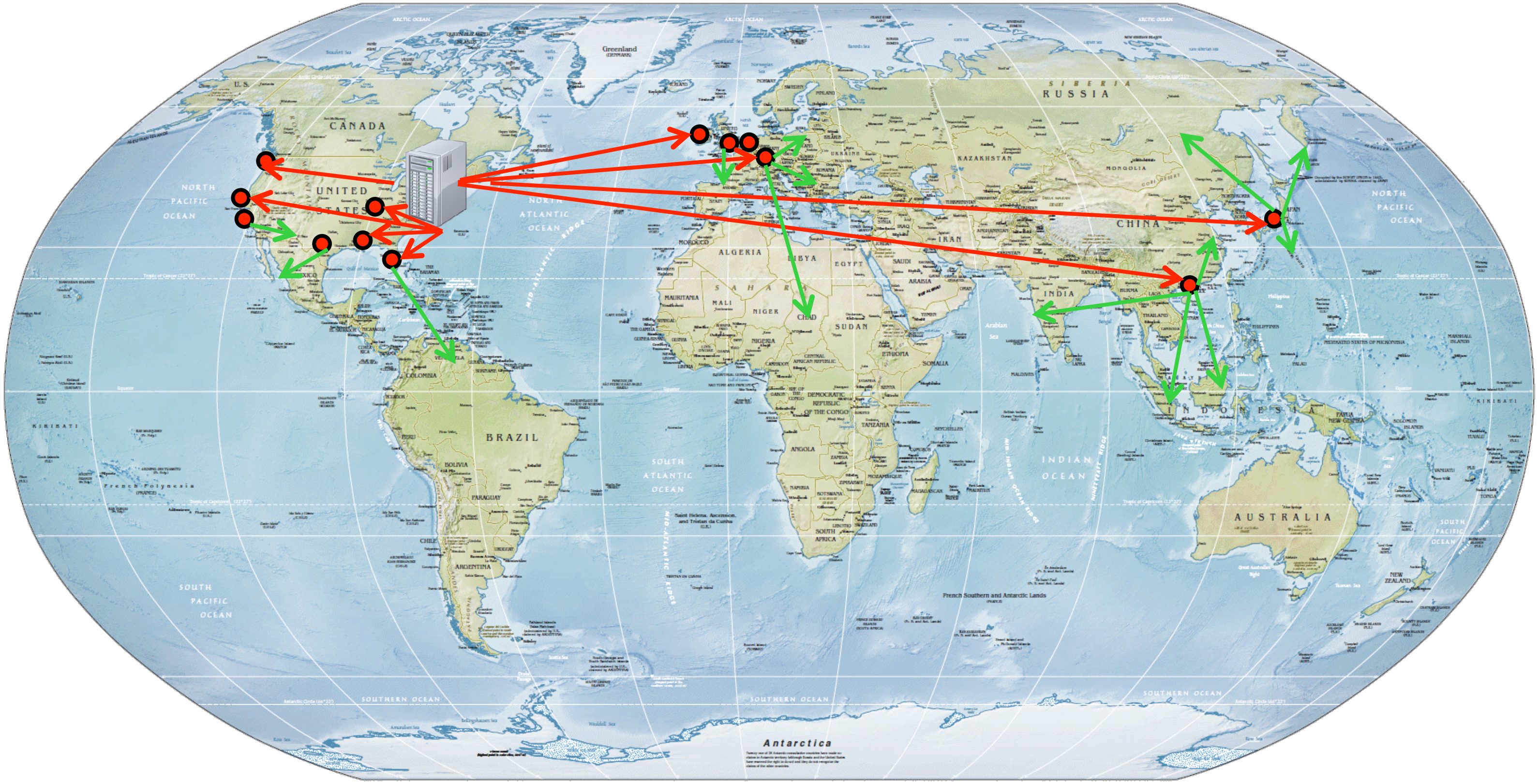


Extract summary information from 6 shopping receipts

Requester:	[redacted]	**HIT Expiration Date:**	Oct 2, 2014 (6 days 23 hours)	**Reward:**	$0.05
		Time Allotted:	24 minutes	**HITs Available:**	5704

- A crowdsourcing marketplace
  - Requesters post small jobs (HIT - Human Intelligence Task), offer small rewards ($0.01-$0.10)

CloudFront



- Content distribution network
  - Caches S3 content at edge locations for low-latency delivery
  - Some similarities to other CDNs like Akamai, Limelight, ...

Plan for today

- Provision INGI2145 Virtual Machines
- Introduce one specific commercial cloud
  - Amazon Web Services (AWS)
  - Elastic Compute Cloud (EC2)
  - Elastic Block Storage (EBS)
  - Other services: Mechanical Turk, CloudFront, ...
  - Next time: S3 and SimpleDB
- Hands on session NEXT

Demo

- Provisioning the VM
- Logging into AWS Management Console
- Launching an instance
- Contacting the instance via ssh
- Creating an EBS volume
- Attaching & mounting the volume on the instance via the CLI
- Terminating an instance
- Details are on your handouts

INGI2145 mandatory conventions

- Instances live in a single namespace
- We need a convention that everyone must follow to share the AWS resources
- 1. Your login on AWS is:
  - ingi2145-<username>
- 2. Tag each virtual machine with a tag
  - key=user
  - value=<username>
- 3. Name your key-pair name as your login
  - ingi2145-<username>
  - NOTE: you need one key-pair in each region!

INGI2145 mandatory conventions

- Use only the following AWS regions
  - us-west-2
  - us-west-1
  - us-east-1

- **Never terminate a virtual machine that you do not own!**

- **Always terminate instances that you don’t need!**
  - Remember each of you has about $100 of credit!