

# CSE 344 Introduction to Data Management

Section 9: AWS, Hadoop, Pig Latin

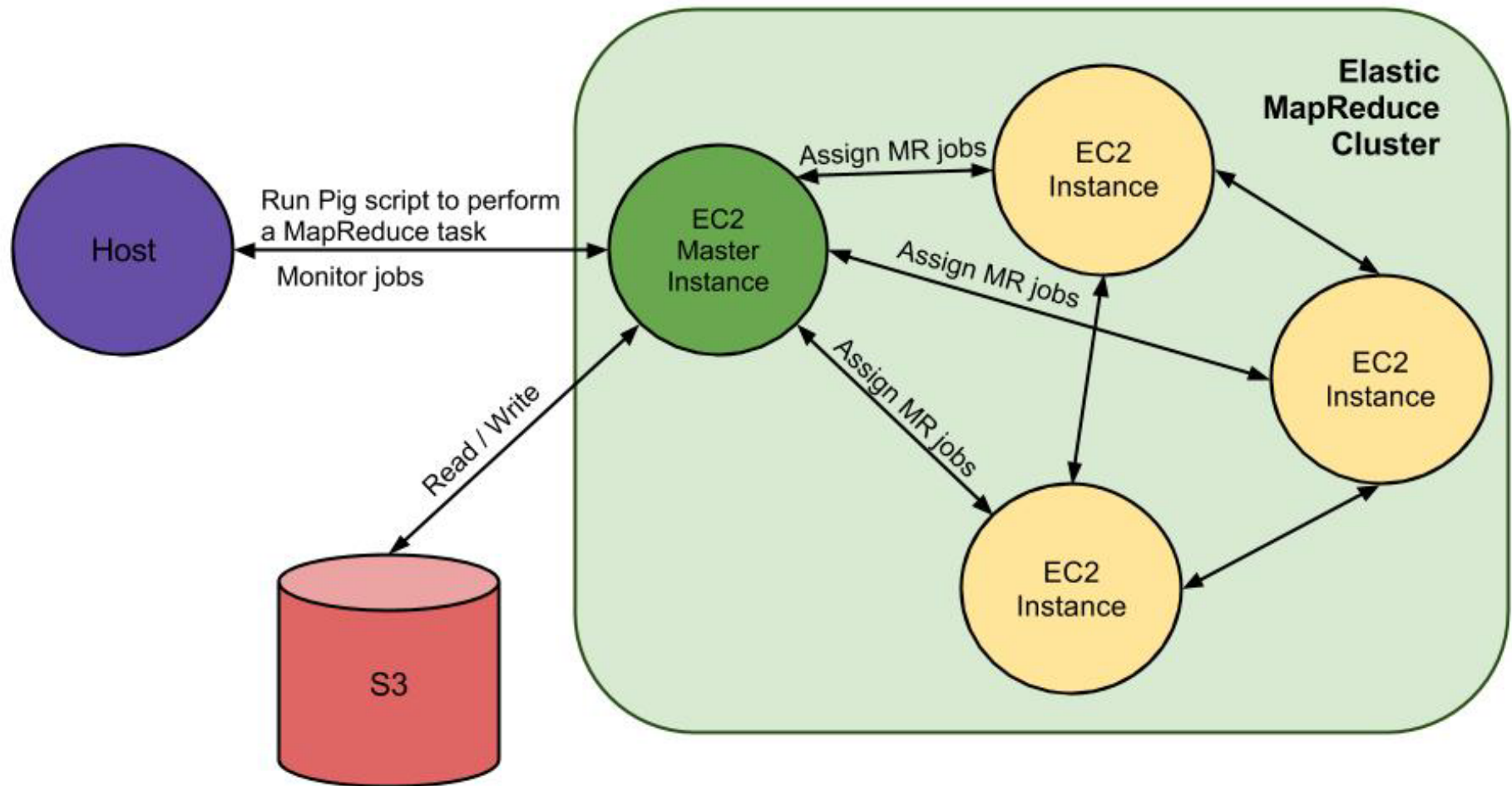
TA: Yi-Shu Wei

# Homework 8

- Big Data analysis on billion triple dataset using Amazon Web Service (AWS)
  - Billion Triple Set: contains web information, obtained by a crawler  
(subject, predicate, object)
  - Working with up to 0.5 TB of data
- You will write pig queries for each task and use MapReduce to perform data analysis.
- Due 3/13 (1 late day allowed)!

# Overview

- AWS offers various cloud computing services. In this assignment, we will use:
  - **Elastic MapReduce**: Managed Hadoop Framework
  - **EC2** (Elastic Computing Cluster): virtual servers in the cloud
  - **S3** (Simple Storage Service): scalable storage in the cloud

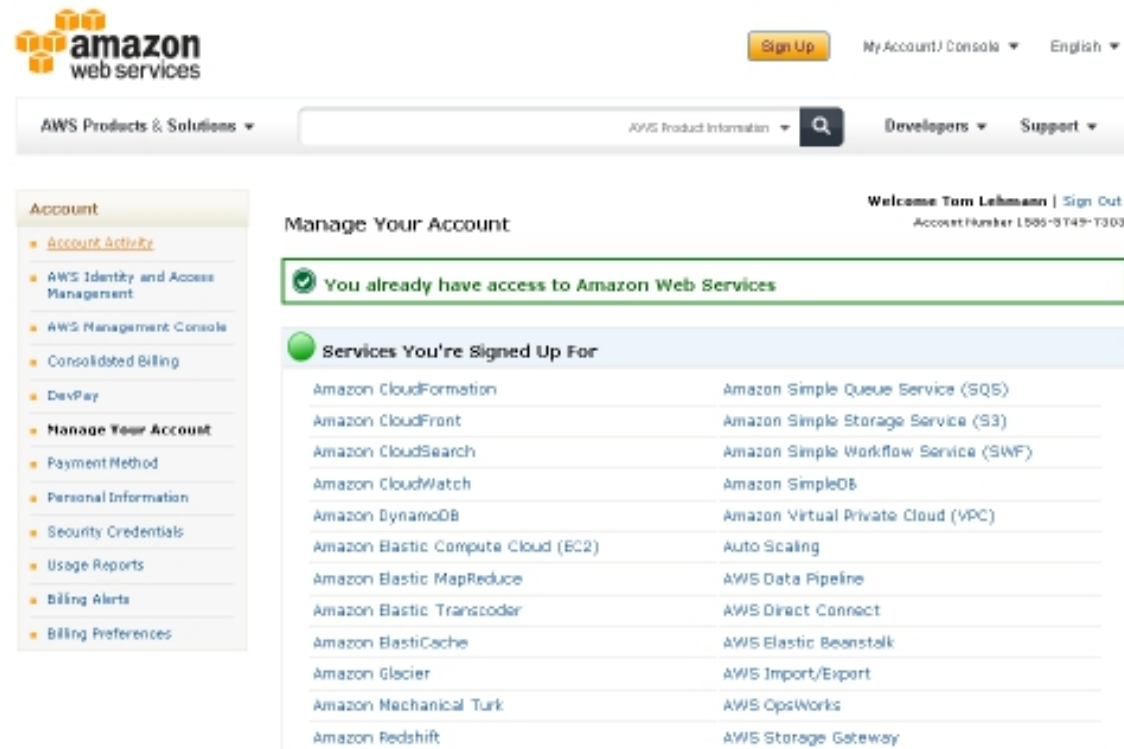


# Set-up

(Details in homework instruction)

# 1. Setting up AWS account

- Sign up/in: <https://aws.amazon.com/>
- Make sure you are signed up for (1) Elastic MapReduce (2) EC2 (3) S3

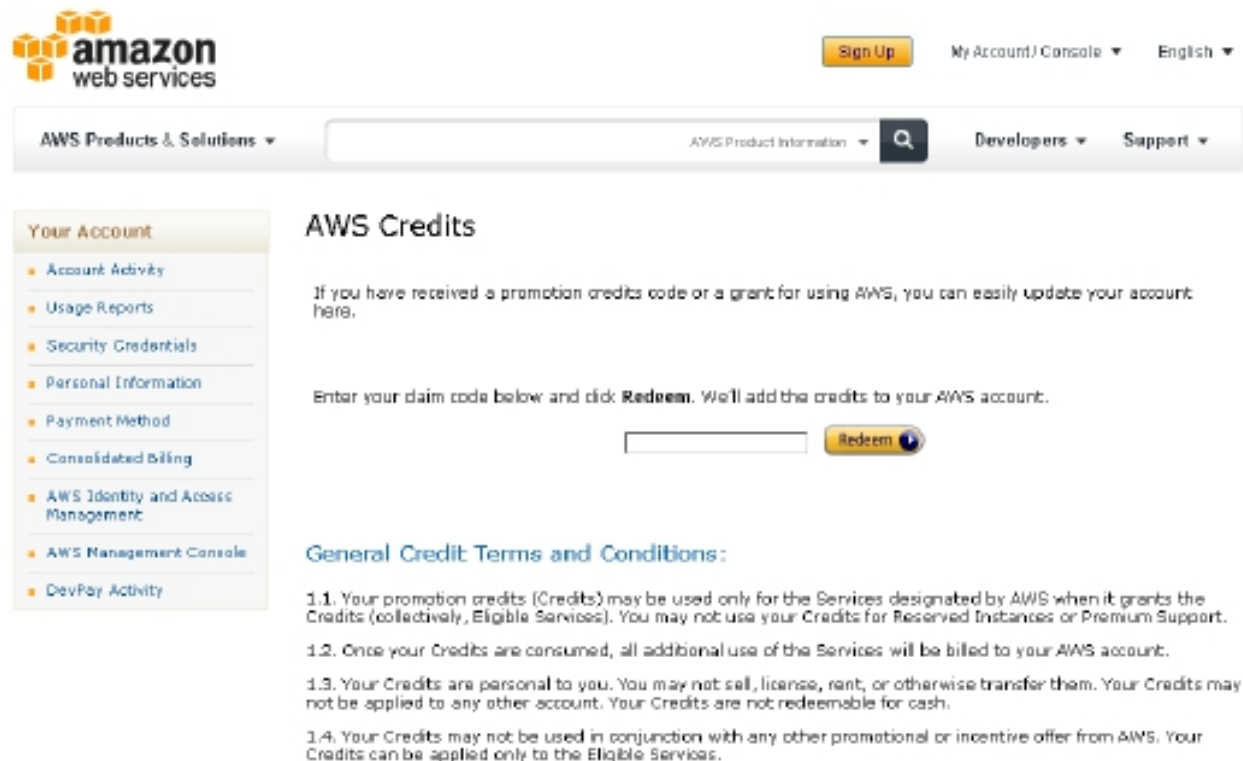


The screenshot displays the AWS Management Console interface. At the top, the Amazon Web Services logo is visible alongside a 'Sign Up' button and links for 'My Account / Console' and 'English'. Below the header, a navigation bar includes 'AWS Products & Solutions', a search bar, and links for 'Developers' and 'Support'. The left-hand navigation pane lists various account management options, with 'Manage Your Account' currently selected. The main content area, titled 'Manage Your Account', features a green success message: 'You already have access to Amazon Web Services'. Below this, a section titled 'Services You're Signed Up For' lists 18 AWS services in two columns. The services listed are: Amazon CloudFormation, Amazon CloudFront, Amazon CloudSearch, Amazon CloudWatch, Amazon DynamoDB, Amazon Elastic Compute Cloud (EC2), Amazon Elastic MapReduce, Amazon Elastic Transcoder, Amazon ElastiCache, Amazon Glacier, Amazon Mechanical Turk, Amazon Redshift, Amazon Simple Queue Service (SQS), Amazon Simple Storage Service (S3), Amazon Simple Workflow Service (SWF), Amazon SimpleDB, Amazon Virtual Private Cloud (VPC), Auto Scaling, AWS Data Pipeline, AWS Direct Connect, AWS Elastic Beanstalk, AWS Import/Export, AWS OpsWorks, and AWS Storage Gateway.

Services You're Signed Up For	
Amazon CloudFormation	Amazon Simple Queue Service (SQS)
Amazon CloudFront	Amazon Simple Storage Service (S3)
Amazon CloudSearch	Amazon Simple Workflow Service (SWF)
Amazon CloudWatch	Amazon SimpleDB
Amazon DynamoDB	Amazon Virtual Private Cloud (VPC)
Amazon Elastic Compute Cloud (EC2)	Auto Scaling
Amazon Elastic MapReduce	AWS Data Pipeline
Amazon Elastic Transcoder	AWS Direct Connect
Amazon ElastiCache	AWS Elastic Beanstalk
Amazon Glacier	AWS Import/Export
Amazon Mechanical Turk	AWS OpsWorks
Amazon Redshift	AWS Storage Gateway

# 1. Setting up AWS account

- Free Credit: <https://aws.amazon.com/awscredits/>
  - Should have received your AWS credit code by email
  - \$100 worth of credits should be enough



The screenshot shows the AWS Credits page in the AWS Management Console. At the top, there is the AWS logo and navigation links for 'Sign Up', 'My Account/Console', and 'English'. Below the navigation bar, there is a search bar and links for 'AWS Products & Solutions', 'AWS Product Information', 'Developers', and 'Support'. On the left side, there is a 'Your Account' sidebar with links to 'Account Activity', 'Usage Reports', 'Security Credentials', 'Personal Information', 'Payment Method', 'Consolidated Billing', 'AWS Identity and Access Management', 'AWS Management Console', and 'DevPay Activity'. The main content area is titled 'AWS Credits' and contains the following text:

If you have received a promotion credits code or a grant for using AWS, you can easily update your account here.

Enter your claim code below and click **Redeem**. We'll add the credits to your AWS account.

Below the text, there is a text input field and a 'Redeem' button with a circular arrow icon.

**General Credit Terms and Conditions:**

- 1.1. Your promotion credits (Credits) may be used only for the Services designated by AWS when it grants the Credits (collectively, Eligible Services). You may not use your Credits for Reserved Instances or Premium Support.
- 1.2. Once your Credits are consumed, all additional use of the Services will be billed to your AWS account.
- 1.3. Your Credits are personal to you. You may not sell, license, rent, or otherwise transfer them. Your Credits may not be applied to any other account. Your Credits are not redeemable for cash.
- 1.4. Your Credits may not be used in conjunction with any other promotional or incentive offer from AWS. Your Credits can be applied only to the Eligible Services.

## 2. Setting up an EC2 key pair

- Go to AWS security credentials page and make sure that you see a key under the access key, if not just click Create a new Access Key.

<https://portal.aws.amazon.com/gp/aws/securityCredentials>



## 2. Setting up an EC2 key pair

- Go to EC2 Management Console  
<https://console.aws.amazon.com/ec2/v2/home>
- Pick region in navigation bar (top right)
- Click on *Key Pairs* and click *Create Key Pair*
- Enter name and click *Create*
- Download of .pem private key
  - lets you access EC2 instance
  - Only time you can download the key

## 2. Setting up an EC2 key pair (Linux/Mac)

- Change the file permission

```
$ chmod 400 </path/to/saved/keypair/file.pem>
```

## 2. Setting up an EC2 key pair (Windows)

- AWS instruction:  
<http://docs.aws.amazon.com/gettingstarted/latest/computebasics-linux/getting-started-deploy-app-connect.html>
- Use PuTTYGen to convert a key pair from .pem to .ppk (part 1 – 2)
- Use PuTTY to establish a connection to EC2 master instance (part 3 – 6)

### 3. Starting an AWS cluster

- <http://console.aws.amazon.com/elasticmapreduce/vnext/home>
- Click *Create Cluster*
- Name the cluster
- Disable Logging

### 3. Starting an AWS Cluster

- Select # of core instances (must be <20)
- Set your previously created Key Pair to be the Amazon EC2 Key Pair
- Add Bootstrap action – Memory intensive configuration
- Create cluster

# 3. Starting an AWS Cluster

- Go back to cluster list
- (It takes a few minutes to start)
- Retrieve the Master Public DNS
- Windows users use PuTTY to connect to cluster
- Everybody else runs this from command line

```
ssh -o "ServerAliveInterval 10" -i </path/to/saved/keypair/file.pem>  
hadoop@<master.public-dns-name.amazonaws.com>
```

## 4. Running Pig interactively

- Once you successfully made a connection to EC2 cluster, type pig, and it will show  
grunt>
- Time to write some pig queries!



## 4. Running Pig interactively

example.pig

- Found in the project archive
- Loads and parses billion triple dataset:  
Triples (subject, predicate, object)
- Group object by attribute, sort in descending order based on count of tuple
- Check out the README for more information



# 5. Monitoring Hadoop jobs

Possible options are:

1. Using ssh tunneling (recommended)
2. Using LYNX
3. Using SOCKS proxy

## 6. Terminating Cluster

- Go to cluster list (Management Console)
- Select cluster
- Click Terminate
- Wait a few minutes ...
- Eventually status should be Terminated
- Don't forget to terminate your cluster to avoid extra charges!

# Where is your input file?

- Your input files come from Amazon S3
- You will use three sets, each of different size
  - `s3n://uw-cse344-test/cse344-test-file` -- 250KB
  - `s3n://uw-cse344/btc-2010-chunk-000` -- 2GB
  - `s3n://uw-cse344` -- 0.5TB
- See `example.pig` for how to load the dataset

```
raw = LOAD 's3n://uw-cse344-test/cse344-test-file' USING TextLoader as (line:chararray);
```

# Where is your output stored?

- Two options

1. Hadoop File System

The AWS Hadoop cluster maintains its own HDFS instance, which dies with the cluster (this is different from other Hadoop provider). **Don't forget to copy** them to your local machine before terminating the cluster.

2. S3

S3 is persistent storage. But S3 costs money while it stores data. **Don't forget to delete** them once you are done.

- A job will output a set of files stored under a directory. (Each file is generated by a reduce worker to avoid contention on a single output file.)

# How can you get the output files?

## 1. Easier and expensive way:

- Create your own S3 bucket (file system), write the output there
- Output filenames become s3n://your-bucket/outdir
- Can download the files via S3 Management Console
- But S3 does cost money, even when the data isn't going anywhere. DELETE YOUR DATA ONCE YOU'RE DONE!

## 2. Harder and cheapskate way:

- Write to cluster's HDFS
- Output directory name is /user/hadoop/outdir. You'll need to create /user/hadoop
- Need to double download
  1. from HDFS to master node's filesystem with *hadoop dfs -copyToLocal*
  2. from master node to local machine with scp

# Final Comment

- Start early
- Important: read the spec carefully!  
If you get stuck or have an unexpected outcome, it is likely that you miss some step or there may be important directions/notes in the spec.
- Running jobs may take up to several hours
  - Problem 4 takes about 4 hours.