Google Cloud and Assignment #3

Computing is Rapidly Changing

- There are many trends putting pressure on conventional computing centers, e.g.
  - Explosive growth in applications: biomedical informatics, space exploration, business analytics, web 2.0 social networking
  - Extreme scale content *generation*: e-science and e-business data deluge
  - Extraordinary rate of digital content *consumption*: digital gluttony: Apple iPhone, iPad, Amazon Kindle
  - Exponential growth in compute capabilities: multi-core, storage, bandwidth, virtual machines (virtualization)
  - Very short cycle of obsolescence in technologies: Windows Vista→ Windows 10; Java versions; C→C#; Python
  - Newer architectures: web services, persistence models, distributed file systems/repositories (Google, Hadoop), multi-core, wireless and mobile
- It is far more difficult for a company to manage this complex situation with a traditional IT infrastructure:

Enter the Cloud

- **Definition (Simple):** *Cloud computing* refers to the practice of using a network of remote servers hosted on the Internet to store, manage, and process data, rather than a local server or a personal computer
- **Definition (Complicated):** *Cloud computing* is an information technology paradigm that enables ubiquitous access to shared pools of configurable system resources and higher-level services that can be rapidly provisioned with minimal management effort, often over the Internet.
- **Definition:** *Cloud computing* is Internet-based computing, whereby shared resources, software and information are provided to computers and other devices on-demand, like the electricity grid.
- **Other names for cloud computing:**
  - **on-demand computing, utility computing, ubiquitous computing, autonomic computing, platform computing, edge computing, elastic computing, grid computing, …**

Cloud Computing

- **Cloud Computing takes place over the Internet,**
  - a collection/group of integrated and networked hardware, software and Internet infrastructure (called a *platform*).
  - Using the Internet for communication and transport provides hardware, software and networking services to clients
- **These platforms hide the complexity and details of the underlying infrastructure from users and applications by providing a "simple" graphical interface or API**

Cloud Computing Platform

- **The platform provides on-demand services, that are always on, anywhere, anytime and any place**
  - Well almost, e.g. Amazon today (03/02/2017) blamed human error for the big AWS outage that took down a bunch of large internet sites for several hours on Tuesday afternoon
  - **http://www.recode.net/2017/3/2/14792636/amazon-aws-internet-outage-cause-human-error-incorrect-command**
- **Pay for use and as needed, *elastic***
  - scale up and down in capacity and functionalities
- **The hardware and software services are available to everyone**
  - general public, enterprises, government

Purpose and Benefits

- **By using the Cloud infrastructure on "pay as used and on demand", companies save in capital and operational investment!**
- **Clients can:**
  - Put their data on the platform instead of on their own desktop PCs and/or on their own servers.
  - They can put their applications on the cloud and use the servers within the cloud to do processing and data manipulations etc.
- **Enables companies and applications, which are system infrastructure dependent, to be infrastructure-less.**

Virtualization Makes Cloud Computing Possible

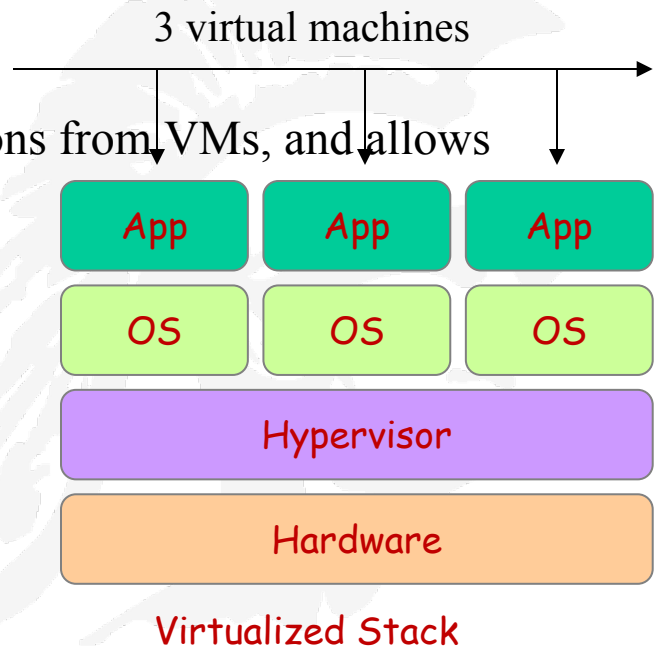
- **Virtualization**: the creation of a virtual -- rather than actual -- version of something, such as an operating system, a server, a storage device or network resources
- **Advantages of virtual machines:**
  1. Run operating systems where the physical hardware is unavailable,
  2. Easier to create new machines, backup machines, etc.,
  3. Software testing using "clean" installs of operating systems and software,
  4. Emulate more machines than are physically available,
  5. Timeshare lightly loaded systems on one host,
  6. Debug problems (suspend and resume the problem machine)
  7. Easy migration of virtual machines (shutdown needed or not)
  8. Run legacy systems!

Hypervisor

- A **hypervisor** or **virtual machine monitor** (**VMM**) is computer software, firmware or hardware that creates and runs virtual machines
- A computer on which a hypervisor runs one or more virtual machines is called a *host machine*, and each virtual machine is called a *guest machine*.
- The hypervisor manages the execution of the guest operating systems.
- Multiple instances of a variety of operating systems may share the virtualized hardware resources: for example, Linux, Windows and MacOS, can all run on a single physical machine.
- Hypervisor Vendors
  - **VMware ESX Server,** ESX301
  - **Xen Project** is a hypervisor providing services that allow multiple computer operating systems to execute on the same hardware concurrently.
    - It was developed by the Univ. of Cambridge and is now being developed by the Linux foundation with support from INTEL XEN3030

Virtualization Makes Cloud Computing Possible

- **Virtual workspaces:**
  - An abstraction of an execution environment that can be made dynamically available to authorized clients by using well-defined protocols,
  - Resource quota (e.g. CPU, memory share),
  - Software configuration (e.g. O/S, provided services).
- **Implement on Virtual Machines (VMs):**
  - Abstraction of a physical host machine,
  - Hypervisor intercepts and emulates instructions from VMs, and allows management of VMs,
  - VMWare, Xen, etc.
- **Provide infrastructure API:**
  - Plug-ins to hardware/support structures

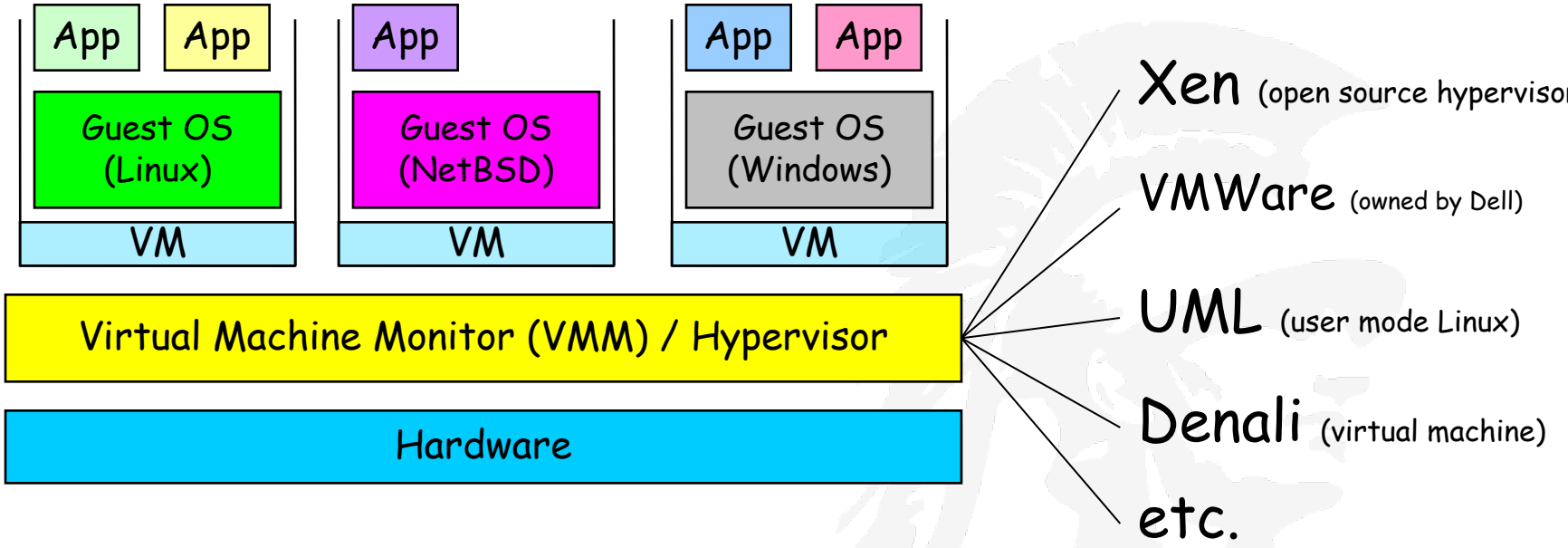


Virtualization Approaches

- The **full virtualization** approach allows datacenters to run an unmodified guest operating system
  - **VMware** uses a combination of direct execution and binary translation techniques to achieve full virtualization of an x86 system
- The **para-virtualization** approach modifies the guest operating system to eliminate the need for binary translation. Therefore it offers potential performance advantages for certain workloads but requires using specially modified operating system kernels
  - The **Xen open source project** was designed initially to support para-virtualized operating systems. While it is possible to modify open source operating systems, such as Linux and OpenBSD, it is not possible to modify "closed" source operating systems such as Microsoft Windows .
- Microsoft Windows is the most widely deployed operating system in enterprise datacenters.
    - For such unmodified guest operating systems, a virtualization hypervisor must either adopt the full virtualization approach or rely on hardware virtualization in the processor architecture.

Virtual Machines

- **VM technology allows multiple virtual machines to run on a single physical machine.**



Performance: Para-virtualization (e.g. Xen) is very close to raw physical performance!

Leading Cloud Vendors

Forbes / Tech / #InTheCloud

NOV 7, 2017 @ 09:06 AM 60,928

The Top 5 Cloud-Computing Vendors: #1 Microsoft, #2 Amazon, #3 IBM, #4 Salesforce, #5 SAP

Bob Evans, CONTRIBUTOR

FULL BIO

Opinions expressed by Forbes Contributors are their own.

CLOUD WARS

Top 10 Rankings – Nov. 7, 2017

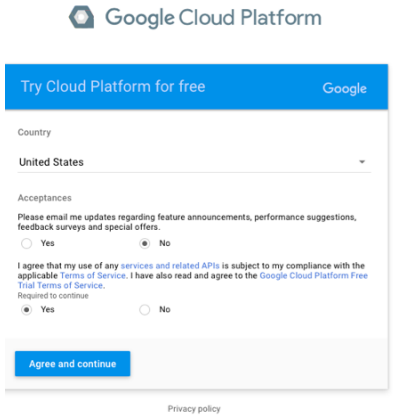
1. **Microsoft** – *Nadella on $20.4B run rate w/end-to-end customer-centric cloud*
2. **Amazon** -- *AWS needs software! 10 software companies Amazon might look at*
3. **IBM** – *Rometty strikes gold helping customers convert legacy IT to private cloud*
4. **Salesforce** – *Benioff must extend SFDC impact from SaaS deeply into PaaS*
5. **SAP** – *McDermott accelerating major product-line overhaul to HANA and cloud*
6. **Oracle** –*Ellison on cybercrime: 'Make no mistake, this is a war—and we're losing'*
7. **Google** – *Tons of potential but still unclear if/how it wants to play in enterprise*
8. **ServiceNow** – *Jumps ahead of Workday: revenue up 40%, new products boom*
9. **Workday** – *Q2 revenue surges 41% as Bhusri jumps into PaaS marketplace*
10. **VMware** – *revenue & stock jump on deals w/AMZN MSFT IBM GOOG for hybrid*

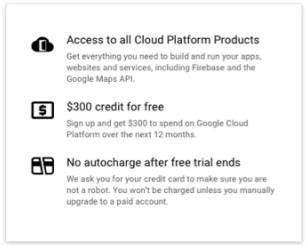
Bob Evans

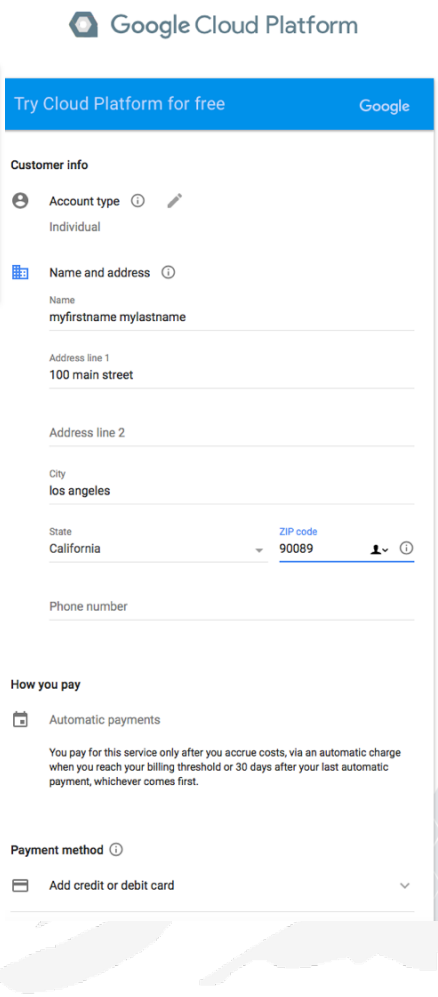
As the Cloud Wars heat up, IBM jumps to #3, Salesforce.com falls from #2 (tie) to #4[+]

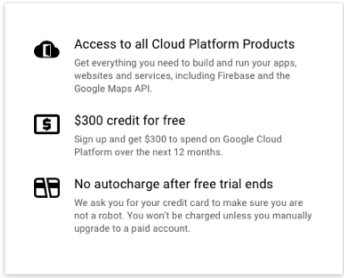
Now Lets Shift to HW#3
Redeeming Google Cloud Credits

Students will sign up for Google's free trial, $300 credit Select Account Type Individual use your @gmail.com address and finish by clicking "Start my free trial"





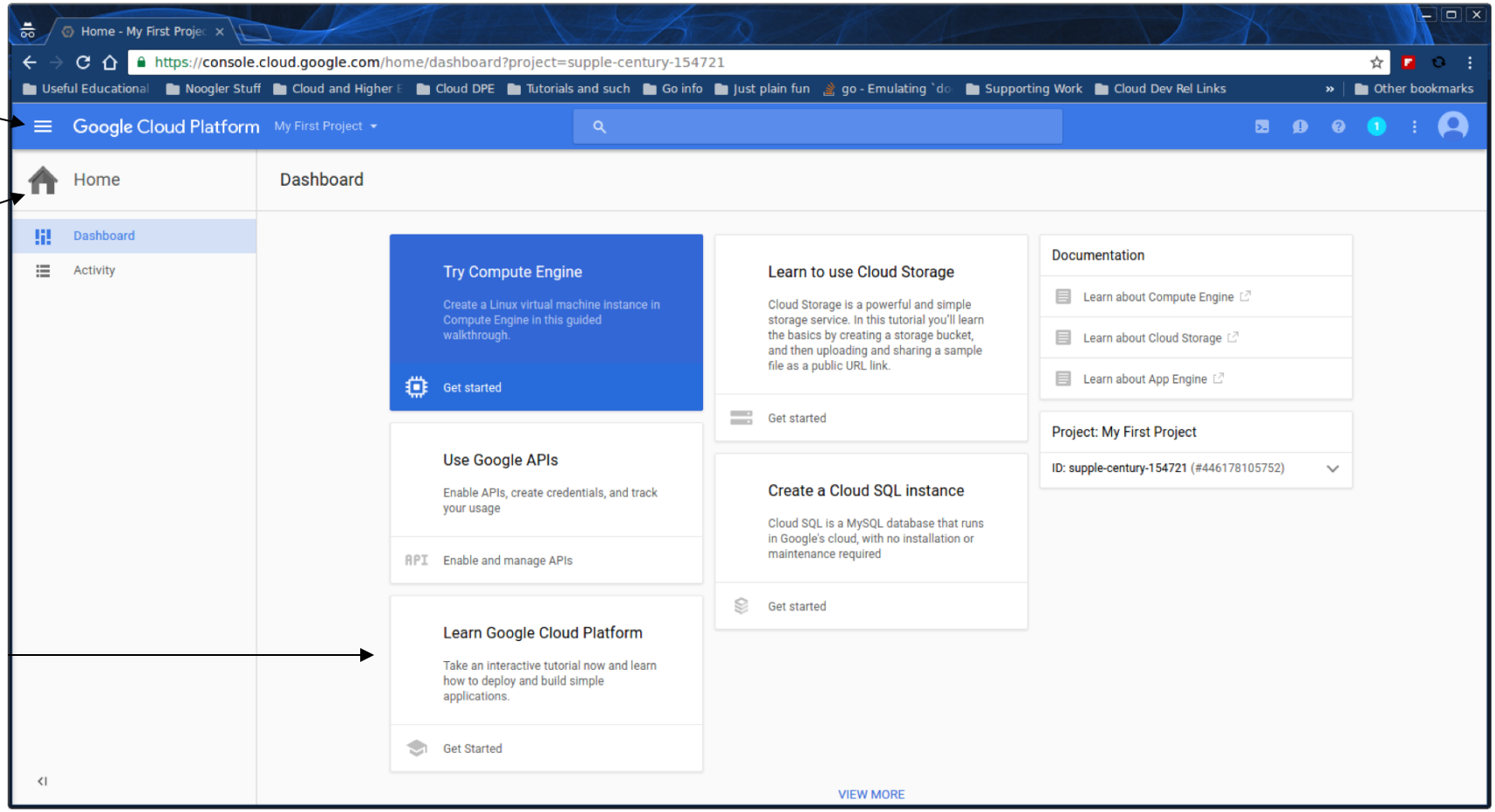




console.cloud.google.com

Google Cloud Home Page
https://console.cloud.google.com

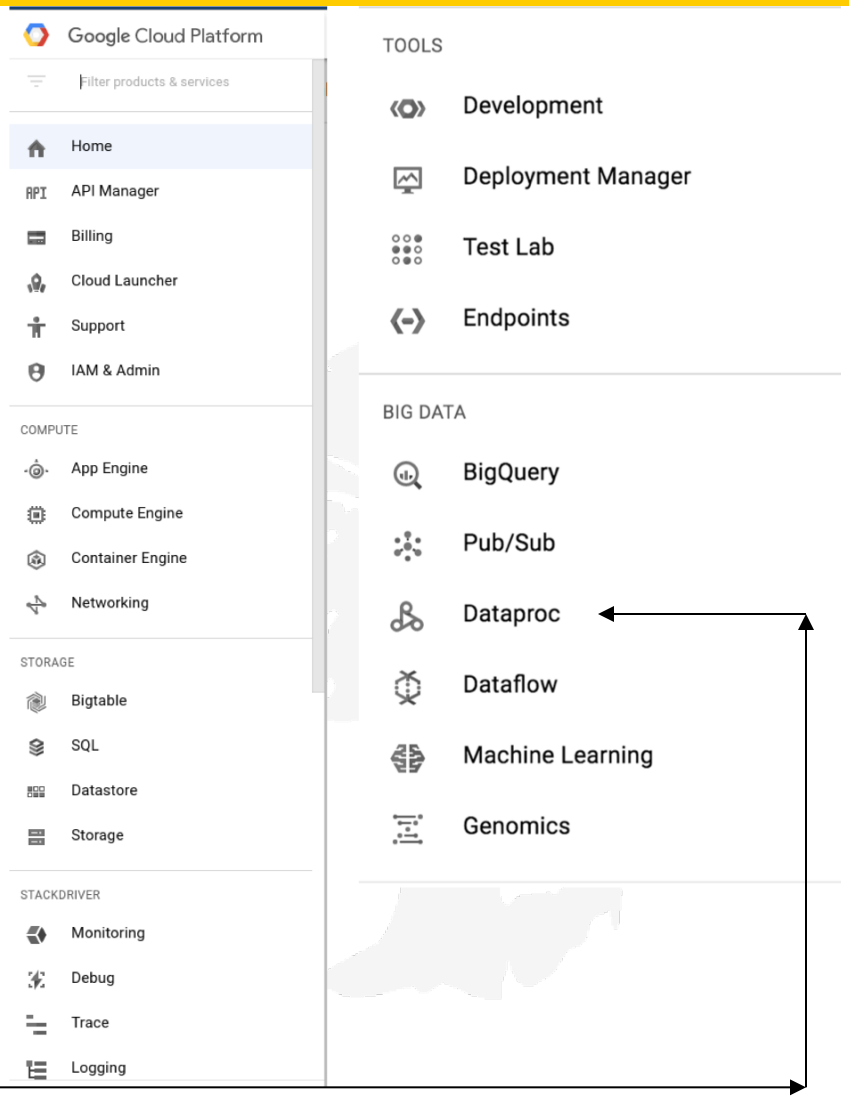
There are two menus available from the console: main and context



Google Cloud Main Menu

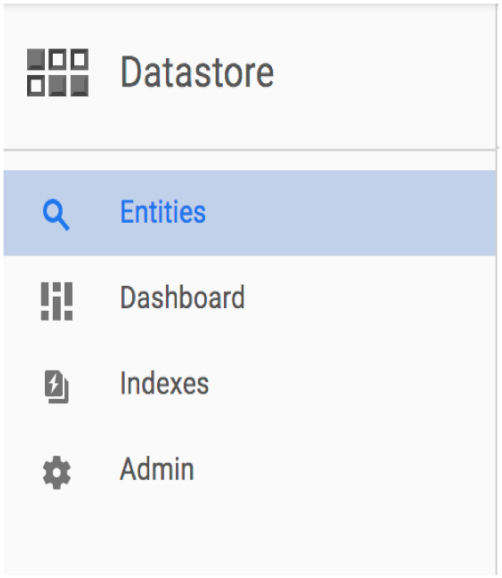
- From the hamburger menu in the top left corner, you can access a menu that brings you to the 5 major components of Google Cloud Platform (GCP):
- Compute
- Storage
- Stackdriver (company to manage distributed apps running on the cloud)
- Tools
- Big Data

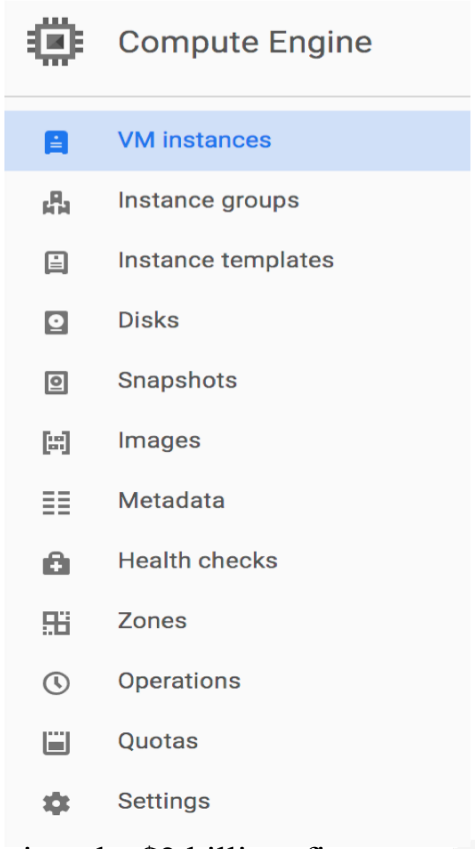
- For this exercise you will use DataProc within BIG DATA to set up a cluster of compute instances

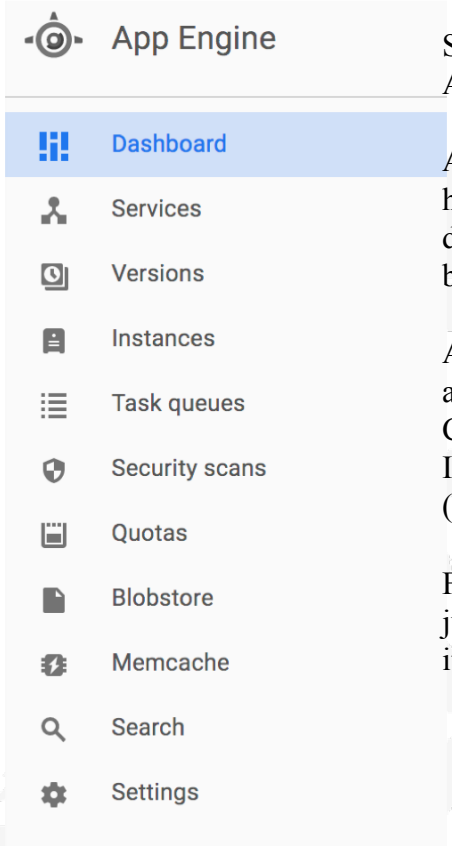


Context Menu

The context menu changes based on the current major component
Here are three examples







Snapchat was built on top of App Engine

App Engine lets clients host their software at datacenters managed by Google

App Engine is a Platform as a Service (PaaS) while Compute Engine is an Infrastructure as a Service (IaaS)

For the App Engine you just write your code and it automatically executes

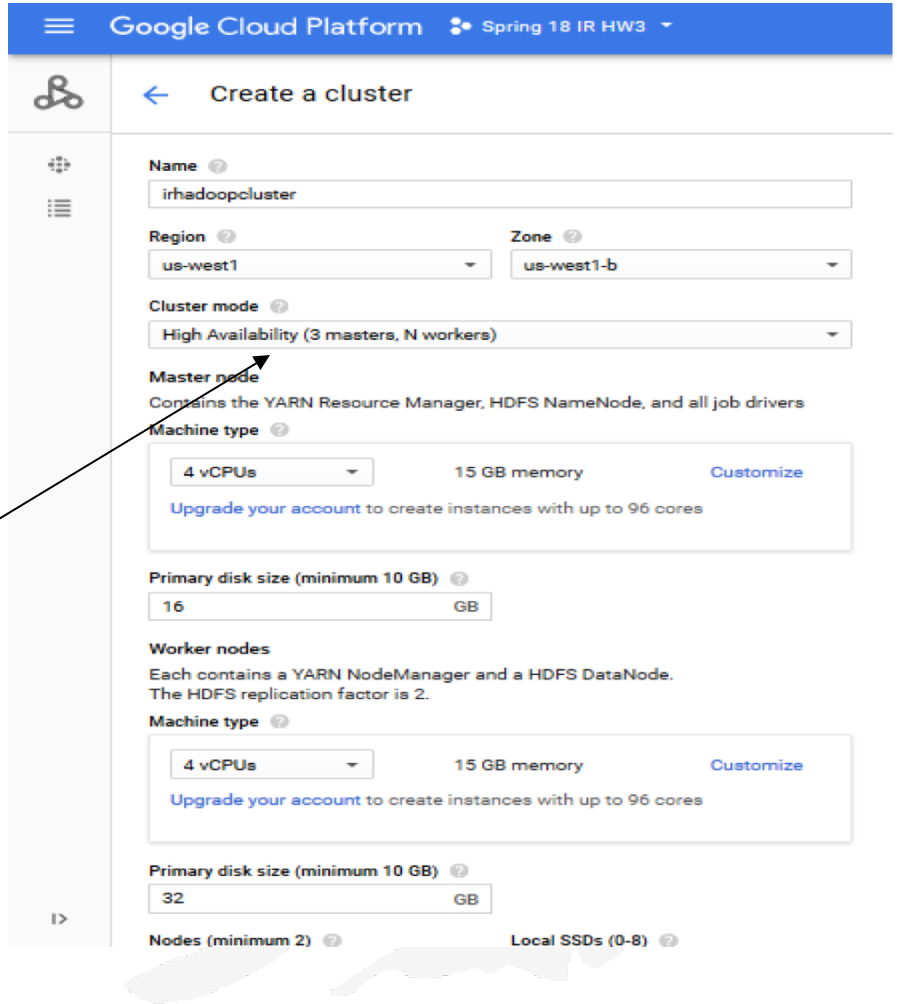
- Snap (Snapchat) recently signed a $2 billion, five year contract with Google for its cloud services, which makes Snap Google's largest customer of its cloud platform
- Apple confirms it is using Google cloud for iCloud services

Create a Cluster

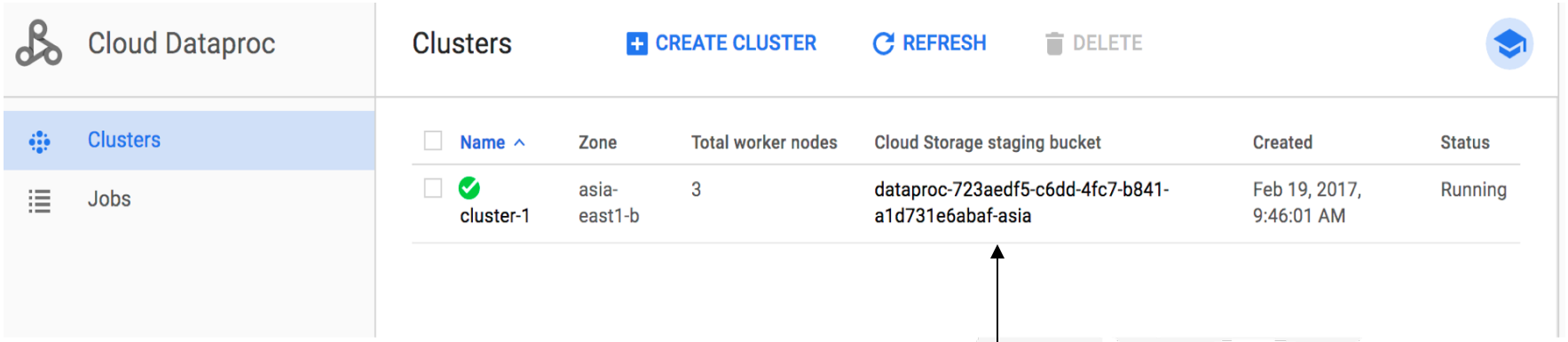
Using the **Google Cloud** Platform Console you can **create** a **cluster** by going to the **Cloud** Platform Console.

Select your project, and then click Continue to open the **Clusters** page.

Contains 1 master node and 3 worker nodes



Successful Creation of a Cluster



This URL will let you ssh to your cluster

Environment Variables Set

Create a home directory

Check env variables
JAVA_HOME
HADOOP_CLASSPATH

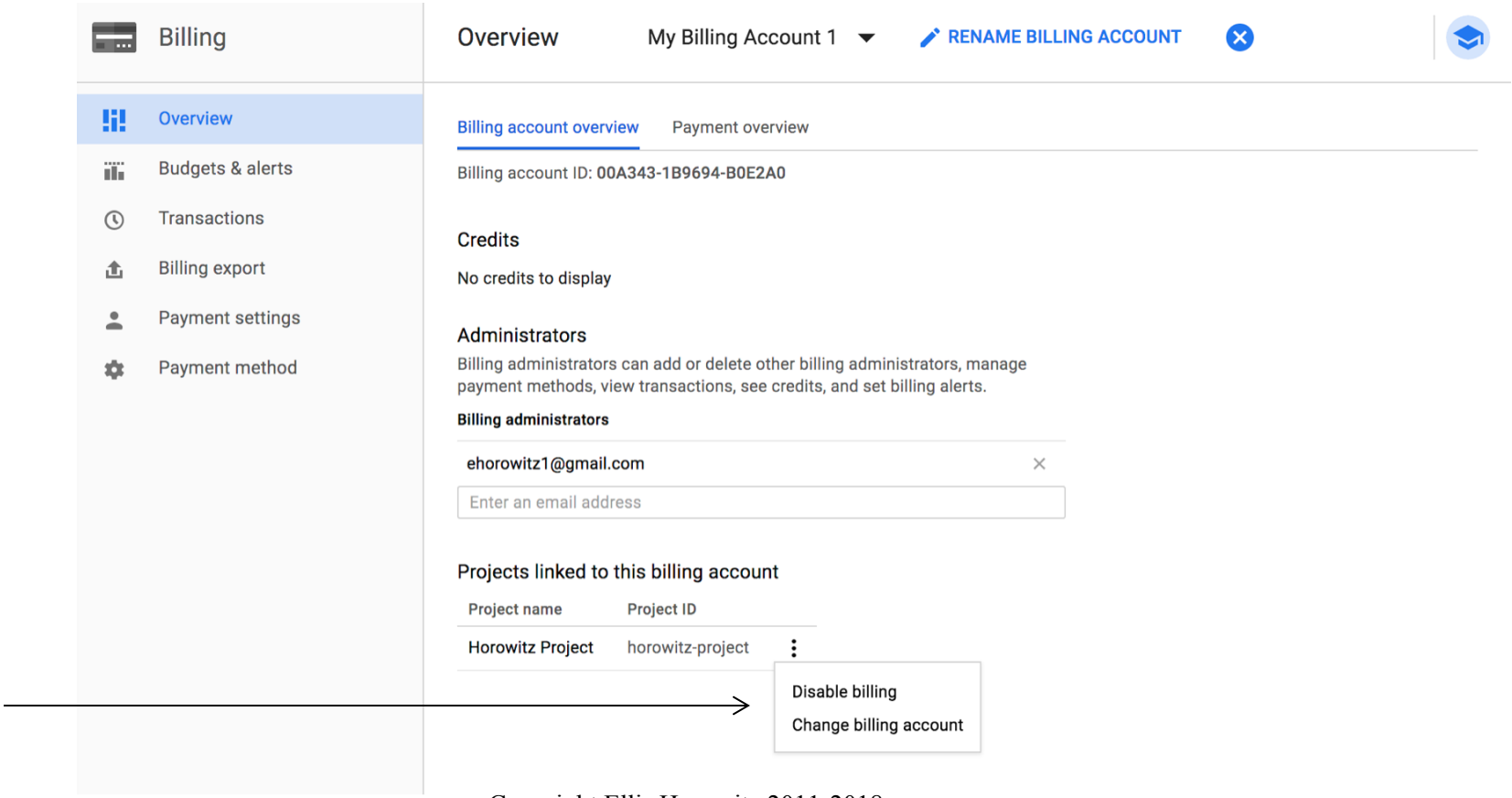
ehorowitz1@cluster-1-m: ~

Secure https://ssh.cloud.google.com/projects/horowitz-project/zones/asia-east1-b/instances/cluster-1-m?authuser=0&hl=en_US&...

```
ehorowitz1@cluster-1-m:~$ env
TERM=xterm-256color
SHELL=/bin/bash
SSH_CLIENT=173.194.90.33 63226 22
SSH_TTY=/dev/pts/0
USER=ehorowitz1
LS_COLORS=rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:bd=40;33;01:cd=40;33;01:or=40;31;01:su=37;41:sg=3
0;43:ca=30;41:tw=30;42:ow=34;42:st=37;44:ex=01;32:*.tar=01;31:*.tgz=01;31:*.arc=01;31:*.arj=01;31:*.taz=01;31:*.lha
=01;31:*.lz4=01;31:*.lzh=01;31:*.lzma=01;31:*.tlz=01;31:*.txz=01;31:*.tzo=01;31:*.t7z=01;31:*.zip=01;31:*.z=01;31:*
.Z=01;31:*.dz=01;31:*.gz=01;31:*.lrz=01;31:*.lz=01;31:*.lzo=01;31:*.xz=01;31:*.bz2=01;31:*.bz=01;31:*.tbz=01;31:*.t
bz2=01;31:*.tz=01;31:*.deb=01;31:*.rpm=01;31:*.jar=01;31:*.war=01;31:*.ear=01;31:*.sar=01;31:*.rar=01;31:*.alz=01;3
1:*.ace=01;31:*.zoo=01;31:*.cpio=01;31:*.7z=01;31:*.rz=01;31:*.cab=01;31:*.jpg=01;35:*.jpeg=01;35:*.gif=01;35:*.bmp
=01;35:*.pbm=01;35:*.pgm=01;35:*.ppm=01;35:*.tga=01;35:*.xbm=01;35:*.xpm=01;35:*.tif=01;35:*.tiff=01;35:*.png=01;35
:*.svg=01;35:*.svgz=01;35:*.mng=01;35:*.pcx=01;35:*.mov=01;35:*.mpg=01;35:*.mpeg=01;35:*.m2v=01;35:*.mkv=01;35:*.we
bm=01;35:*.ogm=01;35:*.mp4=01;35:*.m4v=01;35:*.mp4v=01;35:*.vob=01;35:*.qt=01;35:*.nuv=01;35:*.wmv=01;35:*.asf=01;3
5:*.rm=01;35:*.rmvb=01;35:*.flc=01;35:*.avi=01;35:*.fli=01;35:*.flv=01;35:*.gl=01;35:*.dl=01;35:*.xcf=01;35:*.xwd=0
1;35:*.yuv=01;35:*.cgm=01;35:*.emf=01;35:*.axv=01;35:*.anx=01;35:*.ogv=01;35:*.ogx=01;35:*.aac=00;36:*.au=00;36:*.f
lac=00;36:*.m4a=00;36:*.mid=00;36:*.midi=00;36:*.mka=00;36:*.mp3=00;36:*.mpc=00;36:*.ogg=00;36:*.ra=00;36:*.wav=00;
36:*.axa=00;36:*.oga=00;36:*.spx=00;36:*.xspf=00;36:
DATAPROC_MASTER_HA_COMPONENTS=hadoop-hdfs-journalnode hadoop-hdfs-zkfc zookeeper-server
SSH_AUTH_SOCK=/tmp/ssh-zGxHCr3rJ9/agent.3623
DATAPROC_MASTER_COMPONENTS=hadoop-hdfs-namenode hadoop-yarn-resourcemanager mysql-server
MAIL=/var/mail/ehorowitz1
PATH=/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
PWD=/home/ehorowitz1
JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
HADOOP_CLASSPATH=/lib/tools.jar
LANG=en_US.UTF-8
DATAPROC_COMMON_COMPONENTS=openjdk-8-jdk libjansi-java python-numpy libmysql-java hadoop-client hive pig spark-core
 spark-python spark-r autofs nfs-common libhdfs0 libsnappy1 libatlas3-base libopenblas-base libapr1 vim git bash-co
mpletion spark-yarn-shuffle spark-datanucleus spark-extras hadoop-lzo
DATAPROC_MASTER_STANDALONE_COMPONENTS=hadoop-hdfs-secondarynamenode
ALPN_JAR=/usr/local/share/google/alpn/alpn-boot-8.1.7.v20160121.jar
DATAPROC_WORKER_COMPONENTS=hadoop-hdfs-datanode hadoop-yarn-nodemanager
SHLVL=1
HOME=/home/ehorowitz1
BDUTIL_DIR=/usr/local/share/google/dataproc/bdutil-dataproc-20170214-094528-RC1
LOGNAME=ehorowitz1
SSH_CONNECTION=173.194.90.33 63226 10.140.0.4 22
DATAPROC_AGENT_JAR=/usr/local/share/google/dataproc/agent-20170214-094528-RC1.jar
DATAPROC_MASTER_EXCLUSIVE_COMPONENTS=hadoop-mapreduce-historyserver hive-metastore hive-server2 nfs-kernel-server s
park-history-server
_=/usr/bin/env
```

Disable Billing for Your Cluster

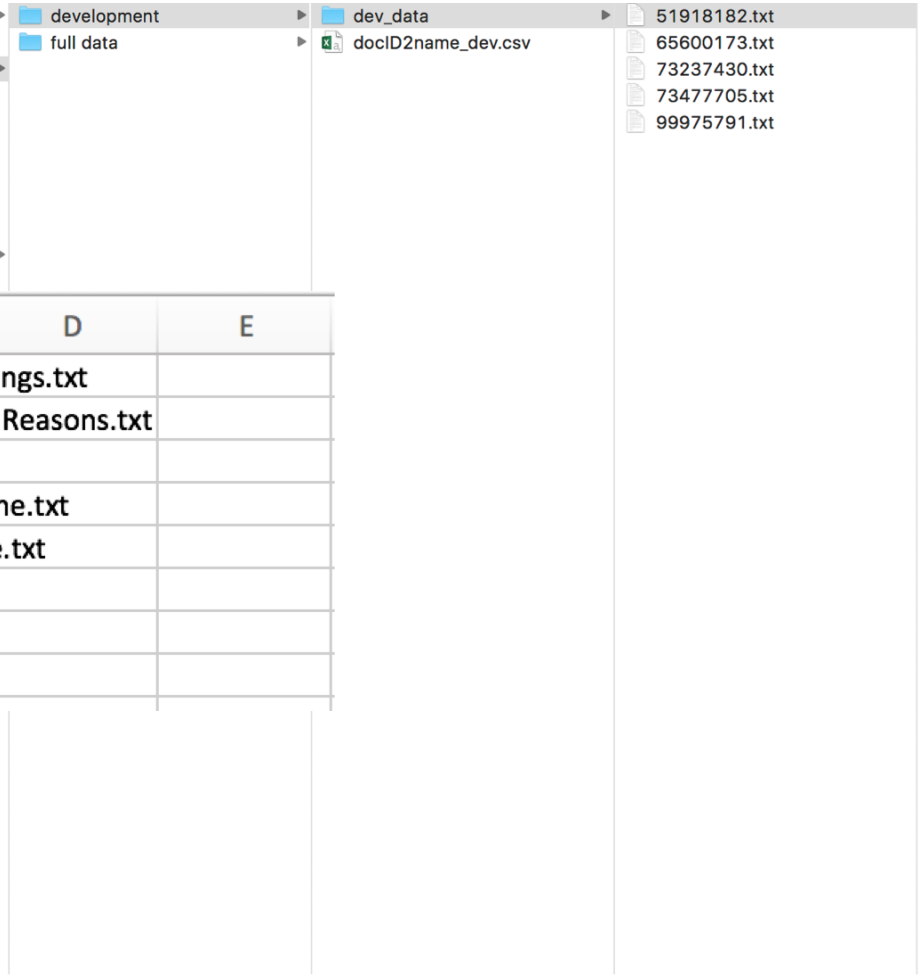
- Please **disable** the billing for the cluster when you are not using it.
- Leaving it running will cost extra credits.
- The cluster is billed based on how many hours it is running and not how much data it is processing



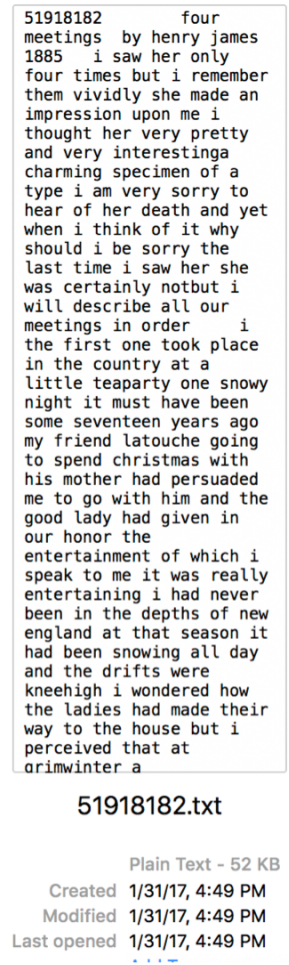
Upload the Data Set

- **We'll be using a collection of 3,036 English books written by 142 authors**
  - The data comes from web.eecs.umich.edu/~lahiri/gutenberg_dataset.html
  - The data has been cleaned of metadata, license information, notes
- **Retrieve the dataset either from the cloud or from the 572 website**
  - http://www-scf.usc.edu/~csci572/2018Spring/hw3/DATA.zip
  - https://drive.google.com/open?id=0B0rRqvGB9KPBU2pJX01uenh2NWc
- **Unzip the contents**
  - two folders inside named 'development' and 'full data'.
  - Each of the folders contains the actual data(books) and a mapper file to map the docID to the file name.

Development Data

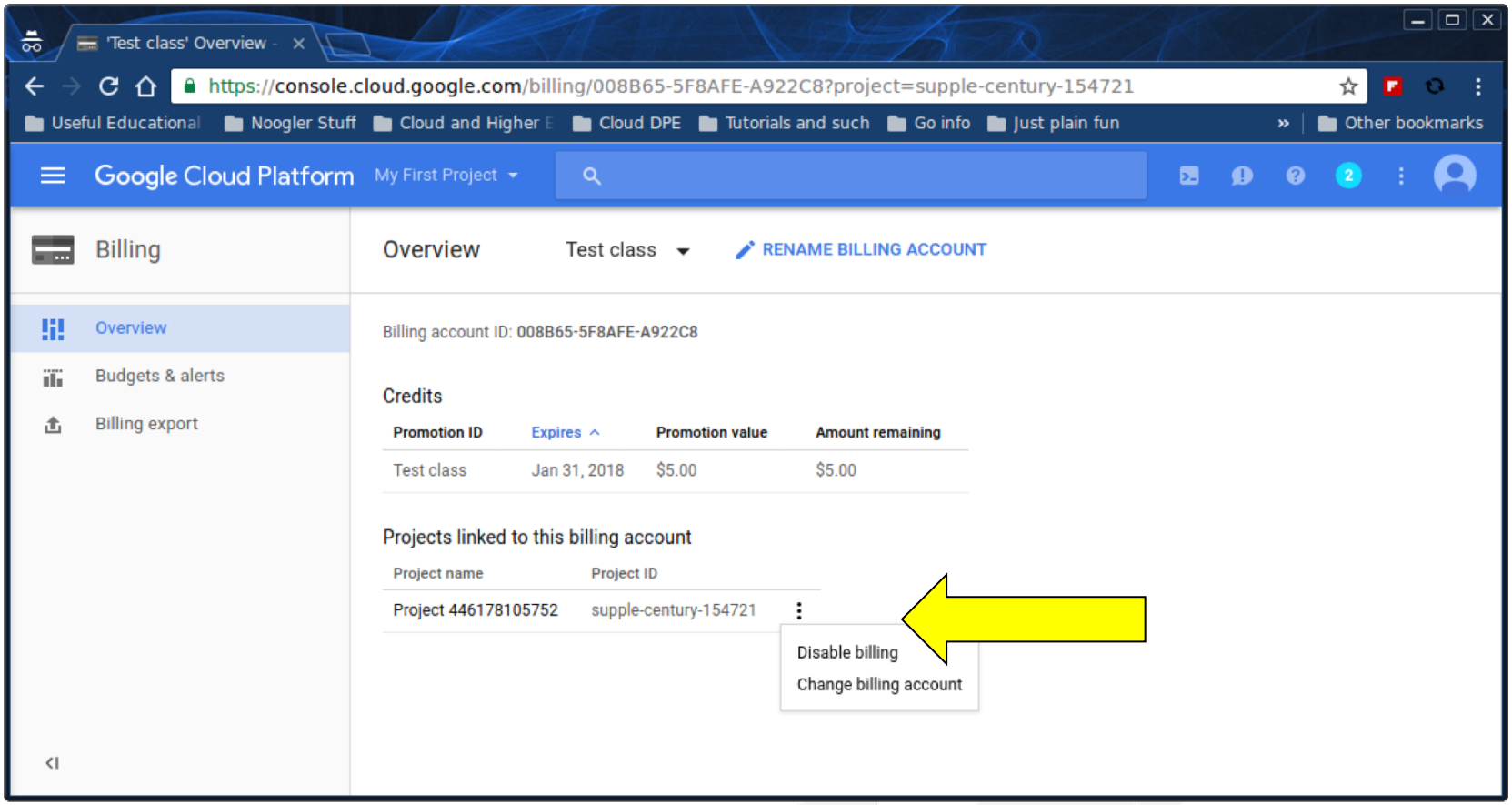


	A	B	C	D	E
1	51918182	Henry James___Four Meetings.txt			
2	99975791	Henry James___Georgina's Reasons.txt			
3	73477705	Henry James___Glasses.txt			
4	73237430	Henry James___Greville Fane.txt			
5	65600173	Henry James___Hawthorne.txt			
6					
7					
8					



REMEMBER TO disable the billing for the cluster when you are not using it

Click here and select Billing



Inverted Index Implementation

- **You need to write some code, in Java, that processes the data file of books and produces an inverted index of the words that occur in the books**
- **Google Cloud requires the code to be packaged as a jar file, e.g.**
- **If your Java program is called** `InvertedIndexJob.java`
  - **first compile the code and then**
  - **run the jar program**
- `hadoop com.sun.tools.javac.Main InvertedIndexJob.java`
- `jar cf invertedindex.jar InvertedIndex*.class`
- **Place this jar file in the default cloud bucket of your cluster in a folder called JAR on your bucket and upload it to that folder**

Mapper Class

The Google cluster requires that you write two routines to implement the Map/Reduce functionality

A lecture on Map/Reduce is coming later

Class is WordCountMapper;
Routine is map(key,value,context)

Program reads a line of text, and for each token (word) that it finds on the line it sends the pair (token, 1) to the collector/ reducer

```
/*
This is the Mapper class. It extends the Hadoop's Mapper class.
This maps input key/value pairs to a set of intermediate(output) key/value pairs.
Here our input key is a LongWritable and input value is a Text.
And the output key is a Text and value is an IntWritable.
*/
class WordCountMapper extends Mapper<LongWritable, Text, Text, IntWritable>
{
    /*
    Hadoop supported data types. This is a Hadoop specific datatype that is used to handle
    numbers and Strings in a hadoop environment. IntWritable and Text are used instead of
    Java's Integer and String datatypes.
    Here 'one' is the number of occurances of the 'word' and is set to the value 1 during the
    Map process.
    */
    private final static IntWritable one = new IntWritable(1);
    private Text word = new Text();

    public void map(LongWritable key, Text value, Context context)
        throws IOException, InterruptedException
    {
        //Reading input one line at a time and tokenizing.
        String line = value.toString();
        StringTokenizer tokenizer = new StringTokenizer(line);

        //Iterating through all the words available in that line and forming the key value pair.
        while (tokenizer.hasMoreTokens())
        {
            word.set(tokenizer.nextToken());
            /*
            Sending to output collector(Context) which in-turn passes the output to Reducer.
            The output is as follows:
                'word1' 1
                'word1' 1
                'word2' 1
            */
            context.write(word, one);
        }
    }
}
```

Reducer Class

Class is WordCountReducer;
Program is reduce(key, values,context);

For each key (word), the number of occurrences are summed together and written out

```
/*
This is the Reducer class. It extends the Hadoop's Reducer class.
This maps the intermediate key/value pairs we get from the mapper to a set
of output key/value pairs, where the key is the word and the value is the word's count.
Here our input key is a Text and input value is a IntWritable.
And the output key is a Text and value is an IntWritable.
*/
class WordCountReducer extends Reducer<Text, IntWritable, Text, IntWritable>
{
    /*
    Reduce method collects the output of the Mapper and adds the 1's to get the word's count.
    */
    public void reduce(Text key, Iterable<IntWritable> values, Context context)
            throws IOException, InterruptedException
    {
        int sum = 0;
        /*
        Iterates through all the values available with a key and add them together and give the
        final result as the key and sum of its values
        */
        for (IntWritable value : values)
        {
            sum += value.get();
        }
        context.write(key, new IntWritable(sum));
    }
}
```

Main Class

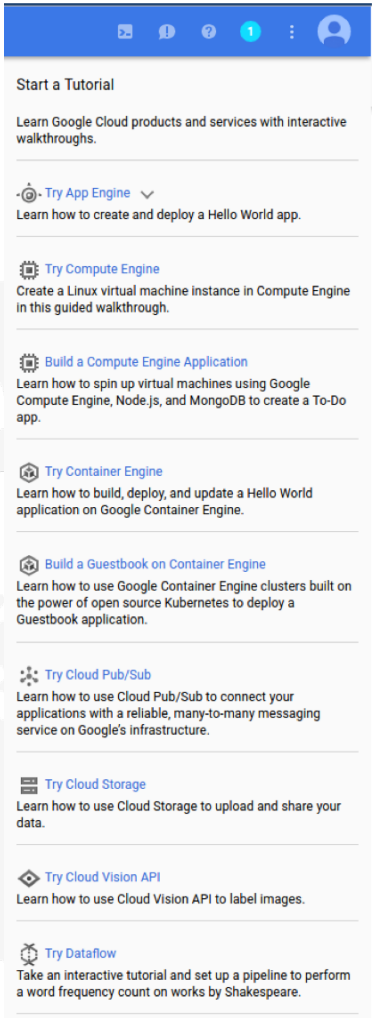
Class WordCount;
Program: main
Create the Hadoop job

```
import java.io.IOException;
import java.util.StringTokenizer;
import org.apache.hadoop.*;
public class WordCount
{
    public static void main(String[] args)
        throws IOException, ClassNotFoundException, InterruptedException {
        if (args.length != 2) {
            System.err.println("Usage: Word Count <input path> <output path>");
            System.exit(-1);
        }
        //Creating a Hadoop job and assigning a job name for identification.
        Job job = new Job();
        job.setJarByClass(WordCount.class);
        job.setJobName("Word Count");
        //The HDFS input and output directories to be fetched from the Dataproc job submission console.
        FileInputFormat.addInputPath(job, new Path(args[0]));
        FileOutputFormat.setOutputPath(job, new Path(args[1]));
        //Providing the mapper and reducer class names.
        job.setMapperClass(WordCountMapper.class);
        job.setReducerClass(WordCountReducer.class);
        //Setting the job object with the data types of output key(Text) and value(IntWritable).
        job.setOutputKeyClass(Text.class);
        job.setOutputValueClass(IntWritable.class);
        job.waitForCompletion(true);
    }
}
```

Built-in Tutorials

Click the tricolon at the top right of the console and select "*Try an interactive tutorial*" to be brought to this list of tutorials.

An advantage of these built-in tutorials is they'll step you through each piece and do necessary project management. It's fine to use the default first project for all of these.



Sample Document and Output

```
1  51918182    four meetings  by henry james  1885   i saw her only four times but i remember them
   vividly she made an impression upon me i thought her very pretty and very interestinga charming
   specimen of a type i am very sorry to hear of her death and yet when i think of it why should i
   be sorry the last time i saw her she was certainly notbut i will describe all our meetings in
   order     i  the first one took place in the country at a little teaparty one snowy night it must
   have been some seventeen years ago my friend latouche going to spend christmas with his mother
```

Sample of Mapper Output

```
james   51918182
people  51918182
people 51918182
of  51918182
of  51918182
```

Sample of Reducer Output

```
1  ably    9931985:1
2  abnegate    85886314:1  80811098:1
3  abney   31694096:3  15109590:1  38583612:1  98115965:98
4  abnormal    47943267:1  94435826:1  80942074:1
5  abroad  73713297:1  11200532:1
```

James occurs 1 time in the document whose ID is 51918182
People occurs 2 times in the document whose ID is 51918182

Ably appears once in document 9931985

Credits Spent

- To check how much you've been charged for your cluster,
  - navigate to the Billing section and
  - click on the project name in the Overview section
  - check this section at least once every 24 hours.