Infrastructure as a Service (IaaS)

- Google Compute Engine
- AWS Elastic Compute Cloud (EC2)
- Azure Virtual Machines
Google Compute Engine (GCE)

- Infrastructure-as-a-Service
  - Hardware service for you to create and run virtual machine instances on
  - Instance = virtual machine running on GCE
  - Runs arbitrary workloads
- Equivalent to AWS EC2 and Digital Ocean
  - Lowest-level abstraction for cloud infrastructure
  - Flexible, but requires management
VM instances on GCE

- Can run any number of Linux, Windows VMs, or custom images
- Can run in different regions globally
- Provides vertical scaling options
  - Number of cores
  - Amount of RAM
  - Video card types
  - Type of disk (standard, SSD)
  - Up to 96 cores, 684 GB! (10/2017)
VM instances on GCE

- Provides networking features
  - Segmentation and filtering for security
  - Load balancing to distribute work across machines globally
- Billed by the minute
  - Options for pre-emptible VMs up to 80% lower
Compute Engine access

- Command-line interface (CLI), using Cloud SDK
- Web UI (console.cloud.google.com)
- API directly, (simply HTTP/JSON with client libraries in many different languages)
- Code libraries
Via Cloud Shell or SDK

- Use cloud shell to list
  
gcloud compute instances list

- Create in cloud shell
  
gcloud compute instances create myinstance
  
  - Select zone
  
  - Creates a VM with default config
  
  - List again

- Delete in cloud shell
  
gcloud compute instances delete myinstance
Via API explorer

- In web console, APIs and Services ➔ Library ➔ Compute Engine API ➔ Try this API in APIs Explorer
- Try listing instances via API (enable OAuth2)
  - compute.instances.list
  - Retrieves the list of instances contained within the specified zone
- REST API URL
  - Results
    
    ```
    200 - Show headers -

    {
    "kind": "compute#instanceList",
    "id": "projects/lateral-array-175417/zones/us-west1-b/instances",
    "items": [
    {
    "kind": "compute#instance",
    "id": "8594540100584034031",
    "creationTimestamp": "2017-08-04T15:06:57.224-07:00",
    "name": "angr2",
    "description": "",
    "tags": {
    "fingerprint": "42WmSpB8rSM="
    },
    "status": "TERMINATED",
    }
    ]
    ```

Portland State University CS 410/510 Internet, Web, and Cloud Systems
GCE and Managed Instance Groups

- Implements autoscaling ("Elastic", Managed VMs)
  - Specify a VM instance template
  - Specify an autoscaling option
  - GCE brings instances up and down automatically
    - Auto-healer reboots and fixes problems
    - Auto-updater distributes new software across VM instances

- Good for…
  - Stateless services such as web frontends
  - Data-parallel workloads such as image processing or financial data analysis
Labs
Compute Engine Lab #1

- Compute the Cosmos (54 min)
  - Demo multiple ways of accessing Compute Engine (command-line, programmatically in Python)
  - Then, perform a large computation
  - Note: You can re-use your course project for this lab
Compute Engine via Command Line

- Set zone to us-west1-b

```
gcloud config set compute/zone us-west1-b
```

- Add Custom VM Image to Project
  - Run the following command to add a custom VM Image to your project named "codelab-image"

```
gcloud compute images create --source-uri http://storage.googleapis.com/codelab-2015-vm-image/1ad8c7f0540790f98eaf87801804feac985676e1.image.tar.gz codelab-image
```

- May take a few minutes
List image

```
gcloud compute images list | egrep codelab
```

```
wuchangfeng@invertible-fin-164222:~$ gcloud compute images list | egrep "codelab|NAME"
NAME          PROJECT        FAMILY DEPRECATED STATUS
```
```
codelab-image invertible-fin-164222 READY
```

Create a persistent disk and use it to build codelab VM

```
gcloud compute disks create disk1 --size 800GB \--zone us-west1-b
```

Note that VM image and disk must be in same zone

```
gcloud compute instances create \  codelab-node \  --image codelab-image \  --machine-type n1-standard-4 \  --scopes compute-rw,storage-full \  --boot-disk-device-name codelab-node \  --disk name=disk1,device-name=disk1,mode=rw \  --zone us-west1-b
```

Portland State University CS 410/510 Internet, Web, and Cloud Systems
Compute Engine via the client API library

- Programmatically list VM instances in your project
  - `ssh` into your VM instance via the console or via

    gcloud compute ssh codelab-node

- Create a simple program `helloworld.py` with the following imports

    from googleapiclient.discovery import build
    from oauth2client.client import GoogleCredentials
    import json

- Then set credentials

    credentials = GoogleCredentials.get_application_default()
    service = build('compute', 'v1', credentials=credentials)

The --scopes of "compute-rw storage-full" you ran with the compute instances create means your VM already has a service account setup with the "scopes of access" you'll need to access the Compute Engine and Cloud Storage services.
Compute Engine via the client API library

- Then enter the following in to the file, filling in PROJECT_ID with your own

```python
response = service.instances().list(project='PROJECT_ID', zone='us-west1-b').execute()
print json.dumps(response, sort_keys=True, indent=4)
```

- Save the file and run it from the command line

```bash
python helloworld.py
```

- Show the JSON that is returned

```json
{
  "instances": [
    {
      "id": "1473399635324631347",
      "kind": "compute#instance",
      "machineType": "https://www.googleapis.com/compute/v1/projects/ion-ion/zones/us-central1-b/machineTypes/n1-standard-4",
      "metadata": {
        "fingerprint": "EnXVWDNq5VM=["kind": "compute#metadata"
      },
      "name": "codelab-node",
```
Computing the Universe

- Find all images that cover one part of the sky
- Align and stack them to create a deeper view

Single image next to the result of combining many images
Overview

- Format and mount persistent disk to your VM Instance
- Request a piece of the Universe from Google Lab’s Tile Server
- Initialize the Image Processing Software
- Process your tile data into an image and view it
Formatting and Mounting Disk Space

- Create a mount point for disk, find persistent disk's ID from previous step, format it, mount it, set it to rwx

```bash
sudo mkdir /mnt/disk1
ls -l /dev/disk/by-id/*
sudo /usr/share/google/safe_format_and_mount -m "mkfs.ext4 -F" /dev/disk/by-id/scsi-0Google_PersistentDisk_disk1 /mnt/disk1
sudo chmod 777 /mnt/disk1
```

- Check disk contents (~800GB available)

```bash
sudo df -h --total
```

Filesystem Size Used Avail Use% Mounted on
/mnt/disk1 total 827G 5.6G 782G 1% -

- Then change directories into it

```bash
cd /mnt/disk1
```
Assign Yourself Part of the Universe

• Run the following command to get a tile assignment

```
curl http://compute-codelab.appspot.com/get-tile > tile-assignment.json
```

• Confirm that your tile assignment is valid.

```
cat tile-assignment.json
```

• File contains JSON object with URLs to `.fits` files.

• Using JSON Pretty Print try running this command

```
cat tile-assignment.json | python -mjson.tool
```

```
{
  "b_list": [
    "6534/6/g/calexp/calexp-006534-g6-0055.fits",
    "5823/6/g/calexp/calexp-005823-g6-0709.fits",
    "5902/6/g/calexp/calexp-005902-g6-0727.fits",
    "2700/6/g/calexp/calexp-002700-g6-0169.fits",
  ]
}
```
Setting Up For Tile Processing

- If you did not get a valid JSON response, try to request another assignment.
- Initialize the LSST Software Stack for astronomical image processing (Large Synoptic Survey Telescope)
  - Download the Python processing script and other input files.
  - Change your present directory to the location of LSST Software Stack.

```bash
cd /opt/lsst/
```

- Initialize LSST software for usage.

```bash
source loadLSST.bash
```
Setting Up For Tile Processing

- Setup the LSST afw Python library.
  ```
  setup afw
  ```

- Download `makeCoaddCloud.py` provided by University of Washington and other input files from Google Cloud Storage to the mounted Persistent Disk.
  ```
  gsutil cp gs://codelab-files/* /mnt/disk1/
  ```

- Change directory back to mounted Persistent Disk.
  ```
  cd /mnt/disk1
  ```
Processing a Tile

- Execute the `makeCoaddCloud.py` script to process the JSON in our tile assignment and generate an image.

```
python makeCoaddCloud.py tile-assignment.json --maxImages=20
```

- Note: Start with `--maxImages=20`, increasing by 5 until it successfully completes, if it still fails abandon this tile and go back to the previous step titled "Assigning yourself part of the Universe" to get assigned a new tile.

- If you want to include all images associated with your tile assignment you can run the command without the `--maxImages` flag but be aware that this will require more time for processing.
Processing a Tile

- Find out the image filename for your newly created image by listing the contents of your current directory.

```bash
[wuchangfeng@codelab-node disk1]$ ls -l
Total 252
-rw-rw-r-- 1 wuchangfeng wuchangfeng 157469 May  7 07:16 49094.png
-rw-rw-r-- 1 wuchangfeng wuchangfeng 5743 May  7 07:13 line.py
```
Processing a Tile

- Create a Google Cloud Storage bucket in your project to host your image for viewing in a web browser.
  
  ```
  gsutil mb gs://viewing-images-PROJECT_ID
  ```

- Copy the composite image file you just created to a Google Cloud Storage bucket in your project.
  
  ```
  gsutil cp image-filename.png gs://viewing-images-PROJECT_ID
  ```

- Make the image public (done via GUI previously)
  
  ```
  gsutil acl set public-read gs://viewing-images-PROJECT_ID/image-filename.png
  ```

- View image in a browser
  
  ```
  http://storage.googleapis.com/viewing-images-PROJECT_ID/image-filename.png
  ```
Compute Engine Lab #1

- Compute the Cosmos (54 min)
  - https://codelabs.developers.google.com/codelabs/cloud-compute-the-cosmos
AWS EC2 Lab #1 (CS 510 only)

- Sign up for AWS
  - https://aws.amazon.com/education/awseducate/
  - AWS Account ($100 per year, needs credit card)
  - AWS Educate Starter Account ($75 per year in credits, no credit card required)
AWS Educate is Amazon's program to help students learn real-world cloud technology skills before graduating. It provides students and educators with the resources needed to accelerate cloud-related learning.
<table>
<thead>
<tr>
<th>Portland State University</th>
<th>United States</th>
</tr>
</thead>
<tbody>
<tr>
<td>Portland</td>
<td></td>
</tr>
<tr>
<td>Oregon</td>
<td></td>
</tr>
<tr>
<td>Lightyear</td>
<td></td>
</tr>
<tr>
<td>Computer Science</td>
<td><a href="mailto:blightyear@pdx.edu">blightyear@pdx.edu</a></td>
</tr>
<tr>
<td>Undergraduate-Adv Courses</td>
<td></td>
</tr>
<tr>
<td>06 2018</td>
<td></td>
</tr>
<tr>
<td>7 1993</td>
<td>Promo Code</td>
</tr>
</tbody>
</table>

Please click the box below to help assure that a person and not an automated program is submitting this application. If a set of letters is displayed enter them on the line. If you have any difficulty with the letters, you can click the reload icon to get a new set of letters, or click the headphones to hear audio of what to enter.

I'm not a robot

NEXT
No credit-card?

- Choose AWS Educate Starter account

  □ Click here to enter an AWS Account ID
  
  Approved students are sent a welcome email and benefits including and AWS promotional code.

  Don't have one? Sign up now

  ■ Click here to select an AWS Educate Starter Account
  
  An AWS Educate Starter Account is a free, capped-account that doesn't require a credit card. There are some Usage limitations including an approximately 25% reduction in access to AWS services. Because Starter Accounts are capped, a separate AWS promotional code is not provided.

  Frequently Asked Questions

  With a Starter Account you will receive access to AWS Educate and a lab account with your usage of AWS services capped at the lab amount. A separate AWS promotional credit will not be provided to you.

- Check e-mail and follow the link to the agreement
Agree that...

- AWS gets your first-born if you go over $75 and cannot pay it back

Agreement contents for the user agreement at www.amazon.com. Any references to the "Corporation," "we," or "us" in this Agreement shall be deemed as referring to AISPL. Additionally, if you are located in India, this Agreement shall be deemed to differ from the above provisions as follows:

(a) The Amazon.com Privacy Notice defined in Section 4.1 shall be deemed to refer to the Amazon.in Privacy Notice located at http://www.amazon.in/gp/help/customer/display.html?nodeId=200534380; and

(b) Under Section 9.4, any notice by you to AISPL under this Agreement must be made by registered or certified mail to Amazon Internet Services Private Limited, Ground Floor, Eros Corporate Towers, Nehru Place, New Delhi - 110 019, India (not to Amazon Web Services, Inc.).

You must scroll through the entire Terms and Conditions before accepting or declining.

I Agree  I Decline

After about three days, get free money!
- and then do the following labs...
AWS EC2 Lab #1 (CS 510 only)

- Elastic Compute 10-minute tutorials (do both Linux and Windows tutorials)
  - Links at end of walkthrough
- Goto https://aws.amazon.com
  - Sign-in to console
  - Click on Compute:EC2
• Launch instance on EC2 using an Amazon Linux AMI and a free-tier machine type (t2.micro)
• Review and launch
• Create a new key pair to use to ssh and download it

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.

Create a new key pair

Key pair name

ec2lab

Download Key Pair

You have to download the **private key file** (*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.
- Wait for it to come up.

- Then, using the key you downloaded, ssh into the instance using its external IP address.
• Terminate the instance


• Repeat for Windows
• [URL]: https://aws.amazon.com/getting-started/tutorials/launch-windows-vm/