OPS 813: Cloud Computing

-Today's plan:

0) Textbook Update
1) Homework #3: Time to get another badge
2) Case #2: Stock Market data repository & secondary analysis
3) Docker Containers, Datalab, Cloud SQL

Google Cloud Study Jam – Qwik Labs (HW3)

1) We are going to get you started on your next Quest:
   Data Engineering:
   https://www.qwiklabs.com/quests/25

2) Remember to please follow these instructions:
   https://support.google.com/qwiklabs/answer/9222527?hl=en&ref_topic=9139328 and email me (ns27) the link to your public profile.

HANDS-ON LAB

Weather Data in BigQuery

In this lab you analyze historical weather observations using BigQuery and use weather data in conjunction with other datasets. This lab is part of a series of labs on processing scientific data.

45m Fundamental 5 Credits

HANDS-ON LAB

Analyzing Natality Data Using Datalab and BigQuery

In this lab you analyze a large (137 million rows) natality dataset using Google BigQuery and Cloud Datalab. This lab is part of a series of labs on processing scientific data.

30m Advanced 7 Credits

HANDS-ON LAB

Bigtable: Qwik Start - Hbase Shell

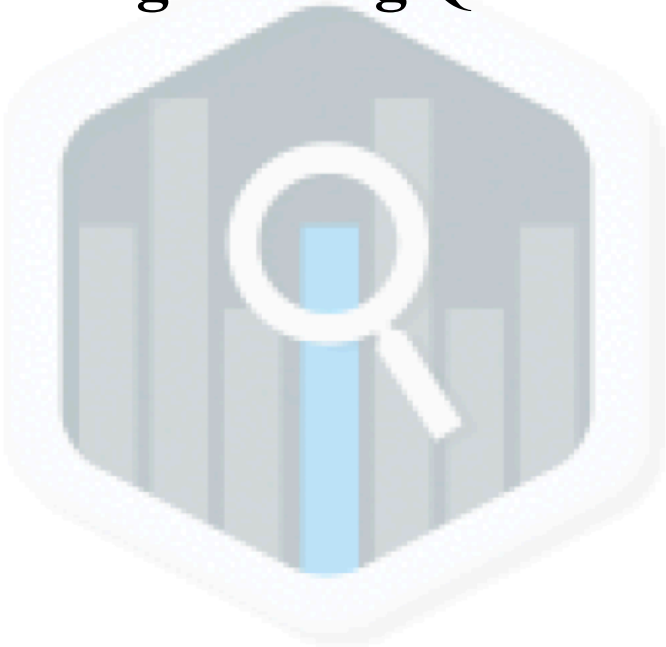
This hands-on lab will show you how to use the HBase shell to connect to a Cloud Bigtable instance. Watch the short video Bigtable: Qwik Start - Qwiklabs Preview.

30m Introductory 1 Credit

HW #3

For your homework #3:

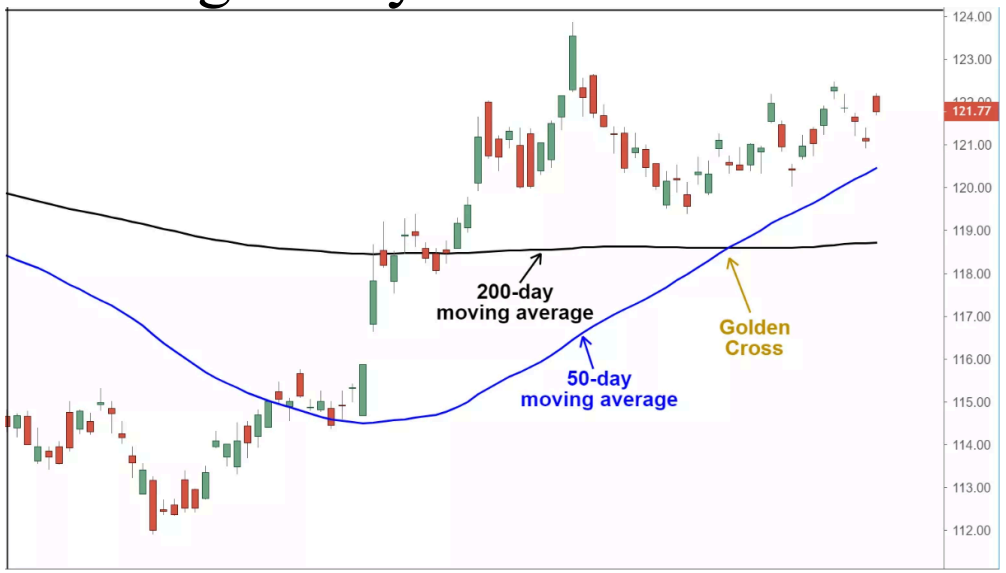
- Please obtain a badge for completing the **Data Engineering** Quest (all the labs in this Quest).
- Read Chapters 6, 7 of textbook



Case #2

For your case #2:

- Please download the stock data from this link to a cloud bucket:
  https://www.kaggle.com/qks1lver/amex-nyse-nasdaq-stock-histories/home
- Put everything into one SQL database
- Then (1) find all stocks that are currently below their 200 day moving average, (2) find all stocks that are currently above their 200 day moving average using Datalab. (3) Also calculate the 50 DMA and see how it relates to the 200 DMA. (4) Is there a correlation between the two and the direction a stock moves? (5) as a group choose 6 or 7 stocks that seem interesting to buy and hold for a month. (6) if you have time, are there are metrics that predict direction?
- Every group will be handing in a report. One group will also present.



Last time: Software on the machine: Using sudo, apt-get, wget, bash

```
ns27@cloudexample:~$ sudo apt-get update
Hit:1 http://us-west1.gce.archive.ubuntu.com/ubuntu xenial InRelease
Get:2 http://us-west1.gce.archive.ubuntu.com/ubuntu xenial-updates InRelease [109 kB]
Get:3 http://us-west1.gce.archive.ubuntu.com/ubuntu xenial-backports InRelease [107 kB]
Hit:4 http://archive.canonical.com/ubuntu xenial InRelease
Hit:5 http://security.ubuntu.com/ubuntu xenial-security InRelease
Fetched 216 kB in 0s (384 kB/s)
Reading package lists... Done
```

If you visit https://repo.continuum.io/archive/ you will see there are many versions of Anaconda for various platforms.

Magic commands:

wget https://repo.continuum.io/archive/Anaconda3-5.3.1-Linux-x86_64.sh

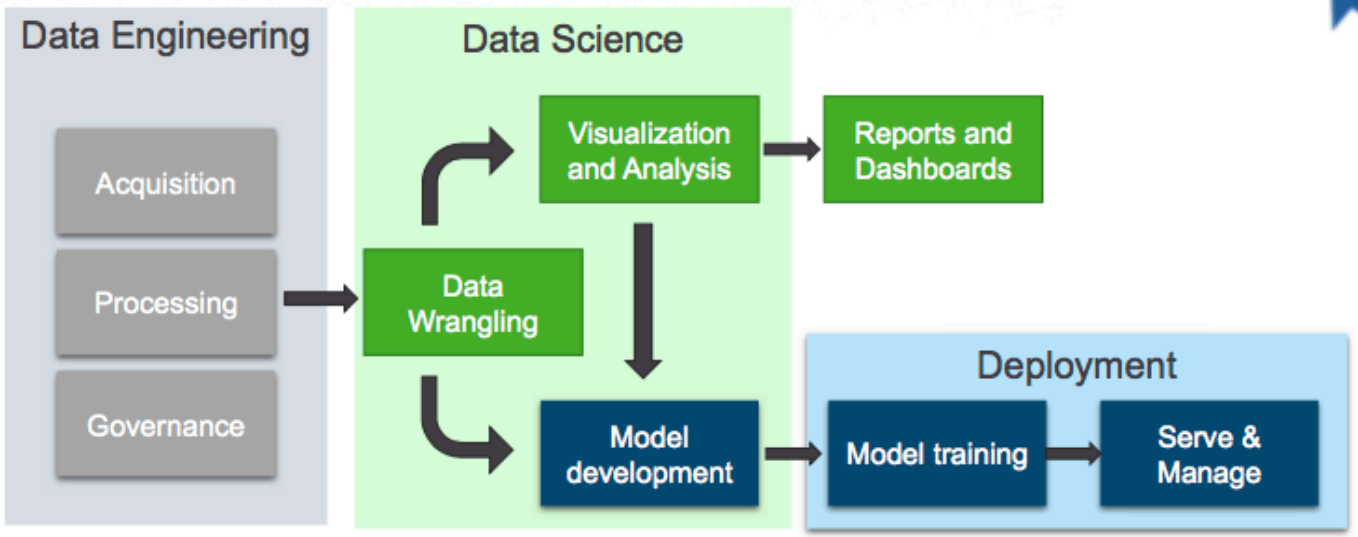
bash Anaconda3-5.3.1-Linux-x86_64.sh

```
installation finished.
Do you wish the installer to initialize Anaconda3
in your /home/ns27/.bashrc ? [yes|no]
[no] >>> yes
```

Docker

A conventional issue in Data Analytics is how do we go around this circle better in terms of quality and speed:

Reproducibility and *portability of compute environment* are critical in analytics and a new challenge is *deployment*:



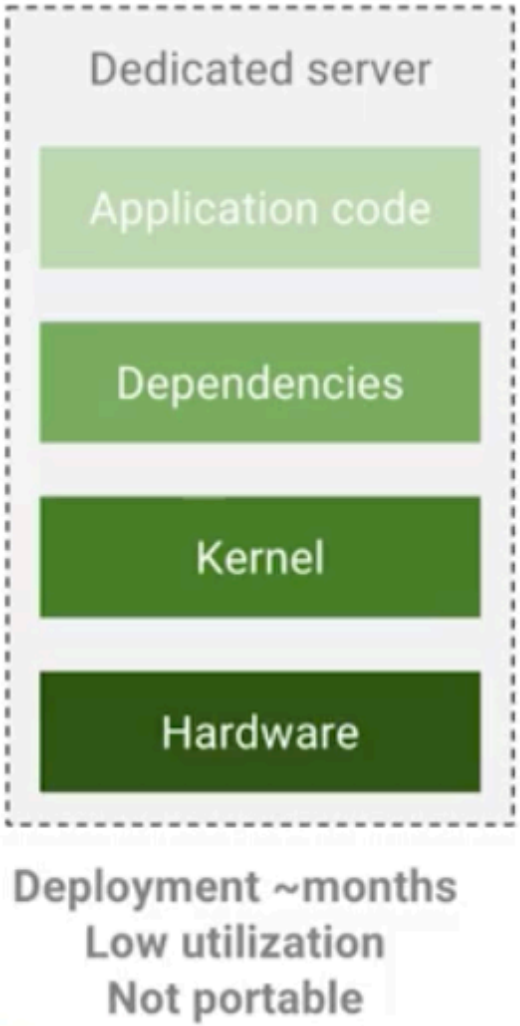
Higher Level of Abstraction: Cloud Native

A new mindset is to have your workloads run in the cloud with as little maintenance and configuration as possible. This is how Google, Netflix, Amazon, and other companies work today.

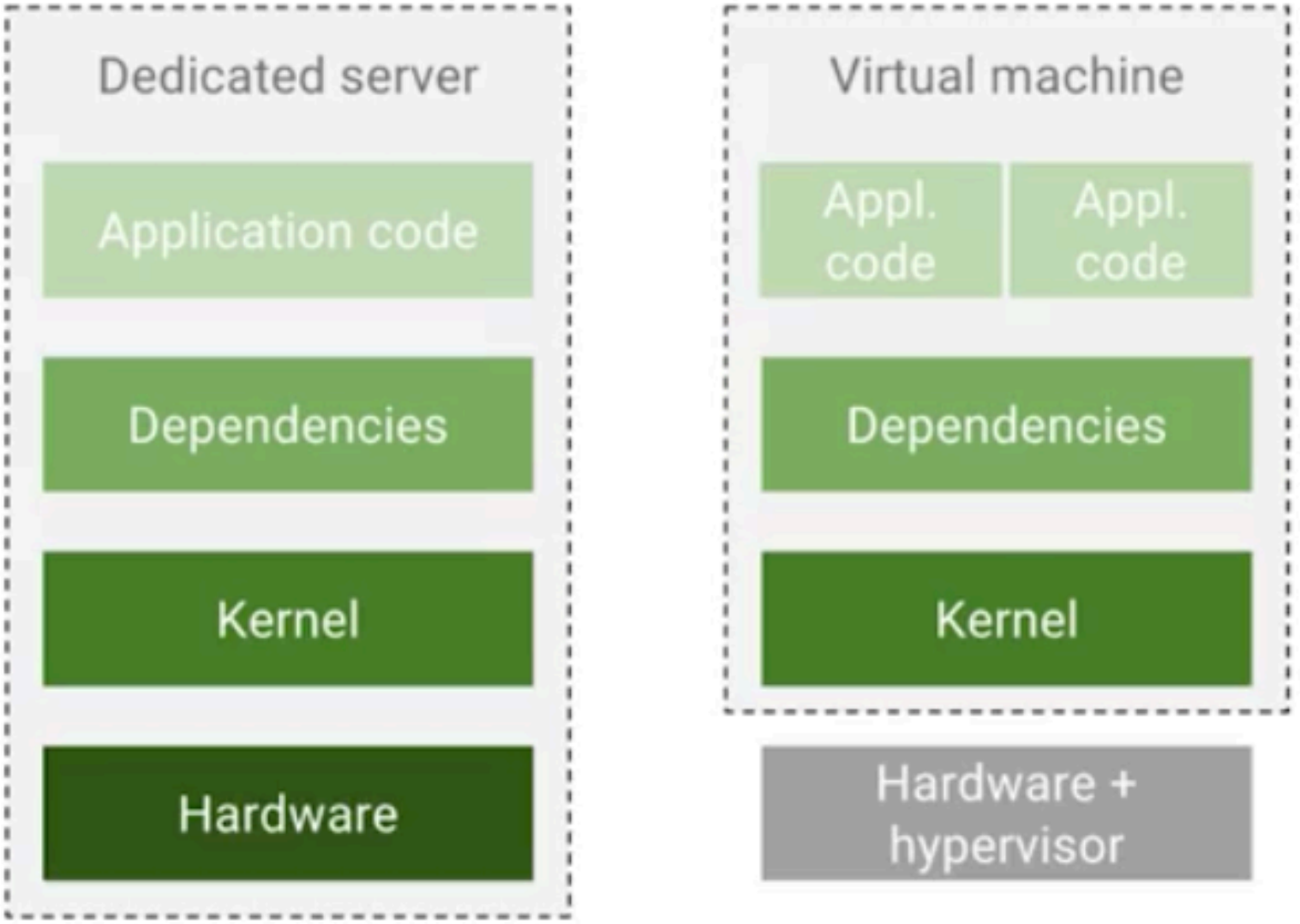
1. Container-based	[Docker] Container as the unit of isolation and scale
2. API-oriented	Loosely-coupled components talk via APIs in a distributed system
3. Dynamically orchestrated	Applications are dynamic and organic: they grow, shrink and adapt

What is a container?

Containers isolate programs from each other and the underlying operating system. They let you move them across systems without reconfiguration. Some history & dependencies:



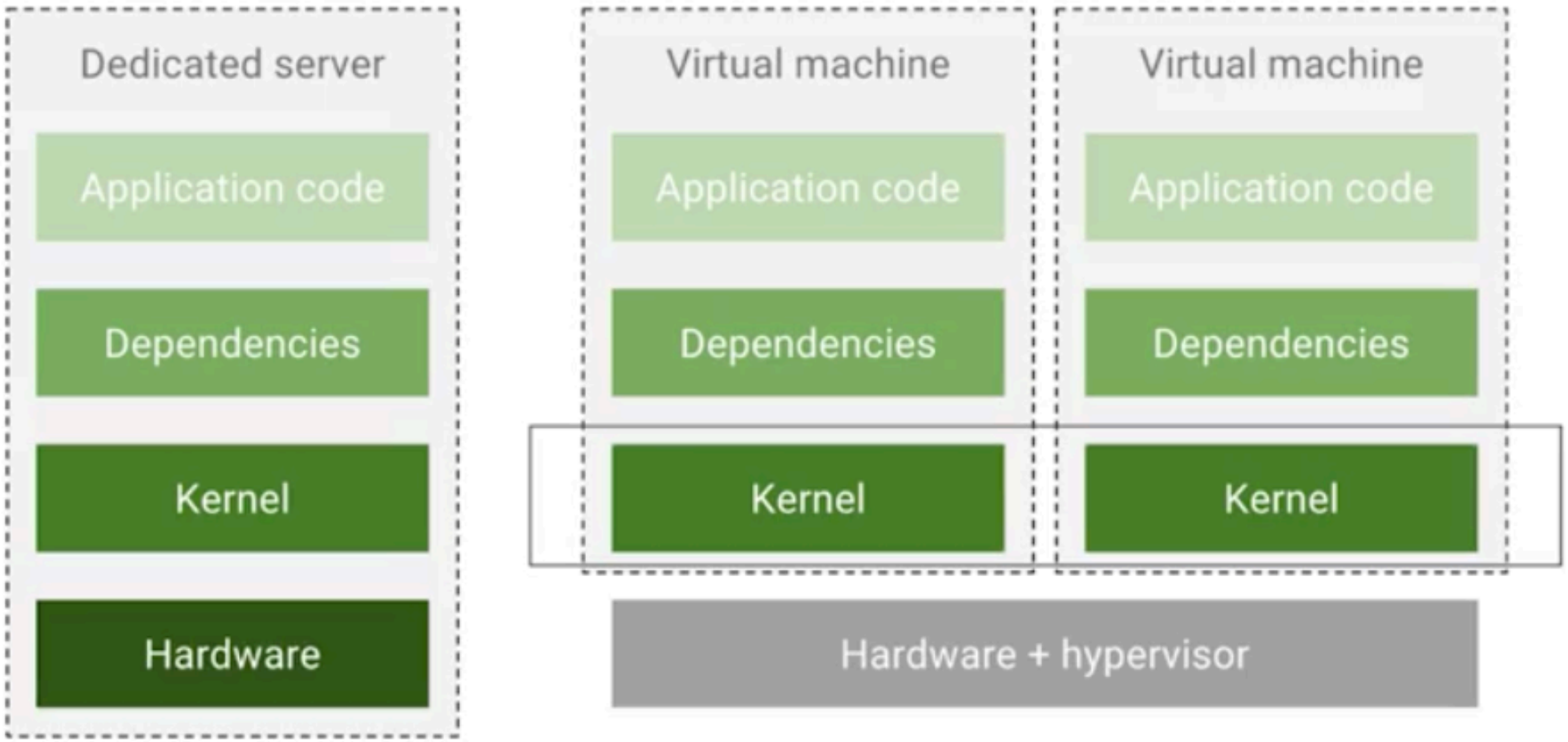
What is a container?



Deployment ~months
Low utilization
Not portable

Deployment ~days (mins)
Hypervisor-specific
Low isolation; tied to OS

What is a container?



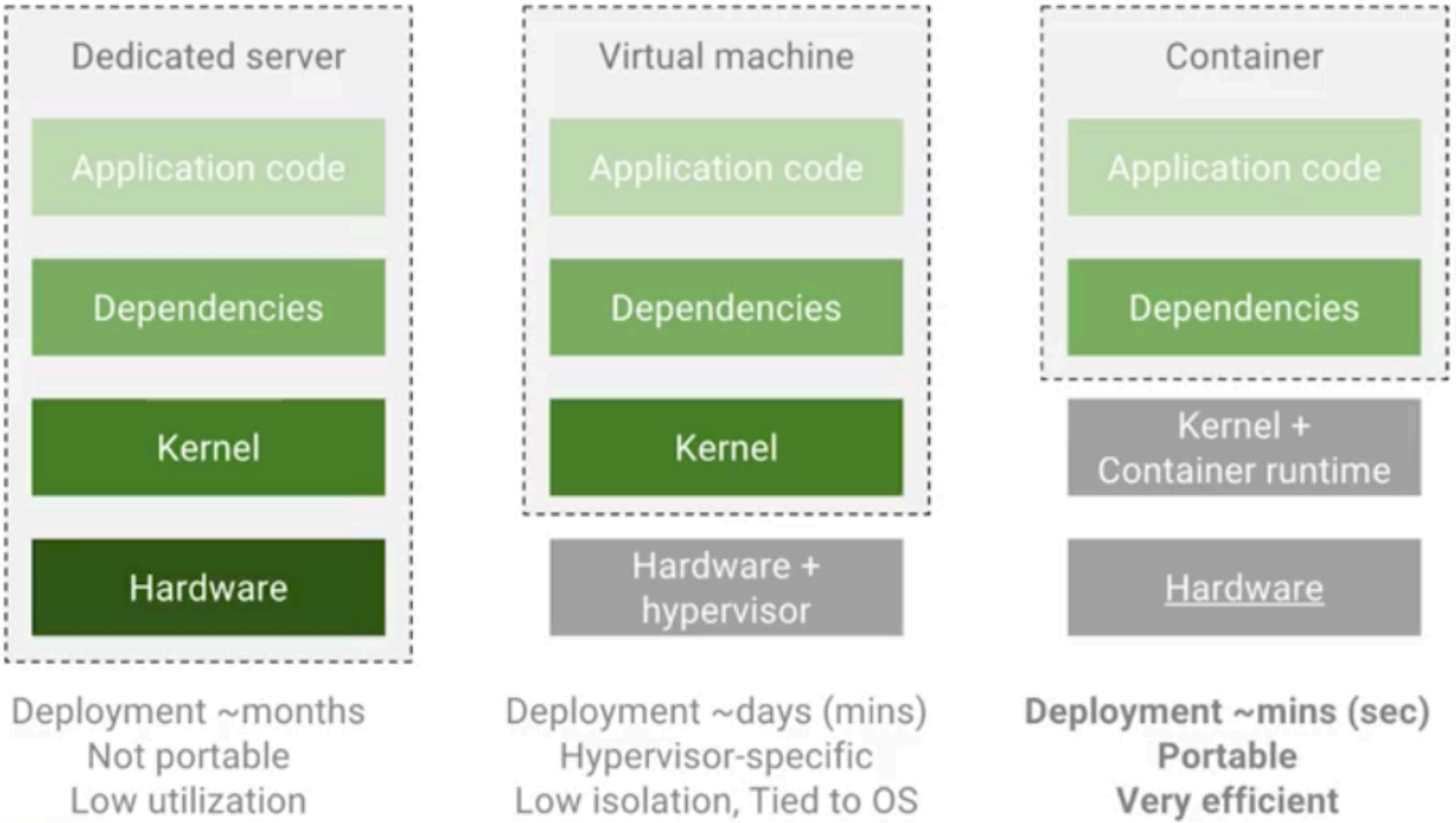
Deployment ~months
Not portable
Low utilization

Deployment ~days (mins)
Hypervisor-specific
Low isolation; tied to OS

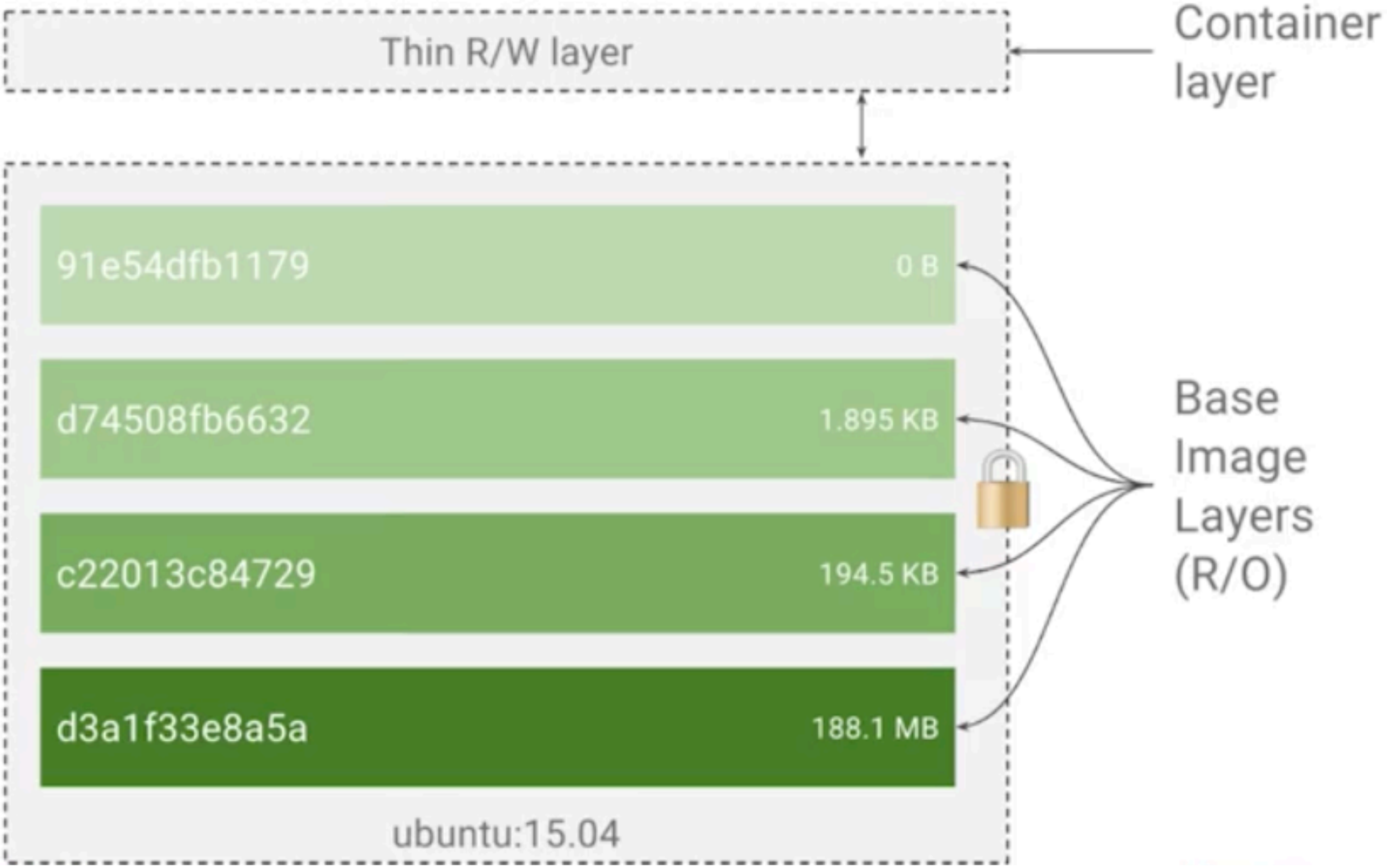
Deployment ~days (mins)
Hypervisor-specific
Redundant OS

A container is a different level of abstraction

Decoupled from Operating System and Hardware (Docker ~ 2013):



Containers use a layered file system with only top layer writeable



Docker – other details

- A docker configuration file is needed to build the container image.

Container lifecycle:

- Build Docker image
- Create and run a container from your Docker image
- Interact with your container by attaching a terminal session
- Save the state of container as new image
- List running/non-running containers
- List all images
- Push your image to a registry (e.g., DockerHub or Google's)

Jupyter vs Cloud Datalab

With Jupyter you have to download, install, and configure the environment. Datalab is a hosted container version of Jupyter on GCP.

To create:

- Open CloudShell and create a compute instance by running:
  datalab create --zone $ZONE $INSTANCE_NAME
  (replace $ZONE and $INSTANCE_NAME, e.g., us-central1-a)
  (use --no-backups if you don't want a backup every 10 minutes)

To connect:

- Use the Web Preview button in CloudShell.

To reconnect (if laptop goes to sleep):

- Type the following in CloudShell:
  datalab connect --zone $ZONE $INSTANCE_NAME

To delete the instance (by default doesn't delete disk):

- Type the following in CloudShell:
  datalab delete --zone $ZONE $INSTANCE_NAME

Cloud Datalab Issues & Billing

Potential Issues:

```
ns27@cloudshell:~ (ops-813-silicon-valley)$ datalab create --zone us-central1-a mytest
Creating the network datalab-network
Creating the firewall rule datalab-network-allow-ssh
Creating the disk mytest-pd
ERROR: (gcloud.source.repos.list) User [ns27@stmarys-ca.edu] does not have permission to access project [ops-813-silicon-valley] (or it may
not exist): Cloud Source Repositories API has not been used in project 1071223620137 before or it is disabled. Enable it by visiting https:/
/console.developers.google.com/apis/api/sourcerepo.googleapis.com/overview?project=1071223620137 then retry. If you enabled this API recentl
y, wait a few minutes for the action to propagate to our systems and retry.
- '@type': type.googleapis.com/google.rpc.Help
  links:
  - description: Google developers console API activation
    url: https://console.developers.google.com/apis/api/sourcerepo.googleapis.com/overview?project=1071223620137
A nested call to gcloud failed, use --verbosity=debug for more info.
```

If you ever want to disable the api later:

https://console.developers.google.com/apis/api/sourcerepo.googleapis.com/overview?project=ops-813-silicon-valley

If you want to stop the instance (not delete):

- Type the following in CloudShell:
  datalab stop --zone $ZONE $INSTANCE_NAME
  (if you want to reconnect, please use the connect command)

Want to update your Cloud Datalab container?

datalab delete --keep-disk ***instance-name***

datalab create ***instance-name***

Want to delete your attached persistent disk too? (otherwise charges)

datalab delete --delete-disk ***instance-name***

Datalab is a docker container

Let's ssh into VM running Datalab:

gcloud compute --project *project-id* ssh --zone *zone instance-name*

(in order to remember the information type: datalab list
after ssh-ing in, see if there are any files on the cloud instance)

See all docker processes running:

docker ps

Open interactive shell session inside container:

docker exec -it *container-id* bash

(remember if you want help, just ask: docker exec --help)

Now explore:

ls

cd /content/datalab/notebooks

Getting files in or out:

ls

cd /content/datalab/notebooks (this is the location for the git repo, you can type: git status)

Getting files in and out of the docker container

You can use git (or ungit)

When you run "datalab create *VM-instance-name*" for the first time, it adds a datalab-notebooks Cloud Source Repository in the project:
https://cloud.google.com/source-repositories/

This is a remote repository for the /content/datalab/notebooks git repository created in the docker container running in your Cloud Datalab VM instance. You can browse the cloud remote repo from the Google Cloud Platform Console Repositories page:
https://console.cloud.google.com/code/develop/repo

Copying it to a directory in your instance

gcloud compute scp --recurse \
datalab@***instance-name***:/mnt/disks/datalab-pd/content/datalab/notebooks \
instance-name-notebooks

Other places

Google Cloud Storage: file and directory access using datalab.storage api,
BigQuery: tables and views can be queried, local file system on VM

Git and Ungit

"Git is known for being a versatile distributed source control system that is a staple of many individuals, communities, and even for the City of Chattanooga to crowd source bicycle parking locations:

However, it is not known for user friendliness or an easy learning curve.

Ungit brings user friendliness to git without sacrificing the versatility of git."

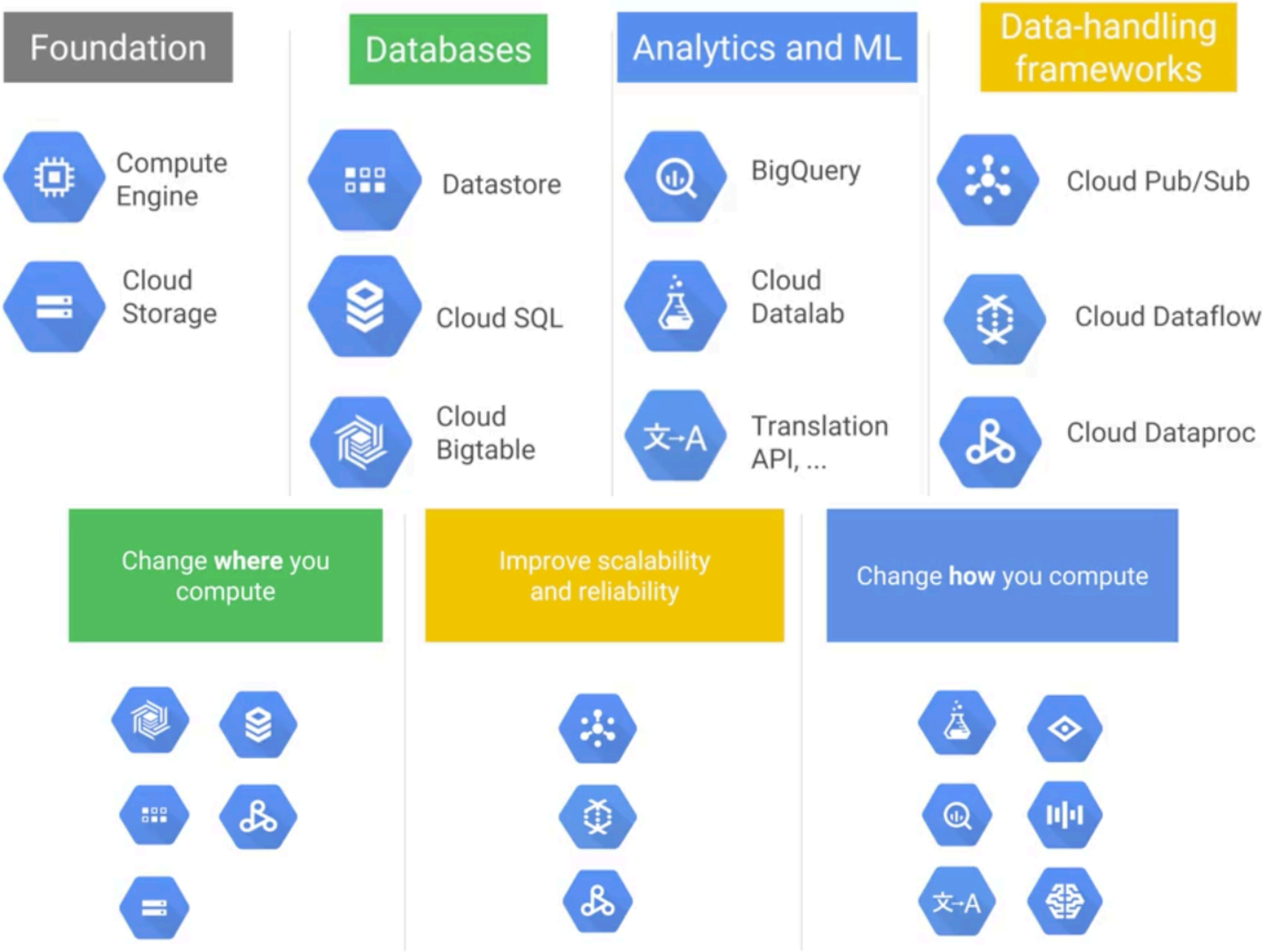
Datalab

What is installed?

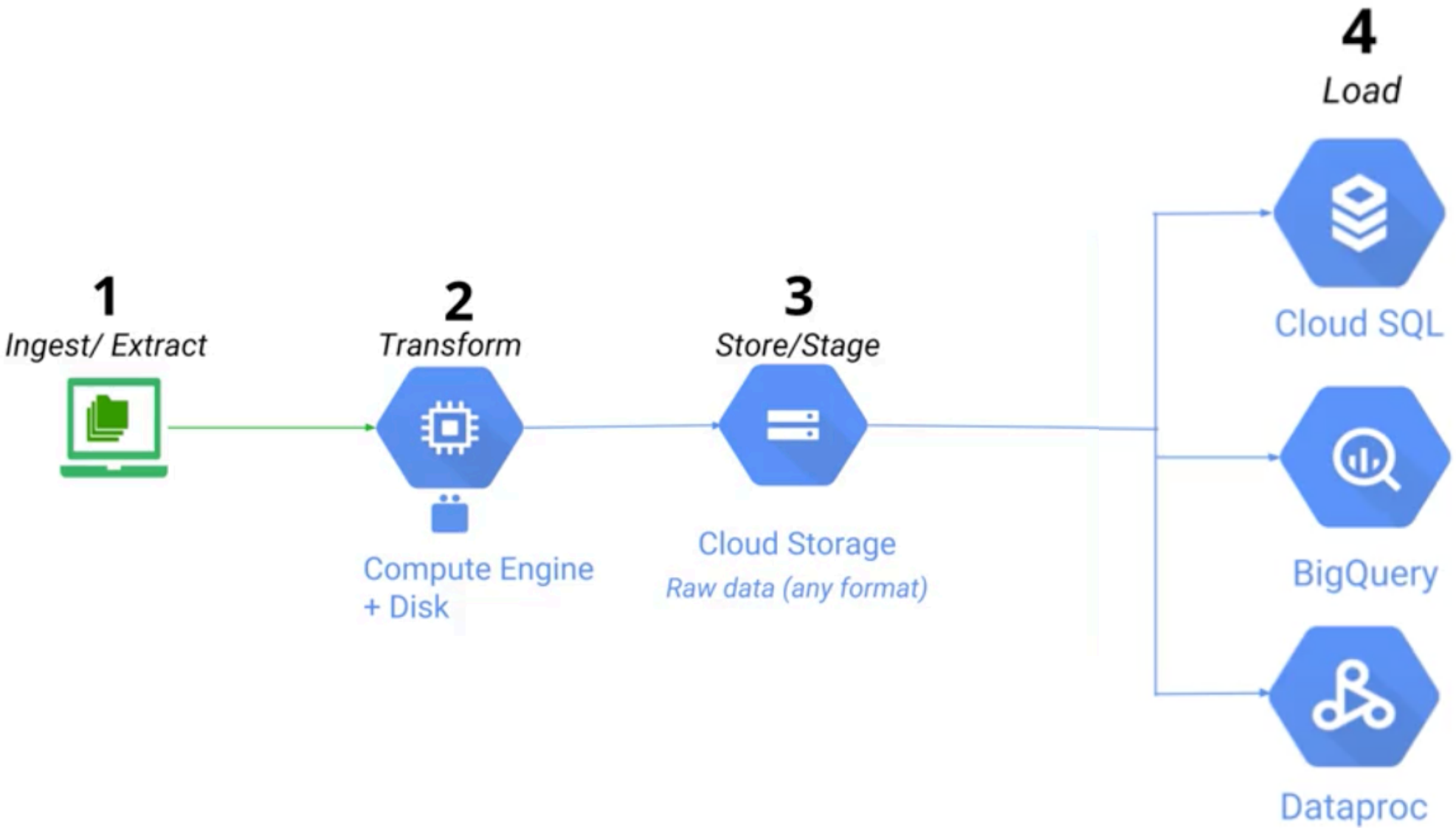
!pip list
!pip install *lib-name*

Bash in a cell?

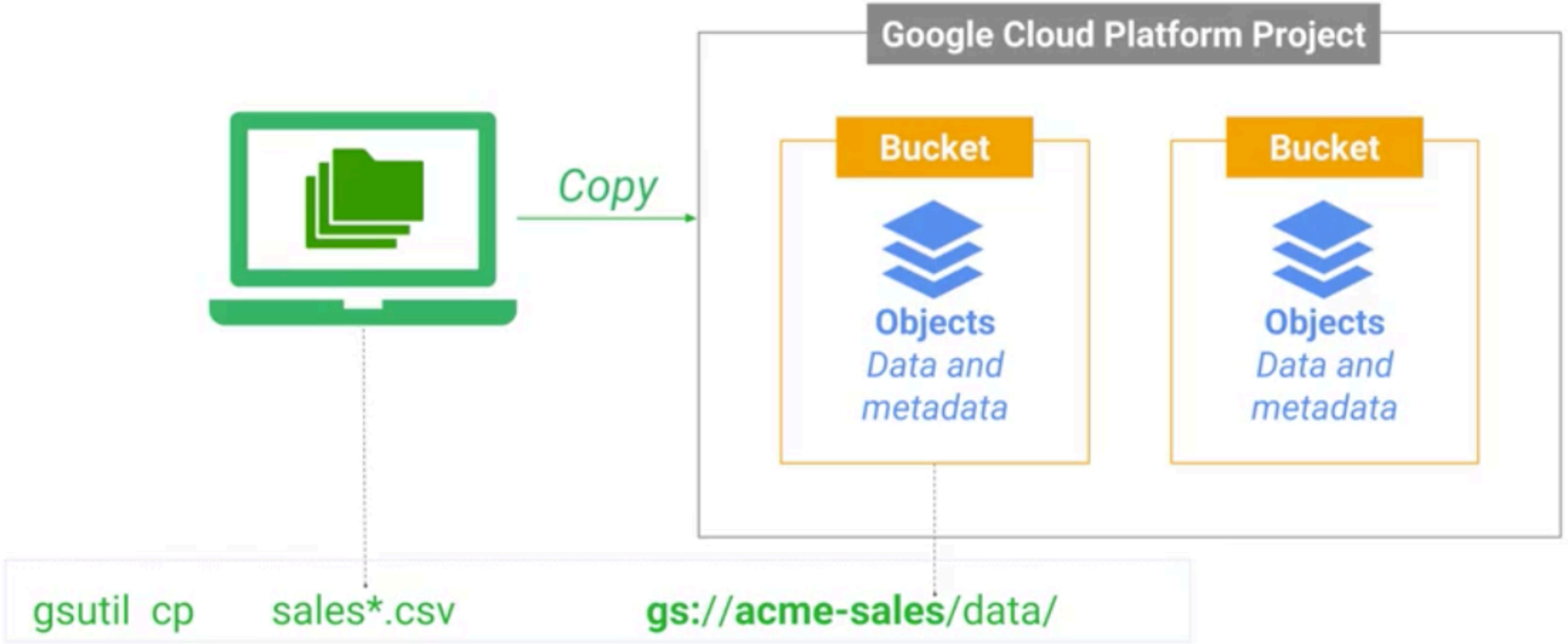
%%bash (also try %%html)


Foundation
Compute Engine
Cloud Storage
Databases
Datastore
Cloud SQL
Cloud Bigtable
Analytics and ML
BigQuery
Cloud Datalab
Translation API, ...
Data-handling frameworks
Cloud Pub/Sub
Cloud Dataflow
Cloud Dataproc
Change where you compute
Improve scalability and reliability
Change how you compute

Data Lifecycle



Copying files to Cloud Storage (GCS)



Earthquake Case Study – Steps we will follow

- upload mtg03.zip to your instance
- ssh into your instance and unzip mtg03
- move into the earthquakes-gcs folder
- inspect and run the ingest.sh script
- inspect the downloaded data
- let's transform the data for a map but first install_missing python packages
- now transform the data (careful which python thy uses)

- let's create a bucket to store on GCS
- copy data from instance to bucket:
  gsutil cp earthquakes.* gs://gcs-lab/earthquakes/
- check that your bucket has the files
- make png available to everyone
  (three dots, Name=allUsers)
- now open up the csv file in pandas, using Datalab

Accessing files in Datalab from GCS

```
!pip install gcsfs

import pandas as pd
import gcsfs
fs = gcsfs.GCSFileSystem(project='my-project')
with fs.open('bucket/path.csv') as f:
    df = pd.read_csv(f)
```

or

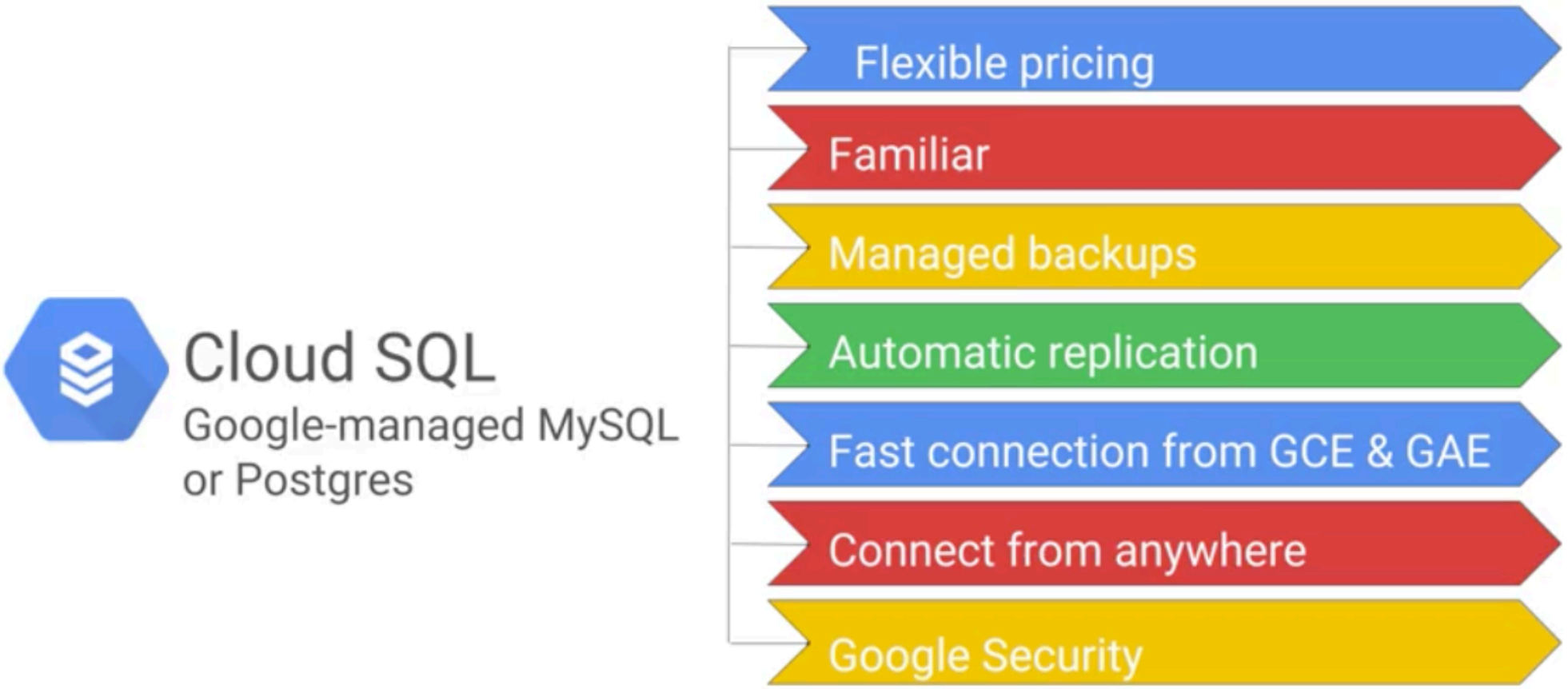
```
# look into dask, https://dask.org/
import dask.dataframe as dd
df = dd.read_csv('gs://bucket/data.csv')
df2 = dd.read_csv('gs://bucket/path/*.csv') # nice!

# df is now Dask dataframe, ready for distributed processing
# If you want to have the pandas version, simply:
df_pd = df.compute()
```

SQL Database in the Cloud

	Cloud Storage	Cloud SQL	Datastore	Bigtable	BigQuery
Capacity	Petabytes +	Gigabytes	Terabytes	Petabytes	Petabytes
Access metaphor	Like files in a file system	Relational database	Persistent Hashmap	Key-value(s), HBase API	Relational
Read	Have to copy to local disk	SELECT rows	filter objects on property	scan rows	SELECT rows
Write	One file	INSERT row	put object	put row	Batch/stream
Update granularity	An object (a "file")	Field	Attribute	Row	Field
Usage	Store blobs	No-ops SQL database on the cloud	Structured data from AppEngine apps	No-ops, high throughput, scalable, flattened data	Interactive SQL* querying fully managed warehouse

What is Cloud SQL?



Rentals Case Study

- upload mtg03.zip to your Cloud Shell (not your instance)
- start your Cloud Shell and unzip mtg03
- move into the rentals-sql folder (within the mtg03 folder)
- check out the files in the cloudsql folder
- try:
  head *.csv
- move these files into a bucket, how?

- now create a Cloud SQL instance named rentals
  (when creating instance, choose second generation, and authorize networks under show configuration options, run script to ip address) – note the password
- keep track of public IP address for SQL instance
- go into the SQL instance and click import, and select SQL file
- go into SQL instance and click import again, and use recommendation spark for the csv files

Rentals Case Study

- You can now use the mysql cli from your Cloud Shell:
  mysql --host=PUBLIC_IP –user=root --password
- Enter your password

- SQL time:
  use recommendation_spark;
  show tables;
  select * from Rating;
  select * from Accomodation where type = ‘castle’ and price < 1500;