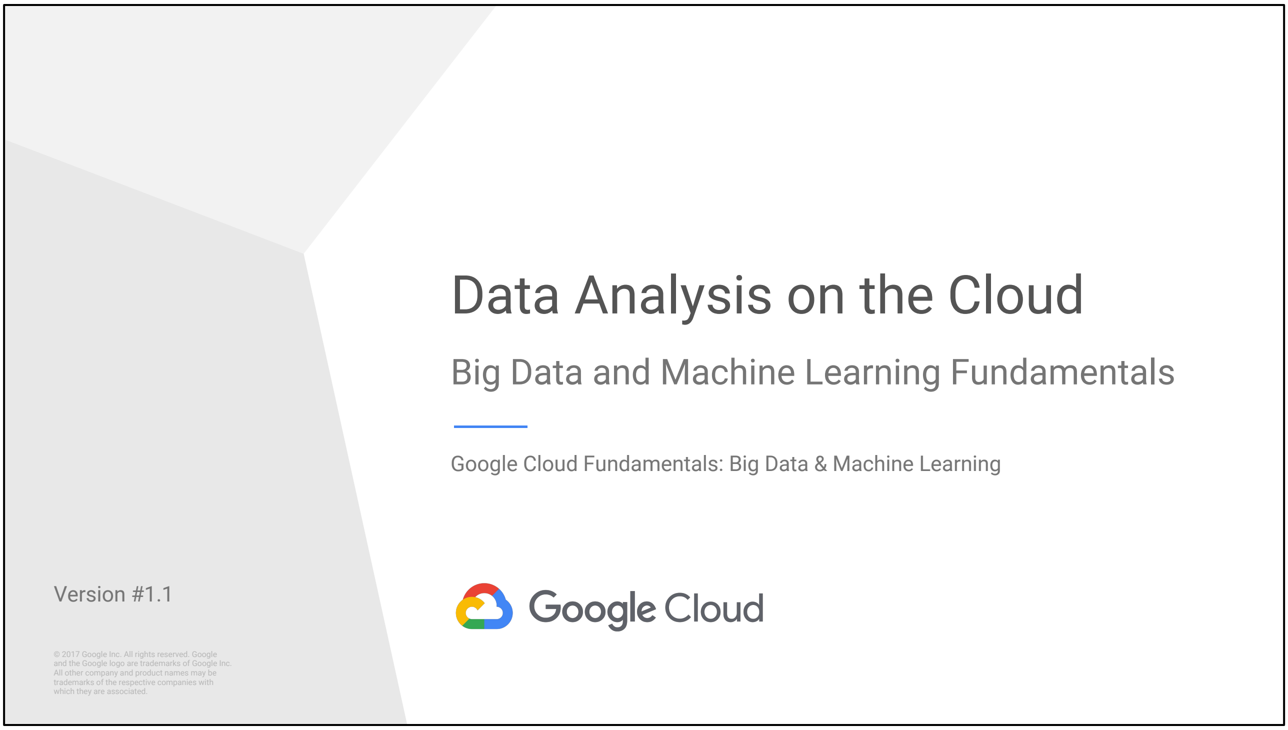


Data Analysis on the Cloud

Big Data and Machine Learning Fundamentals

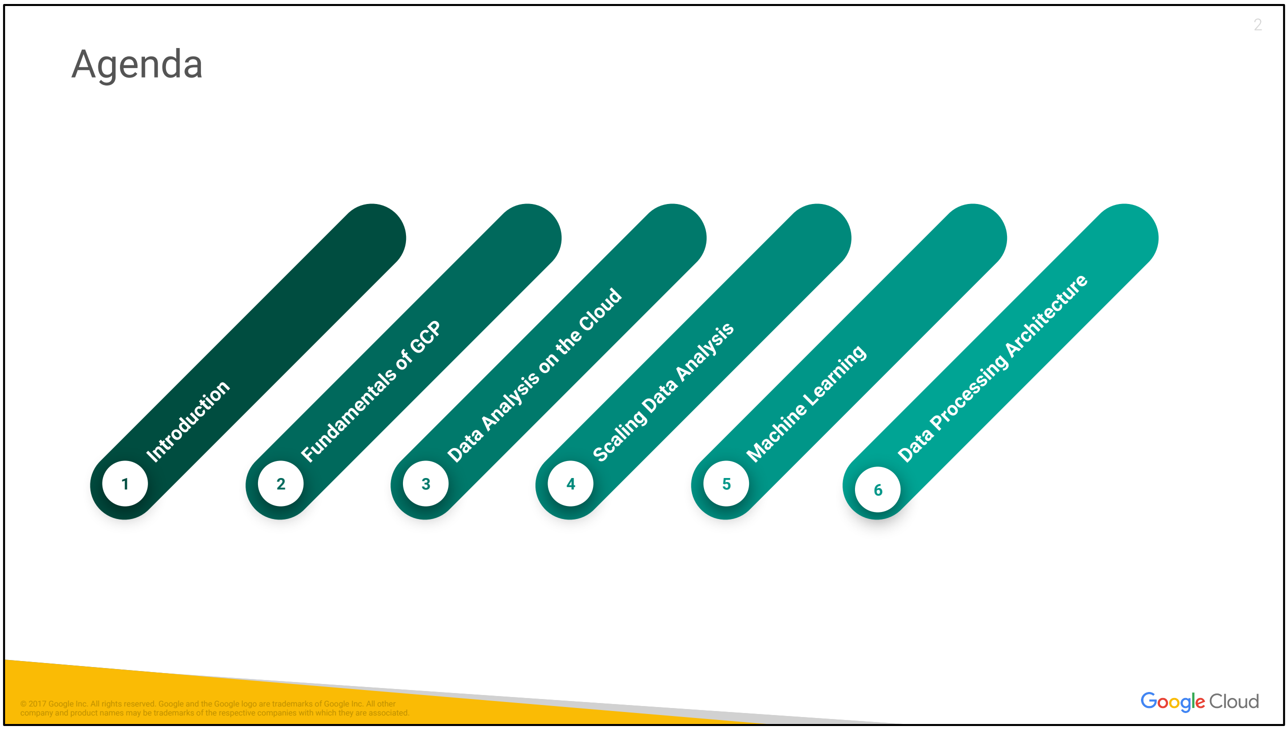
Google Cloud Fundamentals: Big Data & Machine Learning

Version #1.1

Google Cloud

Notes:

60 minutes lecture + 30 minutes lab



Agenda

1 Introduction

2 Fundamentals of GCP

3 Data Analysis on the Cloud

4 Scaling Data Analysis

5 Machine Learning

6 Data Processing Architecture

Notes:

1. Introduction

Overview of GCP as a whole, but with emphasis on the data-handling aspects of the platform

- GCP, GCP Big Data
- Usage scenarios
- Create an account on GCP

2. Foundation of GCP

Compute and Storage with a focus on their value in data ingest, storage, and federated analysis

- Compute Engine
- Cloud Storage
- Start GCE instance
- Upload data to GCS

3. Data analytics on the Cloud

Common use cases that Google manages for you and for which there is an easy migration path to the Cloud

- Cloud SQL
- Dataproc
- Import data into and query Cloud SQL
- Machine Learning with Dataproc

In the morning, we will complete Modules 1 and 2 and get halfway through Module 3.

4a. Scaling data analysis

Change how you compute, not just where you compute with GCP

- Datalab
- Datastore, Big Table
- BigQuery

5. TensorFlow

Change how you compute, not just where you compute with GCP

- TensorFlow
- Datalab instance
- BigQuery
- Demand forecasting with ML

6. Data processing architectures

Scaleable, reliable data processing on GCP

- Pub/Sub
- Dataflow

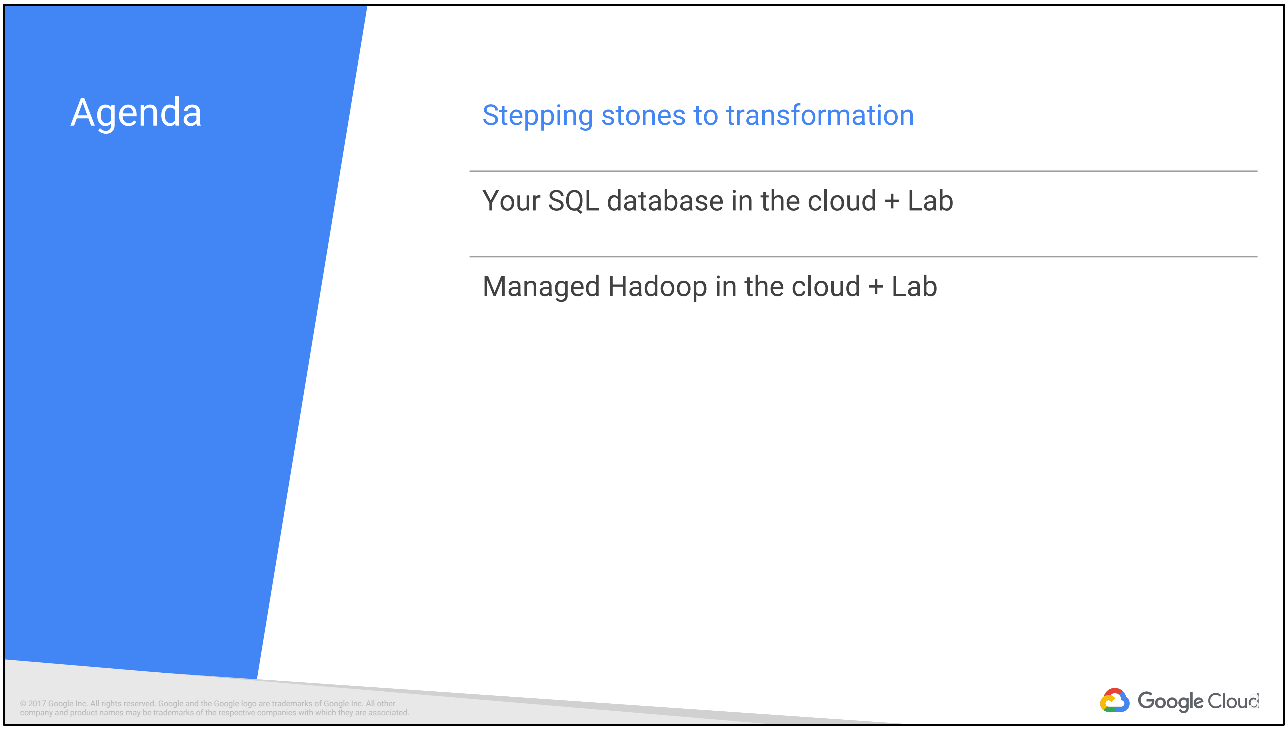
7. Summary

Course summary

- Resources

Please feel free to use the appendixes for self-study.
In the morning, we will get halfway through Module 3.

Please feel free to use the appendixes for self-study.



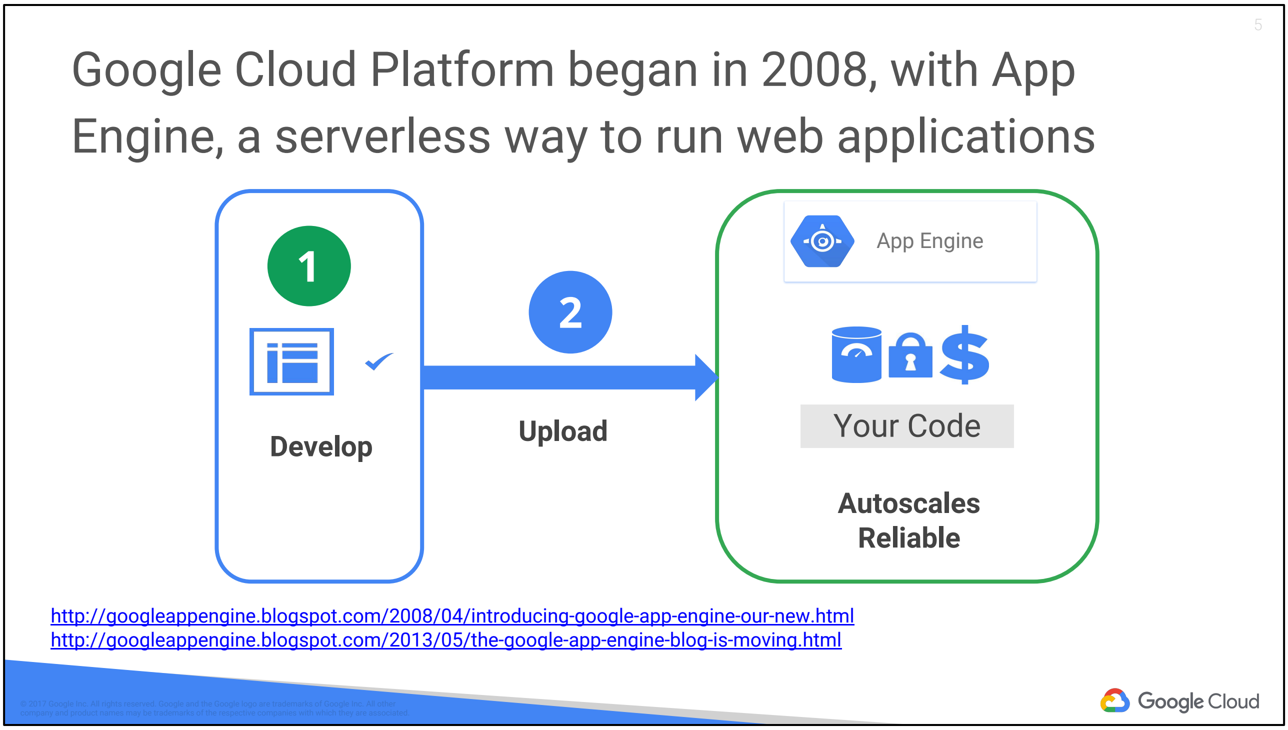
Agenda

Stepping stones to transformation

Your SQL database in the cloud + Lab

Managed Hadoop in the cloud + Lab

Google Cloud



Google Cloud Platform began in 2008, with App Engine, a serverless way to run web applications

1

Develop

2

Upload

App Engine

Your Code

Autoscales
Reliable

http://googleappengine.blogspot.com/2008/04/introducing-google-app-engine-our-new.html
http://googleappengine.blogspot.com/2013/05/the-google-app-engine-blog-is-moving.html

Google Cloud

Notes:
http://googleappengine.blogspot.com/2008/04/introducing-google-app-engine-our-new.html
http://googleappengine.blogspot.com/2013/05/the-google-app-engine-blog-is-moving.html

Why are we starting off with web applications in a module about data analysis? Because it helps tell the story of why we have Cloud SQL and Hadoop support in Google Cloud Platform. Understanding the long-term arc of Google Cloud Platform is helpful. App Engine is a no-ops environment for web applications. Similarly, BigQuery is no-ops data analytics and Bigtable is our noSQL datastore. Why would anyone use anything else? Three years ago, Google would have just built these and asked customers to adapt.

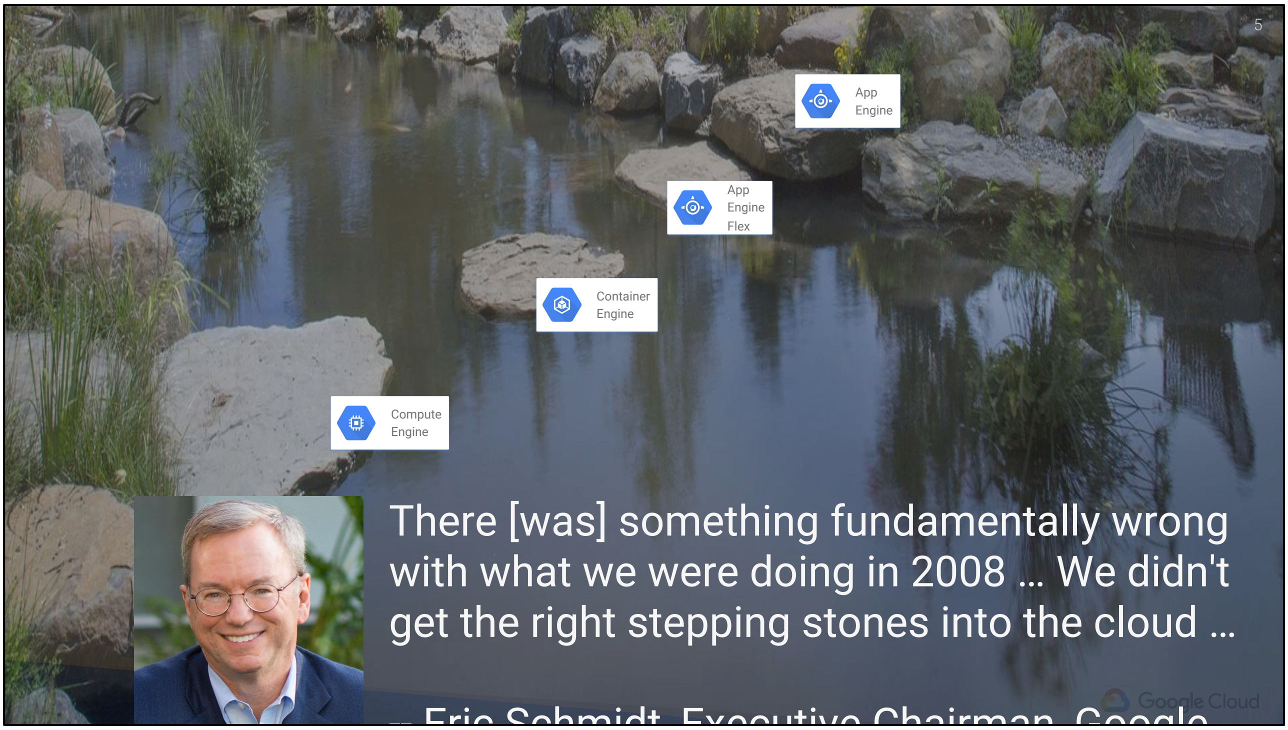
The steps to develop and deploy and app:

1. **Develop** a web application using a supported language and App Engine SDK
2. **Upload** to App Engine

Then:
App Engine automatically **scales & reliably** serves your web application
App Engine manages the runtimes of your code.

Now, though, we meet our customers where they are, and that means building support for SQL databases (even though we know they don't scale, and many of those use cases can be solved with Datastore) and Hadoop (even though we know BigQuery is a better solution).

App Engine with its promise of no-ops was ahead of the market for cloud apps. Businesses understood moving their compute jobs to the cloud as-is for the capital expense cost-savings, but the whole idea of no-ops and autoscaling was ahead of the market. It’s taken 8 years for the market to catch up. Supported languages for App Engine include: Python, Java, Go, PHP, and Node.js. Download and use the App Engine SDK to develop, test and deploy your code.



App Engine

App Engine Flex

Container Engine

Compute Engine

There [was] something fundamentally wrong with what we were doing in 2008 … We didn't get the right stepping stones into the cloud …

-- Eric Schmidt, Executive Chairman, Google

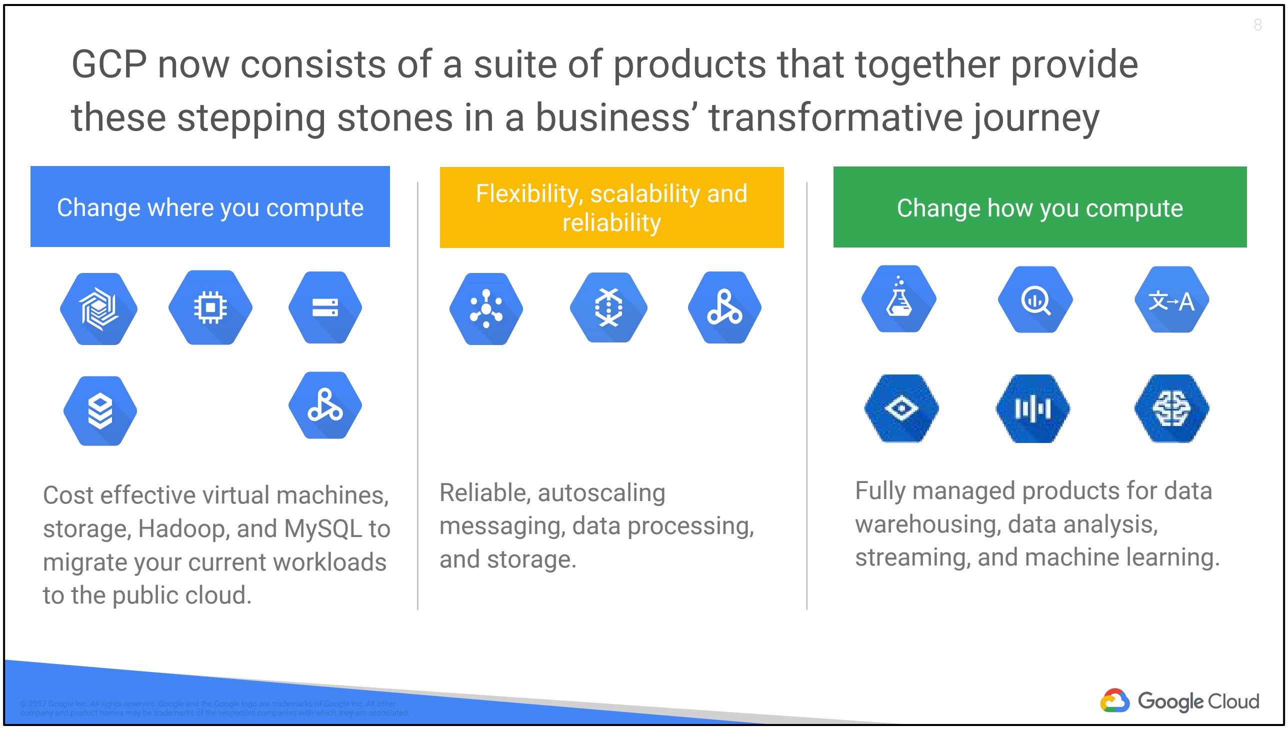
Google Cloud

Notes:
https://pixabay.com/en/natural-pond-stepping-stones-2428106/ (cc0)

We wanted people to just go straight to the transformational stuff. Now, we provide products that help meet our customers where they are. See: http://www.informationweek.com/cloud/infrastructure-as-a-service/google-pumps-up-cloud-platform-with-machine-learning/d/d-id/1324822.

Image from Eric Schmidt’s Google Plus profile.

Of course, the story is not as simple -- even as we were building stepping stones, we were also enhancing AppEngine, building AppEngine Flex, and cross-pollinating features in between. The future is going to be better and more integrated, but we still need to meet customers where they are.



GCP now consists of a suite of products that together provide these stepping stones in a business' transformative journey

Change where you compute

Cost effective virtual machines, storage, Hadoop, and MySQL to migrate your current workloads to the public cloud.

Flexibility, scalability and reliability

Reliable, autoscaling messaging, data processing, and storage.

Change how you compute

Fully managed products for data warehousing, data analysis, streaming, and machine learning.

Google Cloud

Notes:

A more expanded version of a similar slide in Ch 1; we are adding details here.

Icons:
Ist column: Bigtable, Storage, SQL, Dataproc for Hadoop workloads
2nd column: Pub/Sub, Dataflow, Dataproc
3rd column: Datalab, BigQuery, Translation API, Vision API, Speech API, Cloud Machine Learning

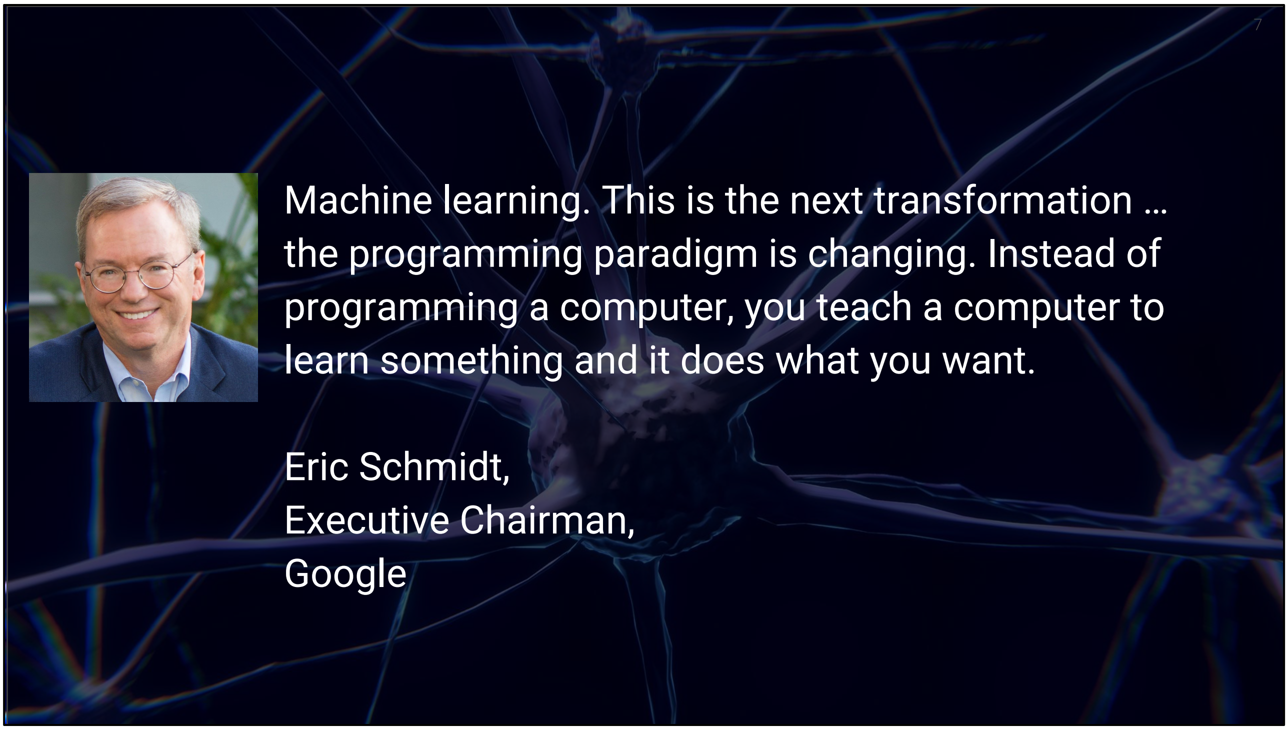
Change where you compute:
Databases, Storage, and Hadoop
Cloud Databases for different needs (relational, key-value, NoSQL) Cloud SQL, Cloud Datastore, Cloud Bigtable
Proven storage platform Cloud Storage: Standard, Durable Reduced Availability
Managed Hadoop/Spark/Pig/Hive Cloud Dataproc

Scalability and Reliability:
Messaging and Data processing
Reliable, large scale messaging Cloud Pub/Sub
Flexible, scalable and reliable data processing Cloud Dataflow, Cloud Dataproc

Change how you compute:

Exploration, analytics and intelligence
Data exploration and business intelligence Cloud Datalab, Cloud Data Studio
Fast & economical data warehouse for large-scale data analytics Google BigQuery
Machine learning Cloud Machine Learning, Vision API, Speech API, Translate API

The previous slide was on the Compute side. On the data side, we know that the end-game for many users will be BigQuery and Cloud Machine Learning, but the stepping stones are Cloud SQL and Cloud Dataproc.



Machine learning. This is the next transformation … the programming paradigm is changing. Instead of programming a computer, you teach a computer to learn something and it does what you want.

Eric Schmidt,
Executive Chairman,
Google

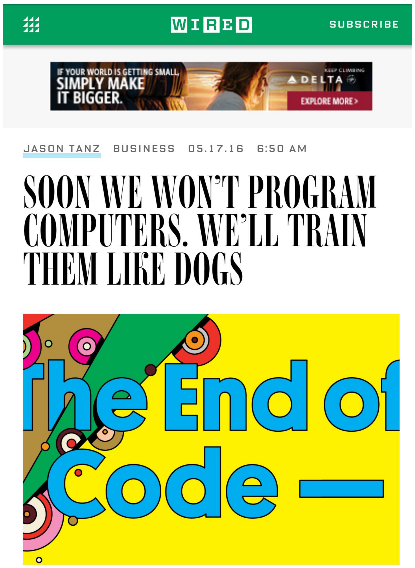
Notes:

http://www.businessinsider.com/eric-schmidt-smart-computers-will-create-wealth-2016-3.

Eric also said this. Use this slide to introduce machine learning -- the process of teaching a computer with data.

WIRED’s headline

“If you want to teach a neural network to recognize a cat, for instance, you don’t tell it to look for whiskers, ears, fur, and eyes. You simply show it thousands and thousands of photos of cats, and eventually it works things out.”



JASON TANZ BUSINESS 05.17.16 6:50 AM

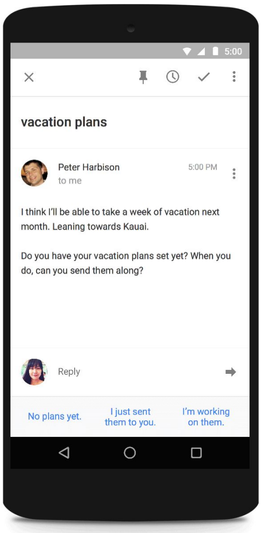
SOON WE WON’T PROGRAM COMPUTERS. WE’LL TRAIN THEM LIKE DOGS

Google Cloud

Notes:

The headline is very cool though. We’ll train computers like dogs to recognize cats!

Machine Learning is not new, but it is now mainstream



vacation plans

Peter Harbison
to me
5:00 PM

I think I'll be able to take a week of vacation next month. Leaning towards Kauai.

Do you have your vacation plans set yet? When you do, can you send them along?

Reply

No plans yet. | I just sent them to you. | I'm working on them.

Search

People who bought ...

Spam filtering

Suggest next video

Route planning

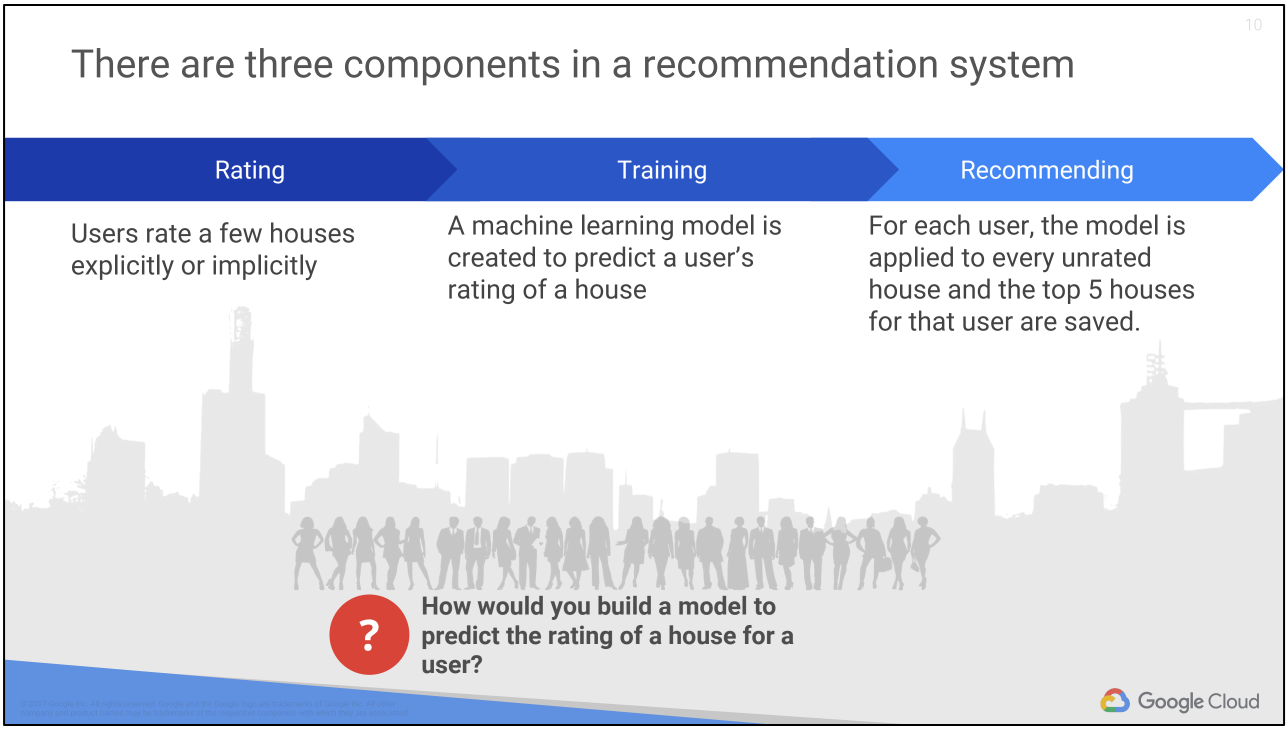
Smart Reply

? **What's common to all of these use cases of Machine Learning?**

Google Cloud

astronomy (1970s) to smart reply in Inbox

Answer: they are all recommendation engines (or at least part of the ML pipeline includes a recommendation engine). They may not believe you at first. But you can come back to it. Anything that involves *personalization* is a recommendation engine -- it's not about products, it's about people. This is why your Google search result is different from the result that your mom gets when she searches for the exact same thing. Of course, things like suggesting next video and smart reply are far more sophisticated the simple ALS algorithm that we do in this section -- the ALS algorithm is most akin to "people who bought …"



There are three components in a recommendation system

Rating → **Training** → **Recommending**

Users rate a few houses explicitly or implicitly

A machine learning model is created to predict a user's rating of a house

For each user, the model is applied to every unrated house and the top 5 houses for that user are saved.

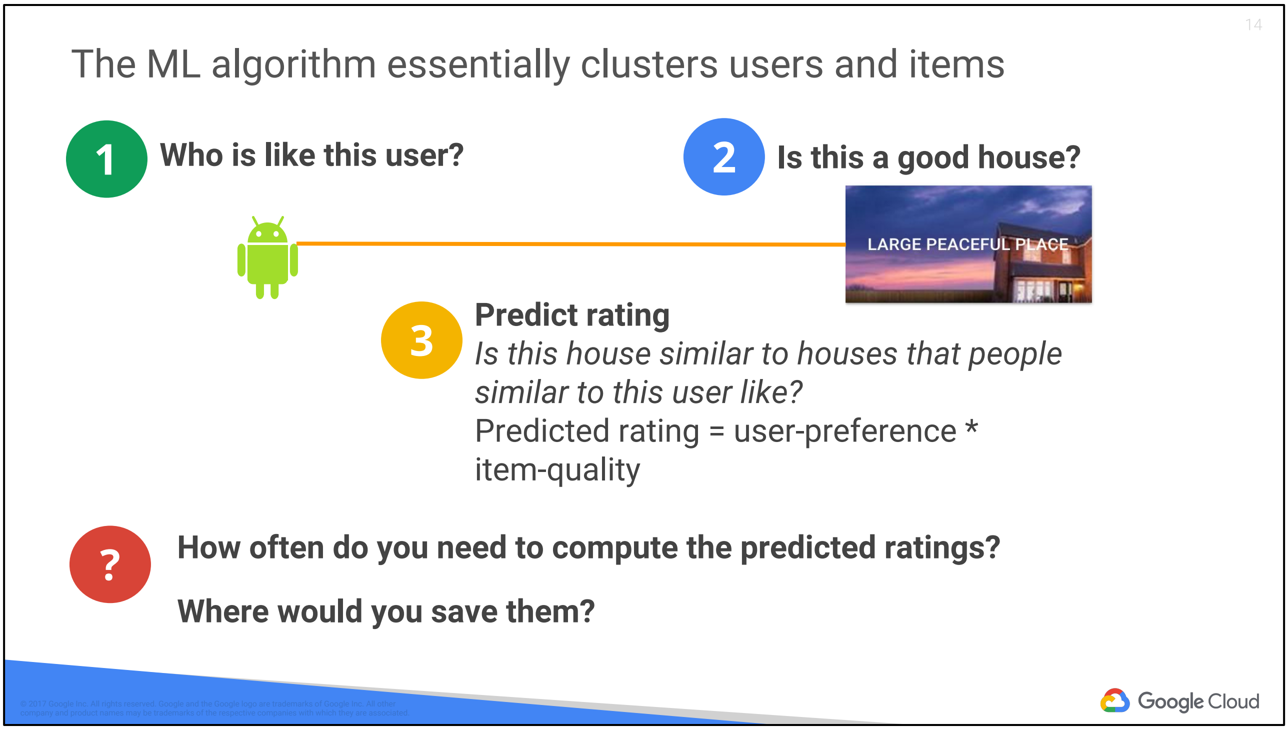
? **How would you build a model to predict the rating of a house for a user?**

Google Cloud

https://pixabay.com/en/skyline-city-blue-town-silhouette-296469/ (cc0)
https://pixabay.com/en/people-group-crowd-line-silhouette-312122/ (cc0)

In order for people to rate houses, you need to add ratings capability to your website. This is why I am calling it a (software) component.

Answer: you need two things: ML infrastructure and a ML algorithm.



The ML algorithm essentially clusters users and items

1 **Who is like this user?**

2 **Is this a good house?**

3 **Predict rating**
Is this house similar to houses that people similar to this user like?
Predicted rating = user-preference * item-quality

? **How often do you need to compute the predicted ratings?**

Where would you save them?

Notes:

We may have 1000s of items and only 2-3 reviews per item. And chances are that those reviewers have nothing in common with the user we want the rating for. So, we need to cluster items and users together. To put this in a more immediate form, if all your friends drive SUVs, and you read an article on Porsche, the car that appears on your feed might be a Porsche SUV even if the article was about Porsche cars and all your friends drive Toyota SUVs. The machine learning model is imputing a rating for a Porsche SUV that is quite high even though none of your friends rated it.

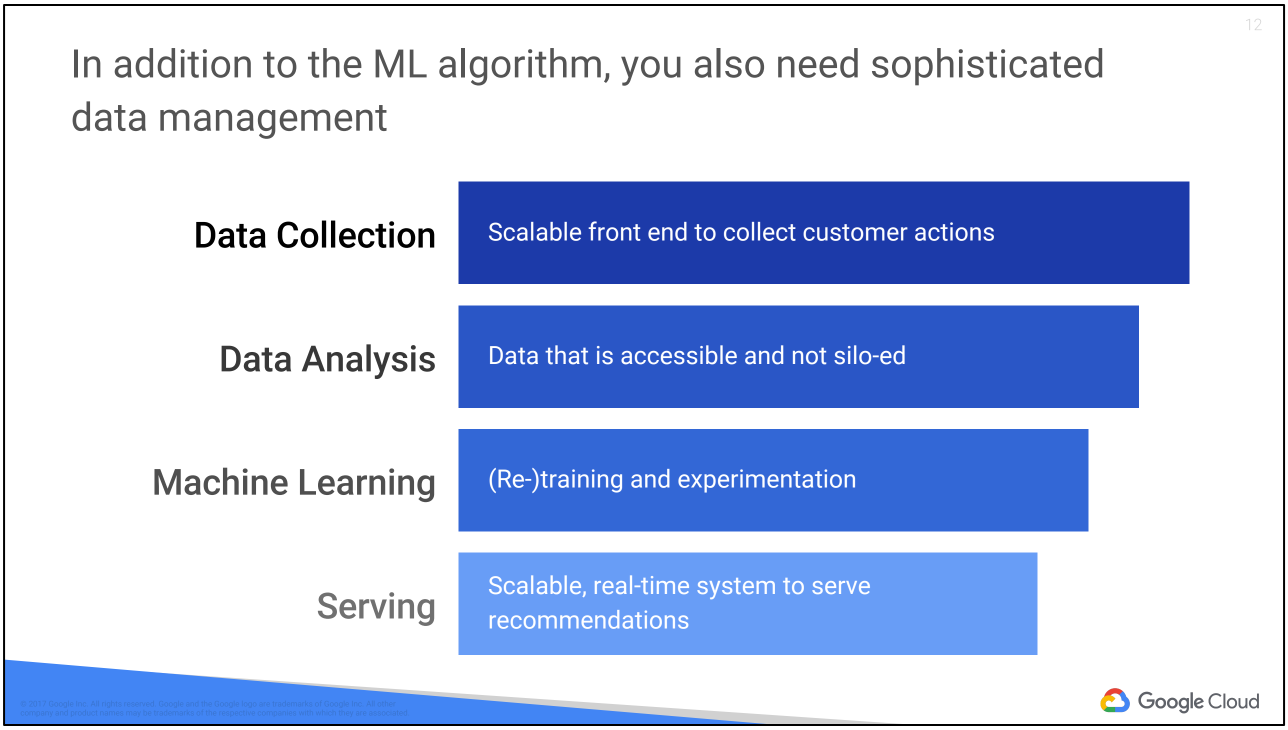
Answer: We can get away with computing predicted ratings once a day. We don't need to get the recommendation in real-time. So Hadoop batch processing is enough -- here we will use SparkML and do it on Cloud Dataproc. Incidentally, this is why the "Smart Reply" in Inbox is so amazing -- that does have to be done in real-time. Needless to say, Inbox is doing something far more sophisticated than what we are talking about here. We could save them in a database since there are only 5 recommendations per user. That's Small Data. Hence our use of CloudSQL.

Some machine learning background (you don't need to get into this, but it might be useful if someone brings it up in a hallway conversation):

The multiplication is actually a matrix multiplication:

- Rating = Preference * Quality   (R = PQ)
- P is a u x r matrix; u = number of users, r is called the rank (loosely the number of user-item clusters)
- Q is a r x c matrix; c = number of items in catalog
- The result R is a u x c matrix i.e. a rating for every user for every item in the catalog
- The training process fills out the matrices P and Q and chooses the r that minimizes R - PQ for the user-item pairs that the company has ratings for.

The optimization method (called Alternating Least Squares) keeps P constant and does gradient descent on Q. Then it keeps Q constant and does a gradient descent of P, alternating until things converge. In essence, we find what each user would rank each cluster (an easier problem than ranking each item) and the extent to which each item falls into a cluster.



In addition to the ML algorithm, you also need sophisticated data management

Data Collection — Scalable front end to collect customer actions

Data Analysis — Data that is accessible and not silo-ed

Machine Learning — (Re-)training and experimentation

Serving — Scalable, real-time system to serve recommendations

Google Cloud

https://cloud.google.com/solutions/recommendations-using-machine-learning-on-compute-engine

It is not just the ML algorithm -- you should also think about the infrastructure you'll need in order to do ML. "If you can not do data analysis, you can not do ML"

In this section, we don't do real-time serving: instead, we pre-compute the top 5.

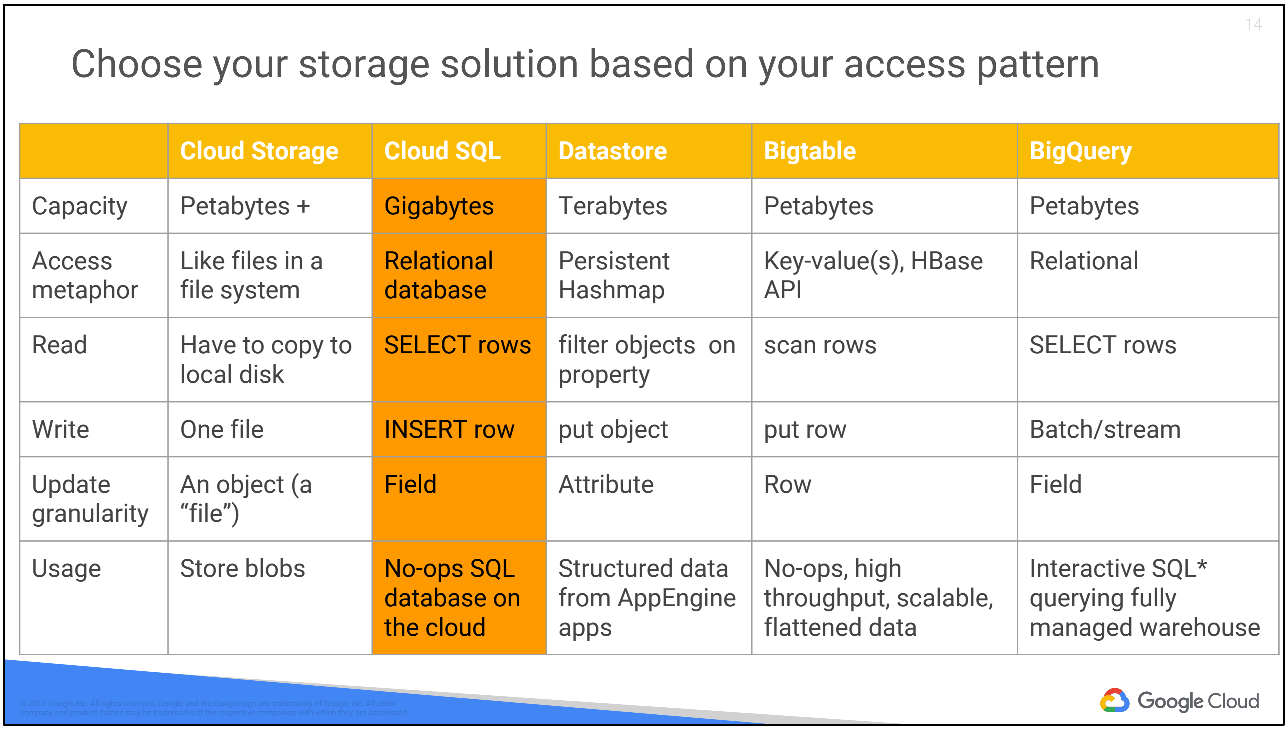
To provide recommendations, whether in real time while customers browse or through email later on, several things need to happen. At first, while you know little about your users' tastes and preferences, you might base recommendations on item attributes alone. But your system needs to be able to learn from your users, collecting data about their tastes and preferences. Over time and with enough data, you can use machine learning algorithms to perform useful analysis and deliver meaningful recommendations. Other users' inputs can also improve the results, enabling the system to be retrained periodically.

Agenda

Stepping stones to transformation

Your SQL database in the cloud + Lab

Managed Hadoop in the cloud + Lab



Choose your storage solution based on your access pattern

	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

Notes:

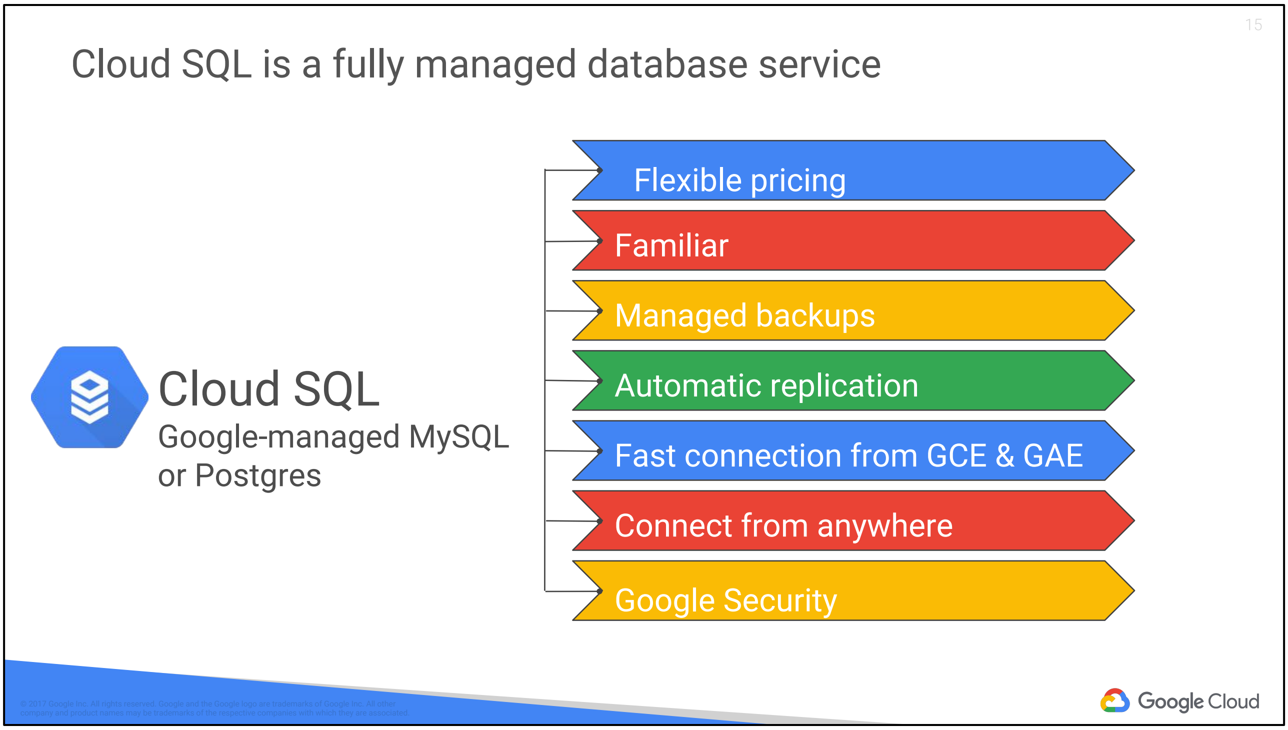
CloudSQL and Datastore both support transactions. 2nd generation of CloudSQL supports upto 10TB, but it's best to remain in the gigabyte range.

Flattened data: Bigtable doesn't support SQL or joins.

The asterix in BigQuery is that it is SQL-like.

We're talking about Cloud SQL here because this chapter is about easy migrations. But you should really look at Datastore/Bigtable/BigQuery as those are better suited to large data and the cloud.

For our rentals recommendation problem, Cloud SQL is the right scale.



Cloud SQL is a fully managed database service

Cloud SQL

Google-managed MySQL or Postgres

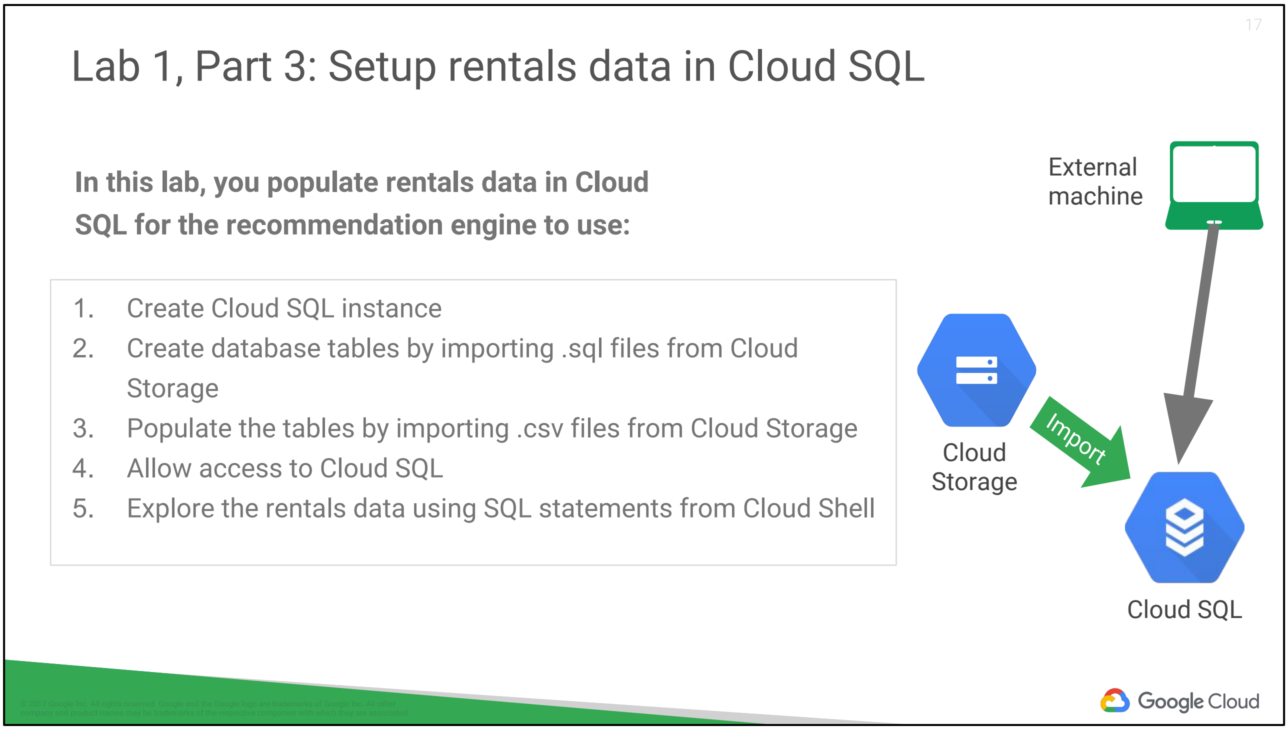
- Flexible pricing
- Familiar
- Managed backups
- Automatic replication
- Fast connection from GCE & GAE
- Connect from anywhere
- Google Security

Notes:

- Familiar: Cloud SQL supports most MySQL statements and functions, even Stored procedures, Triggers and Views.
- Not supported: User-defined functions, MySQL-esque replication, statements, and functions related to files and plugins.
- Flexible pricing: You can pay per use or per hour.
- Backups, replication, and so on are managed for you.
- Connect from anywhere (can assign a static IP address, and use typical SQL connector libraries).
- Fast: You can place your Cloud SQL instance in same region as your App Engine or Compute Engine applications and get great bandwidth.
- Google security: Cloud SQL resides in secure Google datacenters.

Lab: Set up rentals data in Cloud SQL

Google Cloud

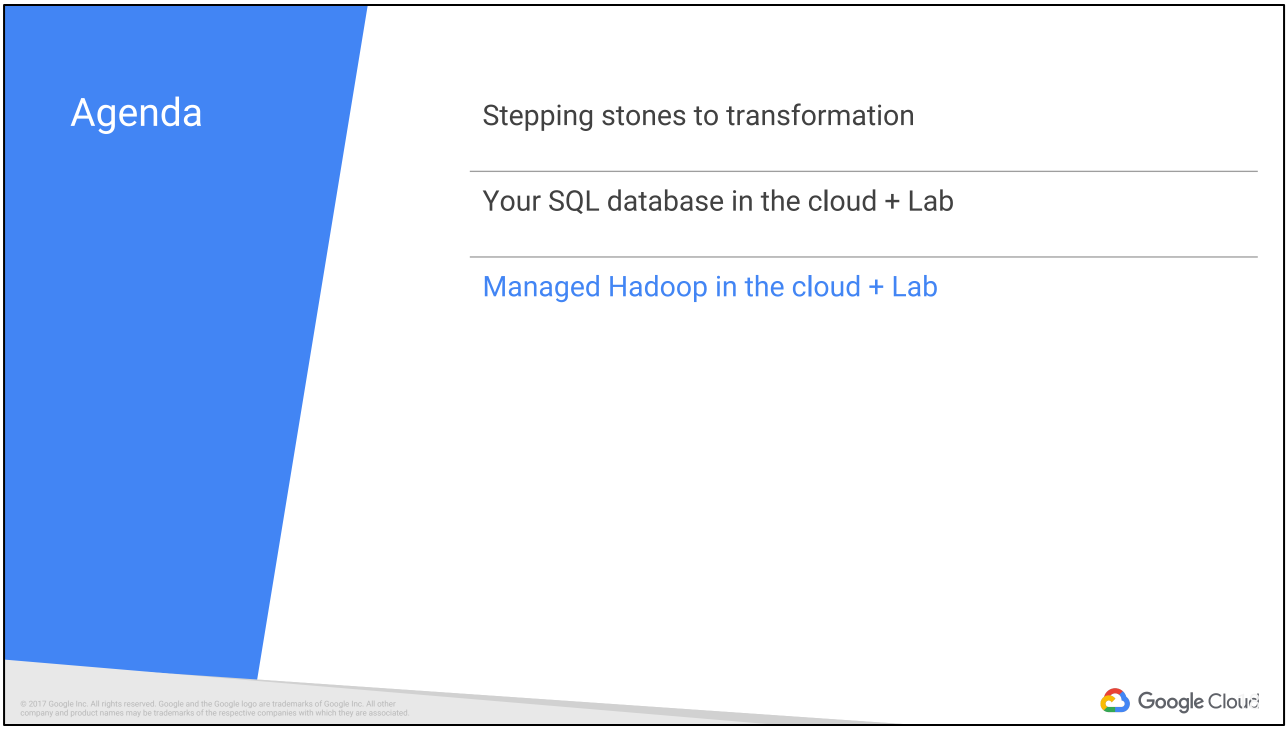


Lab 1, Part 3: Setup rentals data in Cloud SQL

In this lab, you populate rentals data in Cloud SQL for the recommendation engine to use:

1. Create Cloud SQL instance
2. Create database tables by importing .sql files from Cloud Storage
3. Populate the tables by importing .csv files from Cloud Storage
4. Allow access to Cloud SQL
5. Explore the rentals data using SQL statements from Cloud Shell

Google Cloud

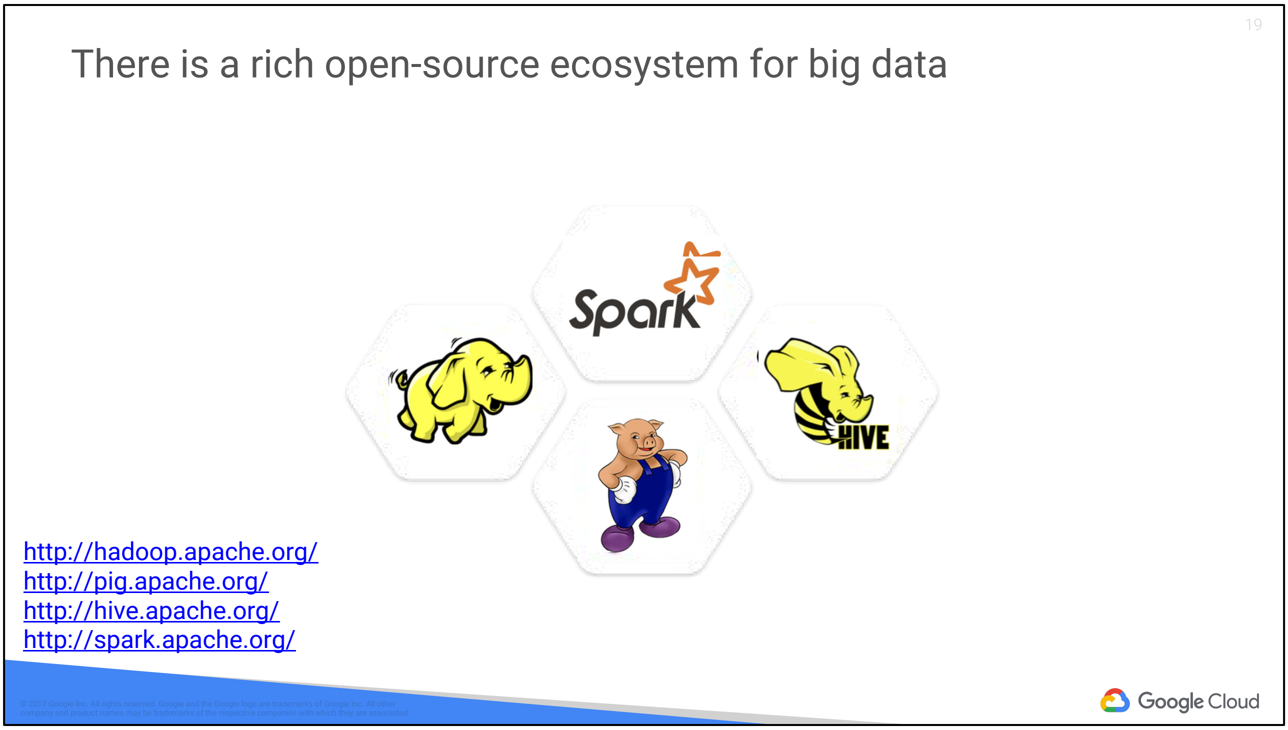


Agenda

Stepping stones to transformation

Your SQL database in the cloud + Lab

Managed Hadoop in the cloud + Lab

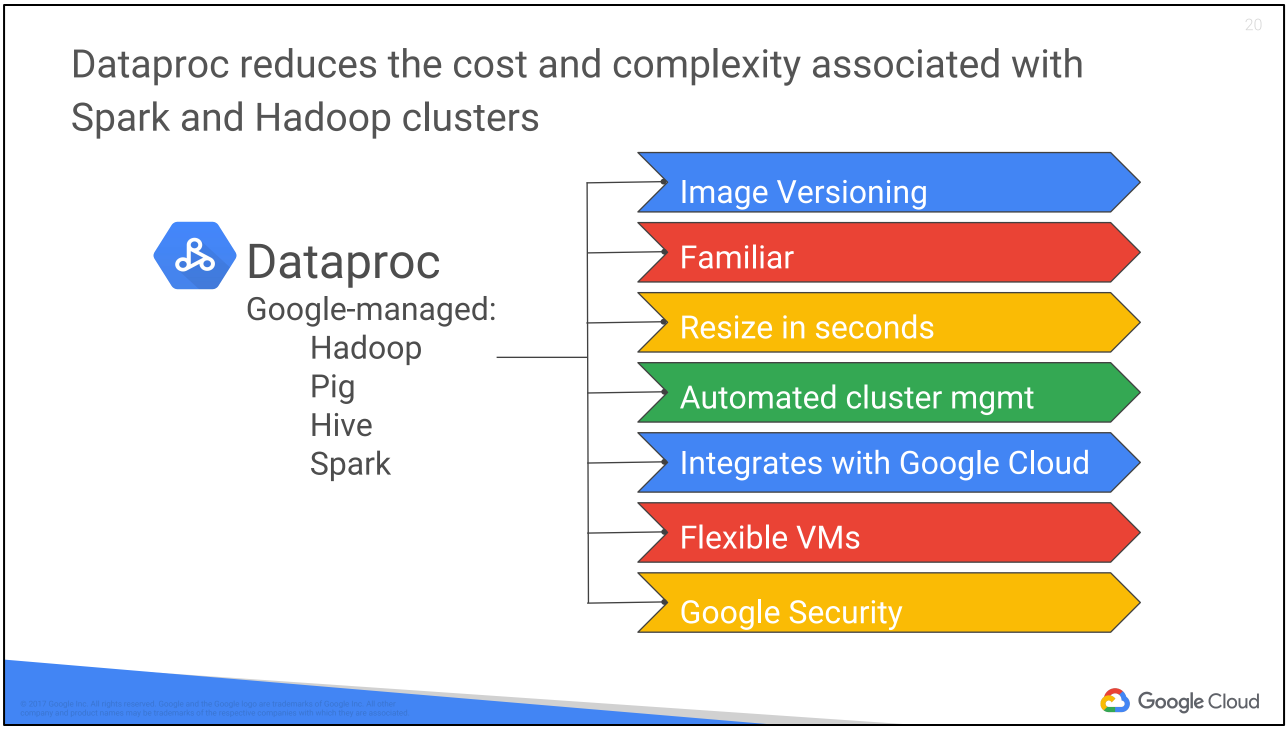


There is a rich open-source ecosystem for big data

http://hadoop.apache.org/
http://pig.apache.org/
http://hive.apache.org/
http://spark.apache.org/

Google Cloud

Notes:
Hadoop is the canonical open-source MapReduce framework
Pig provides a convenient scripting language that can be compiled into Hadoop MapReduce jobs
Hive is a data warehousing system and query language
Spark is a fast, interactive, general-purpose framework for SQL, streaming, machine learning, and so on
Spark hides all the underlying details and can run standalone, over Hadoop or on the cloud.



Dataproc reduces the cost and complexity associated with Spark and Hadoop clusters

Dataproc

Google-managed:

Hadoop

Pig

Hive

Spark

- Image Versioning
- Familiar
- Resize in seconds
- Automated cluster mgmt
- Integrates with Google Cloud
- Flexible VMs
- Google Security

Google Cloud

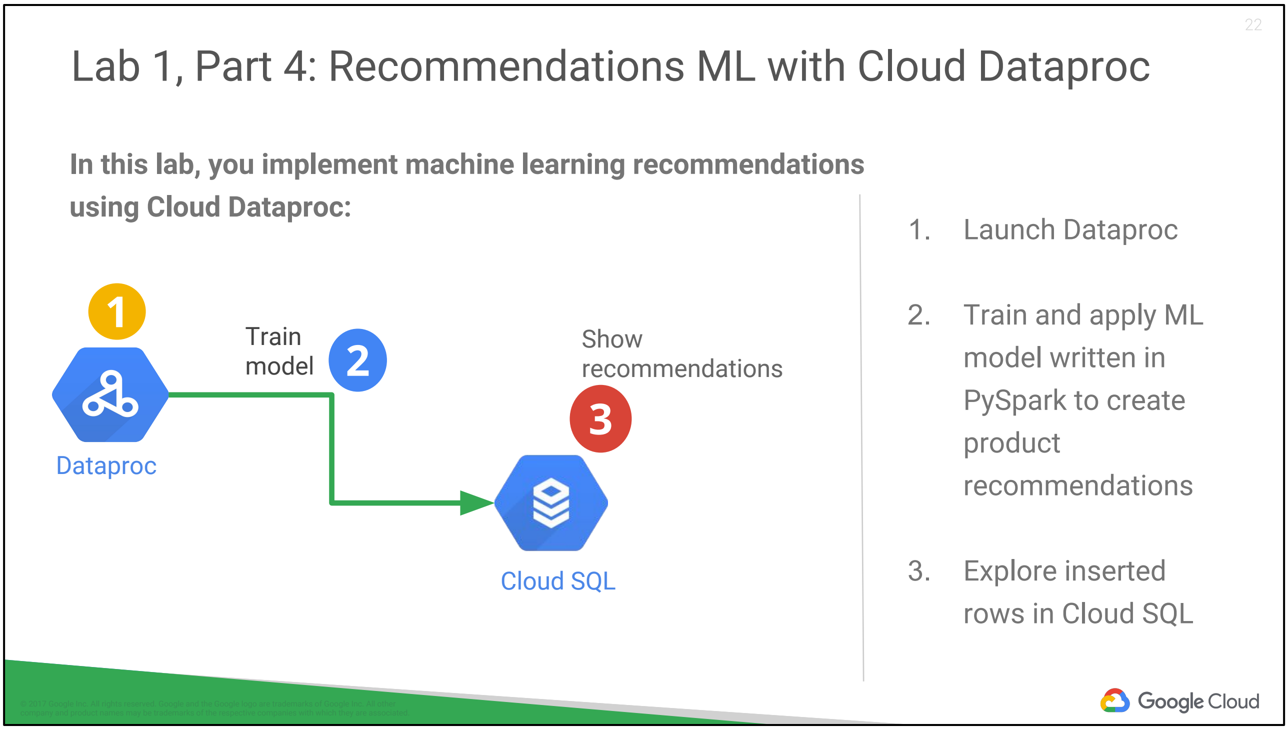
Notes:

https://cloud.google.com/dataproc/dataproc-versions

The diagram is, of course, a visual reminder that this is similar to how Google manages MySQL.

- Familiar: It's the same Pig, Hive, and so on that they use so workloads transfer unchanged.
- Image versioning: Cloud Dataproc has up-to-date versions of the tools, but you can use earlier versions. You also have initialization options to get a cluster initialized the way you need it.
- Integrates with Google Cloud Platform so you get the monitoring, logging, cluster management, fail-over, and so on. And you can talk to Cloud SQL or BigQuery easily.
- Resize in seconds, with custom-machine types and/or preemptible VMs, tear them down when you don't need them. It's all quite cost-effective.

Lab: Recommendations ML with Dataproc



Lab 1, Part 4: Recommendations ML with Cloud Dataproc

In this lab, you implement machine learning recommendations using Cloud Dataproc:

1. Launch Dataproc
2. Train and apply ML model written in PySpark to create product recommendations
3. Explore inserted rows in Cloud SQL

Notes:

The complete flow of information.

1. “Ratings” from App Engine are stored in Cloud SQL (Users rate only a few properties).
2. You run a training job in Cloud Dataproc to build a model.
3. You run the model in Cloud Dataproc to create recommendations and save the top 5 recommendations for each user into Cloud SQL.
4. Recommendations are displayed in App Engine.

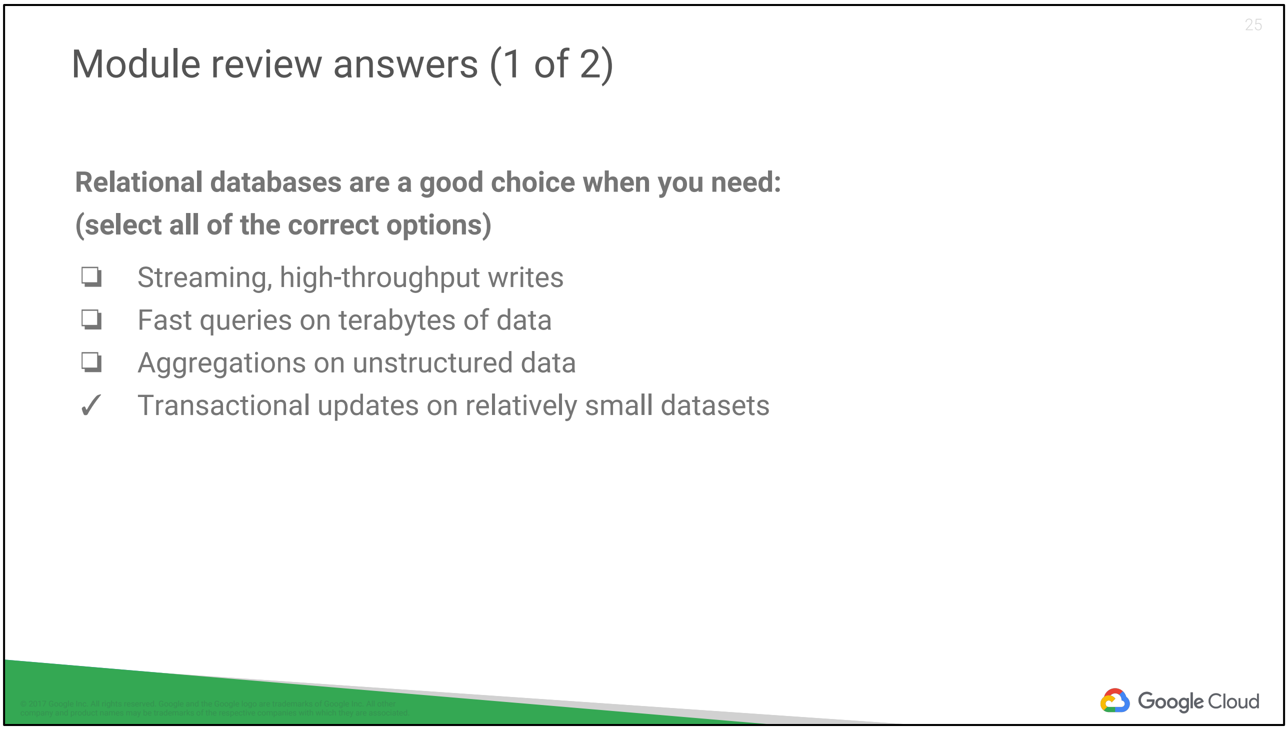
Students won’t do Steps 1 and 4 in this lab because those steps deal mostly with web programming. In this lab, you concentrate on doing steps 2 and 3. If you want a fully complete solution with a website that you can demo, go to https://cloud-training-demos.appspot.com/ and see the App Engine code in https://github.com/GoogleCloudPlatform/spark-recommendation-engine/tree/master/appengine

Module Review

Module review (1 of 2)

Relational databases are a good choice when you need: (select all of the correct options)

- ❑ Streaming, high-throughput writes
- ❑ Fast queries on terabytes of data
- ❑ Aggregations on unstructured data
- ❑ Transactional updates on relatively small datasets



Module review answers (1 of 2)

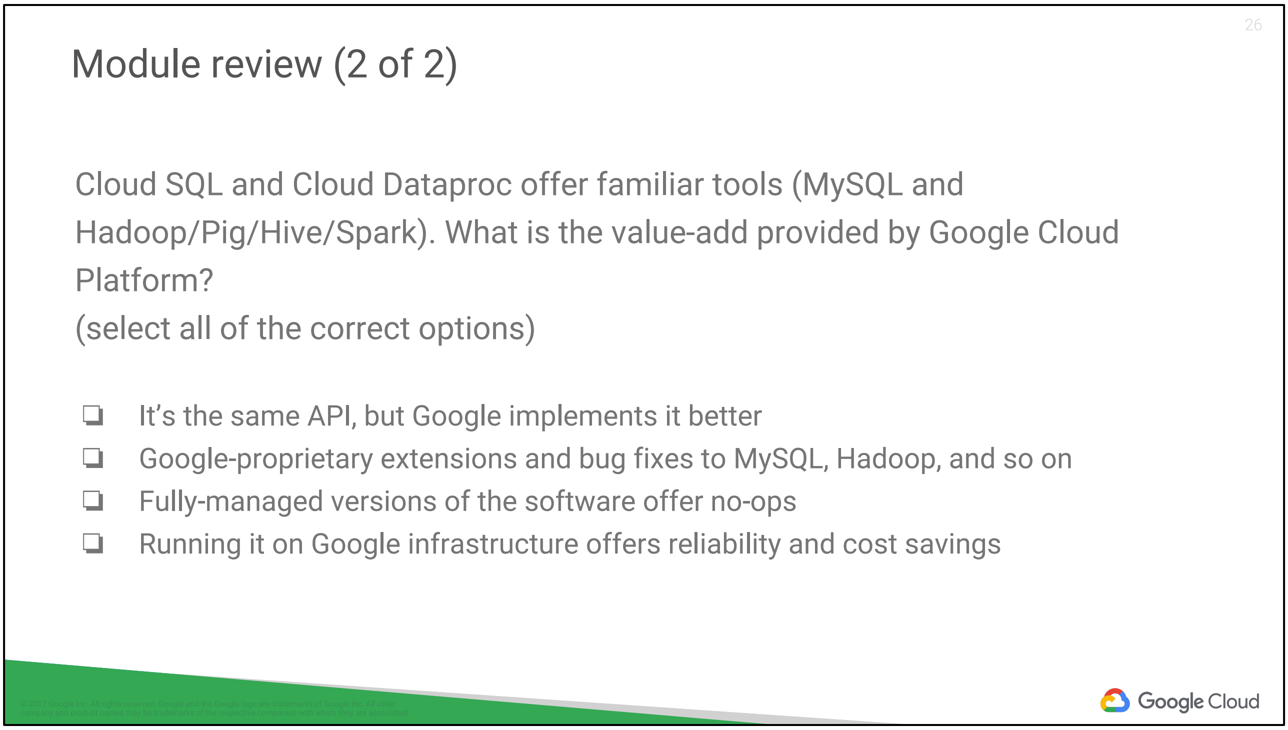
Relational databases are a good choice when you need:
(select all of the correct options)

- ❏ Streaming, high-throughput writes
- ❏ Fast queries on terabytes of data
- ❏ Aggregations on unstructured data
- ✓ Transactional updates on relatively small datasets

Google Cloud

Notes:

1: Bigtable
2: BigQuery
3: Bigtable, Dataflow or Dataproc
4: True

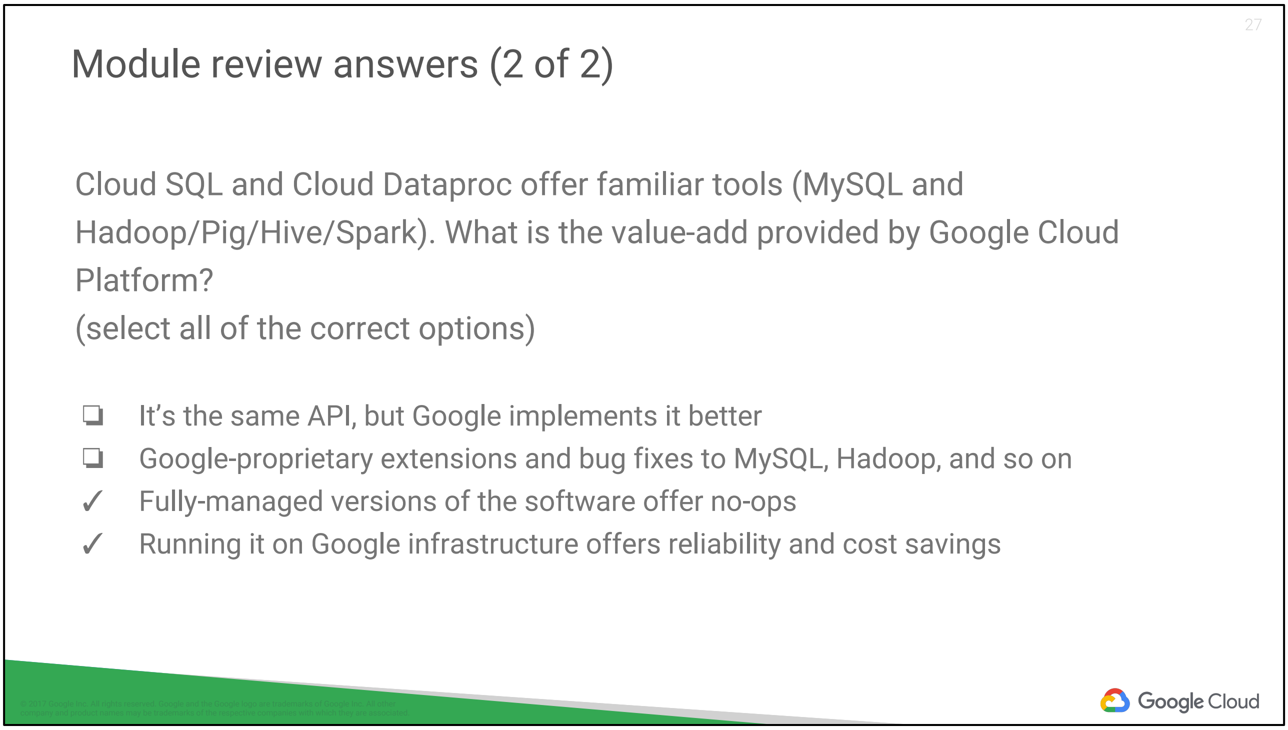


Module review (2 of 2)

Cloud SQL and Cloud Dataproc offer familiar tools (MySQL and Hadoop/Pig/Hive/Spark). What is the value-add provided by Google Cloud Platform?
(select all of the correct options)

- ❑ It’s the same API, but Google implements it better
- ❑ Google-proprietary extensions and bug fixes to MySQL, Hadoop, and so on
- ❑ Fully-managed versions of the software offer no-ops
- ❑ Running it on Google infrastructure offers reliability and cost savings

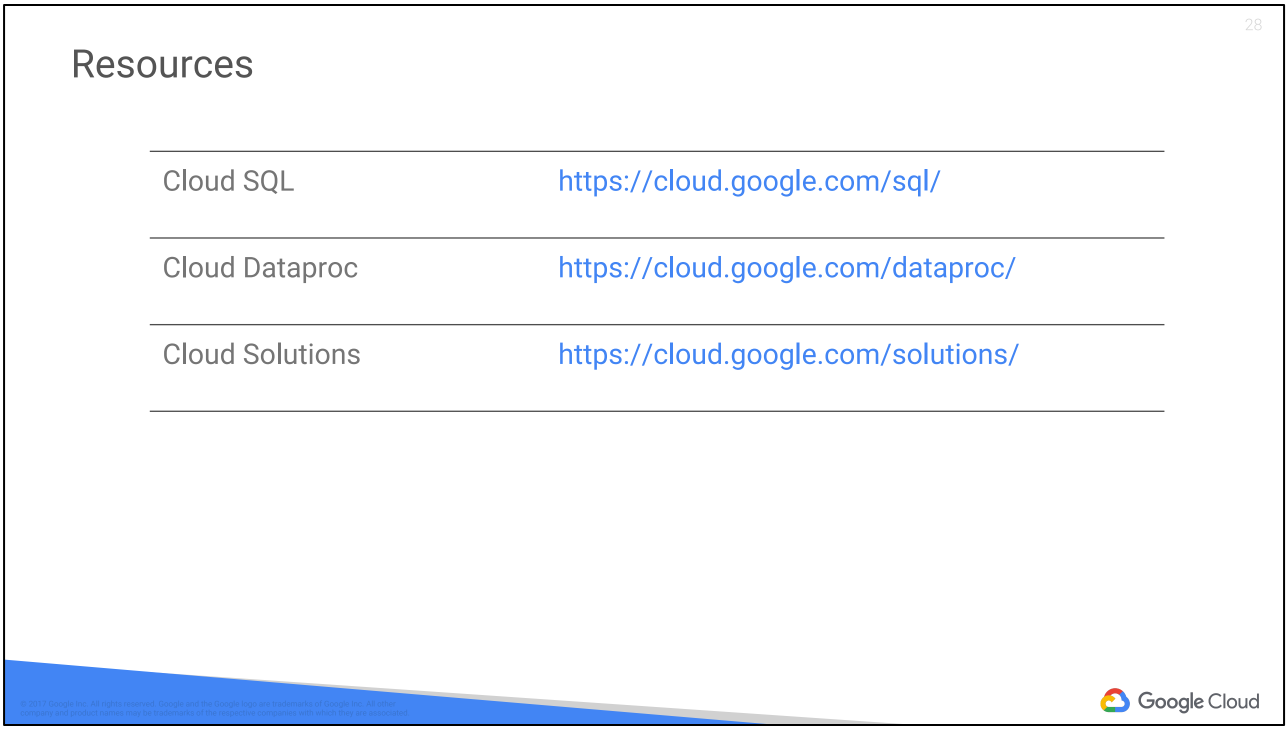
Google Cloud



Module review answers (2 of 2)

Cloud SQL and Cloud Dataproc offer familiar tools (MySQL and Hadoop/Pig/Hive/Spark). What is the value-add provided by Google Cloud Platform?
(select all of the correct options)

- ❏ It’s the same API, but Google implements it better
- ❏ Google-proprietary extensions and bug fixes to MySQL, Hadoop, and so on
- ✓ Fully-managed versions of the software offer no-ops
- ✓ Running it on Google infrastructure offers reliability and cost savings



Resources

Cloud SQL	https://cloud.google.com/sql/
Cloud Dataproc	https://cloud.google.com/dataproc/
Cloud Solutions	https://cloud.google.com/solutions/

Google Cloud

Notes:

Point out that the recommendations ML that they just did is a cloud solution

cloud.google.com

Images by Connie Zhou