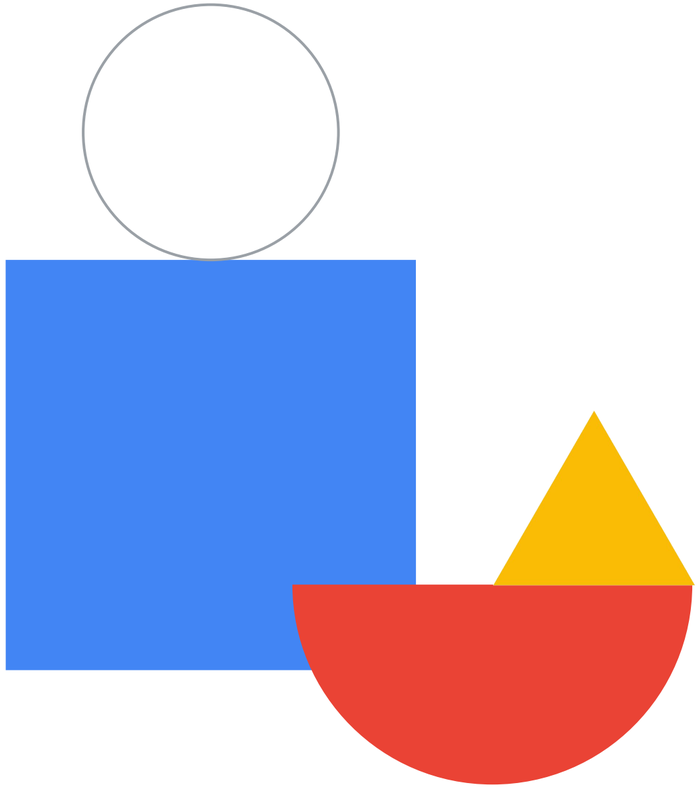
Google Cloud

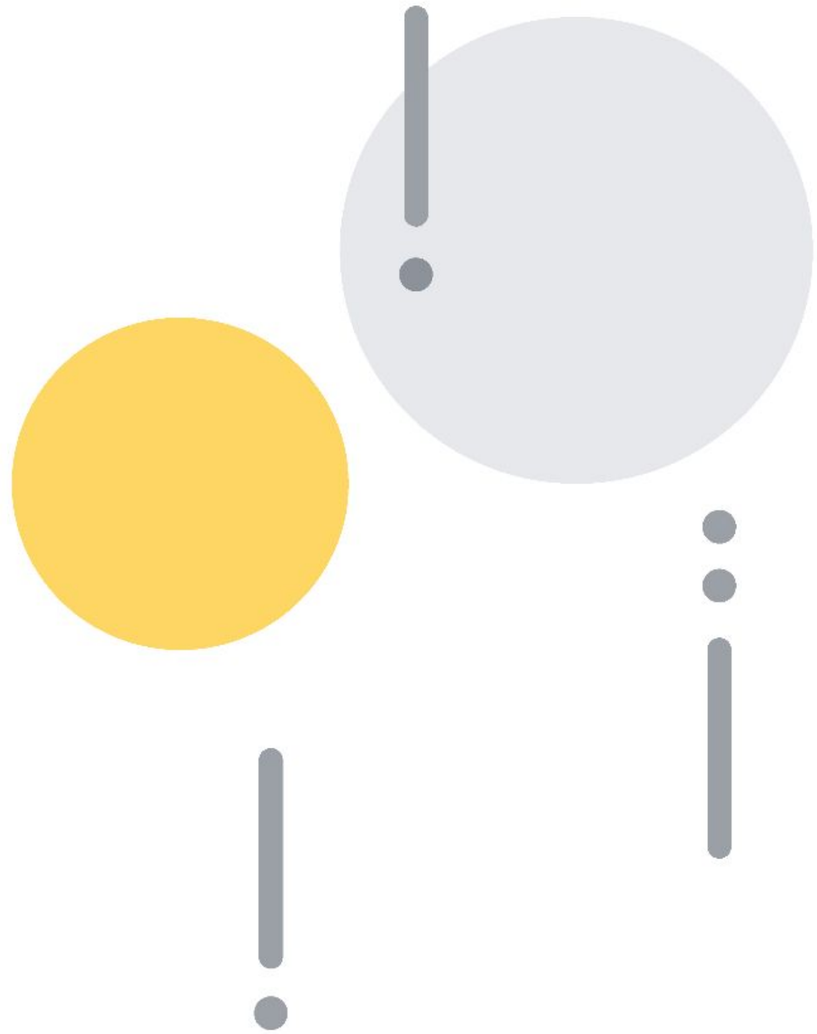


Preparing for Your Professional Data Engineer Journey

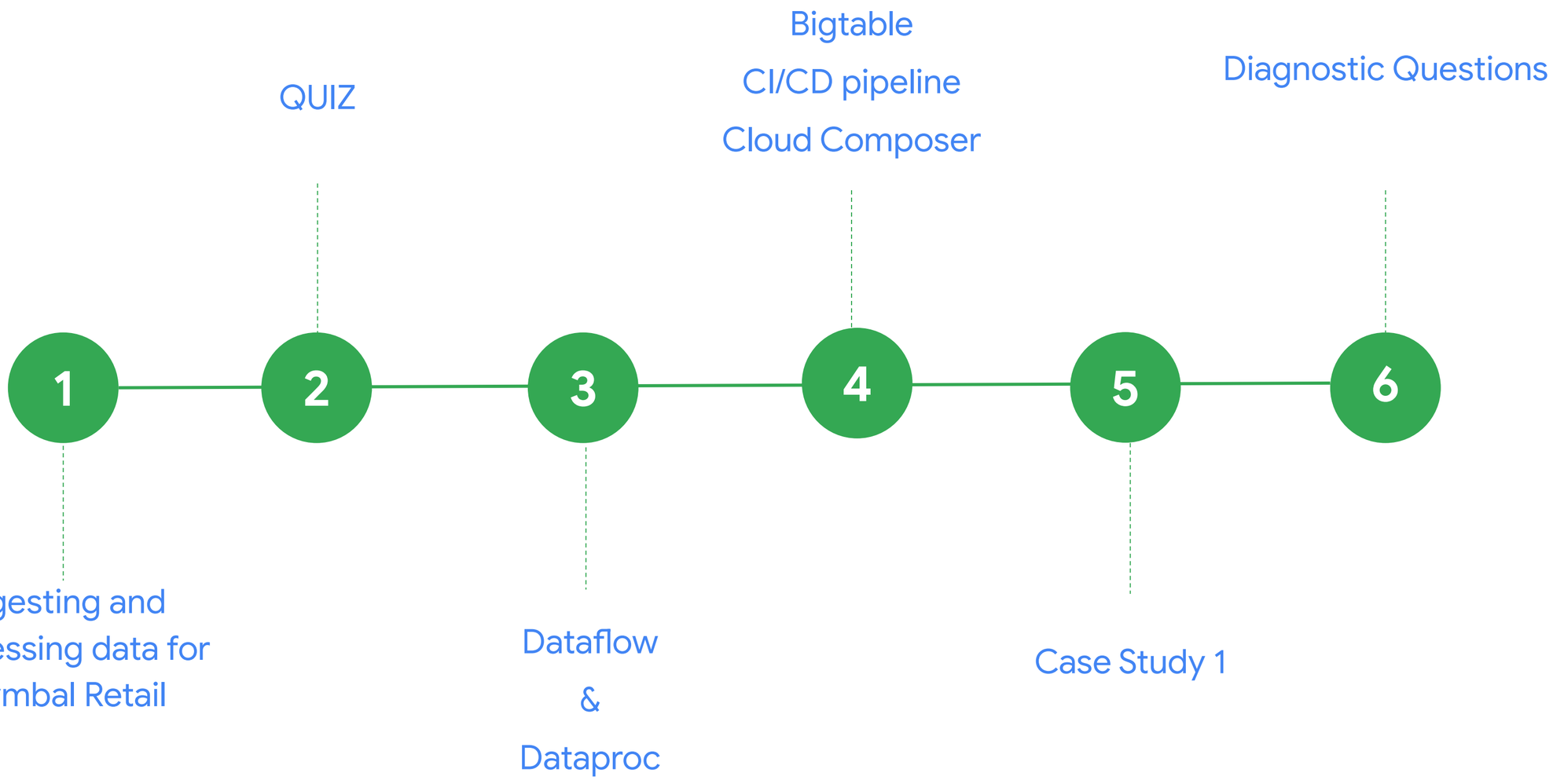
Module 2: Ingesting and Processing the Data

Proprietary + Confidential

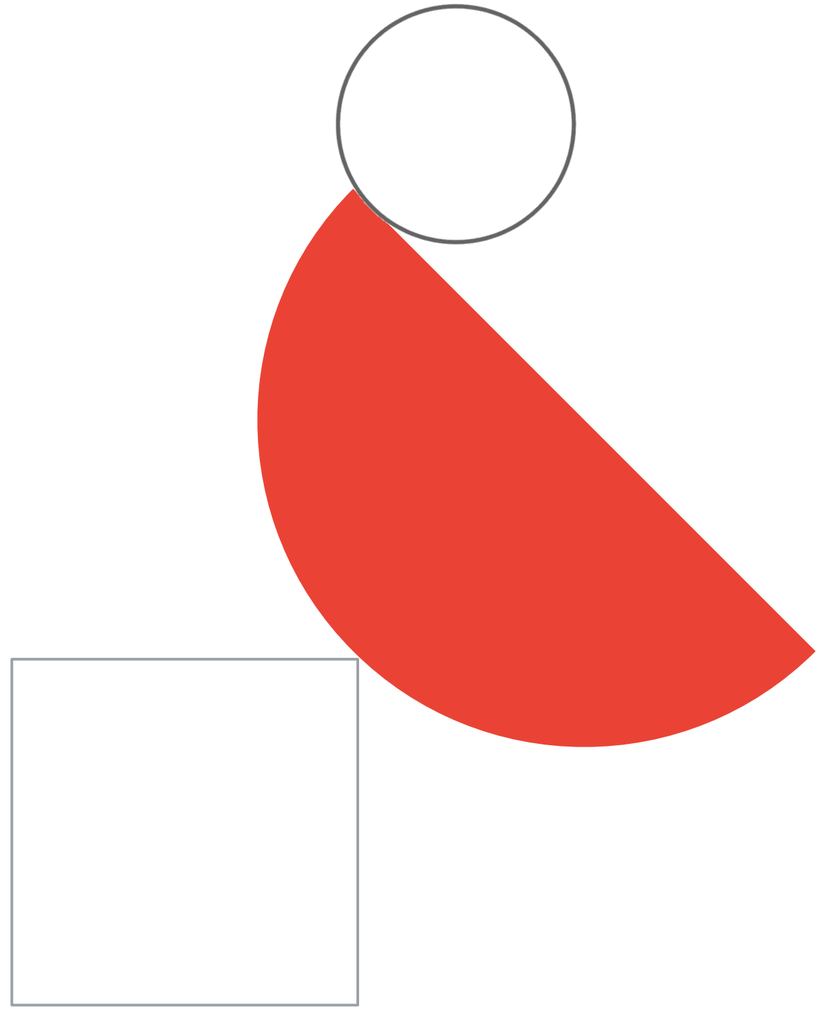
Where we are on our journey



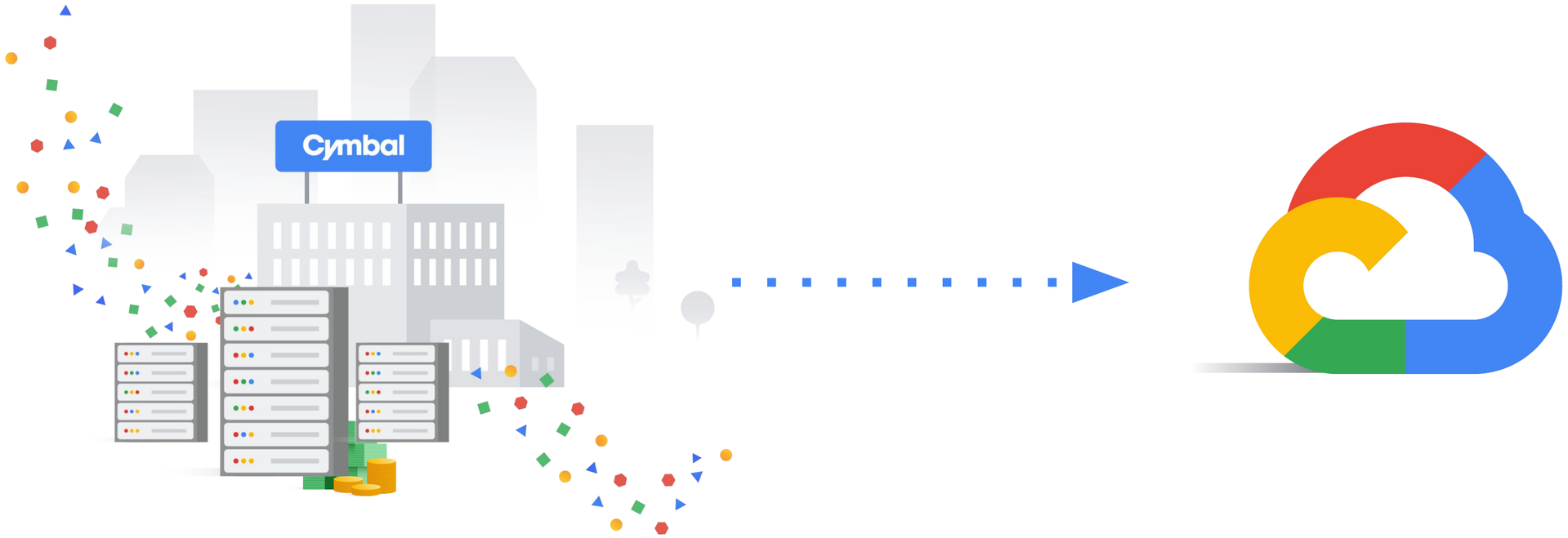
Week 3 agenda



Ingesting and processing data for Cymbal Retail



Migrating existing data processing infrastructure to Google Cloud

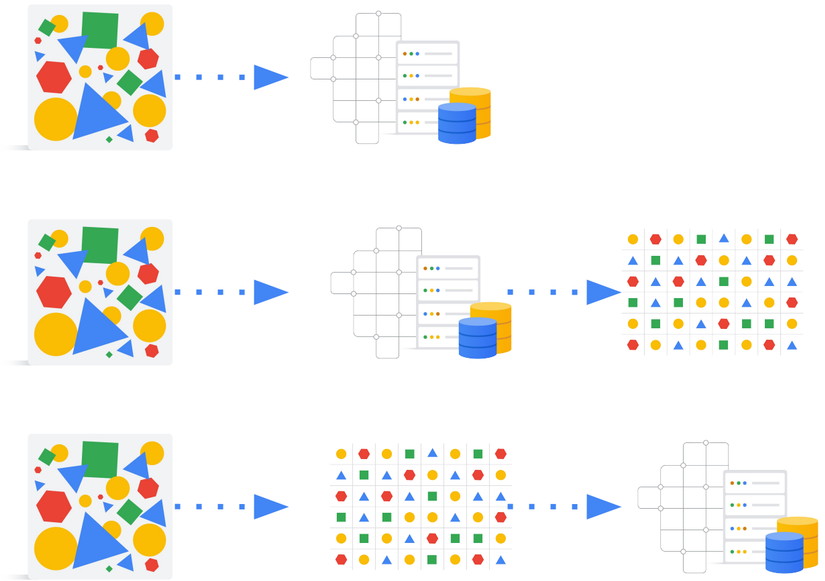


- Cymbal receives data from multiple sources.
- Processing data is complex and costly.
- Spark and Hadoop jobs are executed on static infrastructure.

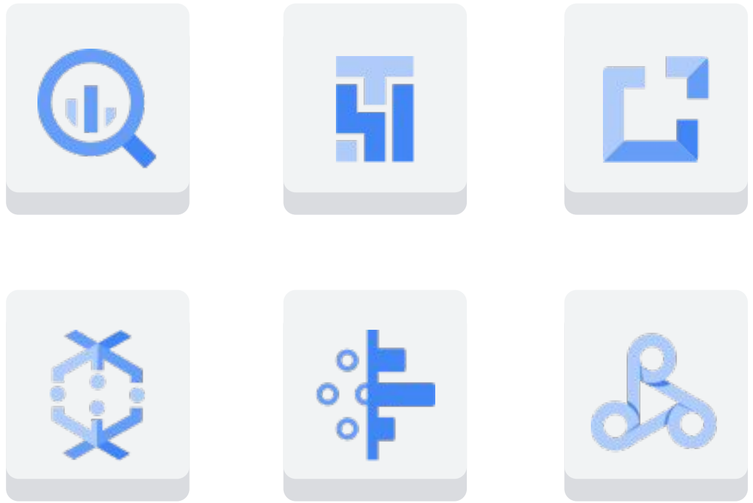
You need to lift-and-shift to Google Cloud.

Designing the architecture for data ingestion and processing

Will you use EL, ELT, or ETL?

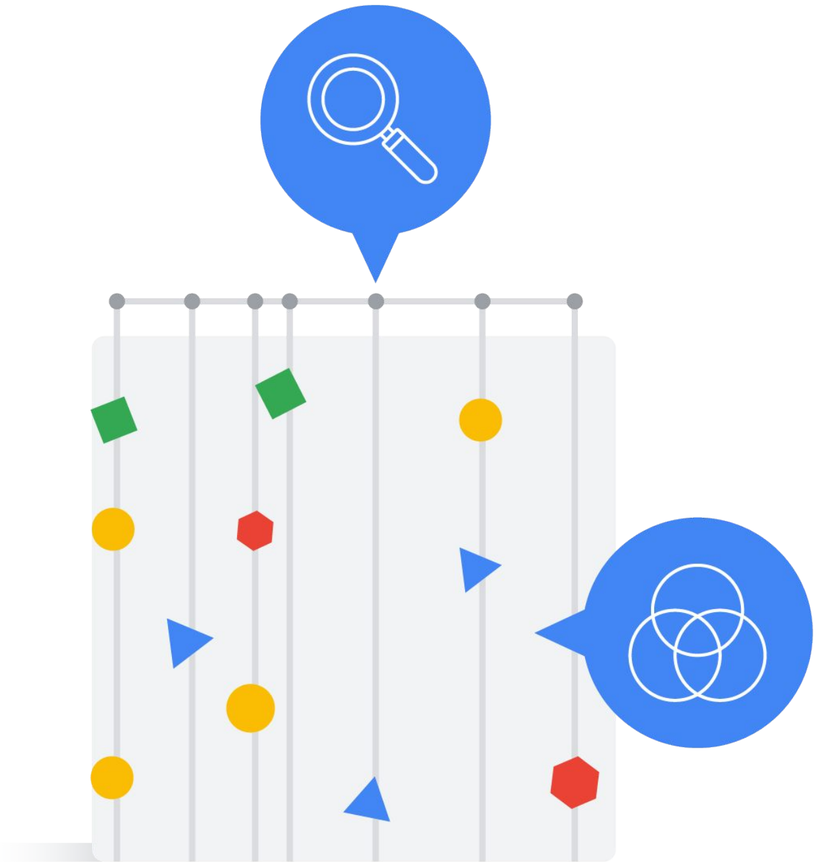


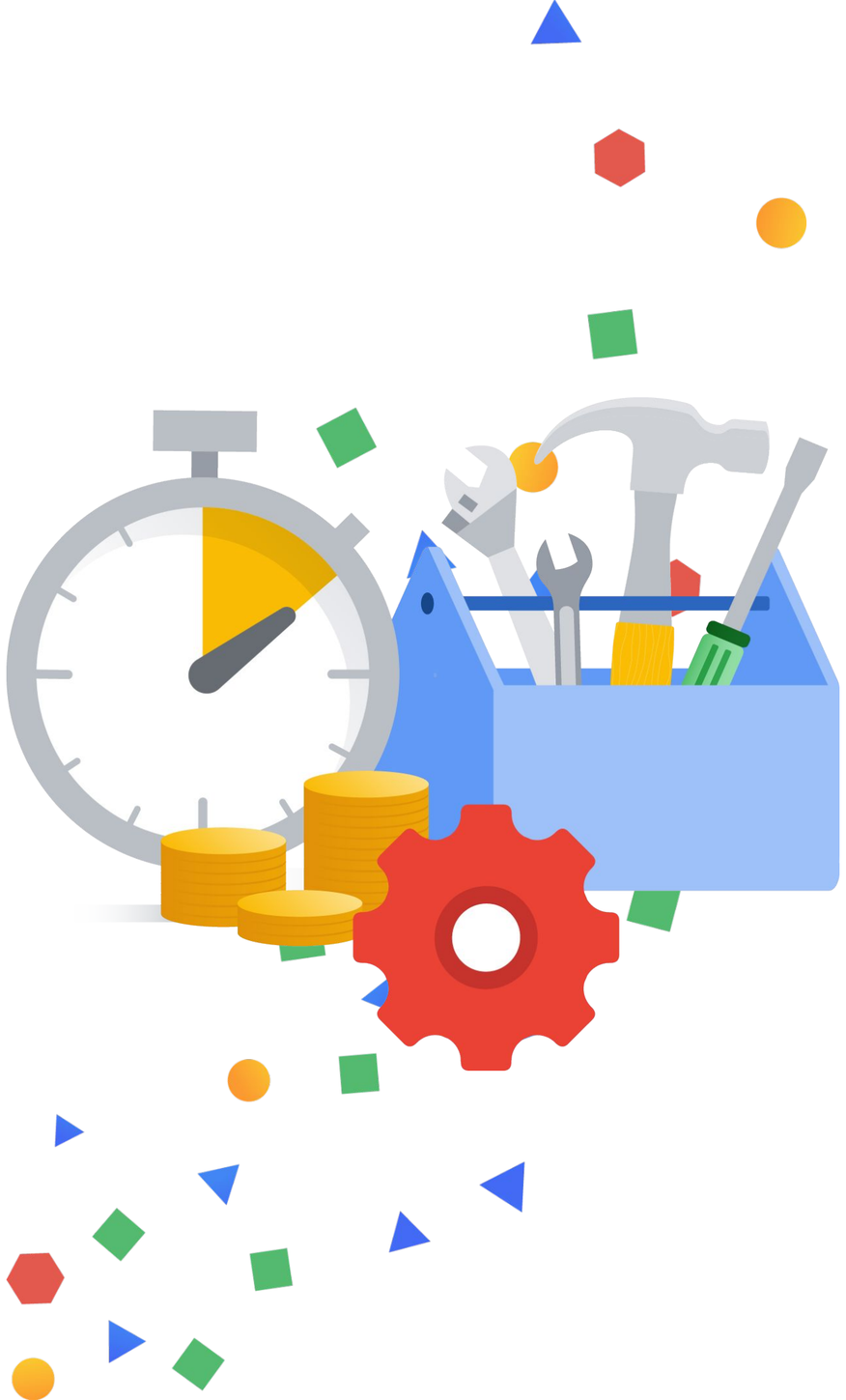
What Google Cloud tools will each pipeline use?



Designing the architecture for real-time data processing

Cymbal Retail needs to use real-time data. You need to choose the tools and types of windowing that will provide the right approach to analyze streaming data.





Optimizing data ingestions and data processing tasks

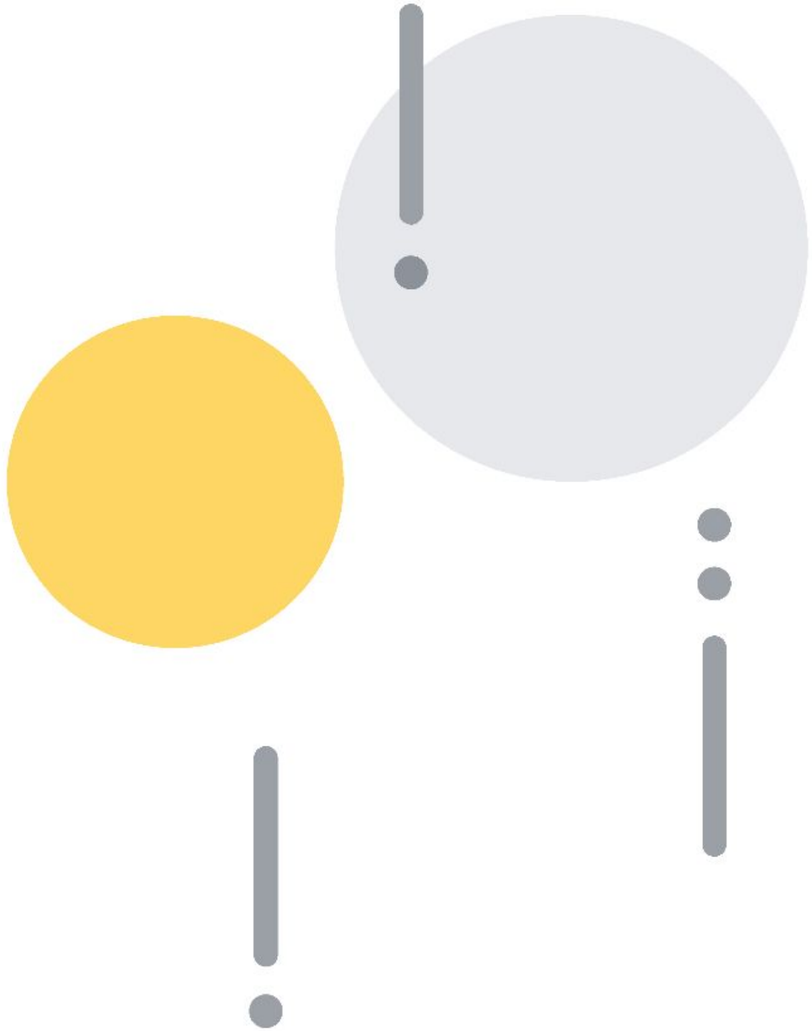
You want to optimize for:

- Savings on effort and cost
- Improvements to availability and responsiveness
- Your design decisions on automation and orchestration will be important as data volume increases.

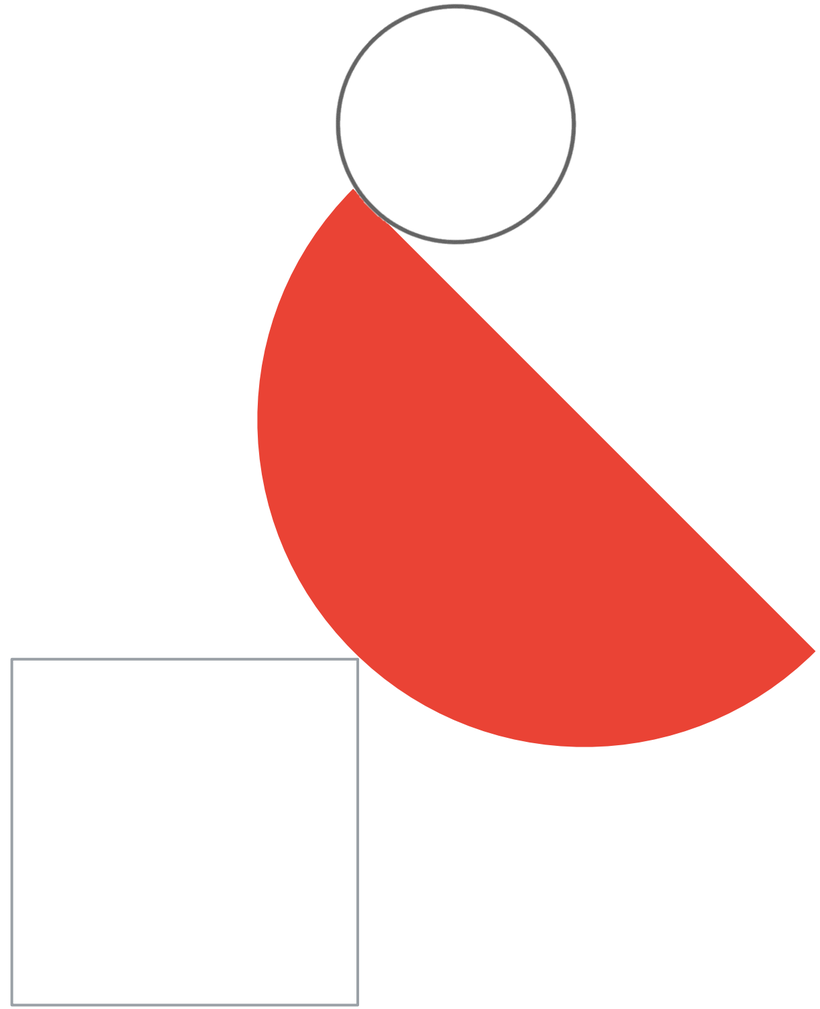
Proprietary + Confidential

Quiz Time

Link

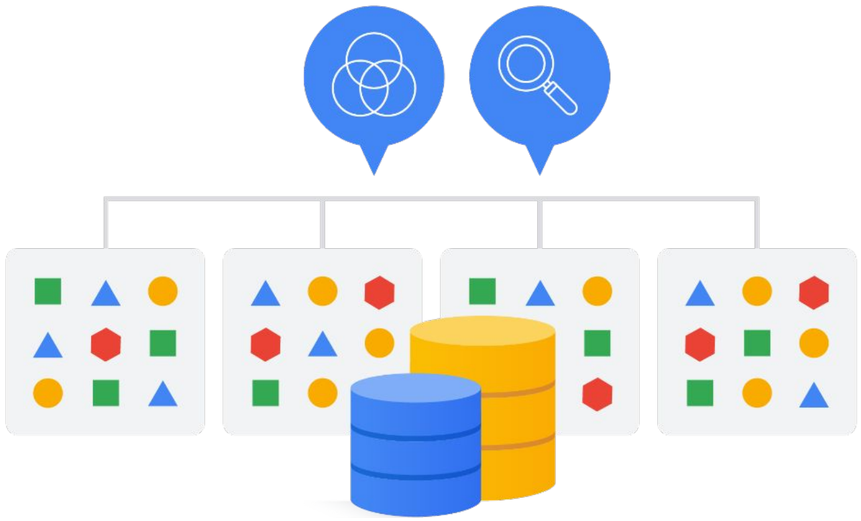


Dataflow



Proprietary + Confidential

Batch processing vs. streaming data processing

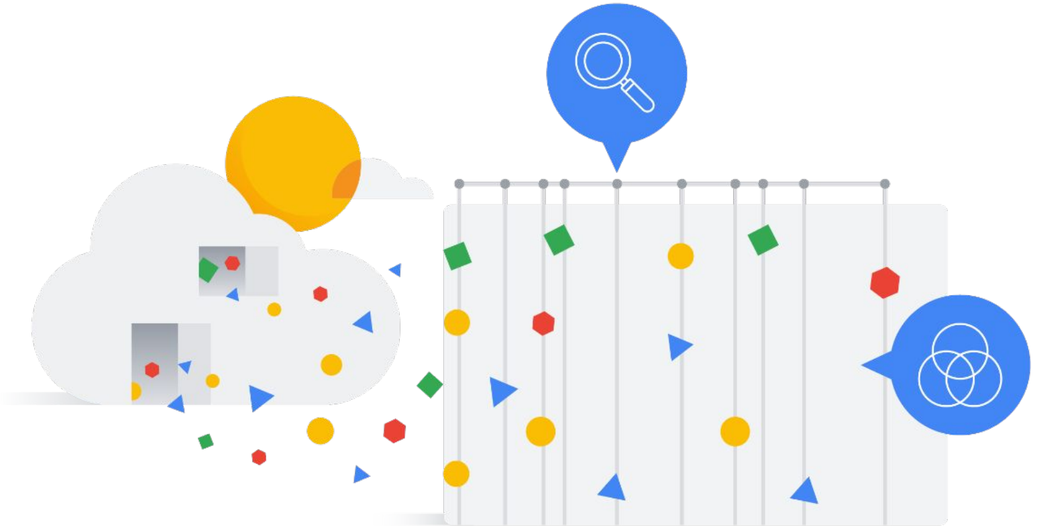


Batch processing

Processing and analysis happens on a set of stored data.

Payroll system

Billing system



Streaming data processing

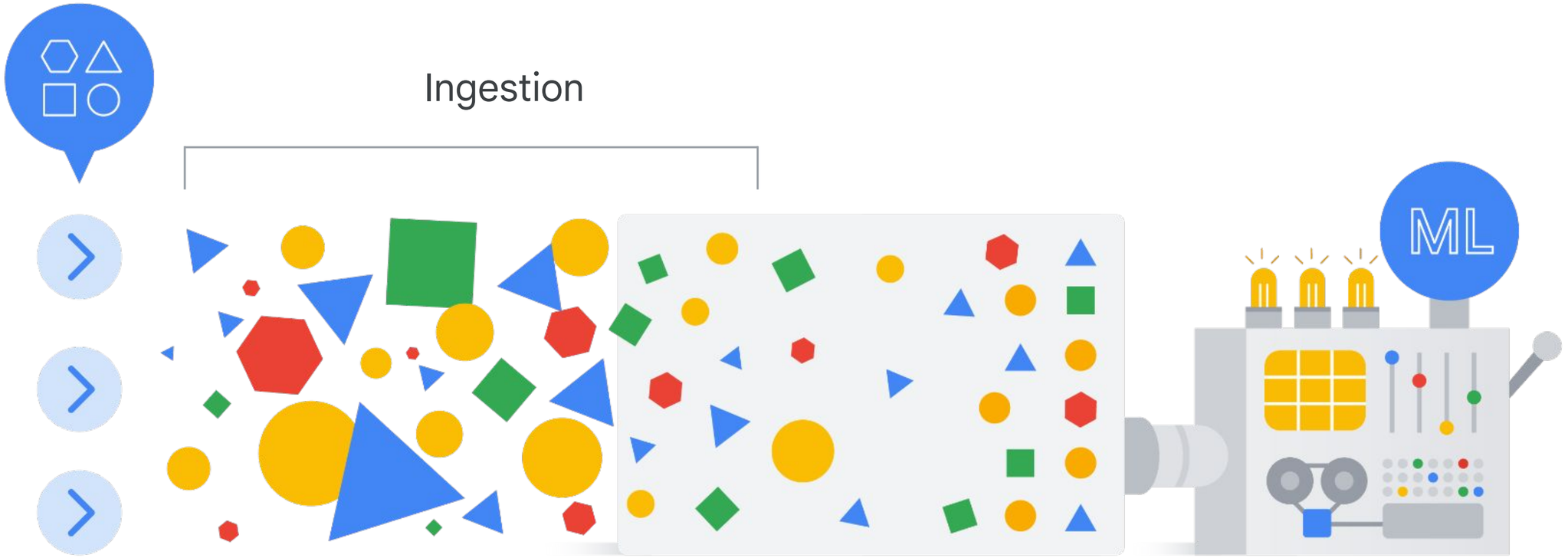
A flow of data records generated by various data sources.

Fraud detection

Intrusion detection

Proprietary + Confidential

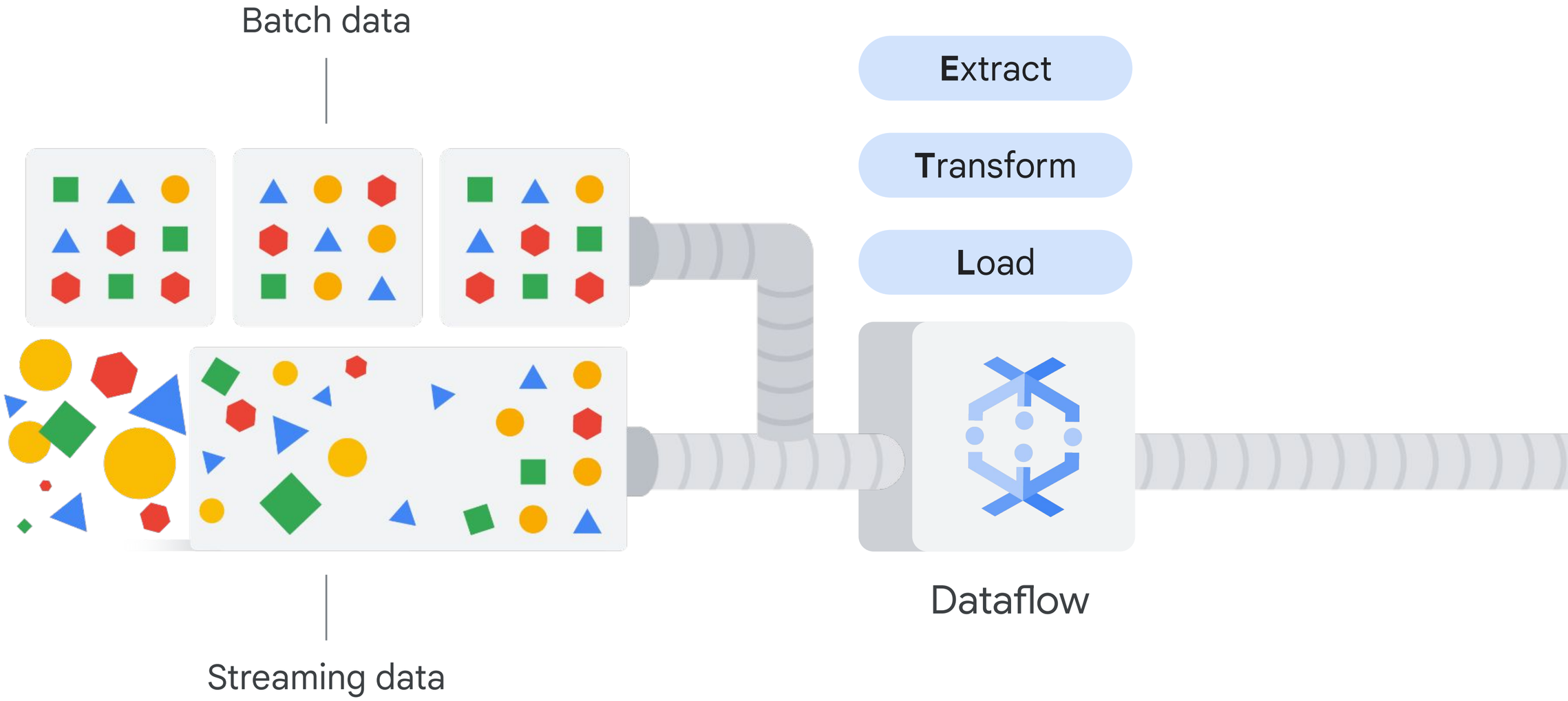
Data ingestion is when stream data is received



Data pipeline

Proprietary + Confidential

Dataflow creates a pipeline to process data



Proprietary + Confidential

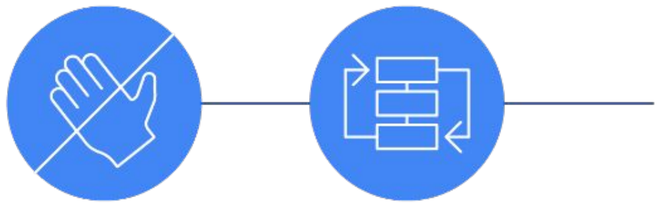
Dataflow is NoOps and serverless

Automated

Maintenance

Monitoring

Scaling



NoOps

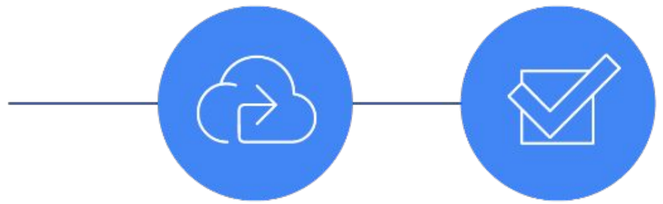
Dataflow

On behalf of the user

Resource provisioning

Performance tuning

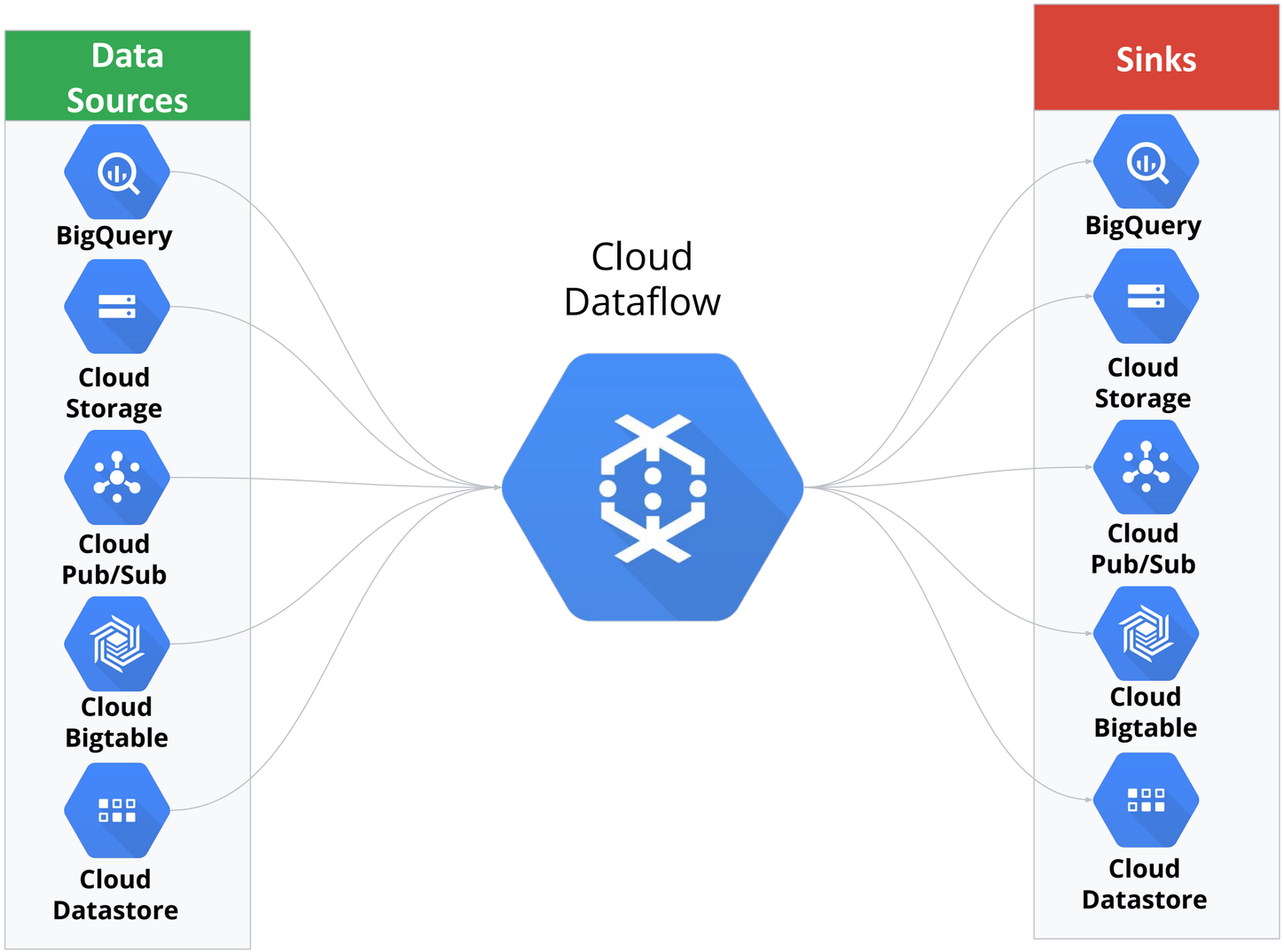
Pipeline reliability



Serverless

Data Sources and Sinks for Dataflow

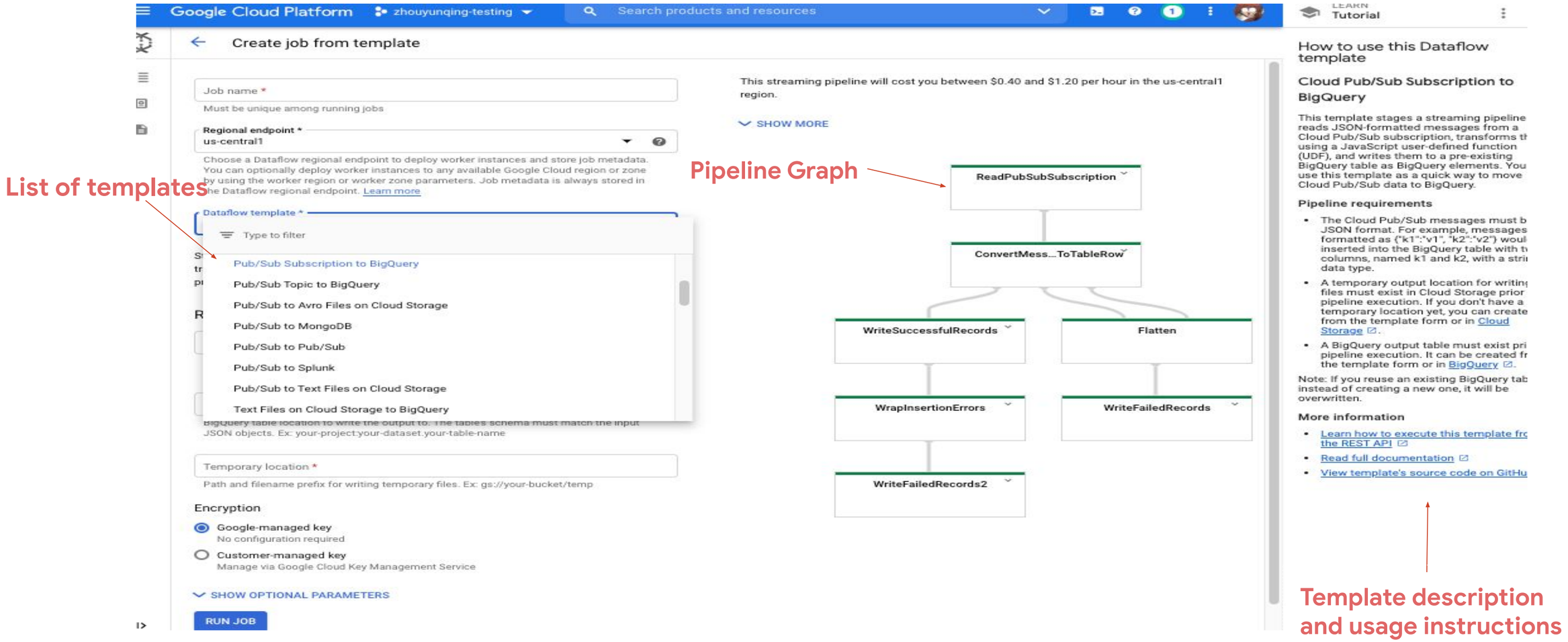
Proprietary + Confidential



Exam Tips:

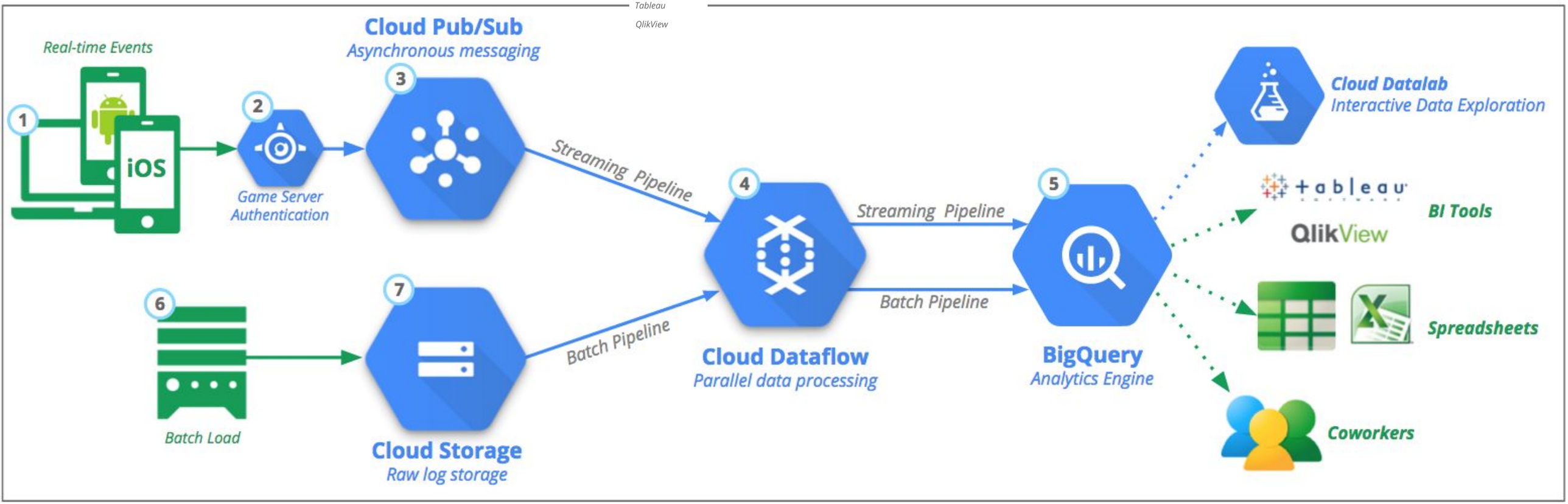
- *Dataflow does NOT store data! There is always a Source and a Sink*

Dataflow: Google Provides Templates for different use-cases

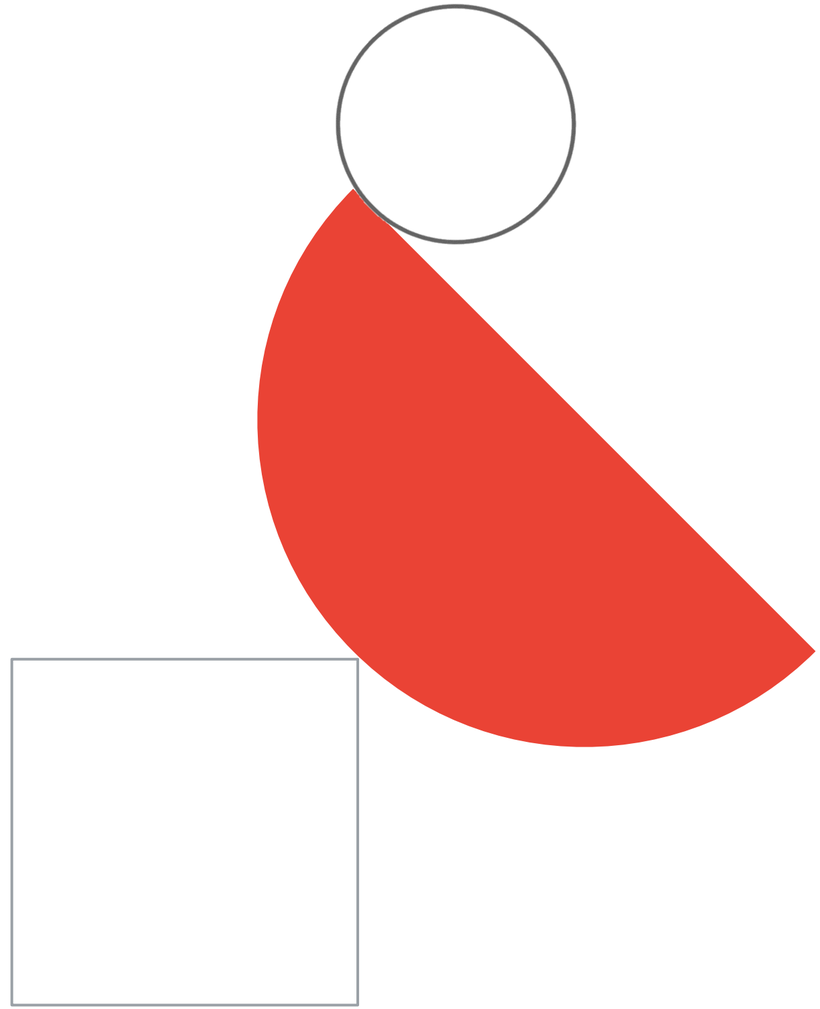


Proprietary + Confidential

Example architecture for data analytics

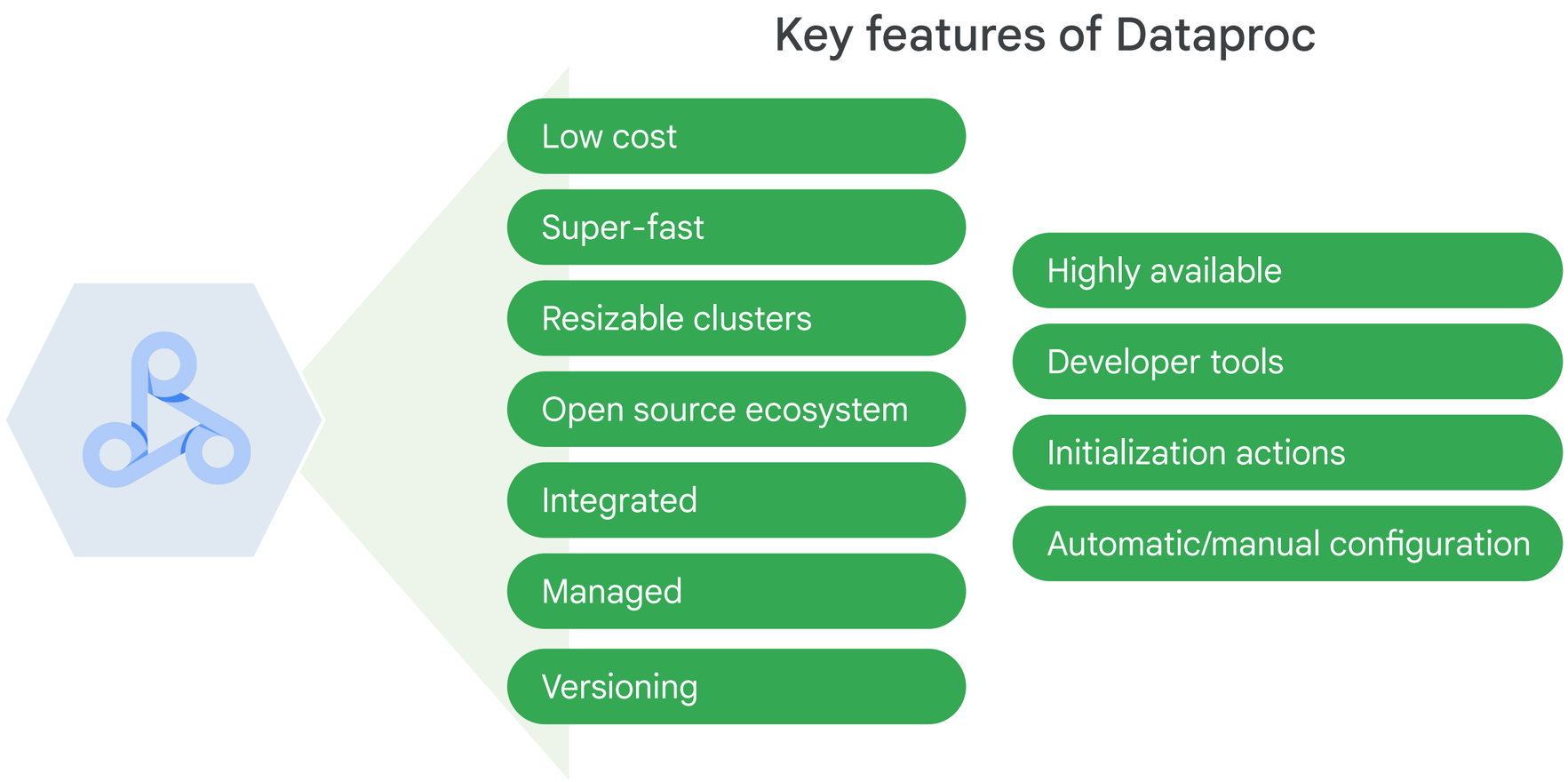


Dataproc



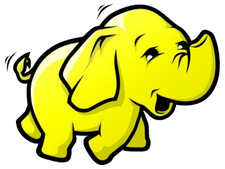
Proprietary + Confidential

Dataproc is a managed service for running Hadoop and Spark data processing workload

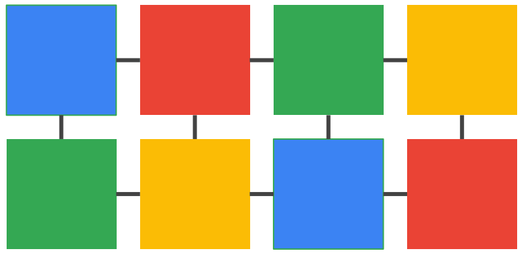


Proprietary + Confidential

With ephemeral clusters, you only pay for what you use

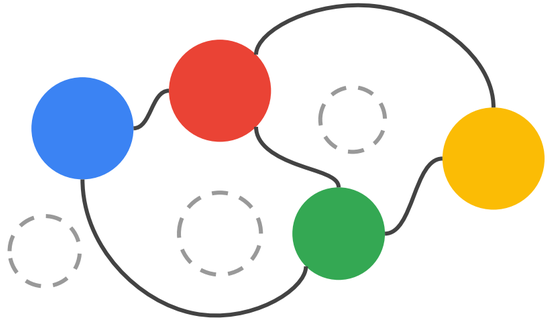


Persistent clusters



Resources are active at all times. You are constantly paying for all available clusters.

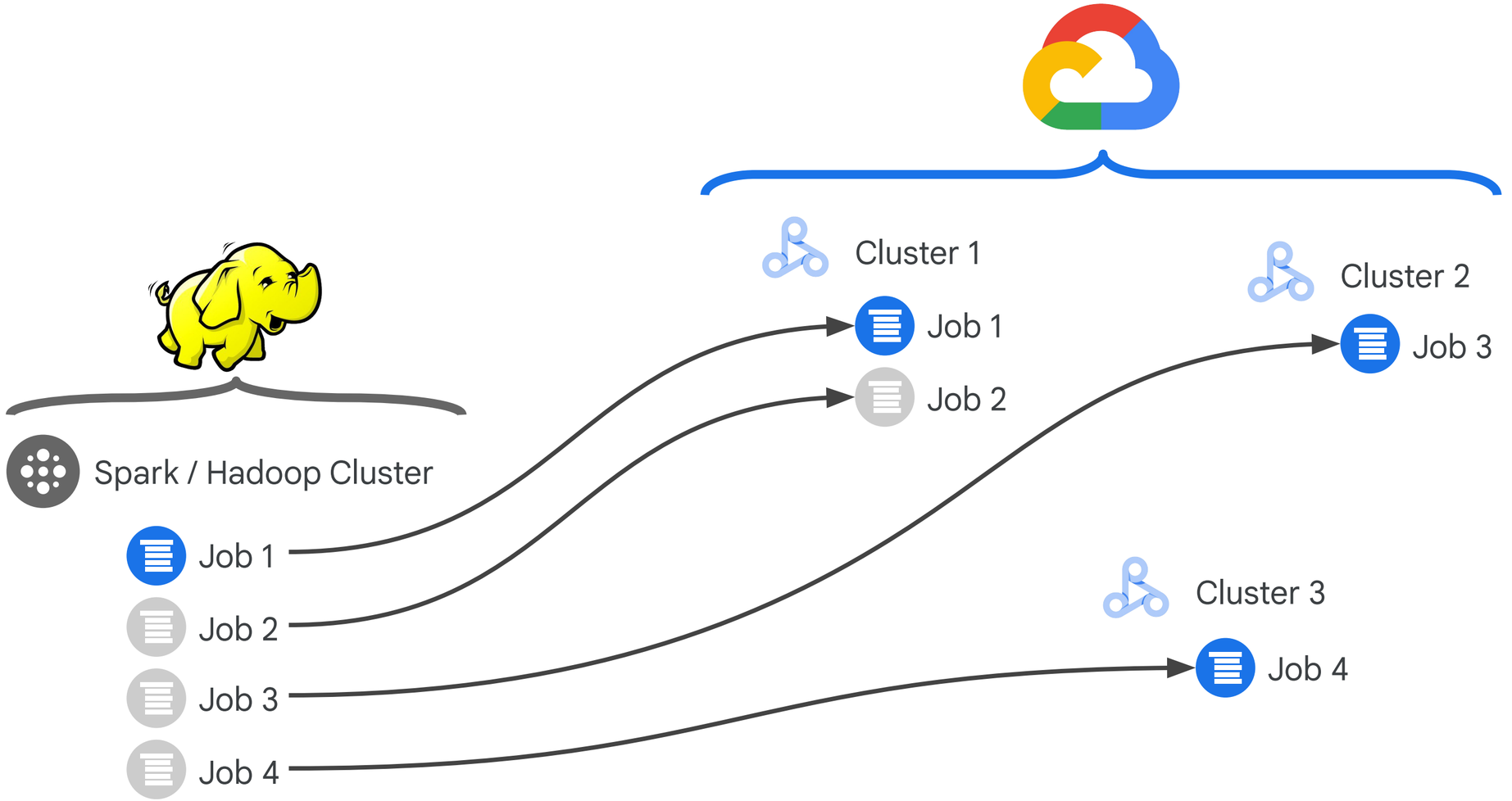
Ephemeral clusters



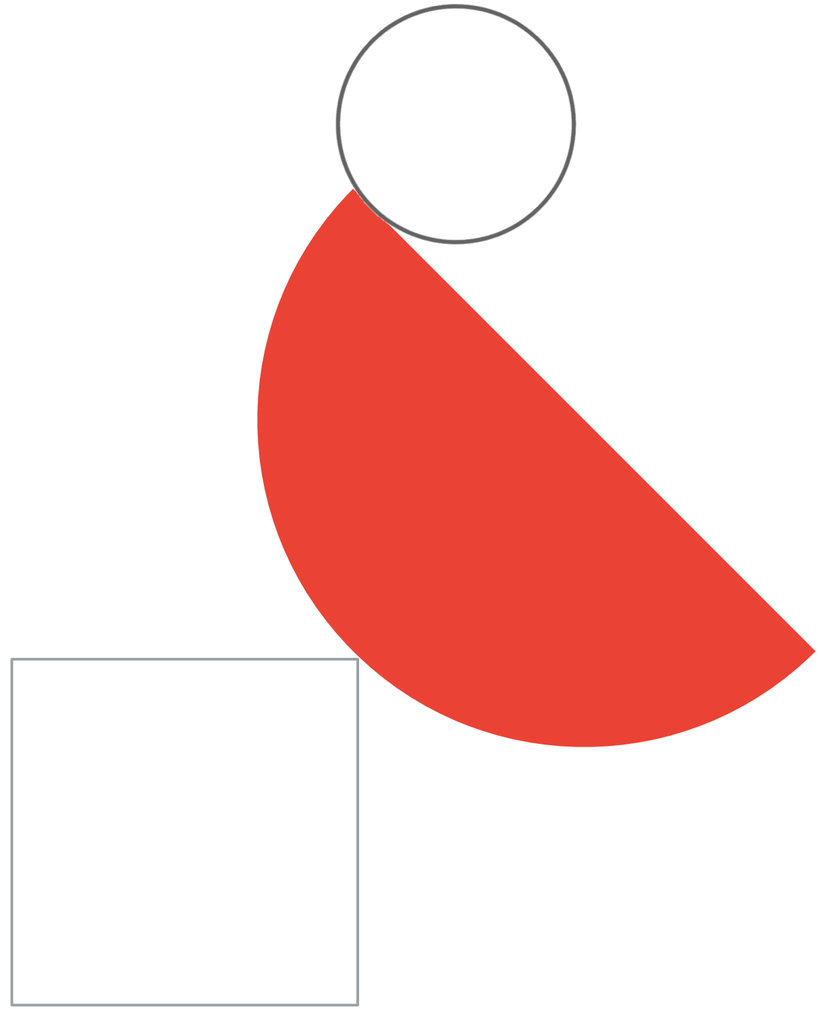
Required resources are active only when being used. You only pay for what you use.

Proprietary + Confidential

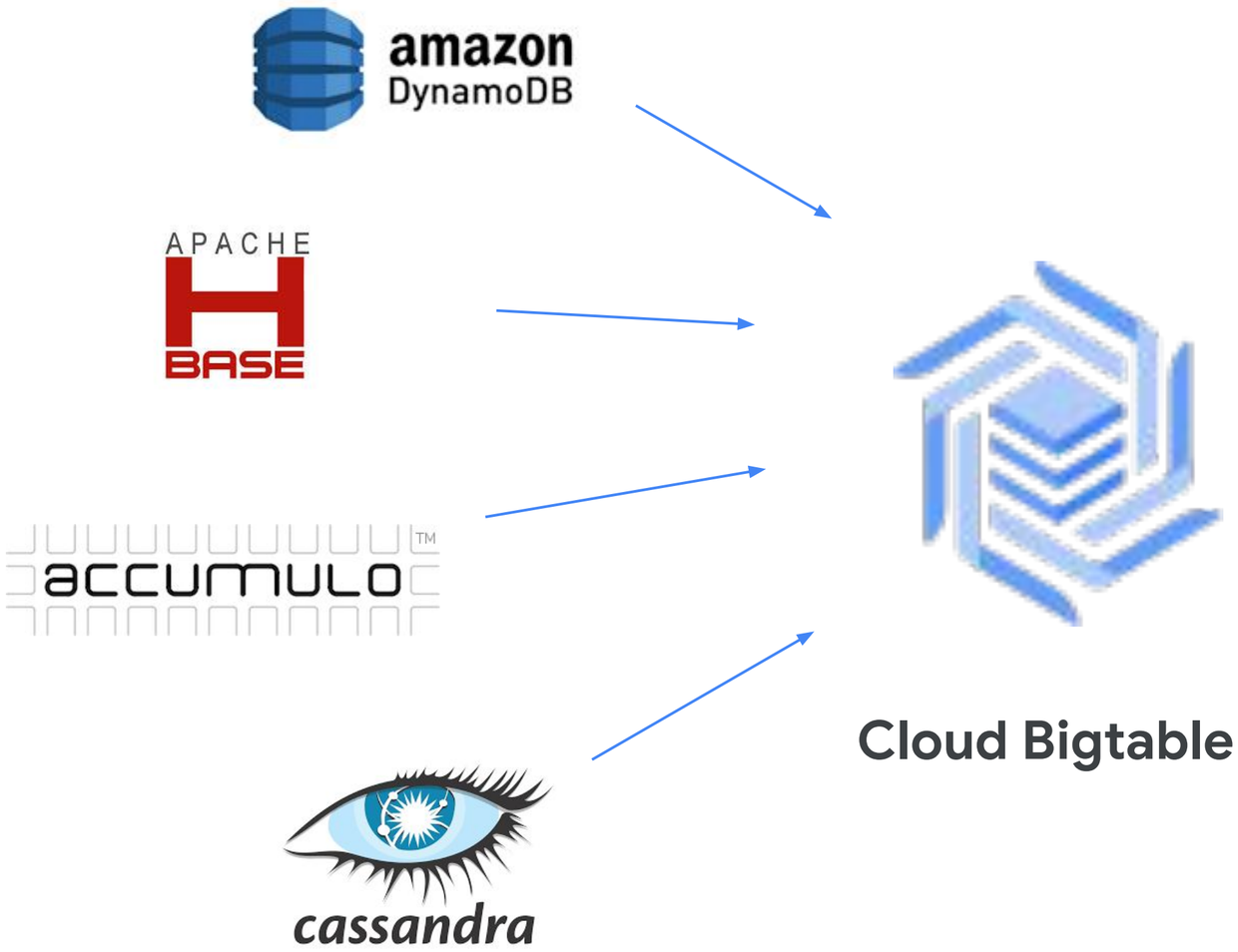
Split clusters and jobs



Bigtable



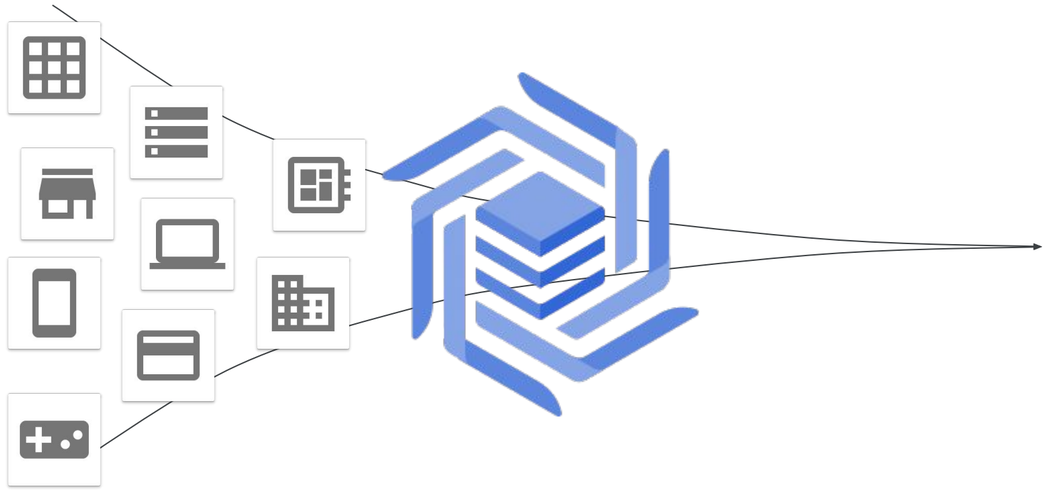
Bigtable is a common migration target for key-value, wide-column and time-series databases



- **Petabyte-scale**
- **fully managed** NoSQL database service for use cases where **low latency** random data access, **scalability** and **reliability** are critical.
- **scales seamlessly**
- **integrates** with the Apache® ecosystem and supports the HBase™ API.

Bigtable is all about speeeeed

High throughput



10MB/s write throughput per node

Up to **220MB/s** scan throughput

10.000 queries (r/w) **per second per node**

Low latency

Consistent 99th percentile **low single digit** read and write latency

What is Bigtable good for?

Use Case Examples

- **Time-series** data, such as CPU and memory usage over time for multiple servers.
- **Marketing data**, such as purchase histories and customer preferences.
- **Financial data**, such as transaction histories, stock prices, and currency exchange rates.
- **Internet of Things** data, such as usage reports from energy meters and home appliances.
- **Graph data**, such as information about how users are connected to one another.

Applications that need...

- Very high throughput
- Scalability
- Non-Structured key/value data where each value is no larger than 10MB

Storage Engine

- Batch MapReduce
- Stream Processing/Analytics
- ML applications

Exam Tip: *types of apps where you'd consider using Bigtable: recommendation engines, personalizing user experience, Internet of Things, real-time analytics, fraud detection, migrating from HBase or Cassandra, Fintech, gaming, high-throughput data streaming for creating / improving ML models.*

What is Bigtable not good for?

Not good for...

- Not a relational database
- No SQL Queries or Joins
- No Multi-Row Transactions

Considerations

- You need full SQL support for OLTP
  → consider Spanner or CloudSQL
- Interactive querying for OLAP
  → consider BigQuery
- Need to store immutable blobs larger than 10MB (e.g. movies, images)
  → consider Cloud Storage

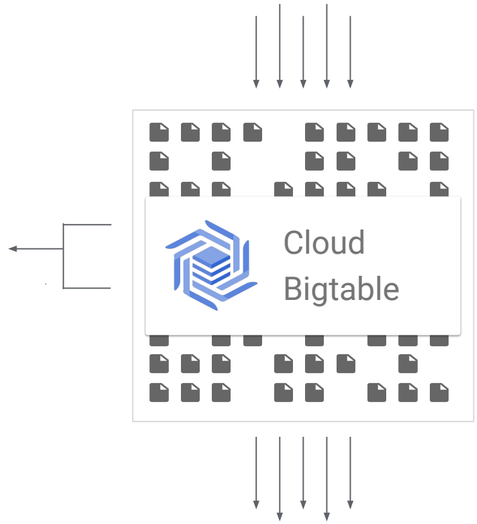
Bigtable for analytics... ?

Bigtable vs BigQuery

NoSQL wide column database

Low latency per-entry access

Heavy read/write events



Enterprise data warehouse for relational structured data

Large scale, ad-hoc SQL-based OLAP analysis

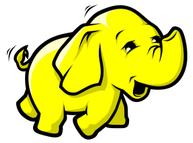
Organizational insights

Analyze data from Cloud Bigtable database

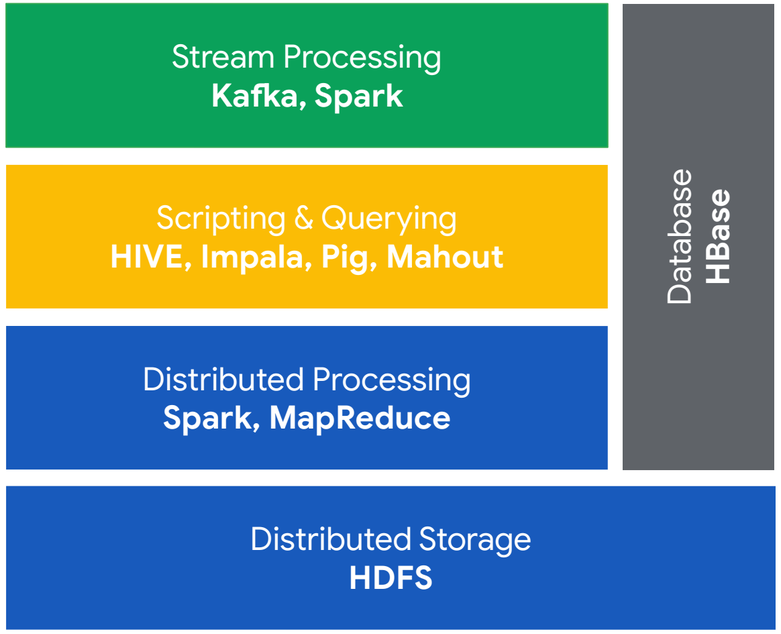
Optimized for	**Point read/write**	**Ad-hoc analysis and reporting**
Typical target	**User/entity level**	**Cohort /population level**

Bigtable: Hadoop migration and modernization

"Before"



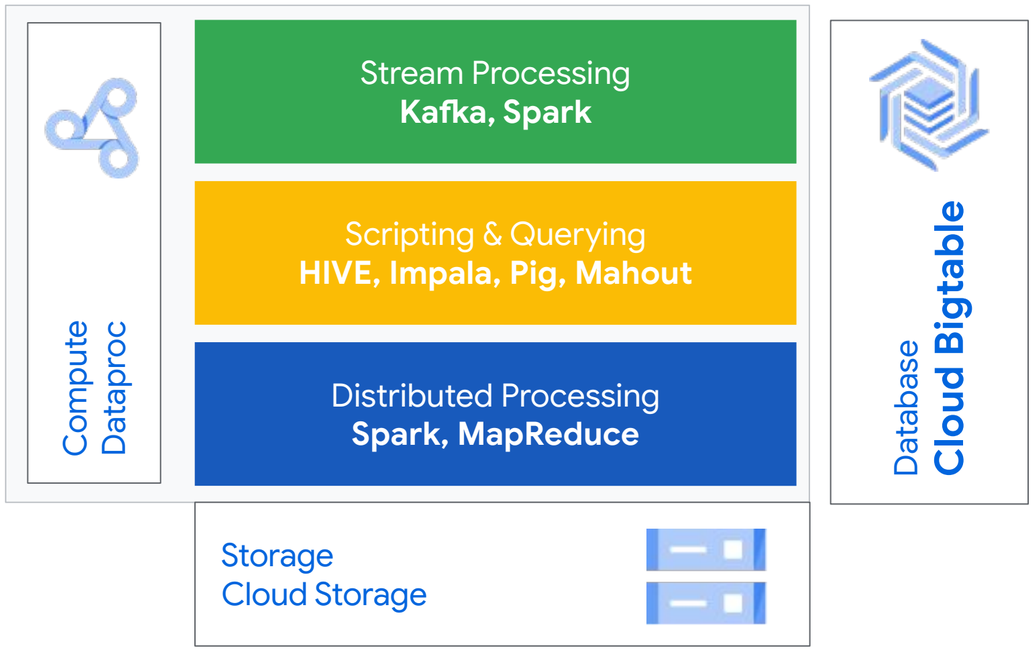
Apache Hadoop/HBase



Simplified
Hadoop Stack

"After"

Data Ecosystem / Cloud Bigtable



Google Cloud
Storage and Databases

Exam Tip: *Main goal: decoupling of storage & compute. As a consequence, you can treat Dataproc clusters as job-specific / ephemeral*

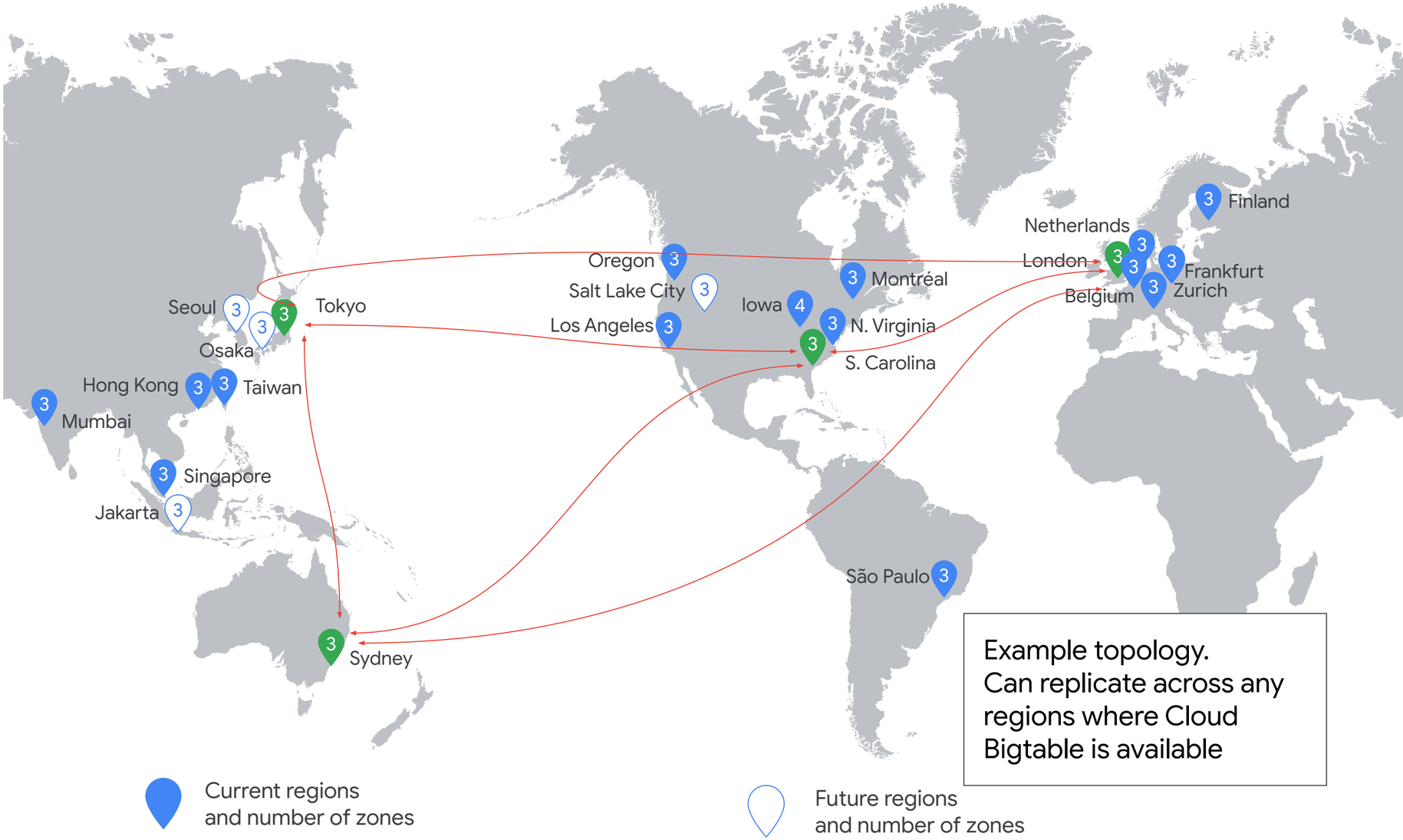
Bigtable: Replication

Regional replication

- SLA up to 99.999%
- Isolate serving and analytics
- Independently scale clusters
- Automatic failover in case of a zonal failure

Global replication

- **Increases durability/availability beyond one region**
- **Fastest region-specific access**
- **Option for DR replica for regulated customers**



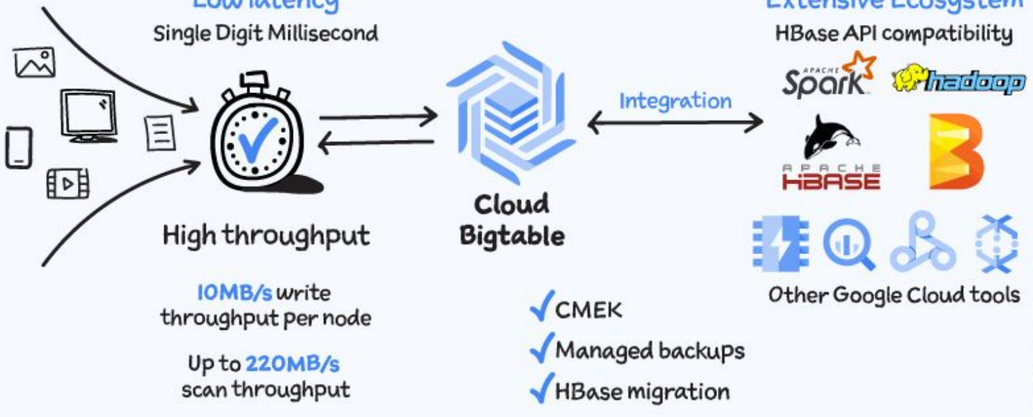
Cloud Bigtable

#GCPSketchnotes

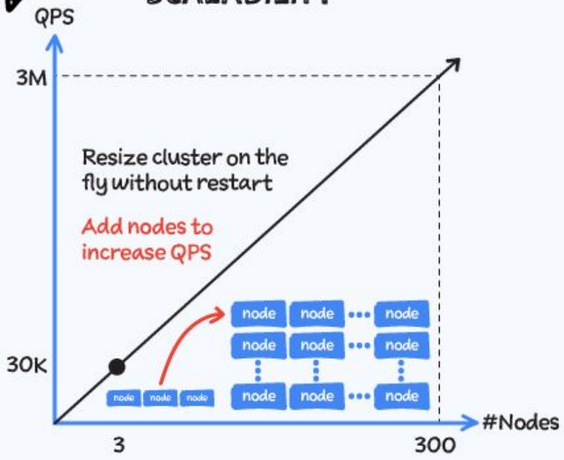
@PVERGADIA
THECLOUDGIRL.DEV 4.14.2021

What is Cloud Bigtable?

FULLY MANAGED PETABYTE-SCALE NOSQL DATABASE
For heavy reads/writes



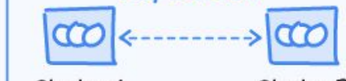
SCALABILITY



Each node provides upto 10,000 QPS (read/write)

FAILOVER FOR HIGH AVAILABILITY (HA)

Single/multi-region/global Replication



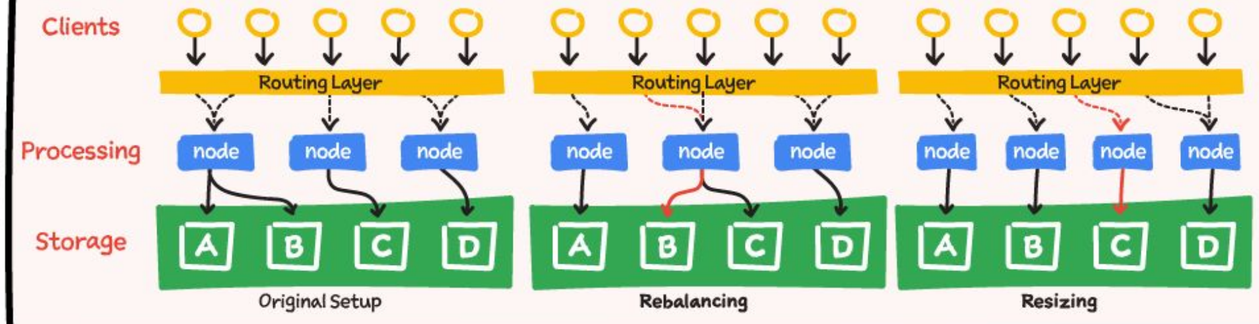
No manual steps to ensure consistency, repair data, or synchronize writes/deletes

How does Cloud Bigtable Optimize throughput?

Cloud Bigtable separates processing from storage, each node has access to a group of database rows

Rebalance load automatically to improve performance

Resize nodes (with no downtime) for best overall throughput

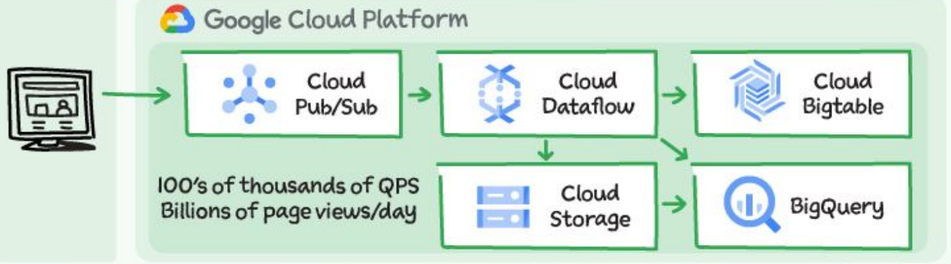


Cloud Bigtable «Use case examples»

REAL-TIME ANALYTICS FOR HUGE WORKLOADS

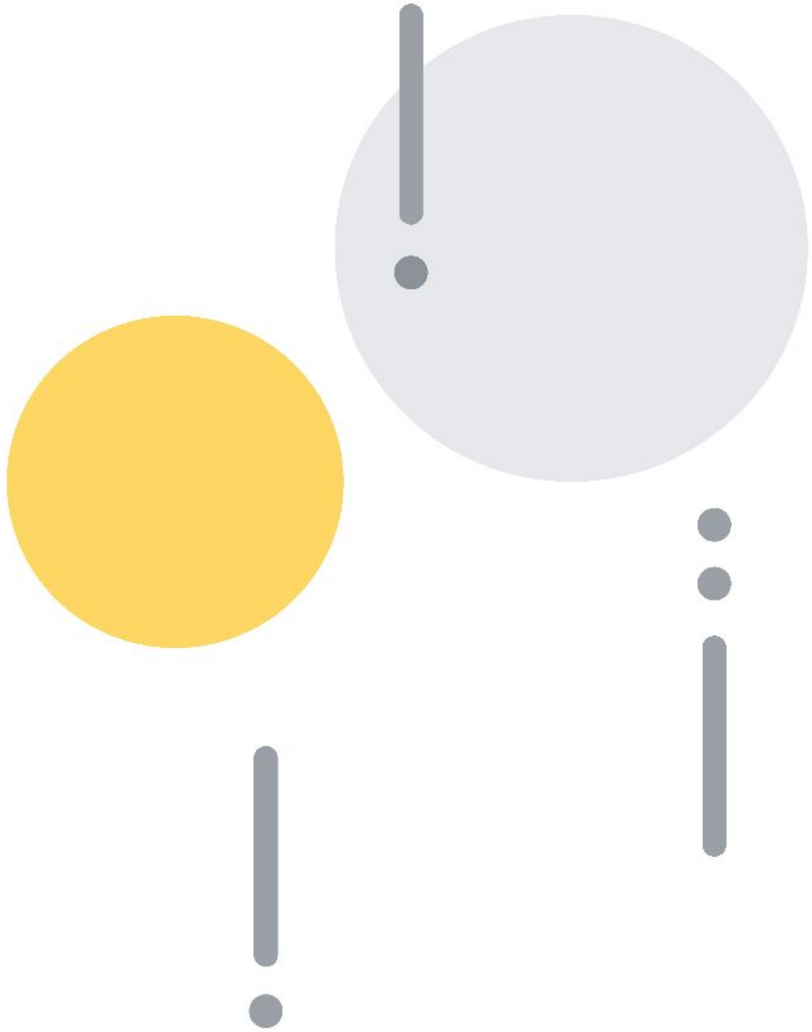
FinTech · AdTech · Gaming · IoT · ML/AI Personalization

REAL-TIME AD BIDDING PLATFORM

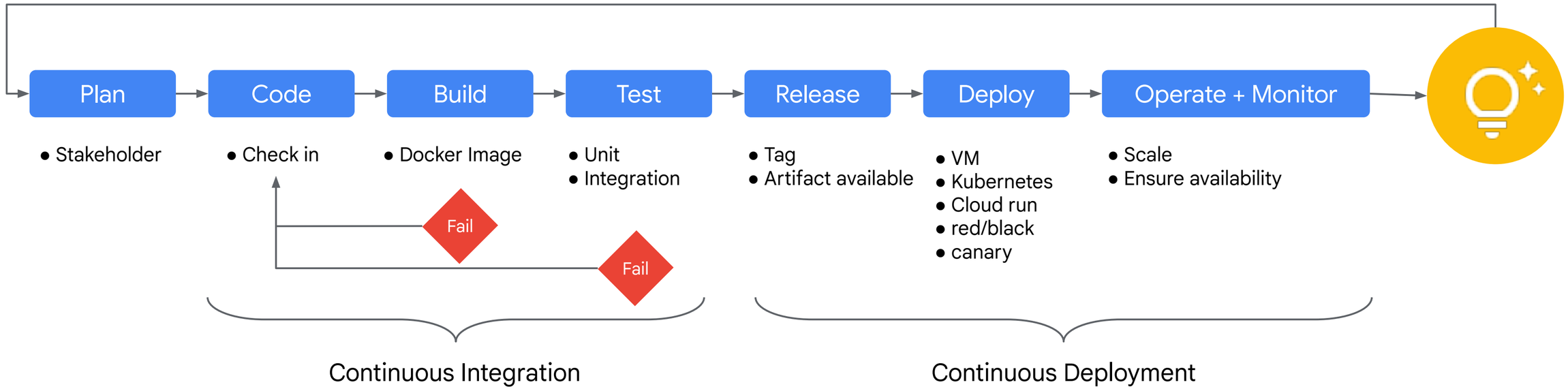


Proprietary + Confidential

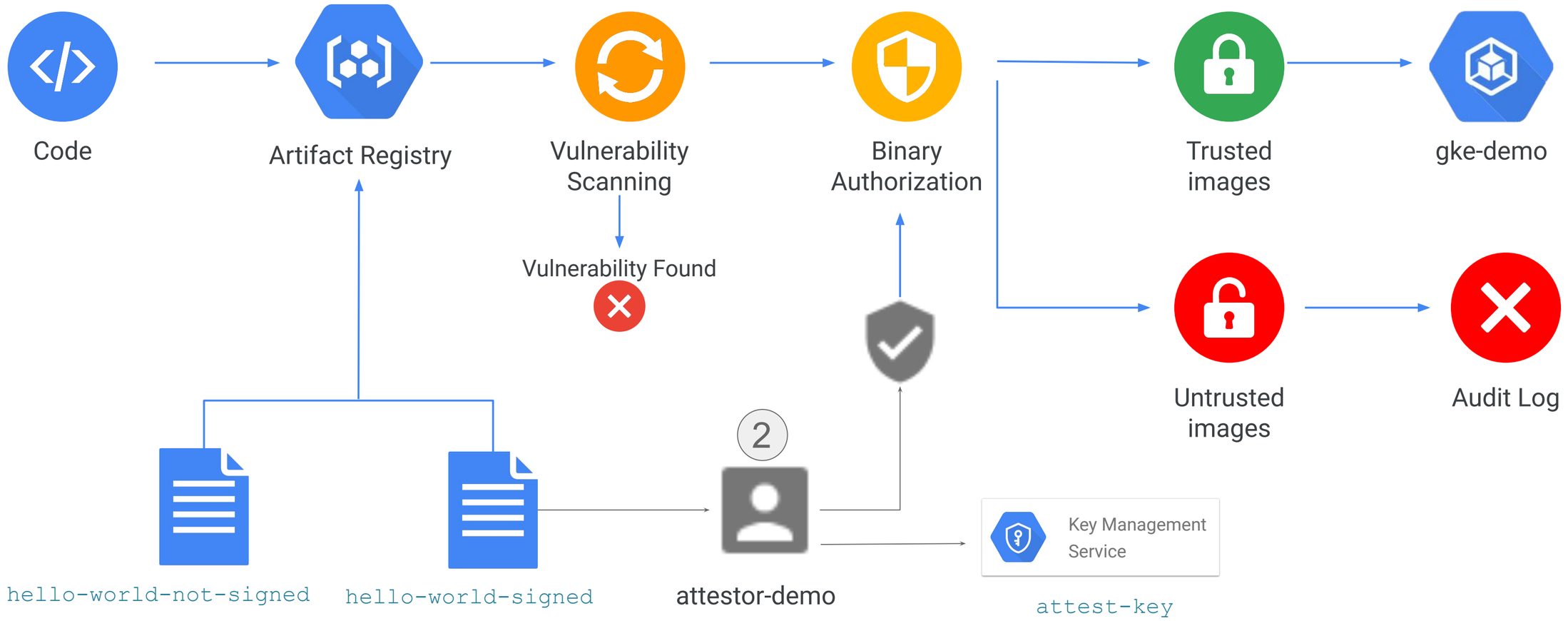
CI/CD Pipeline



Process optimization



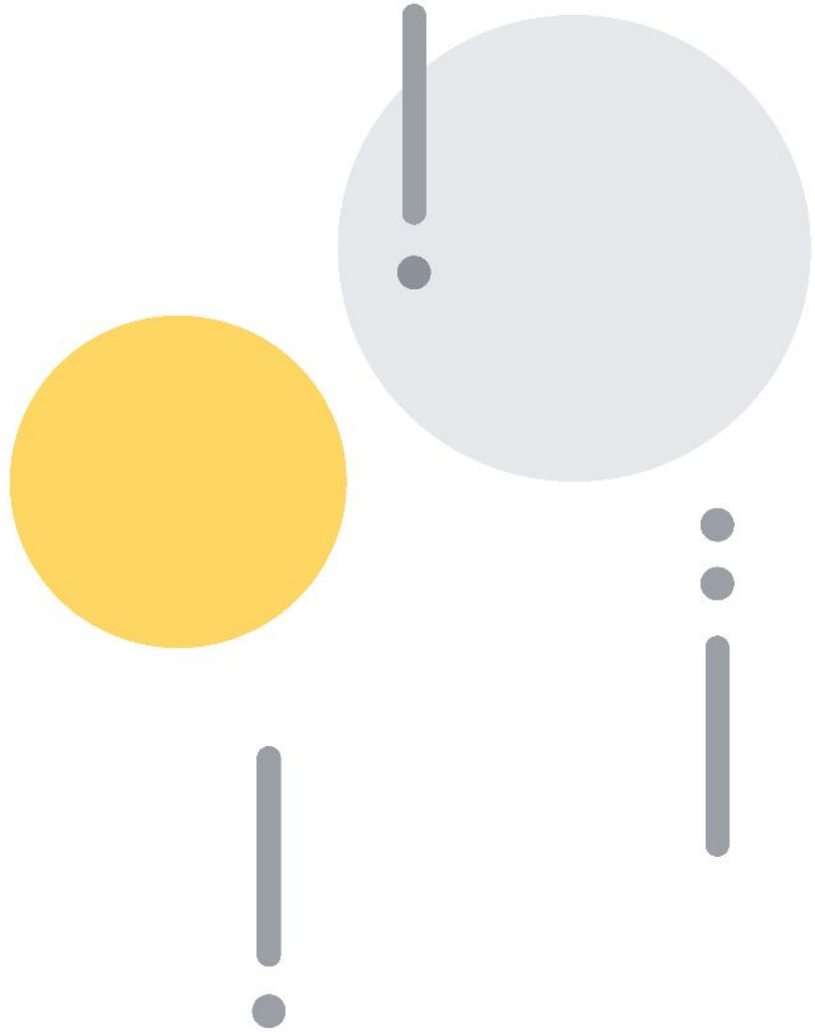
Example of end-to-end CI/CD pipeline



Google Cloud

Proprietary + Confidential

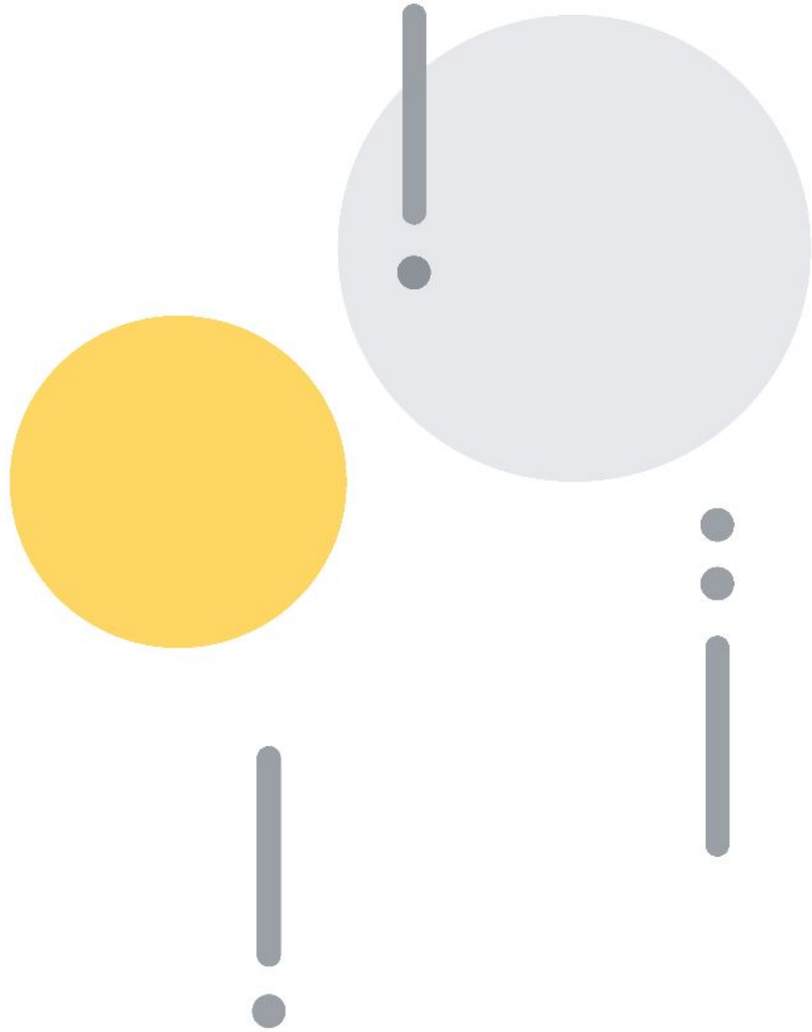
Cloud Composer



Link

Proprietary + Confidential

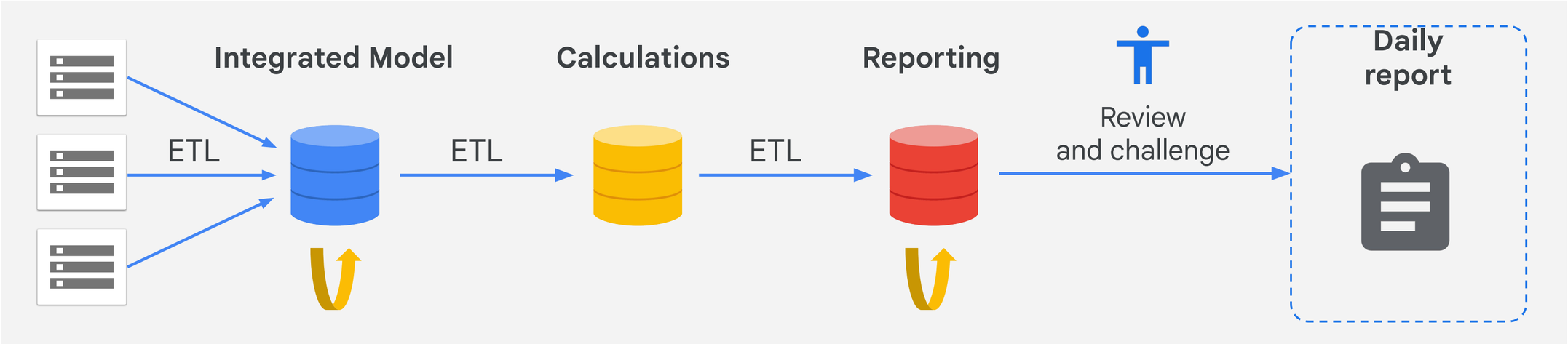
Case Study



Data engineer case study 01:

A customer had this interesting business requirement...

A daily reporting pipeline with multiple sources and complex dependencies
Human intervention to data check quality, inputs, and proceed to next stage
Need daily updates on yesterday's data, but takes >24 hours to run.



OPTIONAL study materials:

[READING]

- ETL
- Performing ETL from a relational database into BigQuery using Dataflow
- Integrations with Bigtable
- Cloud Bigtable Service Level Agreement (SLA)
- Federated queries
- Cloud Data Fusion
- Streaming pipelines

[VIDEOS]

- EL vs ETL vs ELT in Google Cloud Bigquery
- What is Cloud Bigtable? and What can you do with Bigtable?
- Bigtable & Time Series Data
- Querying Cloud SQL from BigQuery
- Querying external data with BigQuery
- Cloud Data Fusion: Concepts of Networking
- Building and executing a pipeline graph in Cloud Data Fusion
- Real-Time Stream Analytics with Google Cloud Dataflow: Common Use Cases & Patterns
- Building stream processing pipelines with Dataflow

OPTIONAL study materials:

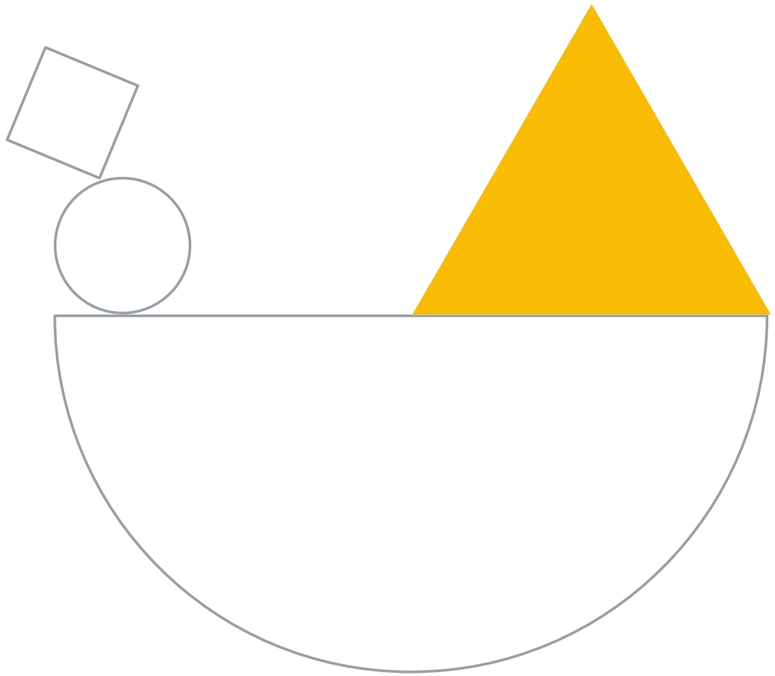
[READING]

- Apache Airflow and Cloud Composer
- Cloud Build

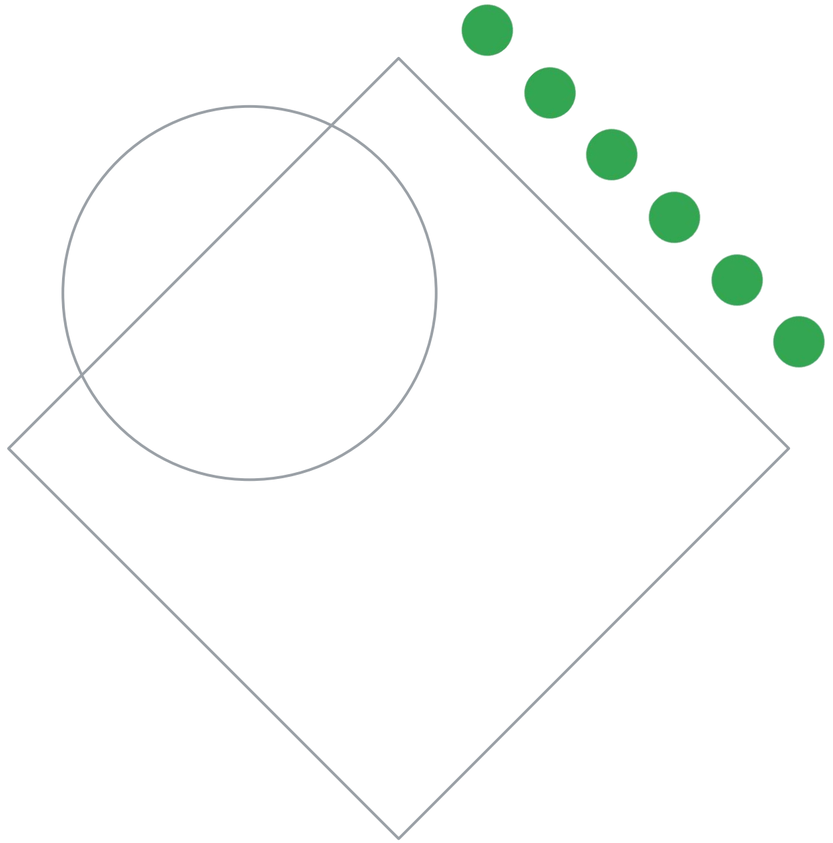
[VIDEOS]

- How to use Cloud Composer for data orchestration
- Deploy to Cloud Run using Cloud Build and Artifact Registry

Diagnostic questions

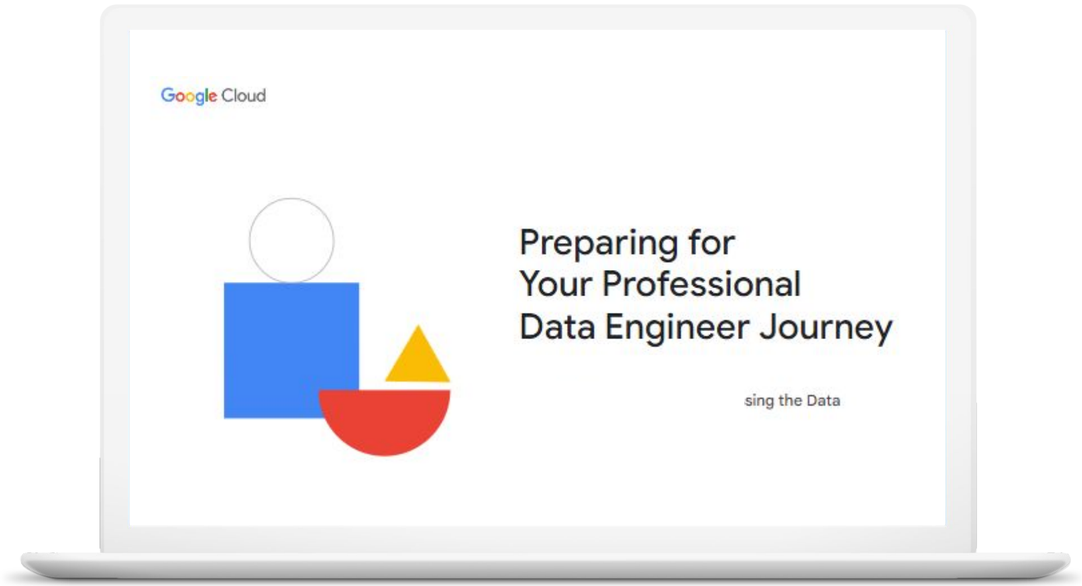


Review and study planning



Your study plan:

Ingesting and processing the data



2.1 Planning the data pipelines

2.2 Building the pipelines

2.3 Deploying and operationalizing the pipelines

2.1 Planning the data pipelines

Considerations include:

- Defining data sources and sinks
- Defining data transformation logic
- Networking fundamentals
- Data encryption

2.1 Diagnostic Question 01 Discussion

Your data engineering team receives data in JSON format from external sources at the end of each day. You need to design the data pipeline.

What should you do?

A. Store the data in Cloud Storage and create an extract, transform, and load (ETL) pipeline.

B. Make your BigQuery data warehouse public and ask the external sources to insert the data.

C. Create a public API to allow external applications to add the data to your warehouse.

D. Store the data in persistent disks and create an ETL pipeline.

2.1 Diagnostic Question 01 Discussion

Your data engineering team receives data in **JSON format from external sources at the end of each day**. You need to design the **data pipeline**.

What should you do?

A. Store the data in Cloud Storage and create an extract, transform, and load (ETL) pipeline.

B. Make your BigQuery data warehouse public and ask the external sources to insert the data.

C. Create a public API to allow external applications to add the data to your warehouse.

D. Store the data in persistent disks and create an ETL pipeline.

2.1 Diagnostic Question 01 Discussion

Your data engineering team receives data in **JSON format from external sources at the end of each day**. You need to design the **data pipeline**.

What should you do?

A. Store the data in Cloud Storage and create an extract, transform, and load (ETL) pipeline.

B. Make your BigQuery data warehouse public and ask the external sources to insert the data.

C. Create a public API to allow external applications to add the data to your warehouse.

D. Store the data in persistent disks and create an ETL pipeline.

2.1 Diagnostic Question 02 Discussion

The first stage of your data pipeline processes tens of terabytes of financial data and creates a sparse, time-series dataset as a key-value pair.

Which of these is a suitable sink for the pipeline's first stage?

A. Cloud Storage
B. Cloud SQL
C. AlloyDB
D. Bigtable

2.1 Diagnostic Question 02 Discussion

The **first stage of your data pipeline processes** tens of terabytes of financial data and creates a **sparse, time-series dataset** as a key-value pair.

Which of these is a suitable sink for the pipeline's first stage?

A. Cloud Storage
B. Cloud SQL
C. AlloyDB
D. Bigtable

2.1 Diagnostic Question 02 Discussion

The **first stage of your data pipeline processes** tens of terabytes of financial data and creates a **sparse, time-series dataset** as a key-value pair.

Which of these is a suitable sink for the pipeline's first stage?

A. Cloud Storage
B. Cloud SQL
C. AlloyDB
D. Bigtable

Link

2.1 Diagnostic Question 03 Discussion

You are processing large amounts of input data in BigQuery. You need to combine this data with a small amount of frequently changing data that is available in Cloud SQL.

What should you do?

A. Copy the data from Cloud SQL to a new BigQuery table hourly.

B. Copy the data from Cloud SQL and create a combined, normalized table hourly.

C. Use a federated query to get data from Cloud SQL.

D. Create a Dataflow pipeline to combine the BigQuery and Cloud SQL data when the Cloud SQL data changes.

2.1 Diagnostic Question 03 Discussion

You are processing large amounts of input data in BigQuery. You need to combine this data with a small amount of **frequently changing data** that is available in **Cloud SQL**.

What should you do?

A. Copy the data from Cloud SQL to a new BigQuery table hourly.

B. Copy the data from Cloud SQL and create a combined, normalized table hourly.

C. Use a federated query to get data from Cloud SQL.

D. Create a Dataflow pipeline to combine the BigQuery and Cloud SQL data when the Cloud SQL data changes.

2.1 Diagnostic Question 03 Discussion

You are processing large amounts of input data in BigQuery. You need to combine this data with a small amount of **frequently changing data** that is available in **Cloud SQL**.

What should you do?

A. Copy the data from Cloud SQL to a new BigQuery table hourly.

B. Copy the data from Cloud SQL and create a combined, normalized table hourly.

C. Use a federated query to get data from Cloud SQL.

D. Create a Dataflow pipeline to combine the BigQuery and Cloud SQL data when the Cloud SQL data changes.

Link

2.1 Planning the data pipelines

Courses

Modernizing Data Lakes and Data Warehouses on Google Cloud
- Introduction to Data Engineering
- Building a Data Lake
- Building a Data Warehouse

Building Batch Data Pipelines on Google Cloud
- Executing Spark on Dataproc
- Manage Data Pipelines with Cloud Data Fusion and Cloud Composer

Building Resilient Streaming Analytics Systems on Google Cloud
- High-Throughput BigQuery and Bigtable Streaming Features

Serverless Data Processing with Dataflow: Develop Pipelines
- Beam Concepts Review
- Sources and Sinks
- Schemas

Skill Badges

Perform Foundational Data, ML, and AI Tasks in Google Cloud
Engineer Data with Google Cloud

Documentation

What Data Pipeline Architecture should I use? | Google Cloud Blog

Bigtable overview

Cloud SQL federated queries | BigQuery

Exploring new features in BigQuery federated queries | Google Cloud Blog

2.2 Building the pipelines

Considerations include:

- Data cleansing
- Identifying the services (e.g., Dataflow, Apache Beam, Dataproc, Cloud Data Fusion, BigQuery, Pub/Sub, Apache Spark, Hadoop ecosystem, and Apache Kafka)
- Transformations
  - Batch
  - Streaming (e.g., windowing, late arriving data)
  - Language
  - Ad hoc data ingestion (one-time or automated pipeline)
- Data acquisition and import
- Integrating with new data sources

2.2 Diagnostic Question 04 Discussion

Your company has multiple data analysts but a limited data engineering team. You need to choose a tool where the analysts can build data pipelines themselves with a graphical user interface.

Which of these products is the most appropriate?

A. Dataflow
B. Cloud Data Fusion
C. Dataproc
D. Cloud Composer

2.2 Diagnostic Question 04 Discussion

Your company has multiple data analysts but a limited data engineering team. You need to choose a **tool** where the analysts can **build data pipelines themselves** with a **graphical user interface.**

Which of these products is the most appropriate?

A. Dataflow
B. Cloud Data Fusion
C. Dataproc
D. Cloud Composer

2.2 Diagnostic Question 04 Discussion

Your company has multiple data analysts but a limited data engineering team. You need to choose a **tool** where the analysts can **build data pipelines themselves** with a **graphical user interface.**

Which of these products is the most appropriate?

A. Dataflow

B. Cloud Data Fusion

C. Dataproc

D. Cloud Composer

Link

2.2 Diagnostic Question 05 Discussion

You manage a PySpark batch data pipeline by using Dataproc. You want to take a hands-off approach to running the workload, and you do not want to provision and manage your own cluster.

What should you do?

A. Configure the job to run on Dataproc Serverless.
B. Configure the job to run with Spot VMs.
C. Rewrite the job in Spark SQL.
D. Rewrite the job in Dataflow with SQL.

2.2 Diagnostic Question 05 Discussion

You manage a **PySpark batch data pipeline** by using Dataproc. You want to take a hands-off approach to running the workload, and you **do not want to provision and manage your own cluster.**

What should you do?

A. Configure the job to run on Dataproc Serverless.
B. Configure the job to run with Spot VMs.
C. Rewrite the job in Spark SQL.
D. Rewrite the job in Dataflow with SQL.

2.2 Diagnostic Question 05 Discussion

You manage a **PySpark batch data pipeline** by using Dataproc. You want to take a hands-off approach to running the workload, and you **do not want to provision and manage your own cluster.**

What should you do?

A. Configure the job to run on Dataproc Serverless.
B. Configure the job to run with Spot VMs.
C. Rewrite the job in Spark SQL.
D. Rewrite the job in Dataflow with SQL.

LInk

2.2 Diagnostic Question 06 Discussion

You need to run batch jobs, which could take many days to complete. You do not want to manage the infrastructure provisioning.

A. Use Cloud Scheduler to run the jobs.
B. Use Workflows to run the jobs.
C. Run the jobs on Batch.
D. Use Cloud Run to run the jobs.

What should you do?

2.2 Diagnostic Question 06 Discussion

You need to run **batch jobs**, which could take many days to complete. You do not want to manage the infrastructure provisioning.

What should you do?

A. Use Cloud Scheduler to run the jobs.

B. Use Workflows to run the jobs.

C. Run the jobs on Batch.

D. Use Cloud Run to run the jobs.

2.2 Diagnostic Question 06 Discussion

You need to run **batch jobs**, which could take many days to complete. You do not want to manage the infrastructure provisioning.

What should you do?

A. Use Cloud Scheduler to run the jobs.

B. Use Workflows to run the jobs.

C. Run the jobs on Batch.

D. Use Cloud Run to run the jobs.

2.2 Diagnostic Question 07 Discussion

You are creating a data pipeline for streaming data on Dataflow for Cymbal Retail's point of sales data. You want to calculate the total sales per hour on a continuous basis.

Which of these windowing options should you use?

A. Hopping windows (sliding windows in Apache Beam)
B. Session windows
C. Global window
D. Tumbling windows (fixed windows in Apache Beam)

2.2 Diagnostic Question 07 Discussion

You are creating a data pipeline for **streaming data** on Dataflow for Cymbal Retail's point of sales data. You want to calculate the **total sales per hour** on a continuous basis.

Which of these windowing options should you use?

A. Hopping windows (sliding windows in Apache Beam)
B. Session windows
C. Global window
D. Tumbling windows (fixed windows in Apache Beam)

2.2 Diagnostic Question 07 Discussion

You are creating a data pipeline for **streaming data** on Dataflow for Cymbal Retail's point of sales data. You want to calculate the **total sales per hour** on a continuous basis.

Which of these windowing options should you use?

A. Hopping windows (sliding windows in Apache Beam)
B. Session windows
C. Global window
D. Tumbling windows (fixed windows in Apache Beam)

Link

2.2 Diagnostic Question 08 Discussion

You want to build a streaming data analytics pipeline in Google Cloud. You need to choose the right products that support streaming data.

Which of these would you choose?

A. Pub/Sub, Dataflow, BigQuery
B. Pub/Sub, Dataprep, BigQuery
C. Cloud Storage, Dataflow, Cloud SQL
D. Cloud Storage, Dataprep, AlloyDB

2.2 Diagnostic Question 08 Discussion

You want to build a streaming data analytics pipeline in Google Cloud. You need to choose the **right products that support streaming data.**

Which of these would you choose?

A. Pub/Sub, Dataflow, BigQuery
B. Pub/Sub, Dataprep, BigQuery
C. Cloud Storage, Dataflow, Cloud SQL
D. Cloud Storage, Dataprep, AlloyDB

2.2 Diagnostic Question 08 Discussion

You want to build a streaming data analytics pipeline in Google Cloud. You need to choose the **right products that support streaming data.**

Which of these would you choose?

A. Pub/Sub, Dataflow, BigQuery

B. Pub/Sub, Dataprep, BigQuery

C. Cloud Storage, Dataflow, Cloud SQL

D. Cloud Storage, Dataprep, AlloyDB

2.2 Building the pipelines

Courses

Building Batch Data Pipelines on Google Cloud
- Introduction to Building Batch Data Pipelines
- Executing Spark on Dataproc
- Serverless Data Processing with Dataflow
- Manage Data Pipelines with Cloud Data Fusion and Cloud Compose

Building Resilient Streaming Analytics Systems on Google Cloud
- Serverless Messaging with Pub/Sub
- Dataflow Streaming Features

Serverless Data Processing with Dataflow: Foundations
- Separating Compute and Storage with Dataflow

Serverless Data Processing with Dataflow: Develop Pipelines
- Windows, Watermarks, and Triggers
- States and Timers
- Dataflow SQL and DataFrames

Serverless Data Processing with Dataflow: Operations
- Performance
- Testing and CI/CD
- Flex Templates

Skill Badges

Perform Foundational Data, ML, and AI Tasks in Google Cloud

Documentation

Cloud Data Fusion overview

What is Dataproc Serverless?

Introduction to Google Batch

Get started with Batch | Google Cloud

Streaming pipelines | Cloud Dataflow

Basics of the Beam model

Streaming analytics solutions | Google Cloud

2.3 Deploying an operationalizing the pipelines

Considerations include:

- Job automation and orchestration (e.g., Cloud Composer and Workflows)
- CI/CD (Continuous Integration and Continuous Deployment

2.3 Diagnostic Question 09 Discussion

You have a data pipeline that requires you to monitor a Cloud Storage bucket for a file, start a Dataflow job to process data in the file, run a shell script to validate the processed data in BigQuery, and then delete the original file. You need to orchestrate this pipeline by using recommended tools.

Which product should you choose?

A. Cloud Tasks
B. Cloud Composer
C. Cloud Scheduler
D. Cloud Run

2.3 Diagnostic Question 09 Discussion

You have a data pipeline that requires you to **monitor a Cloud Storage bucket for a file, start a Dataflow job to process data in the file, run a shell script to validate the processed data in BigQuery, and then delete the original file.** You need to **orchestrate** this pipeline by using recommended tools.

Which product should you choose?

A. Cloud Tasks
B. Cloud Composer
C. Cloud Scheduler
D. Cloud Run

2.3 Diagnostic Question 09 Discussion

You have a data pipeline that requires you to **monitor a Cloud Storage bucket for a file, start a Dataflow job to process data in the file, run a shell script to validate the processed data in BigQuery, and then delete the original file.** You need to **orchestrate** this pipeline by using recommended tools.

Which product should you choose?

A. Cloud Tasks
B. Cloud Composer
C. Cloud Scheduler
D. Cloud Run

Link

2.3 Diagnostic Question 10 Discussion

You are running Dataflow jobs for data processing. When developers update the code in Cloud Source Repositories, you need to test and deploy the updated code with minimal effort.

A. Terraform
B. Compute Engine
C. Cloud Code
D. Cloud Build

Which of these would you use to build your continuous integration and delivery (CI/CD) pipeline for data processing?

2.3 Diagnostic Question 10 Discussion

You are running Dataflow jobs for data processing. When developers update the code in Cloud Source Repositories, you need to **test and deploy the updated code with minimal effort.**

Which of these would you use to build your continuous integration and delivery (CI/CD) pipeline for data processing?

A. Terraform
B. Compute Engine
C. Cloud Code
D. Cloud Build

2.3 Diagnostic Question 10 Discussion

You are running Dataflow jobs for data processing. When developers update the code in Cloud Source Repositories, you need to test and deploy the updated code with minimal effort.

A. Terraform
B. Compute Engine
C. Cloud Code
D. Cloud Build

Which of these would you use to build your continuous integration and delivery (CI/CD) pipeline for data processing?

Link

2.3 Deploying and operationalizing the pipelines

Courses

[Building Batch Data Pipelines on Google Cloud]
- Manage Data Pipelines with Cloud Data Fusion and Cloud Composer

[Serverless Data Processing with Dataflow: Operations]
- Testing and CI/CD

Skill Badges

[Engineer Data with Google Cloud]

Documentation

[How to use Cloud Composer for data orchestration]

[Cloud Composer overview]

[Use a CI/CD pipeline for data-processing workflows | Google Cloud]

Knowledge Check 1

A company wants to improve productivity and decides to programmatically schedule and monitor workflows. What tool can you use to automate your workflows?

A. Dataproc

B. Apache Beam and Dataflow

C. Data Fusion

D. Cloud Composer

Knowledge Check 1

A company wants to improve productivity and decides to programmatically schedule and monitor workflows. What tool can you use to automate your workflows?

A. Dataproc

B. Apache Beam and Dataflow

C. Data Fusion

D. Cloud Composer

Knowledge Check 2

A company collects large amounts of data that is useful for improving business operations. The collected data is already clean and is in a format that is suitable for the further analysis. The company uses Google Cloud Bigquery as a data warehouse. What approach will you recommend to move this data to BigQuery?

A. Directly load the data using Extract and Load approach (EL).

B. Do transformation using Extract, Load & Transform (ELT).

C. Implement Extract, Transform and Load (ETL) pipelines using tools like Dataflow.

D. Split the data into smaller files and then move to Google Cloud.

Knowledge Check 2

A company collects large amounts of data that is useful for improving business operations. The collected data is already clean and is in a format that is suitable for the further analysis. The company uses Google Cloud Bigquery as a data warehouse. What approach will you recommend to move this data to BigQuery?

A. Directly load the data using Extract and Load approach (EL).

B. Do transformation using Extract, Load & Transform (ELT).

C. Implement Extract, Transform and Load (ETL) pipelines using tools like Dataflow.

D. Split the data into smaller files and then move to Google Cloud.