



High-Throughput BigQuery and Bigtable Streaming Features

High-Throughput BigQuery and Bigtable Streaming Features

01

Streaming into BigQuery and visualizing results

02

High-throughput streaming with Cloud Bigtable

03

Optimizing Cloud Bigtable performance



High-Throughput BigQuery and Bigtable Streaming Features

01

Streaming into BigQuery and visualizing results

02

High-throughput streaming with Cloud Bigtable

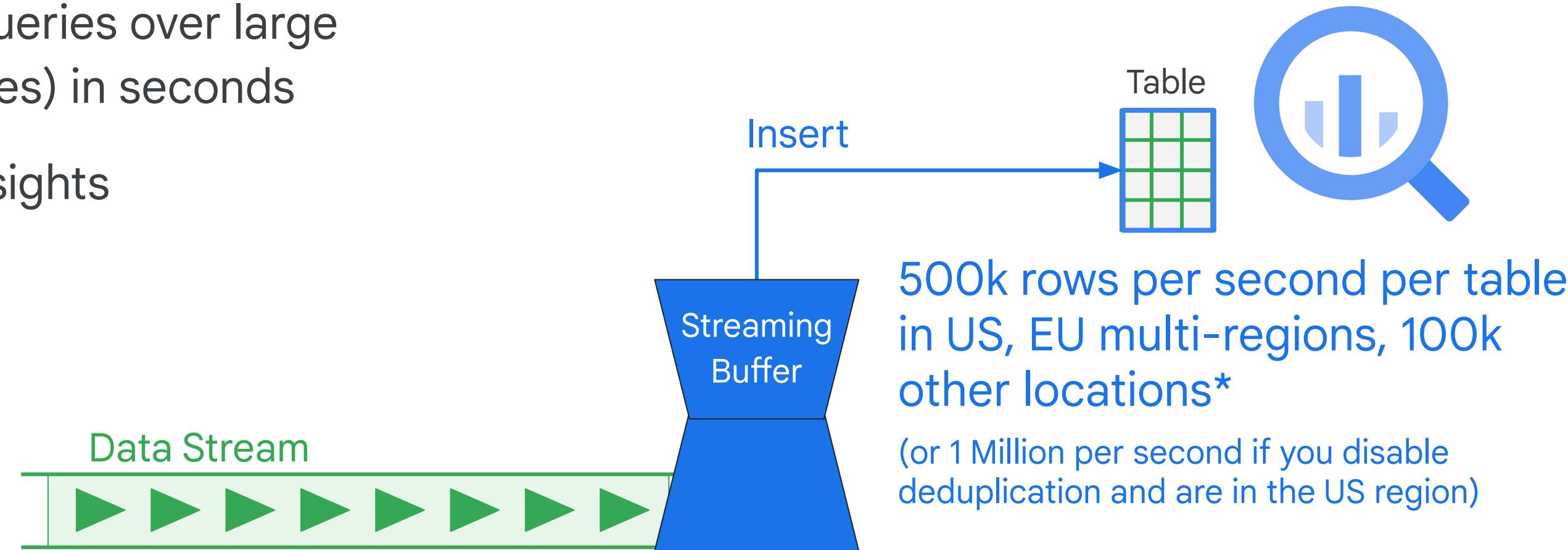
03

Optimizing Cloud Bigtable performance



BigQuery allows you to stream records into a table; query results incorporate latest data

- Interactive SQL Queries over large datasets (petabytes) in seconds
- Near-real-time insights



Note:

Unlike load jobs, there is a cost for streaming inserts (see [quota and limits](#))

Insert streaming data into a BigQuery table

```
export GOOGLE_APPLICATION_CREDENTIALS="/home/user/Downloads/[FILE_NAME].json"
```

Credentials

```
pip install google-cloud-bigquery
```

Install API

The service must have
Cloud IAM permissions
set in the Web UI

```
from google.cloud import bigquery
client = bigquery.Client(project='PROJECT_ID')

dataset_ref = bigquery_client.dataset('my_dataset_id')
table_ref = dataset_ref.table('my_table_id')
table = bigquery_client.get_table(table_ref) ----- Get table access from API

# read data from Cloud Pub/Sub and place into row format
# static example customer orders in units:
rows_to_insert = [
    (u'customer 1', 5),
    (u'customer 2', 17),
]
----- Insert rows into table
errors = bigquery_client.insert_rows(table, rows_to_insert)
```

Python

Create a client

Access dataset
and table

Perform insert

Review streaming data in BigQuery

Query editor

```
1 select * from cloud-training-demos.demos.current_conditions;
```

Run Save query Save view Schedule query More

Query results [SAVE RESULTS](#) [EXPLORE DATA](#)

Query complete (1.2 sec elapsed, 14.3 MB processed)

Job information Results JSON Execution details

Row	timestamp	latitude	longitude	highway	direction	lane	speed	sensorid
1	2008-11-01 11:55:00 UTC	33.191415	-117.363042	5	N	4	10.5	33.191415,-117.363042,5,N,4
2	2008-11-01 11:55:00 UTC	33.191415	-117.363042	5	N	4	10.5	33.191415,-117.363042,5,N,4
3	2008-11-01 09:35:00 UTC	33.191415	-117.363042	5	N	4	11.0	33.191415,-117.363042,5,N,4
4	2008-11-01 12:30:00 UTC	33.191415	-117.363042	5	N	4	11.0	33.191415,-117.363042,5,N,4
5	2008-11-01 09:40:00 UTC	33.191415	-117.363042	5	N	4	11.0	33.191415,-117.363042,5,N,4
6	2008-11-01 10:55:00 UTC	33.191415	-117.363042	5	N	4	11.0	33.191415,-117.363042,5,N,4

Want to visualize insights? Explore Google Data Studio insights right from within BigQuery

Query editor

```
1 select * from cloud-training-demos.demos.current_conditions;
```

Run Save query Save view Schedule query More

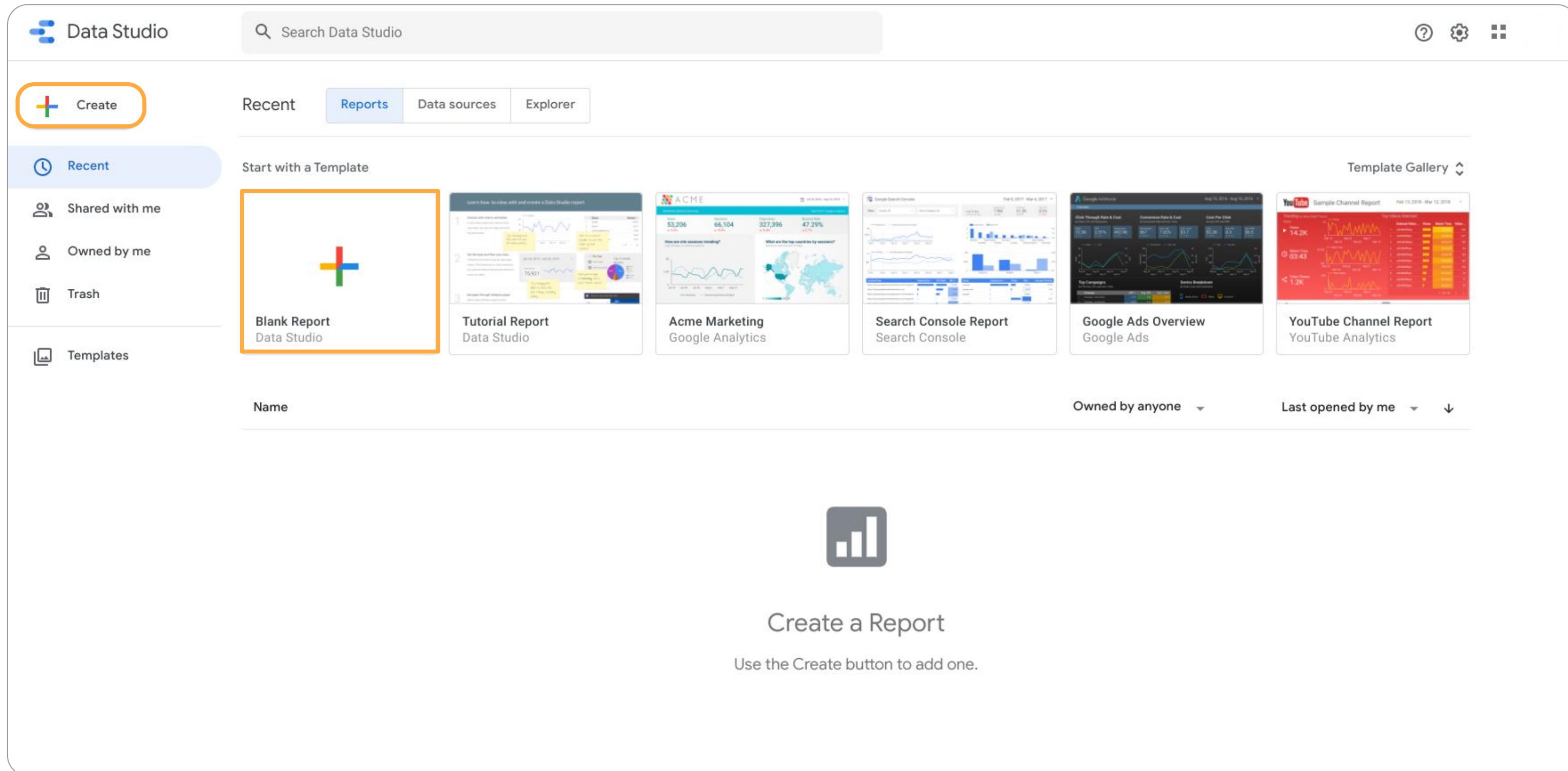
Query results [SAVE RESULTS](#) [EXPLORE DATA](#)

Query complete (1.2 sec elapsed, 14.3 MB processed)

Job information Results JSON Execution details

Row	timestamp	latitude	longitude	highway	direction	lane	speed	sensorid
1	2008-11-01 11:55:00 UTC	33.191415	-117.363042	5	N	4	10.5	33.191415,-117.363042,5,N,4
2	2008-11-01 11:55:00 UTC	33.191415	-117.363042	5	N	4	10.5	33.191415,-117.363042,5,N,4
3	2008-11-01 09:35:00 UTC	33.191415	-117.363042	5	N	4	11.0	33.191415,-117.363042,5,N,4
4	2008-11-01 12:30:00 UTC	33.191415	-117.363042	5	N	4	11.0	33.191415,-117.363042,5,N,4
5	2008-11-01 09:40:00 UTC	33.191415	-117.363042	5	N	4	11.0	33.191415,-117.363042,5,N,4
6	2008-11-01 10:55:00 UTC	33.191415	-117.363042	5	N	4	11.0	33.191415,-117.363042,5,N,4

Create new reports in the Data Studio UI



The screenshot shows the Google Data Studio interface. At the top left is the "Data Studio" logo. To its right is a search bar with the placeholder "Search Data Studio". On the far right of the header are three icons: a question mark, a gear, and a grid. Below the header, there are four tabs: "Recent" (grayed out), "Reports" (blue, indicating it's selected), "Data sources" (grayed out), and "Explorer" (grayed out). A large orange oval highlights the "Create" button, which has a plus sign icon and the word "Create". Below the tabs, there are two sections: "Recent" (highlighted with a blue oval) and "Start with a Template". The "Recent" section includes links for "Shared with me", "Owned by me", and "Trash". The "Templates" section includes a link for "Blank Report Data Studio" (highlighted with an orange oval). The "Start with a Template" section displays six template cards: "Blank Report Data Studio", "Tutorial Report Data Studio", "Acme Marketing Google Analytics", "Search Console Report Search Console", "Google Ads Overview Google Ads", and "YouTube Channel Report YouTube Analytics". To the right of the templates is a "Template Gallery" dropdown menu. At the bottom of the interface, there are filters for "Name", "Owned by anyone", and "Last opened by me". In the center, there's a large "Create a Report" button with a bar chart icon and the text "Create a Report" and "Use the Create button to add one."

Connect to multiple different types of data sources

The screenshot shows the Google Data Studio interface. At the top, there's a toolbar with options like File, View, Page, Help, and various buttons for adding pages, data, charts, controls, and themes. Below the toolbar is a large workspace area with a grid pattern. A modal window titled "Add data to report" is open in the center. Inside the modal, there are two tabs: "Connect to data" (which is selected) and "My data sources". Below the tabs is a search bar with a magnifying glass icon and the placeholder text "Search". The main content area is titled "Google Connectors (22)" and includes a sub-note: "Connectors built and supported by Data Studio [Learn more](#)". The connectors are displayed in a grid format:

- Google Analytics** By Google. Connect to Google Analytics.
- Google Ads** By Google. Connect to Google Ads performance report data.
- Google Sheets** By Google. Connect to Google Sheets.
- BigQuery** By Google. Connect to BigQuery tables and custom queries.
- File Upload** By Google. Connect to CSV (comma-separated values) files.
- Campaign Manager 360** By Google. Connect to Campaign Manager 360 data.
- Cloud Spanner** By Google. Connect to Google Cloud Spanner databases.
- Cloud SQL for MySQL** By Google. Connect to Google Cloud SQL for MySQL databases.
- Display & Video 360** By Google.
- Extract Data** By Google.
- Google Ad Manager 360** By Google.
- Google Cloud Storage** By Google.

Add the data source to your report

The screenshot shows the Looker interface with the following details:

- Top Bar:** Untitled Report, File, View, Page, Help, Reset, Share, View, More.
- Toolbar:** Add page, Add data, Add a chart, Add a control, Theme and layout.
- Left Sidebar:** Add data to report, Data credentials: Steve Leonard.
- Google Sheets Connector:** By Google, The Google Sheets connector allows you to access data stored in a Google Sheets worksheet. Options: LEARN MORE, REPORT AN ISSUE.
- Table View:** ALL ITEMS, OWNED BY ME, SHARED WITH ME, STARRED, URL, OPEN FROM GOOGLE DRIVE. A specific item is selected: Natural_disasters_climate_change - Sheet1.
- Modal Dialog:** You are about to add data to this report.
 - Selected item: Natural_disasters_climate_change - Sheet1.
 - Note: Note that Report Editors can create charts using the new data source(s), and can add dimensions and metrics not currently included in the report.
 - Checkboxes:
 - Don't show me this again
 - Buttons: CANCEL, ADD TO REPORT (highlighted with a yellow border).
 - Text: Optional Range, e.g. A1:B52.
- Bottom Buttons:** Cancel, Add.

Select your data fields to build your visualizations

Untitled Report

File Edit View Insert Page Arrange Resource Help

Reset Share View

Add page Add data Add a chart Add a control Theme and layout

The screenshot shows a data visualization interface with a table of natural disaster data and a sidebar for selecting fields.

Table Data:

	Year	Earthquake	Epidemic	Storm	Wildfire	Volcanic ...	Insect inf...	Extreme t...	Landslide	Mass mo...	Flood	Drought
1.	2018	20	15	94	10	7	1	26	13	1	127	14
2.	2017	22	27	130	15	2	1	10	25	1	126	9
3.	2016	30	25	86	10	null	1	12	13	null	159	14
4.	2015	23	16	121	12	6	1	12	20	1	162	28
5.	2014	26	21	99	4	6	1	17	15	null	135	18
6.	2013	29	23	105	10	3	1	14	11	1	149	9
7.	2012	27	25	91	6	1	1	51	13	1	136	21
8.	2011	30	27	84	8	6	1	16	17	null	156	17

1 - 30 / 30 < >

Available Fields:

- 01 Earthquake
- 01 Epidemic
- 01 Storm
- 01 Wildfire
- 01 Volcanic activity
- 01 Insect infestation
- 01 Extreme temperature
- 01 Landslide
- 01 Mass movement (dry)
- 01 Flood
- 01 Drought

Dimensions:

- 02 Year

Metrics:

- 03 SUM Earthquake
- 03 SUM Epidemic
- 03 SUM Storm
- 03 SUM Wildfire
- 03 SUM Volcanic activity
- 03 CTD Insect infestation
- 03 SUM Extreme temperat...
- 03 SUM Landslide
- 03 SUM Mass movement (...)
- 03 SUM Flood
- 03 SUM Drought

Chart > Table

DATA

Data source: Natural_disasters...

Date Range Dimension: Add dimension

Dimension: Year

Drill down: Off

STYLE

Available Fields: Type to search

01 Drought

01 Earthquake

01 Epidemic

01 Extreme temperature

01 Flood

01 Insect infestation

01 Landslide

01 Mass movement (dry)

01 Storm

01 Volcanic activity

01 Wildfire

01 Year

01 Record Count

ADD A FIELD

ADD A PARAMETER

Edit your data source fields, if necessary

Screenshot of the Data Studio interface showing a report titled "Untitled Report". The report displays a table of natural disaster data from 2012 to 2018. A large orange arrow points from the "Data source" field in the right sidebar to the "Year" field in the table's header.

Report Header:

- File Edit View Insert Page Arrange Resource Help
- Reset Share View

Table Data:

	Year	Earthquake	Epidemic	Storm	Wildfire	Volcanic ...	Insect inf...	Extreme t...	Landslide	Mass mo...	Flood	Drought
1.	2018	20	15	94	10	7	0	26	13	1	127	14
2.	2017	22	27	130	15	2	0	10	25	1	126	9
3.	2016	30	25	86	10	null	0	12	13	null	159	14
4.	2015	23	16	121	12	6	0	12	20	1	162	28
5.	2014	26	21	99	4	6	0	17	15	null	135	18
6.	2013	29	23	105	10	3	0	14	11	1	149	9
7.	2012	27	25	91	6	1	0	51	13	1	136	21

Right Sidebar (Chart > Table):

- DATA** (selected)
- STYLE**
- Data source**: Natural_disasters...
- Available Fields** (list: Drought, Earthquake, Epidemic)
- BLEND DATA**
- Date Range Dimension**
- Dimension**
- ADD A FIELD**
- ADD A PARAMETER**

Bottom Navigation:

- Natural_disasters_climate_change - She...
- Data credentials: Steve Leonard
- Data freshness: 15 minutes
- Community visualizations access: On
- DONE

Fields Table:

Field	Type	Default Aggregation	Description	Search fields
Insect infestation	123 Number	Sum		
Landslide	123 Number	Sum		
Mass movement (dry)	123 Number	Sum		
Storm	123 Number	Sum		
Volcanic activity	123 Number	Sum		
Wildfire	123 Number	Sum		
Year	Year (YYYY)	None		

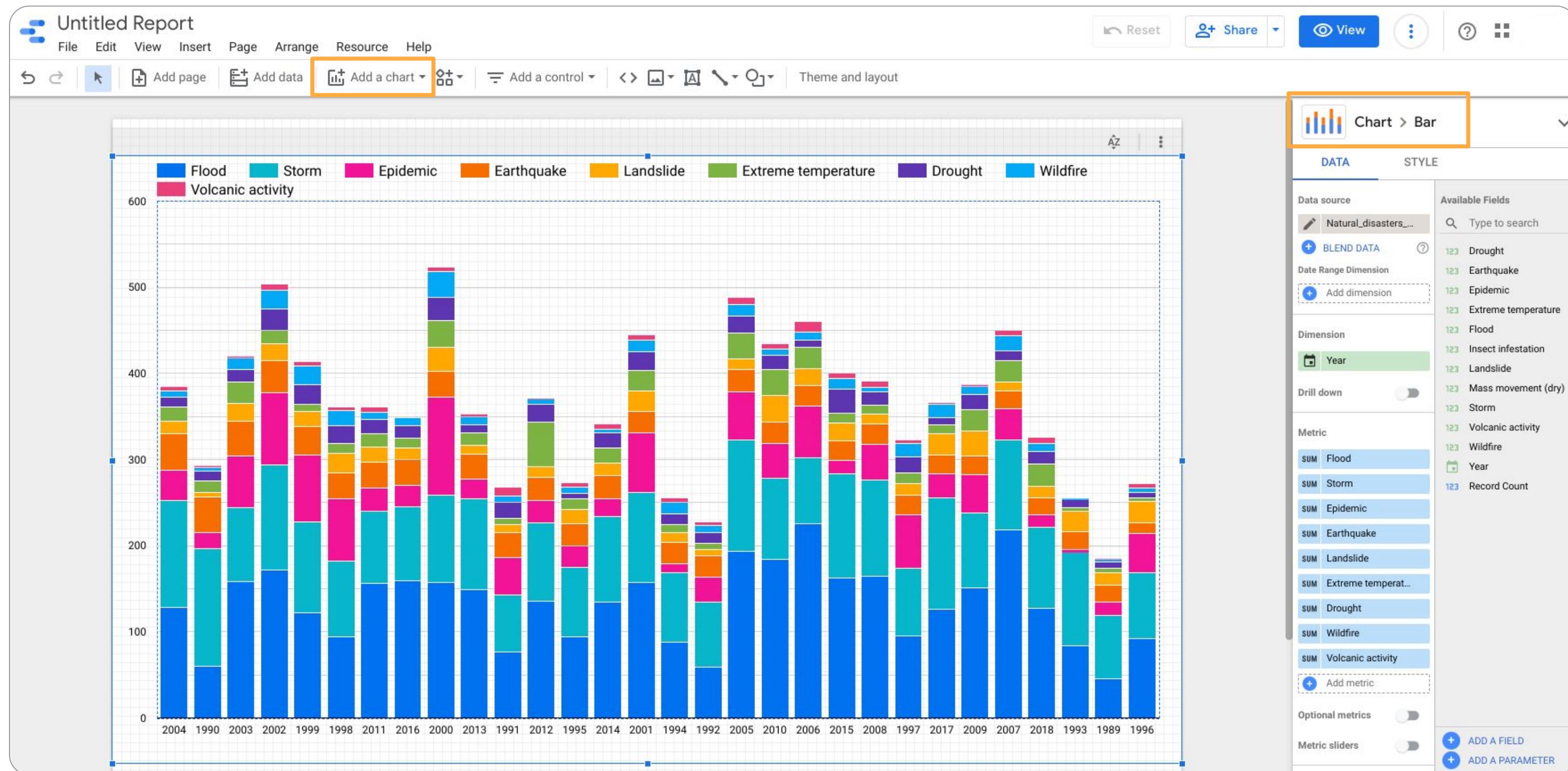
METRICS (1):

Record Count	123 Number	Auto
--------------	------------	------

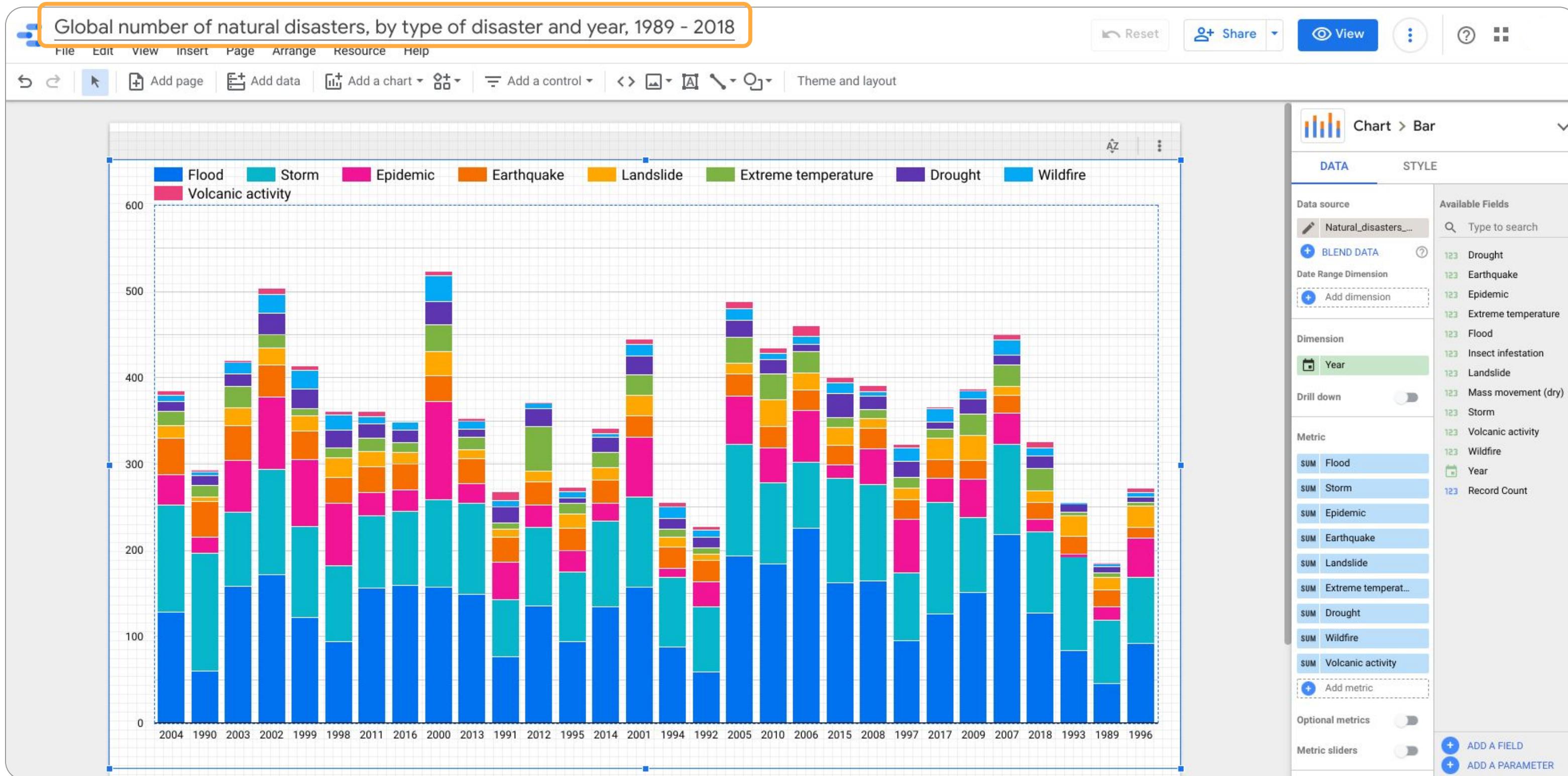
Bottom Buttons:

- REFRESH FIELDS
- 13 / 13 Fields

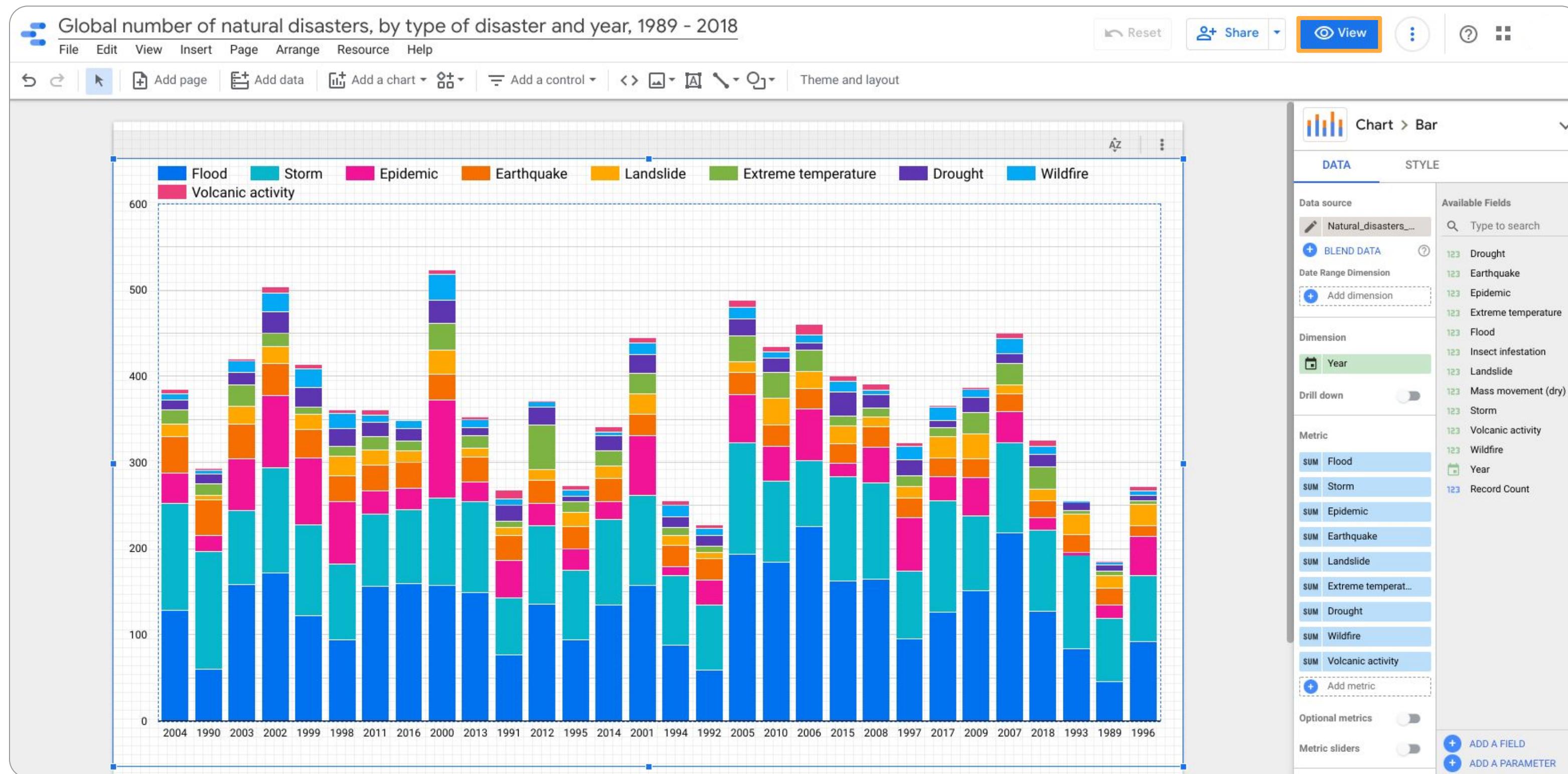
Create charts to visualize data relationships



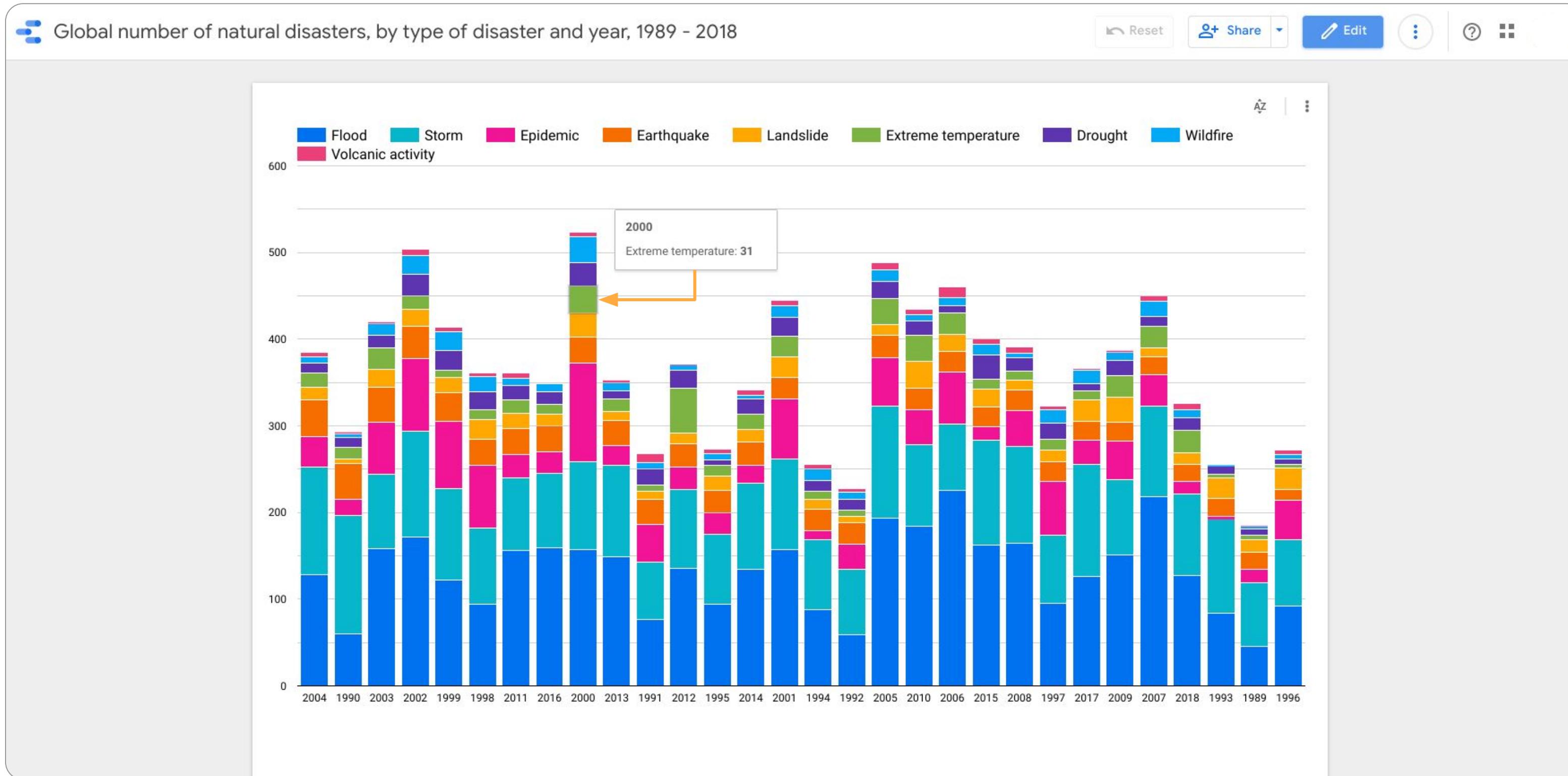
Add a descriptive name to your report



View the end-user version of the report



View your report as an end-user



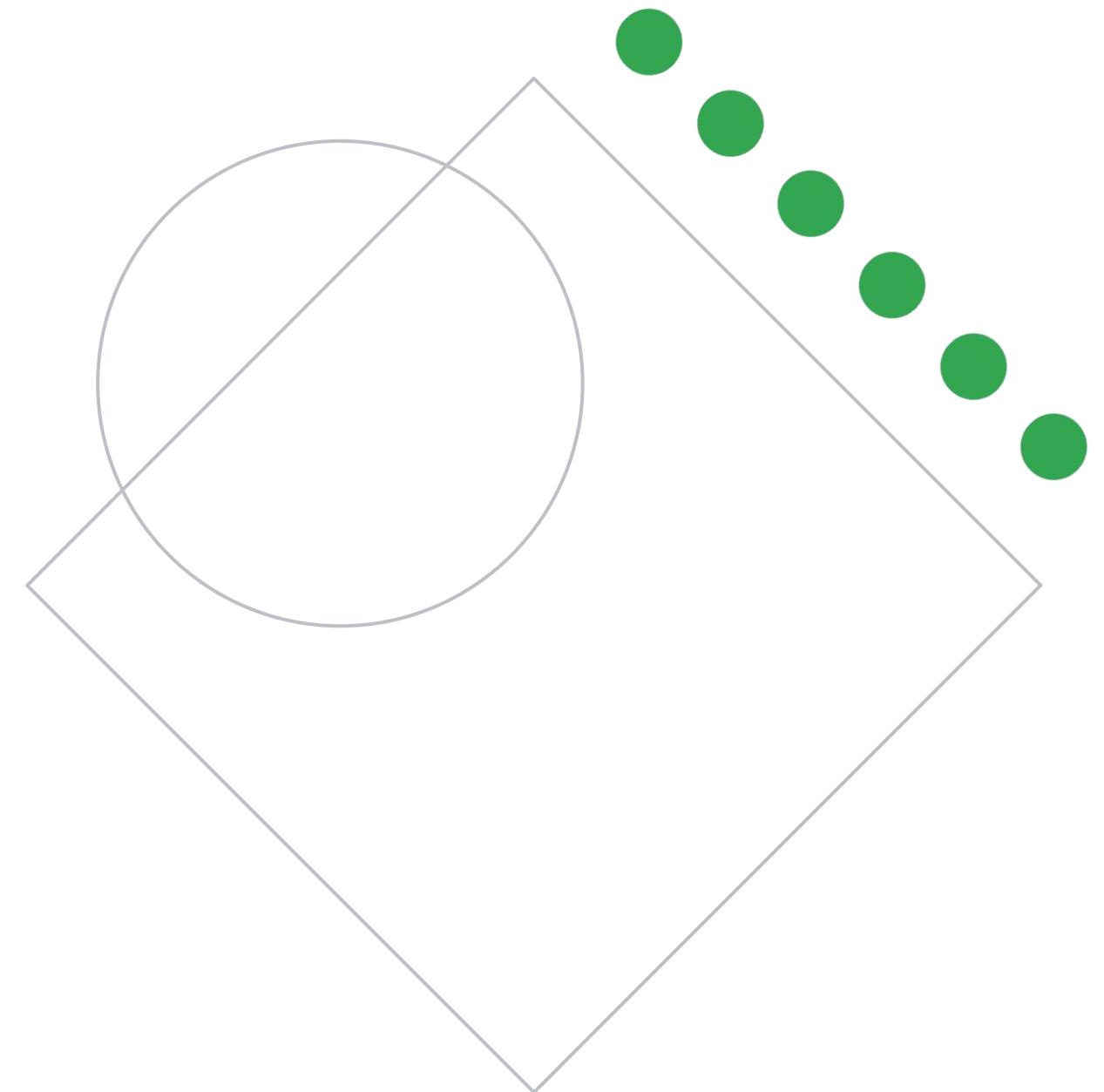
Add value: BI Engine for dashboard performance

- No need to manage OLAP cubes or separate BI servers for dashboard performance.
- Natively integrates with BigQuery streaming for real-time data refresh.
- Column oriented in-memory BI execution engine.



Lab Intro

Streaming Data Processing:
Streaming Analytics and Dashboards



Lab objectives

01

Connect to BigQuery data source
from Google Data Studio

02

Create reports and charts to
visualize the BigQuery data



High-Throughput BigQuery and Bigtable Streaming Features

01

Streaming into BigQuery and visualizing results

02

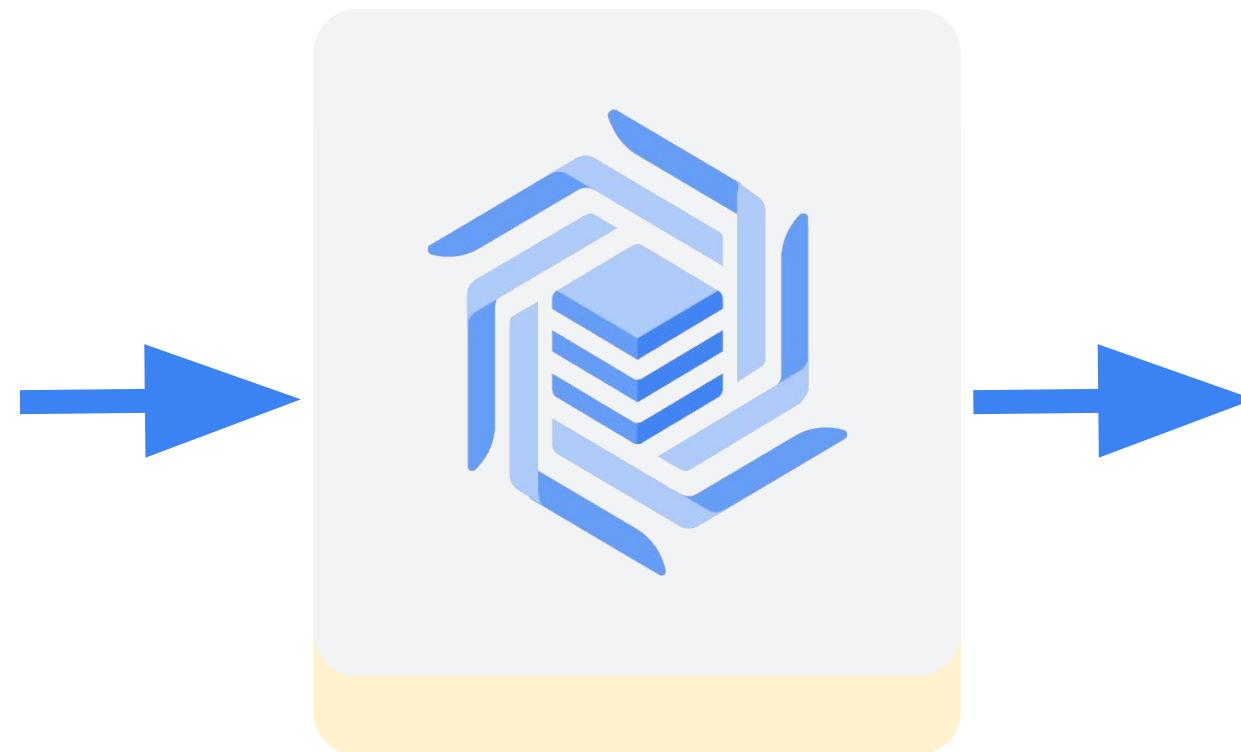
High-throughput streaming with Cloud Bigtable

03

Optimizing Cloud Bigtable performance



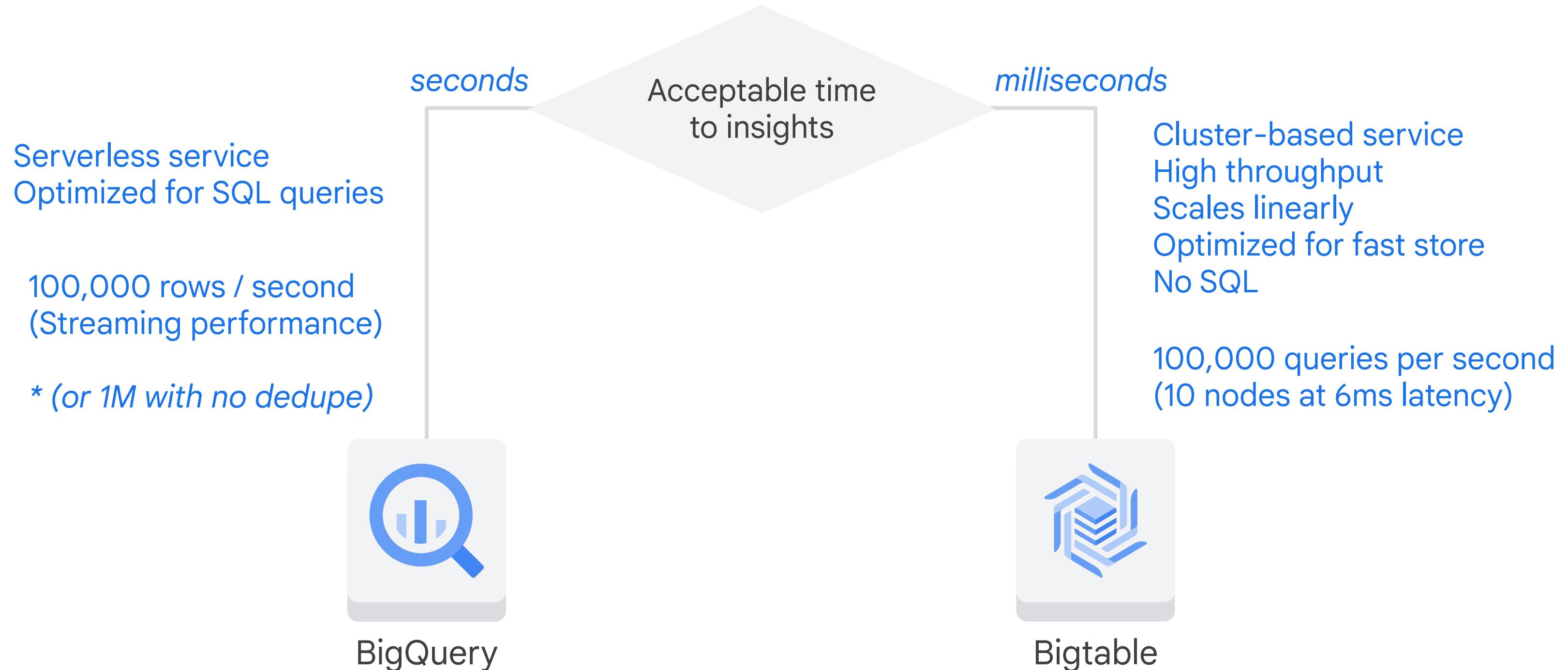
Cloud Bigtable



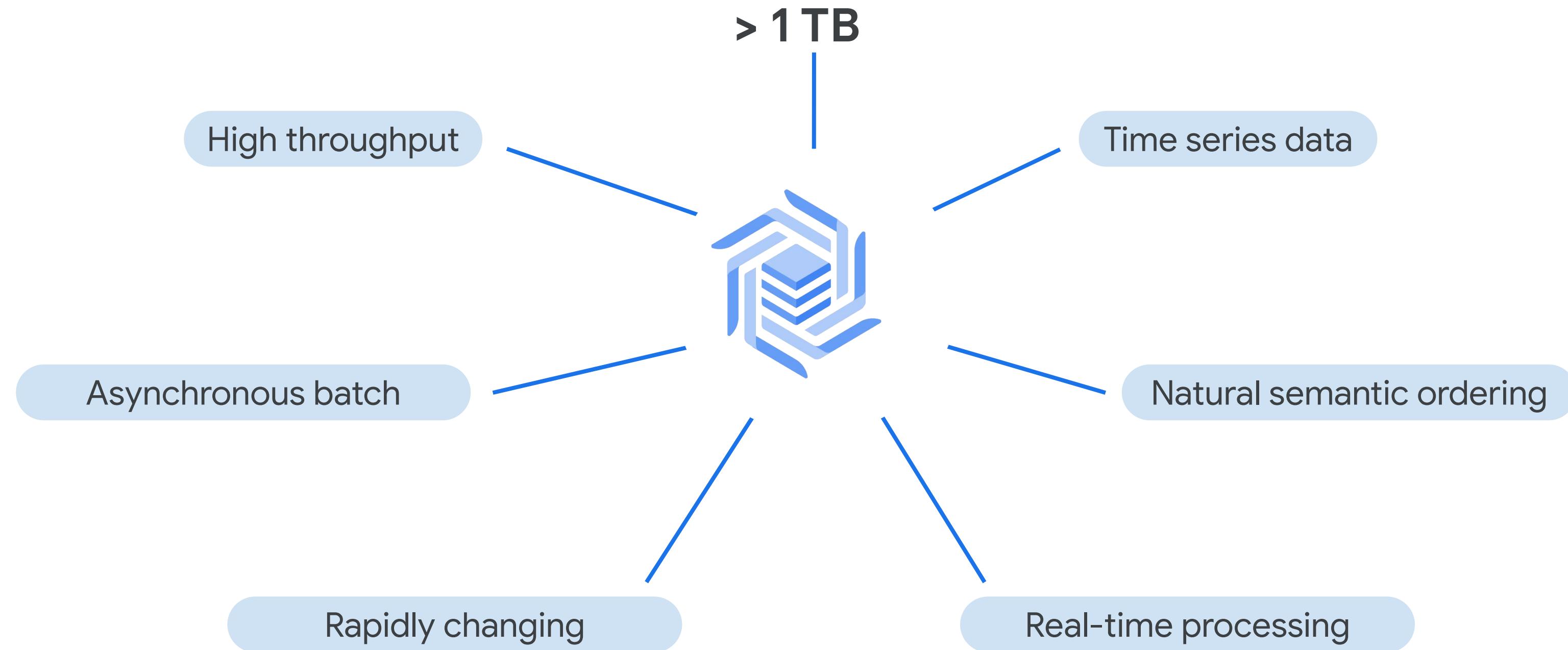
Qualities that Bigtable contributes to data engineering solutions:

- NoSQL Queries over large datasets (petabytes) in milliseconds
- Very fast for specific cases

How to choose between Bigtable and BigQuery



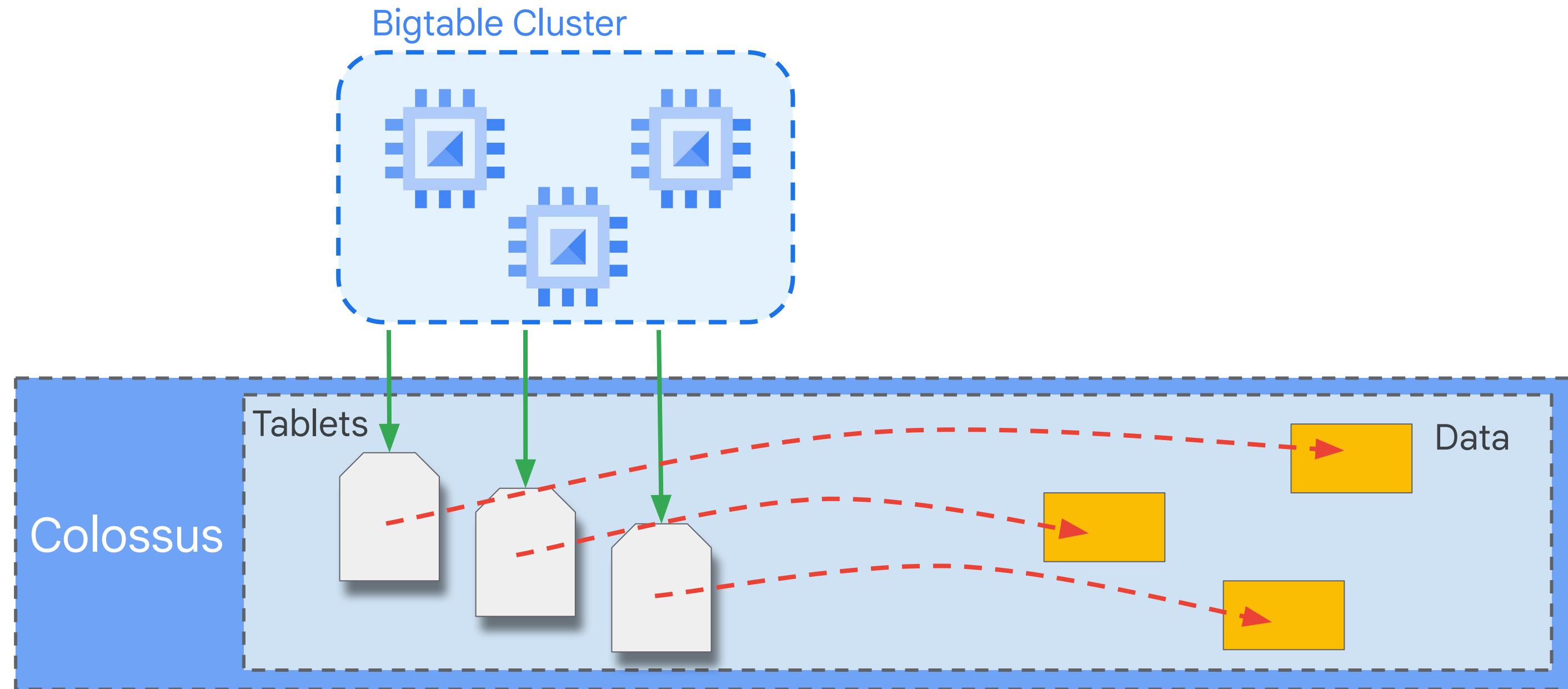
Consider Bigtable for these requirements



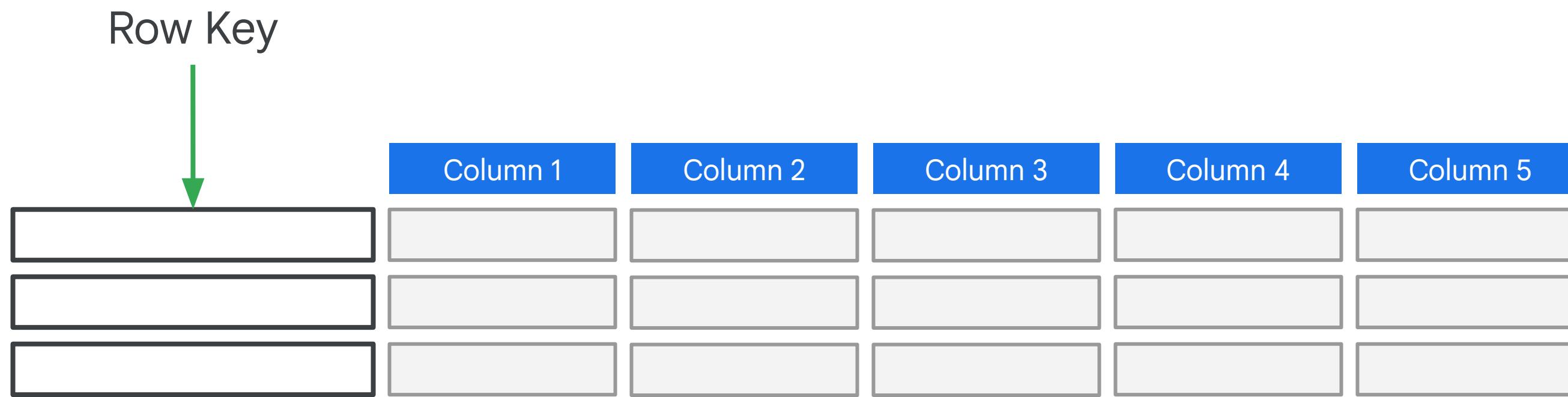
The most common use of Bigtable

Productionize a real-time lookup as part of an application, where speed and efficiency are desired beyond that of other databases.

How does Bigtable work?

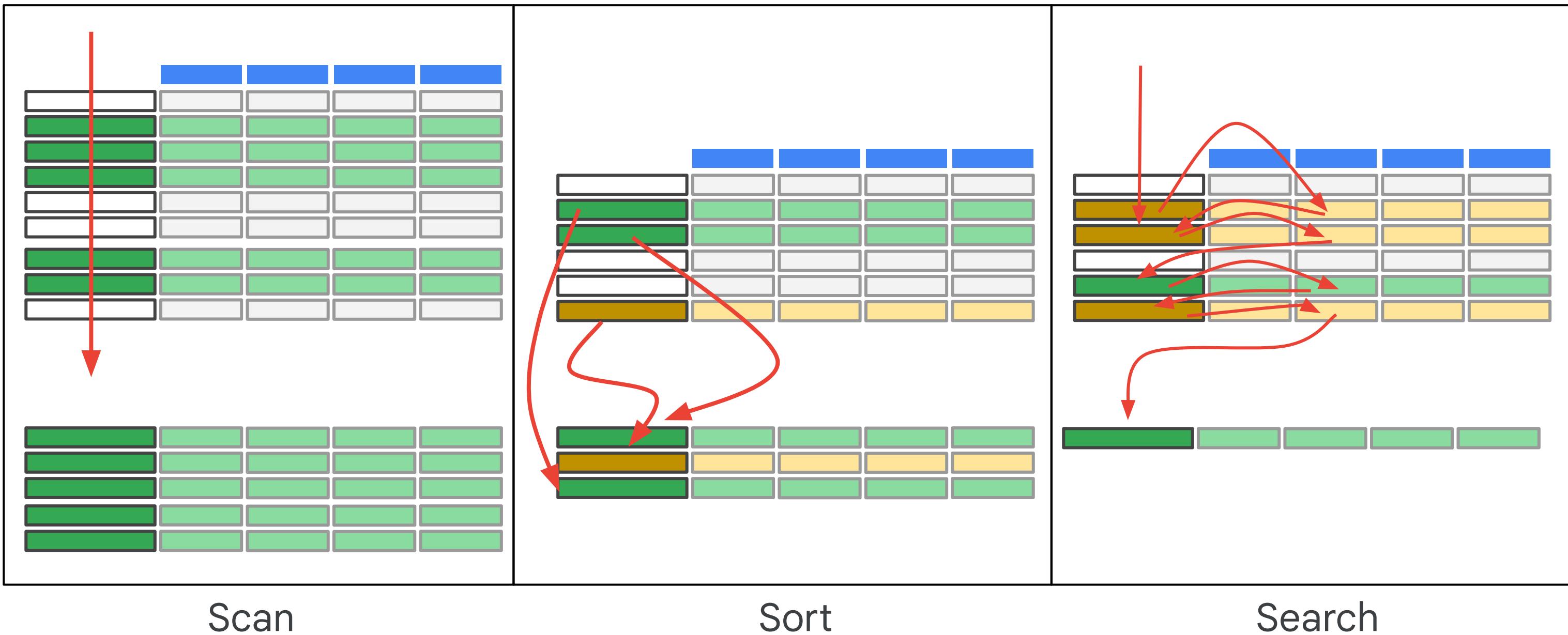


Bigtable design idea is "simplify for speed"

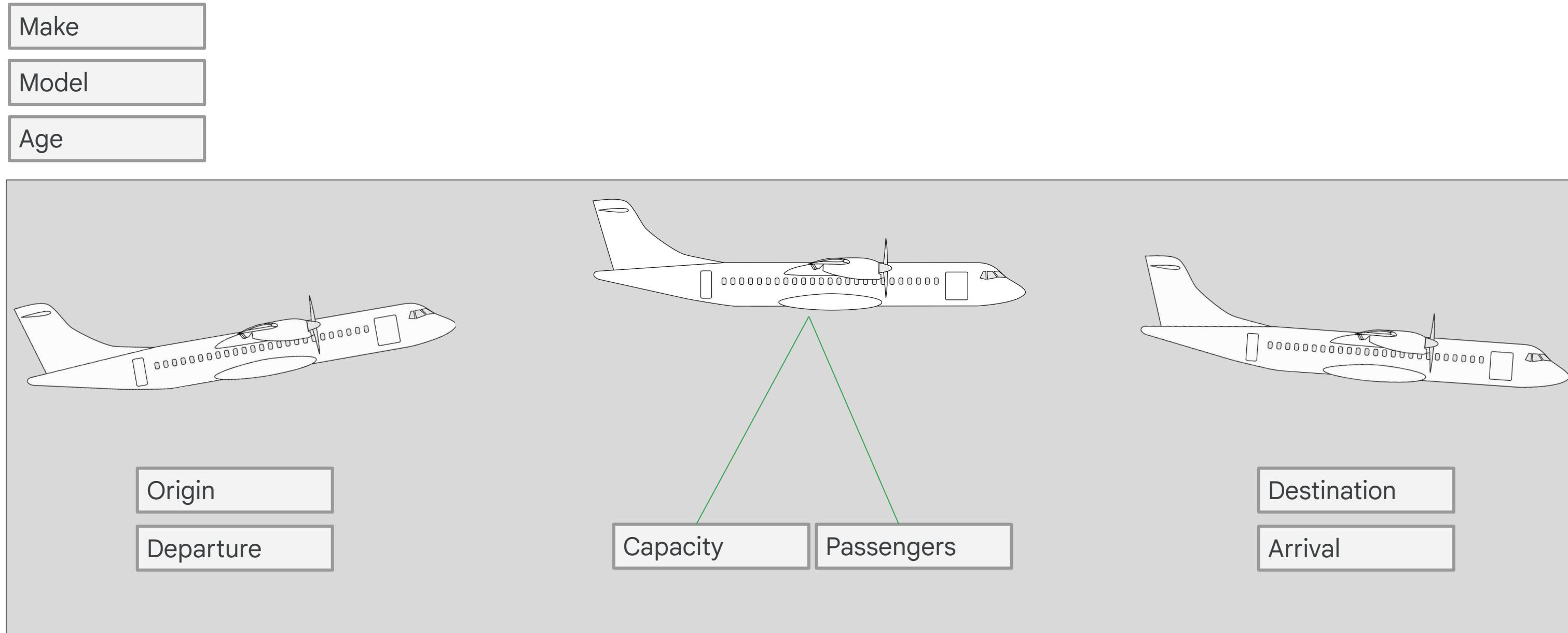


The Row Key is the index.
And you get only one.

But speed depends on your data and Row Key

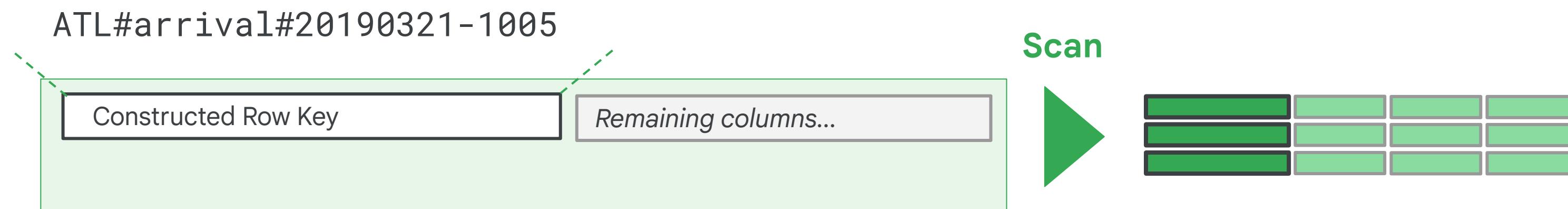
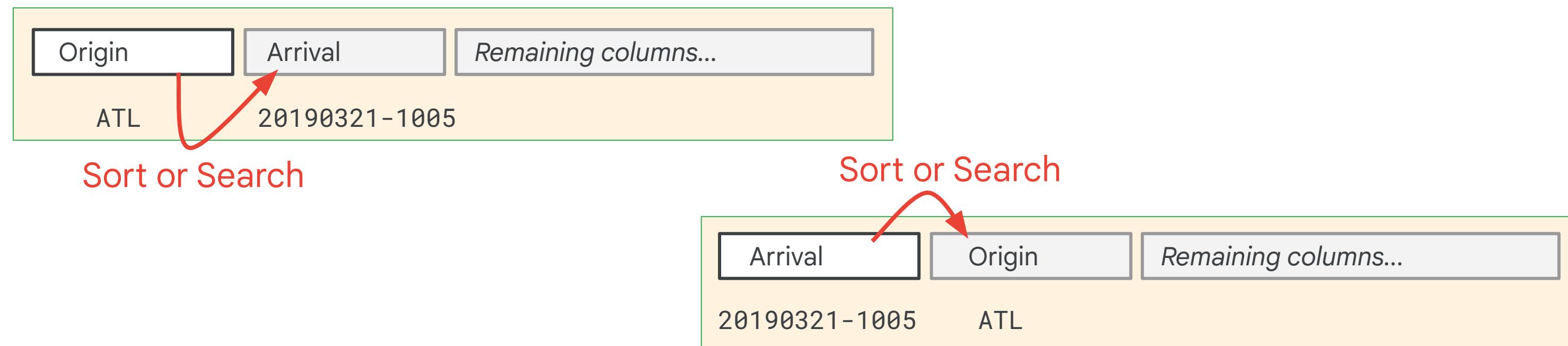


Flights of the world: Reviewing the data



What is the best Row Key?

Query: All flights originating in Atlanta and arriving between March 21st and 29th



Bigtable schema organization



Column Families

Row Key	Flight_Information					Aircraft_Information			
	Origin	Destination	Departure	Arrival	Passengers	Capacity	Make	Model	Age
ATL#arrival#20190321-1121	ATL	LON	20190321-0311	20190321-1121	158	162	B	737	18
ATL#arrival#20190321-1201	ATL	MEX	20190321-0821	20190321-1201	187	189	B	737	8
ATL#arrival#20190321-1716	ATL	YVR	20190321-1014	20190321-1716	201	259	B	757	23

Queries that use the row key, a row prefix, or a row range are the most efficient

Query: Current arrival delay for flights from Atlanta

1

row key based on atlanta arrivals

e.g. ORIGIN#arrival

(ATL#arrival#20190321-1005)

Puts latest flights at bottom
of table

2

reverse timestamp to the rowkey

e.g. ORIGIN#arrival#RTS

(ATL#arrival#560549313)

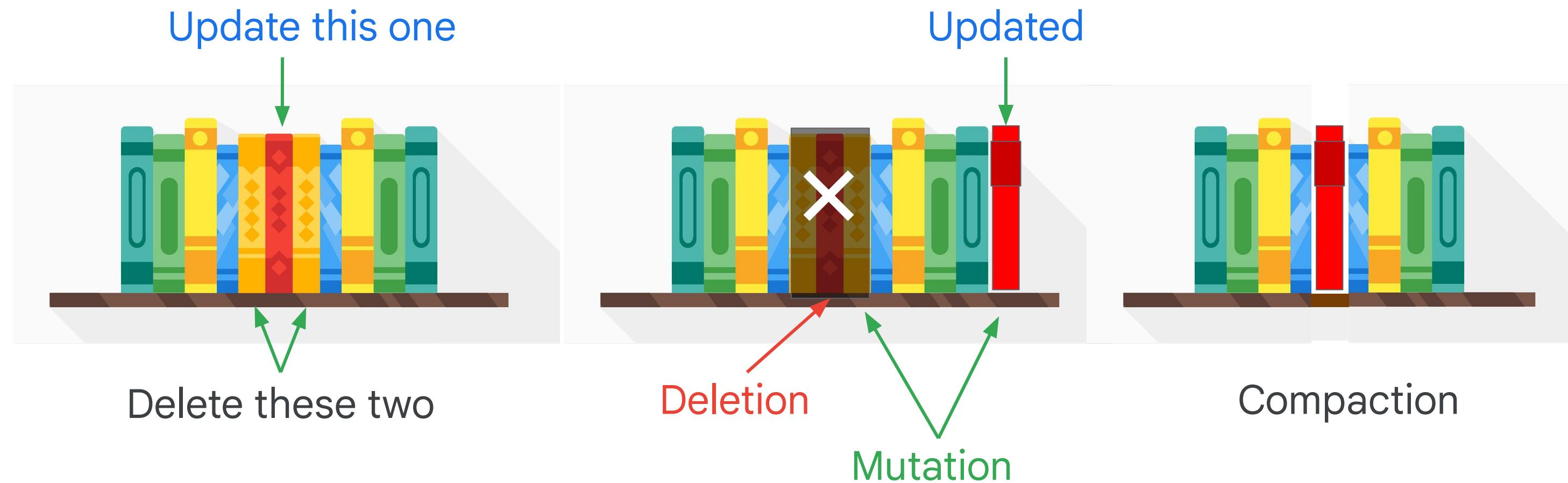
Puts latest flights at top
of table

Use reverse timestamps when your most common query is for the latest values

Query: Current arrival delay for flights from Atlanta

```
// key is ORIGIN#arrival#REVTS
String key = info.getORIGIN() //
+ "#arrival" //
+ "#" + (Long.MAX_VALUE - ts.getMillis()); // reverse timestamp
```

What happens when data in Bigtable is changed?



Optimizing data organization for performance



Group related data for more efficient reads

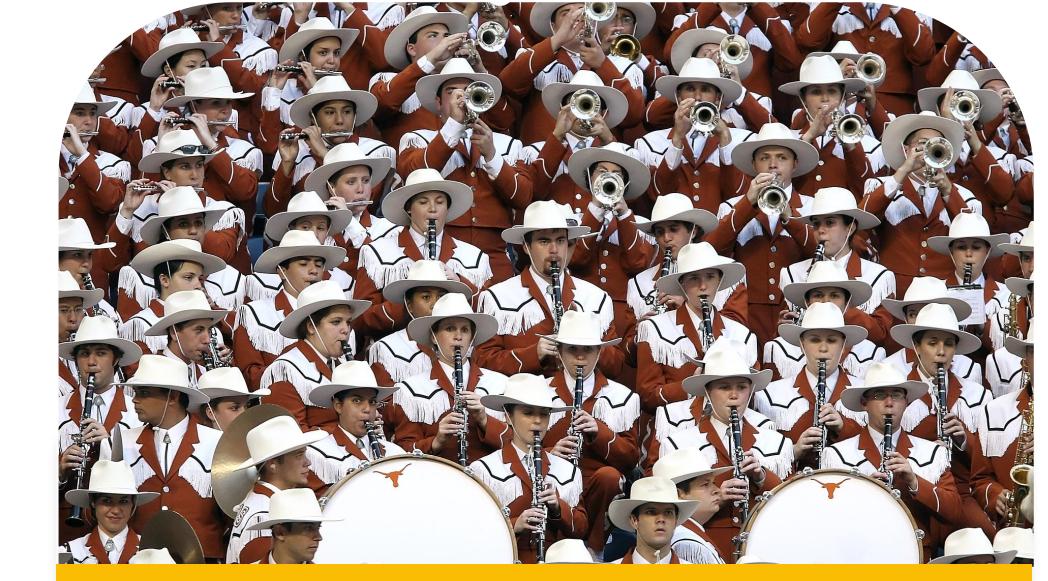
Example row key:

DehliIndia#2019031411841

Use column families



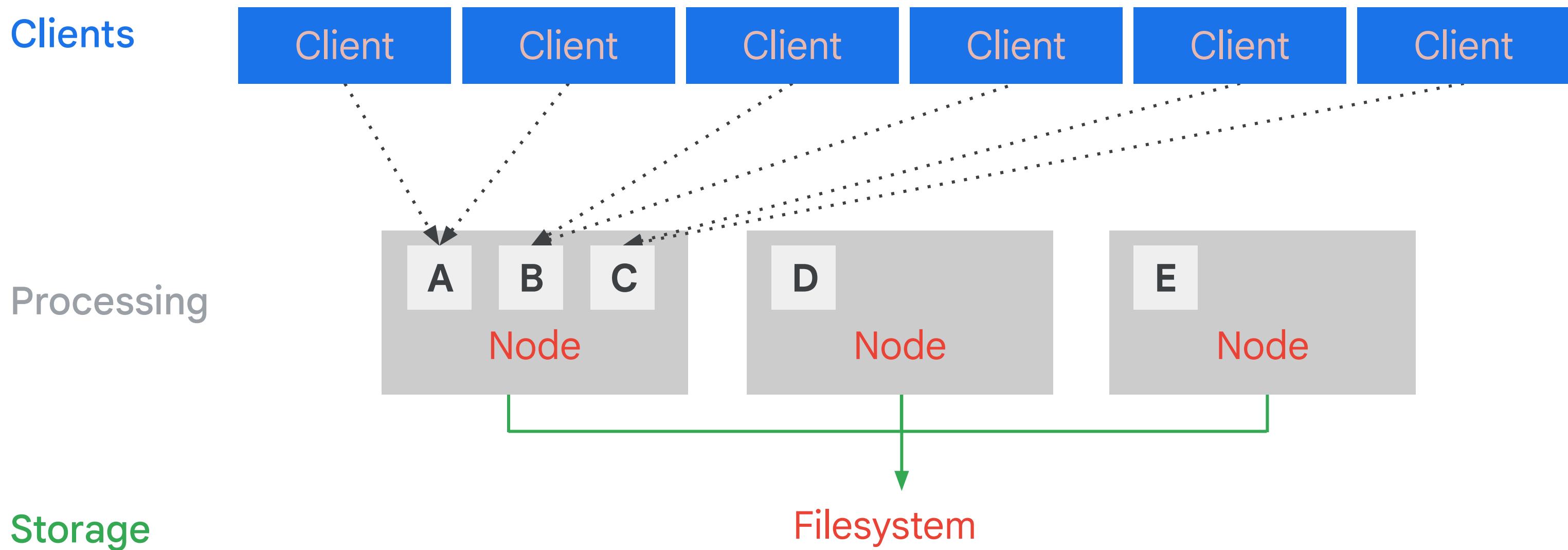
Distribute data evenly for more efficient writes



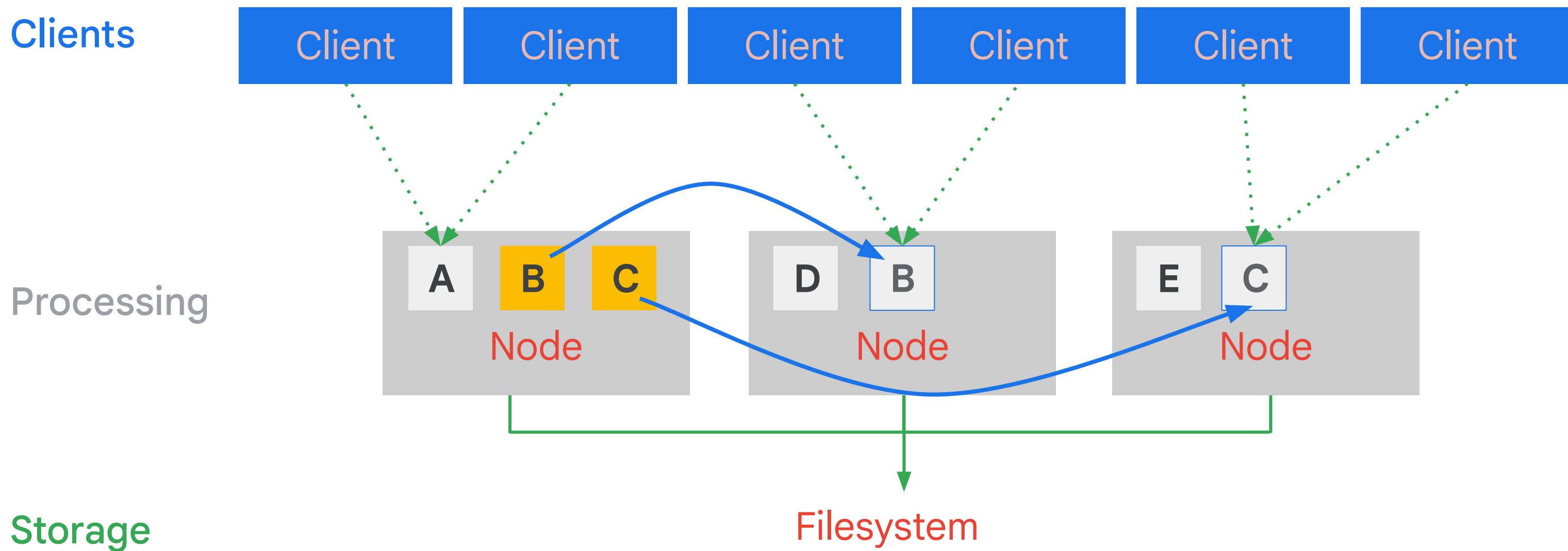
Place identical values in the same row or adjoining rows for more efficient compression

Use row keys to organize identical data

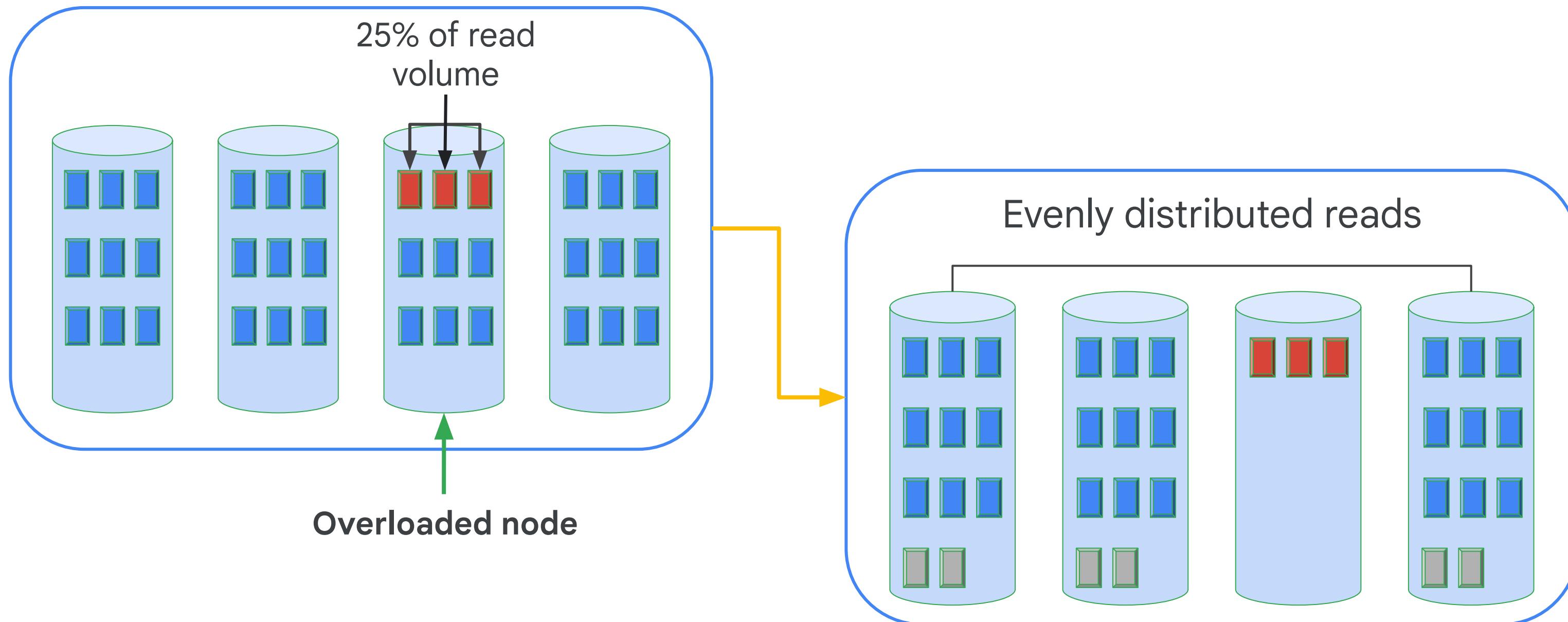
Bigtable self-improves by learning access patterns ...



...and rebalances data accordingly



Rebalance strategy: Distribute reads



Real world use case: Spotify



In 2019, Spotify ran the largest Dataflow job ever at the time with Bigtable "...used as a remediation tool between Dataflow jobs in order for them to process and store more data in a parallel way, rather than the need to always regroup the data"



High-Throughput BigQuery and Bigtable Streaming Features

01

Streaming into BigQuery and visualizing results

02

High-throughput streaming with Cloud Bigtable

03

Optimizing Cloud Bigtable performance



Optimizing Bigtable performance

Tune the schema

Cloud Bigtable learning behavior

Tune the resources

Change schema to minimize data skew

Takes a while after scaling up nodes for performance improvement to be seen

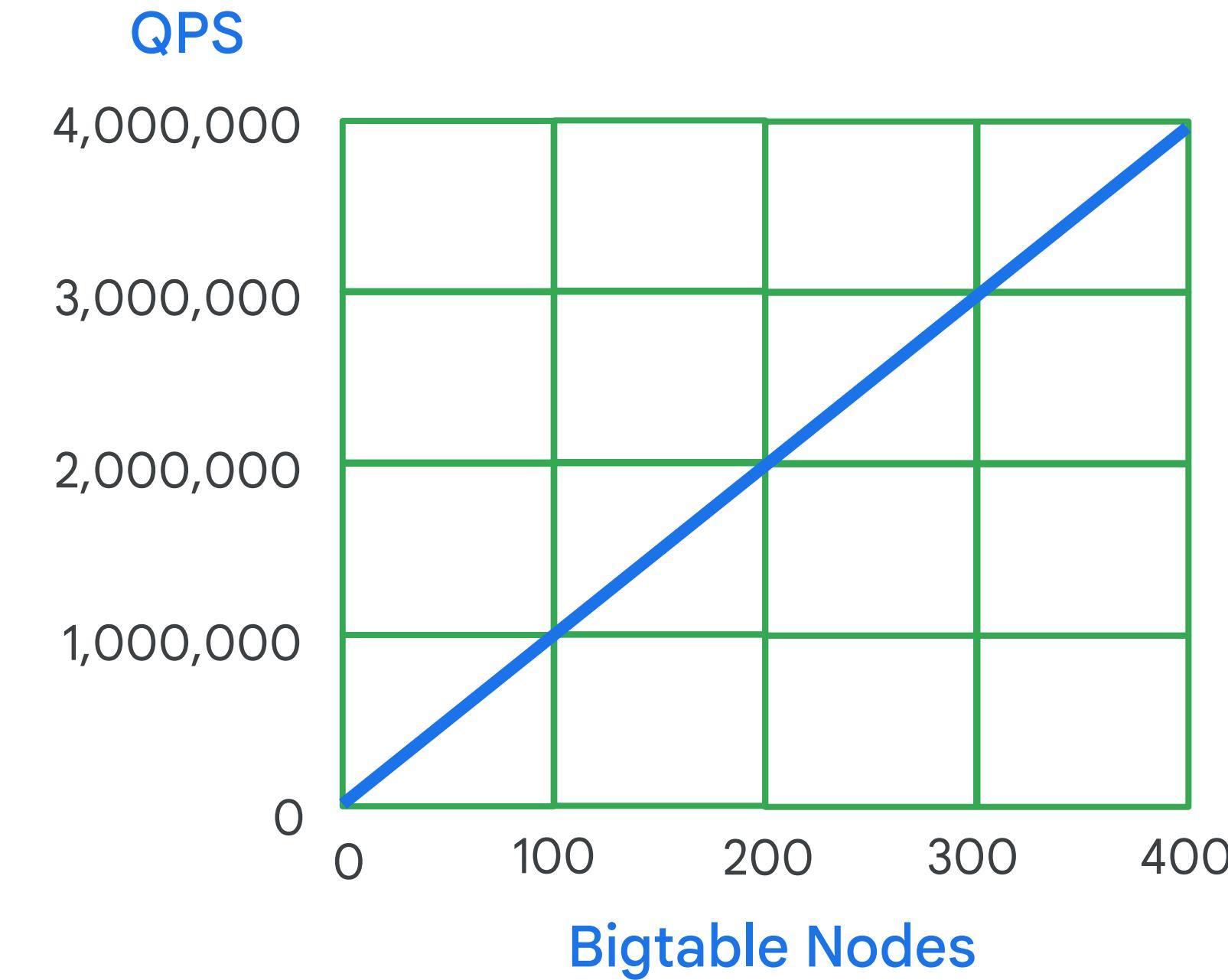
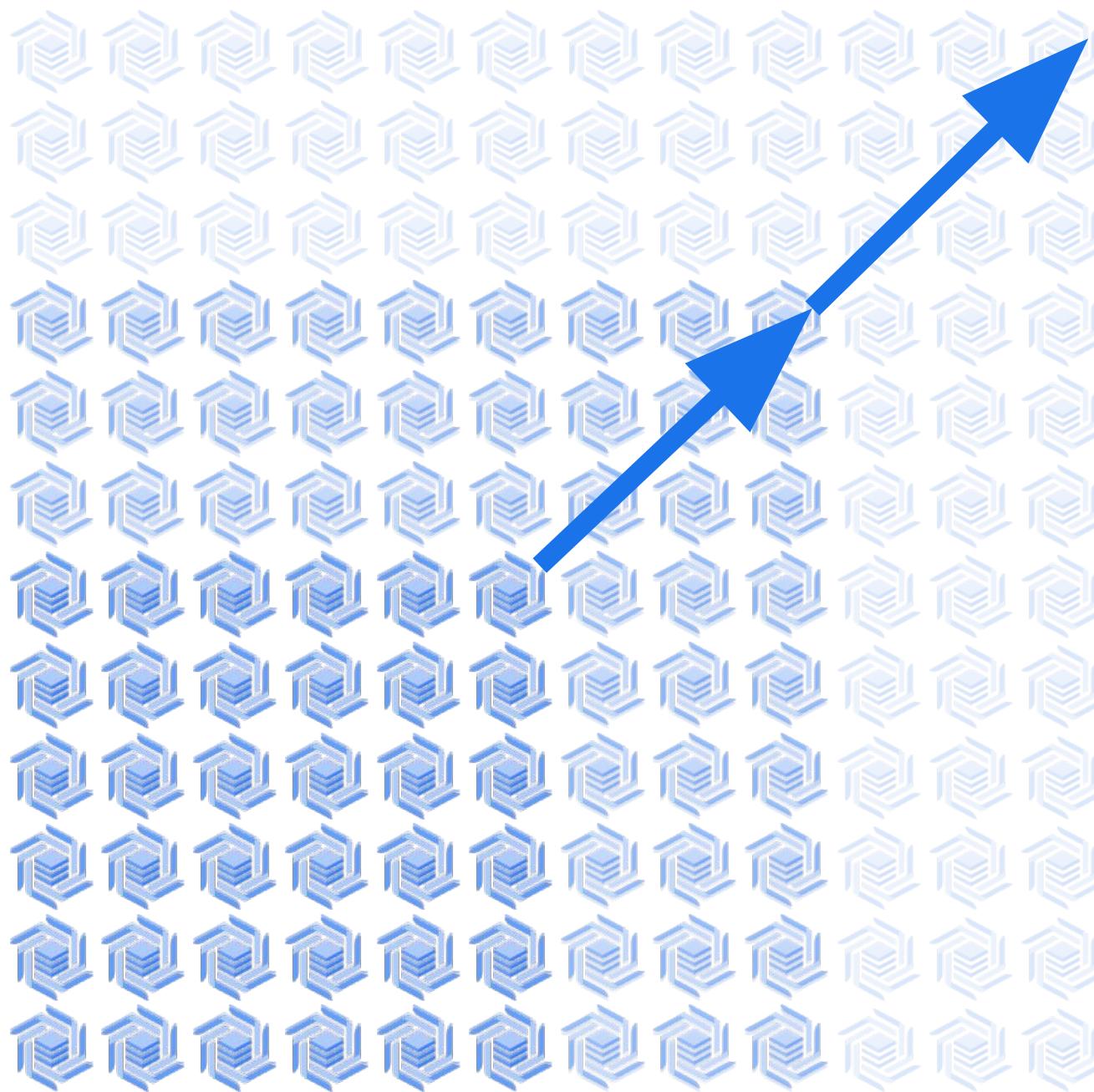
Test with > 300 GB and for **minutes-to-hours** to give time for Bigtable to balance and learn

Make sure clients and Cloud Bigtable are in **same zone**

Disk speed on VMs in the cluster: SSD is faster than HDD

Performance increases linearly with **number of nodes**

Throughput can be controlled by node count

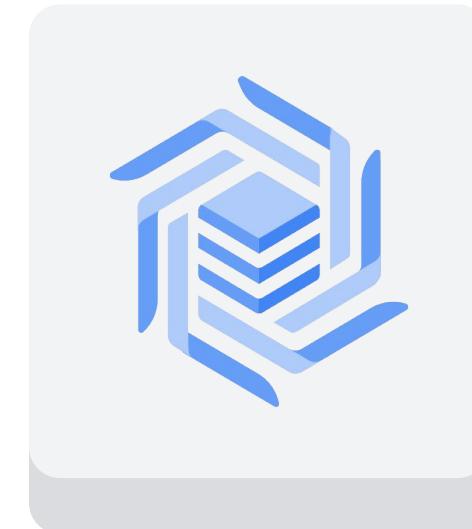


Features for Bigtable streaming

Incoming streaming
data is independent



Writing

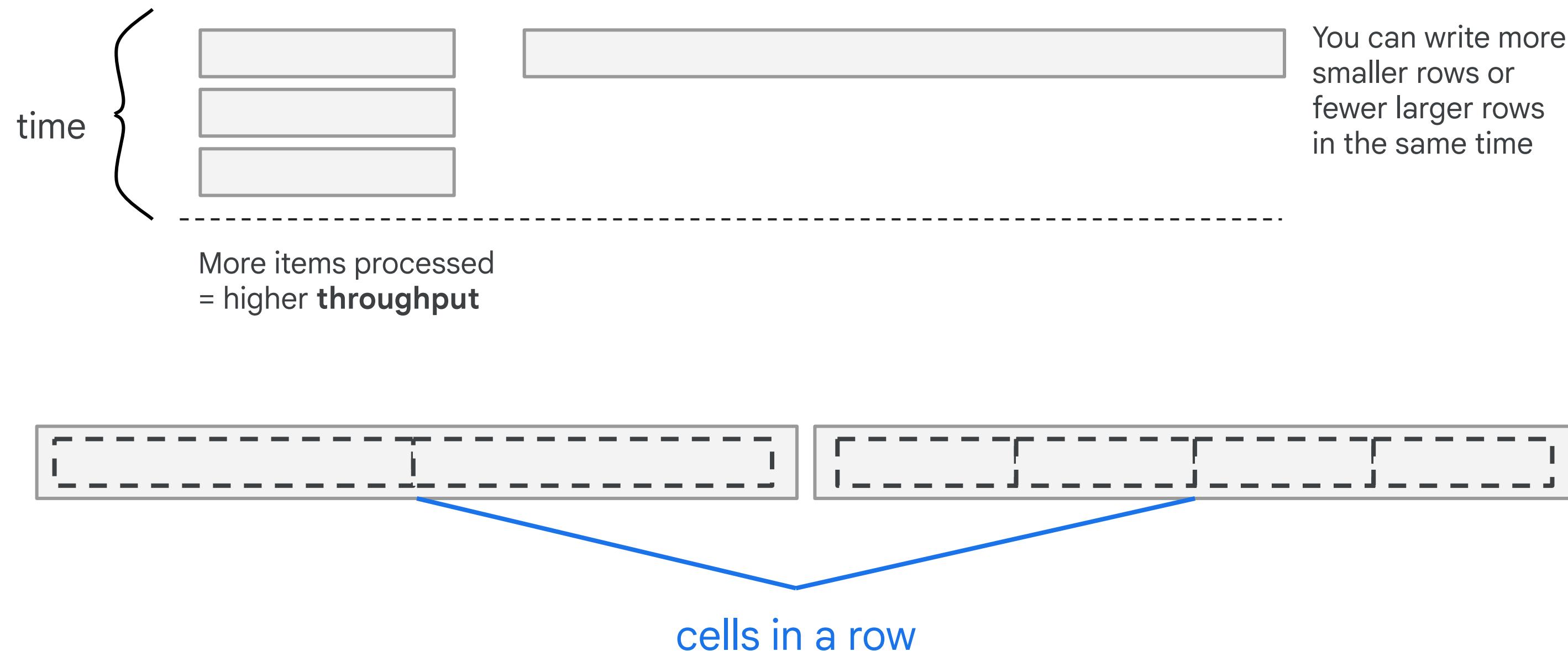


Application reading of
data is controllable



Reading

Schema design is the primary control for streaming

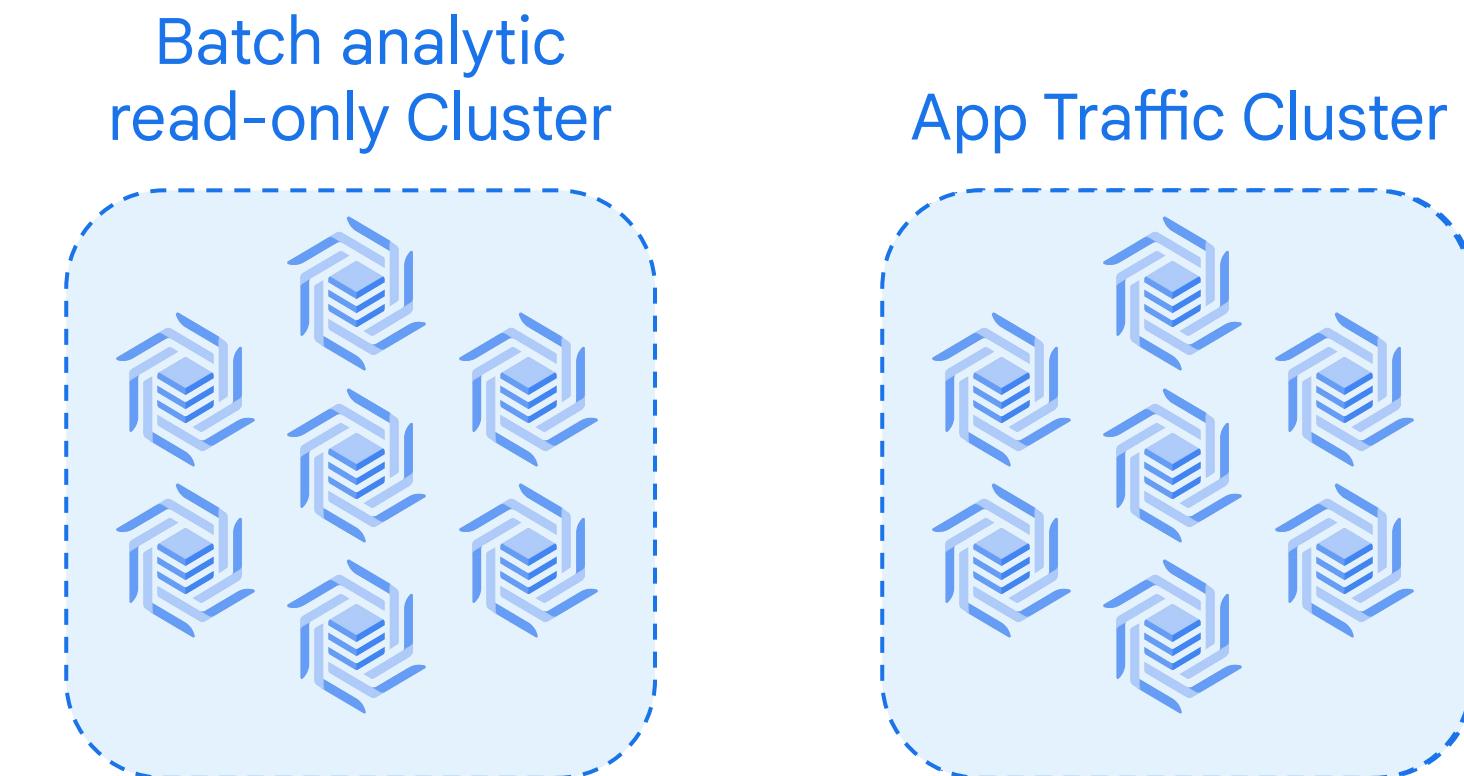


Use Bigtable replications to improve availability

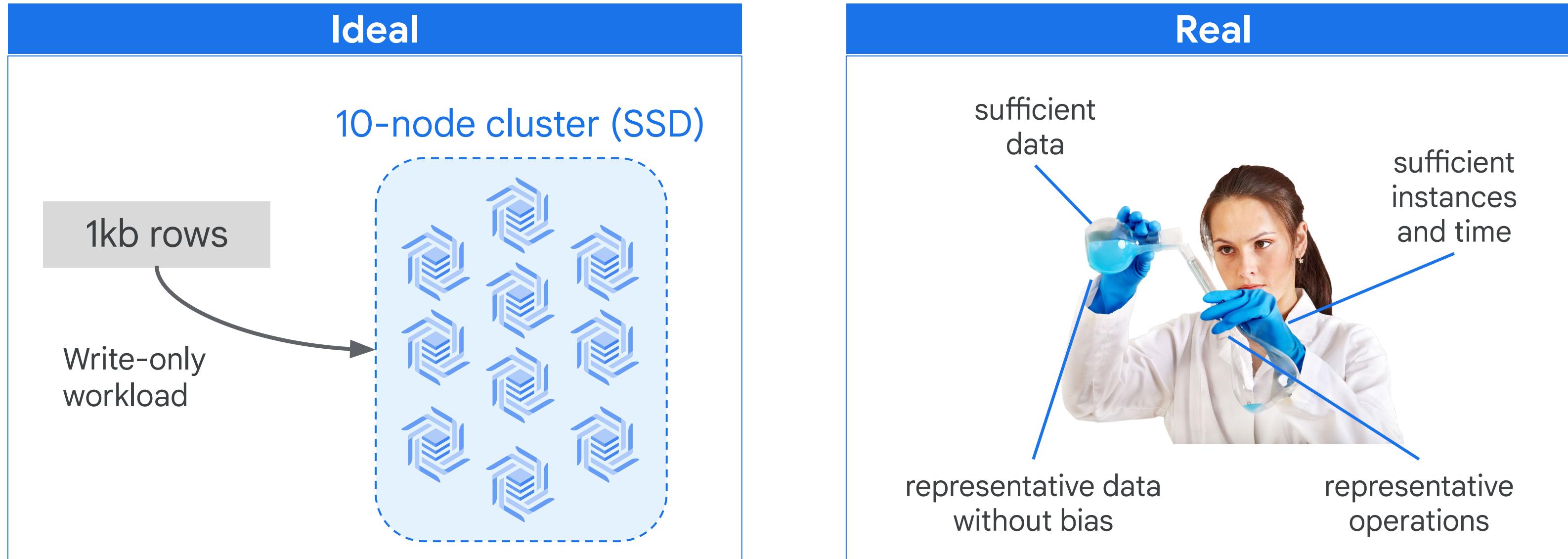
Why perform replication?

- Isolate serving applications from batch reads
- Improve availability
- Provide near-real-time backup
- Ensure your data has a global presence

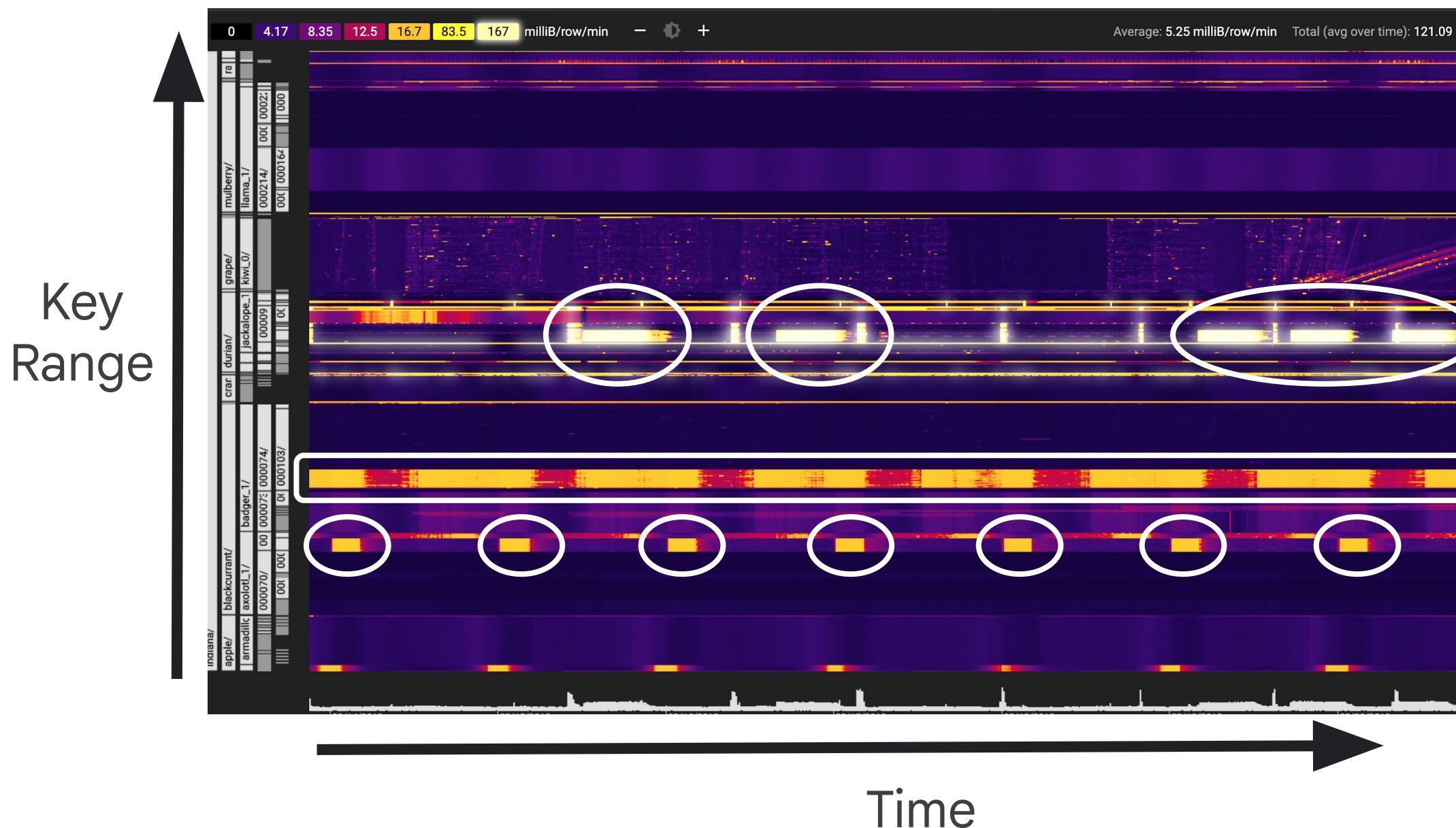
```
gcloud bigtable clusters create CLUSTER_ID \
    --instance=INSTANCE_ID \
    --zone=ZONE \
    [--num-nodes=NUM_NODES] \
    [--storage-type=STORAGE_TYPE]
```



Run performance tests carefully for Bigtable streaming



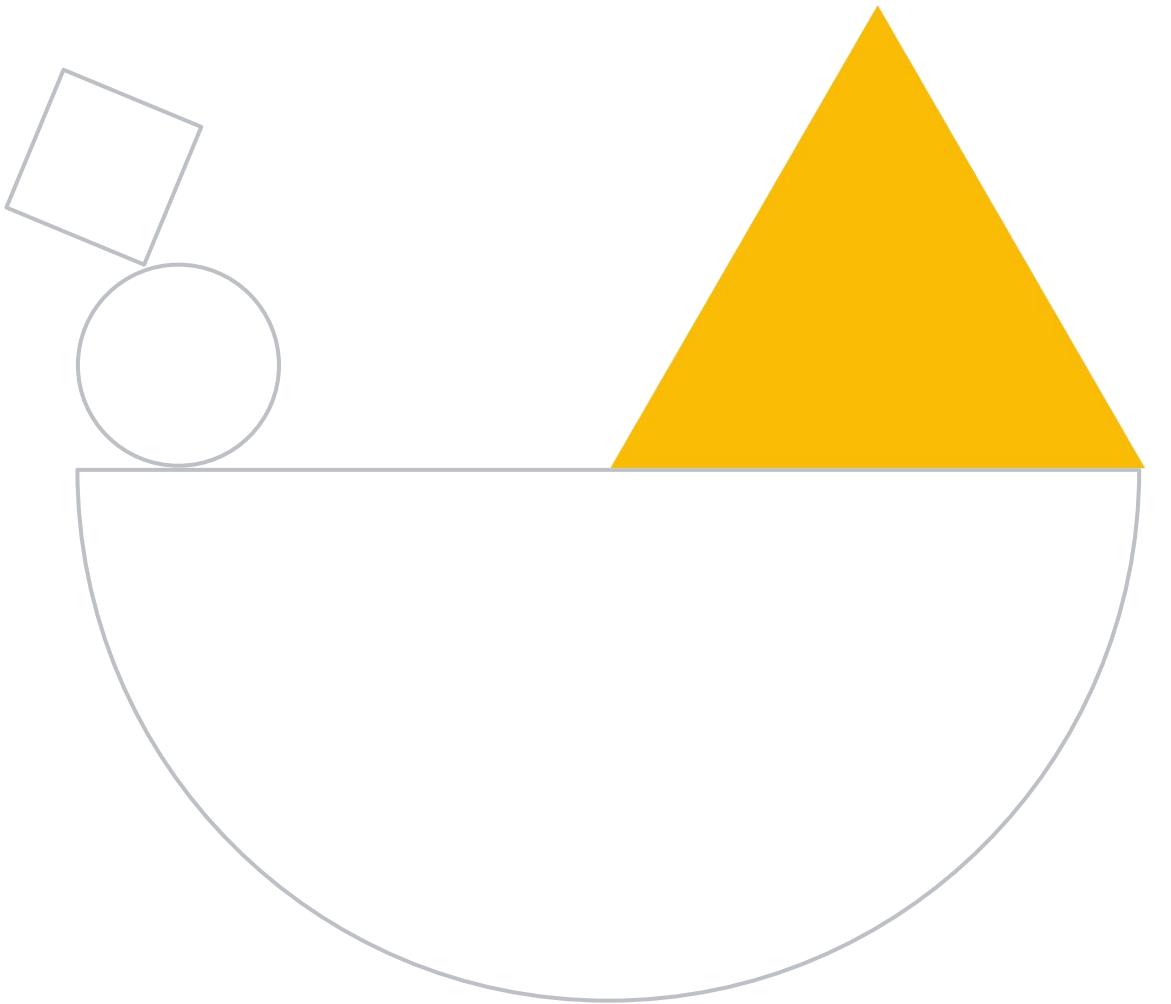
Key Visualizer exposes read/write access patterns over time and key space



- find/prevent hotspots
- find rows with too much data
- see if your key schema is balanced

Lab Intro

Streaming Data Processing:
Streaming Data Pipelines into Bigtable



Lab objectives

01

Launch Dataflow pipeline to read from Pub/Sub and write into Bigtable

02

Open an HBase shell to query the Bigtable database



