



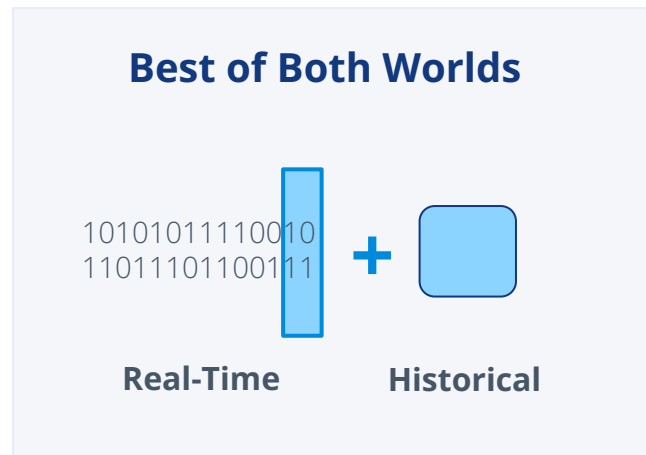
# Build Analytics Apps on Lakes **and** Streams with Apache Druid

Dremio Subsurface BO366, 3 March 2022  
[carl.dubler@imply.io](mailto:carl.dubler@imply.io)

# How does Netflix do it?

- Non-disruptive software updates
- Diagnostic signals
- Viewing activity
- Plan changes
- MyList updates
- Profile changes

Instantly across all devices for 230M+ members!



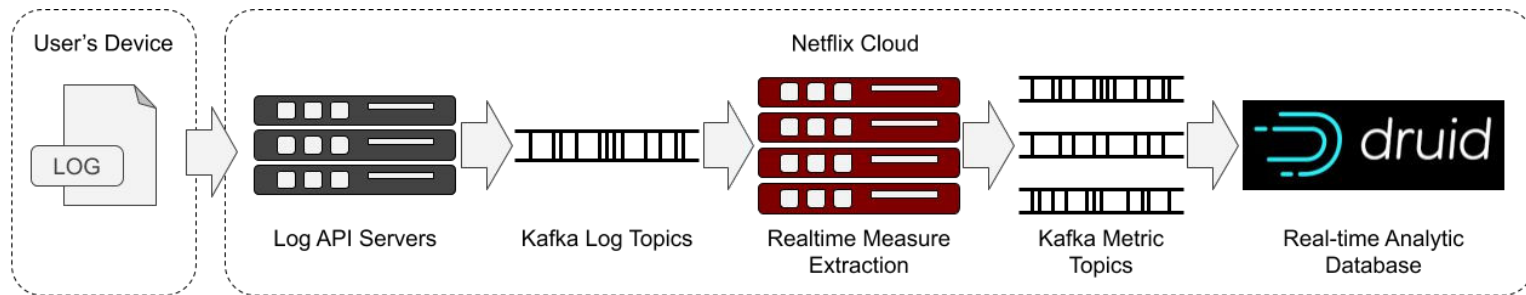
# What if you need real-time and historical data?

Q: How can Netflix update 230 million active subscribers without harm?

- Enable new versions by subset
- 7 platforms
- 5 device types

A: Custom app using Apache Druid to get insights from playback device logs.

- Kafka streams of >2M events/sec
- 1.5T rows of detailed event data
- Query response in 10s of milliseconds



Source: [Netflix Technology Blog](#)

# Lakes **and** streams are needed for modern analytic apps

## Going beyond traditional BI and dashboards

### Internal Apps

#### Deliver the Best Cloud Service & Experience

Real-time application, cloud, and product monitoring & diagnostics



#### Deliver Next-Gen IT/Security Insights

Real-time visibility to improve business, service, or security ops



#### Maximize Customer Sales and Revenue

Real-time insights for digital-native businesses, gaming, and advertising



### External Apps

#### Deliver Analytics to External Customers

Real-time analytics and visibility delivered to customers



# What Is Druid?



Apache Druid is the best database for **modern analytics applications**

**10+**  
Years

**400+**  
Contributors

**1,000+**  
Customers deployed

**11,000+**  
Community members

# Why Do Developers Use Druid?



## Interactive analytics at any scale

- Provide the best analytics experience
- **Cisco** ThousandEyes: 180 metrics, 5-20 QPS, <200ms



## High concurrency at the best value

- Reach more users without surprise bills
- **Ippen Digital**: 30x more users, 25% less cost



## Insights on real-time and historical data

- Simplify architecture with past and present data together
- **Target**: 3,500+ sources (PB+) 70K DAU, 50 QPS, <600ms



# What Sets Druid Apart?

How we achieve sub-second, interactive query power at any scale



Flexible

**Elastic and distributed architecture** to build any application at any scale

- Unique storage-compute
- Independent components



Efficient

**Optimized storage and query engine** unified to maximize valuable compute resources

- Multi-level indexes
- Massively parallel



Resilient

**Automatic recovery** enables non-disruptive growth and always-on operations

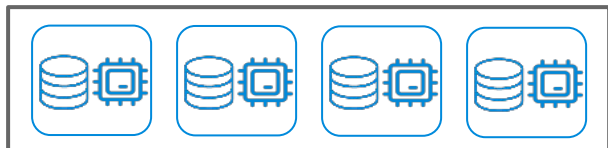
- Multi-level replication
- Automatic load balancing

# Elastic and Distributed Architecture

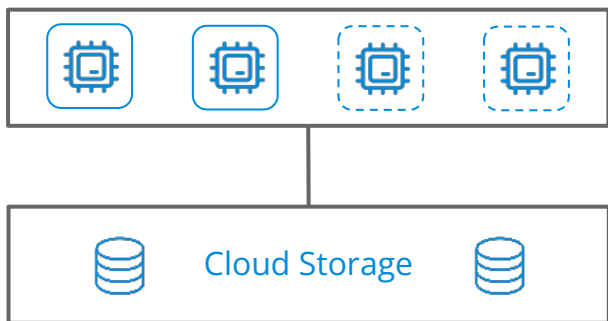


Flexible

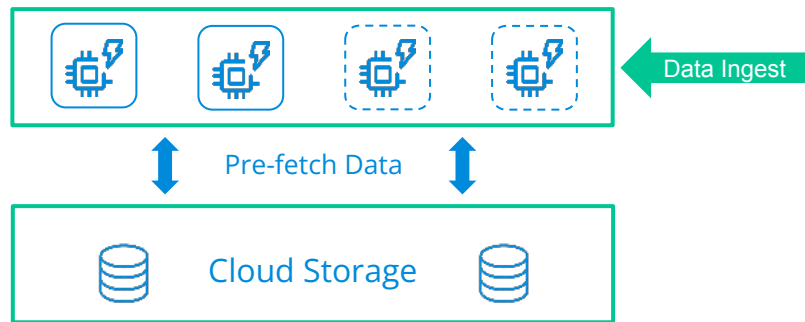
Some systems *share nothing* in order to maximize query **performance**:



Others *separate storage and compute* to maximize **elasticity**:



Druid maximizes both **performance** and **elasticity**:



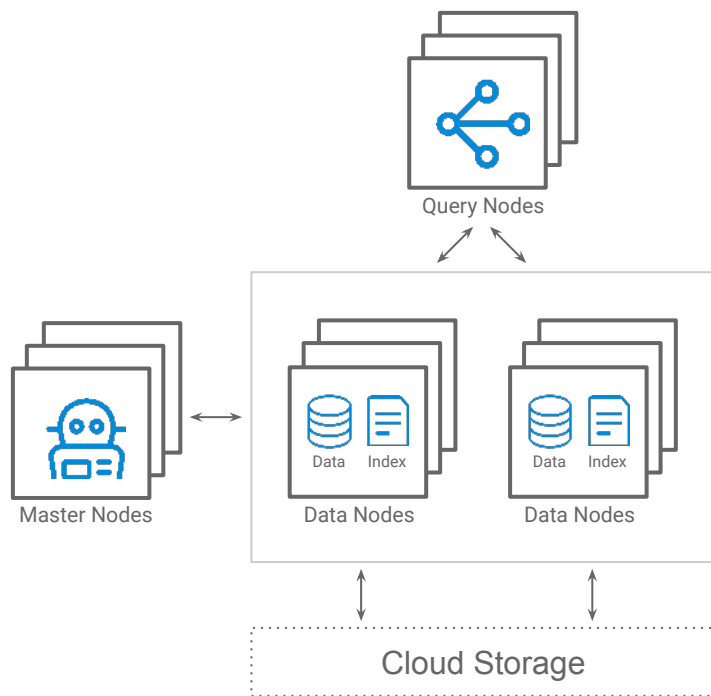
- On ingest, data are segmented, indexed, optionally aggregated, and stored (typically in cloud or can be local)
- Segments are pre-fetched to compute nodes, ready for query (recent streaming data can be queried immediately)
- Clusters can grow to thousands of nodes for limitless scale (ingest and query)



# Optimized Storage and Compute Engine



Efficient



A unique storage format and query engine designed together for best performance while maximizing valuable compute resources

- Data are placed in segments, with each column stored separately and compressed
- Streaming data are held in data node memory temporarily to allow for immediate querying before persisting to cloud storage
- Historical data (batch) are persisted in storage first then pre-fetched back to the data nodes
- Indexes are automatically created (global, data dictionary, bitmap)
- Queries are fanned-out across the cluster for massively parallel processing (scatter/gather)

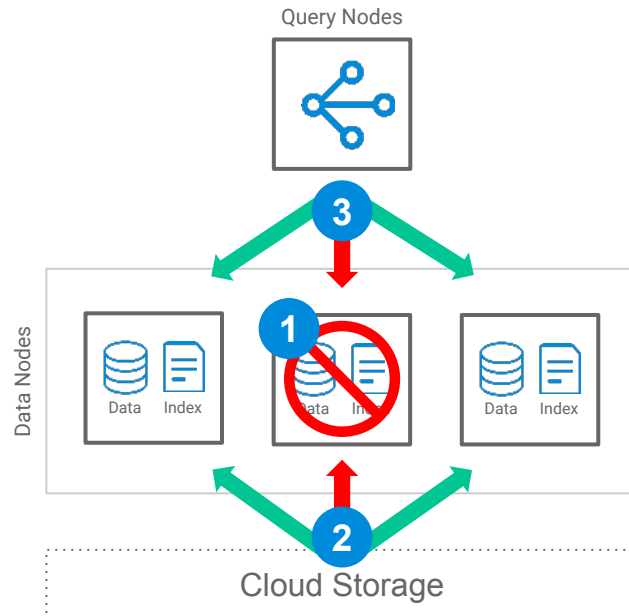
# Automatic Recovery

Druid recovers from faults automatically and enables maintenance without downtime windows.

1. If a node fails, no action is required by an administrator
2. Data are automatically retrieved from cloud storage and distributed across remaining nodes
3. Queries are automatically re-distributed across remaining nodes to maintain parallel performance

When a failed node comes back online, the data and queries are automatically re-balanced.

Because data are stored in the cloud as well, nothing is lost if the entire cluster fails or goes offline.



Free trial: [imply.io/polaris](https://imply.io/polaris)

# Imply Polaris

The cloud database service, built from Apache Druid



## Fully-Managed Cloud Service

Fully-Managed cloud service for Apache Druid. Build modern analytics apps without worrying about the underlying infrastructure.



## Database Optimization

Get all the performance out of Apache Druid without being an expert.



## Single Development Experience

Bring data to analytic apps with a complete, integrated experience.

# Expanding the reach of Druid for applications

## Expanded Capabilities in 2022

### Interactivity

Sub-sec response on PB+ data  
100K+ queries per second  
Streaming and batch ingestion



### Reports

Very large result sets  
Long-running queries

### Alerts

Millions of alerts  
Large #s of objects  
Complex conditions



The Database for Modern Analytics Applications