BITS Pilani
Pilani Campus

Popular scaling approaches

Prof. Akanksha Bharadwaj
CSIS Department

innovate achieve lead

BITS Pilani
Pilani Campus

SE ZG583, Scalable Services
Lecture No. 2

innovate achieve lead

BITS Pilani
Pilani Campus

Partitioning and Sharding

Introduction

- Partitioning and Sharding are scalable approaches to manage and distribute large volumes of data in a database system.
- In many large-scale solutions, data is divided into *partitions* that can be managed and accessed separately.

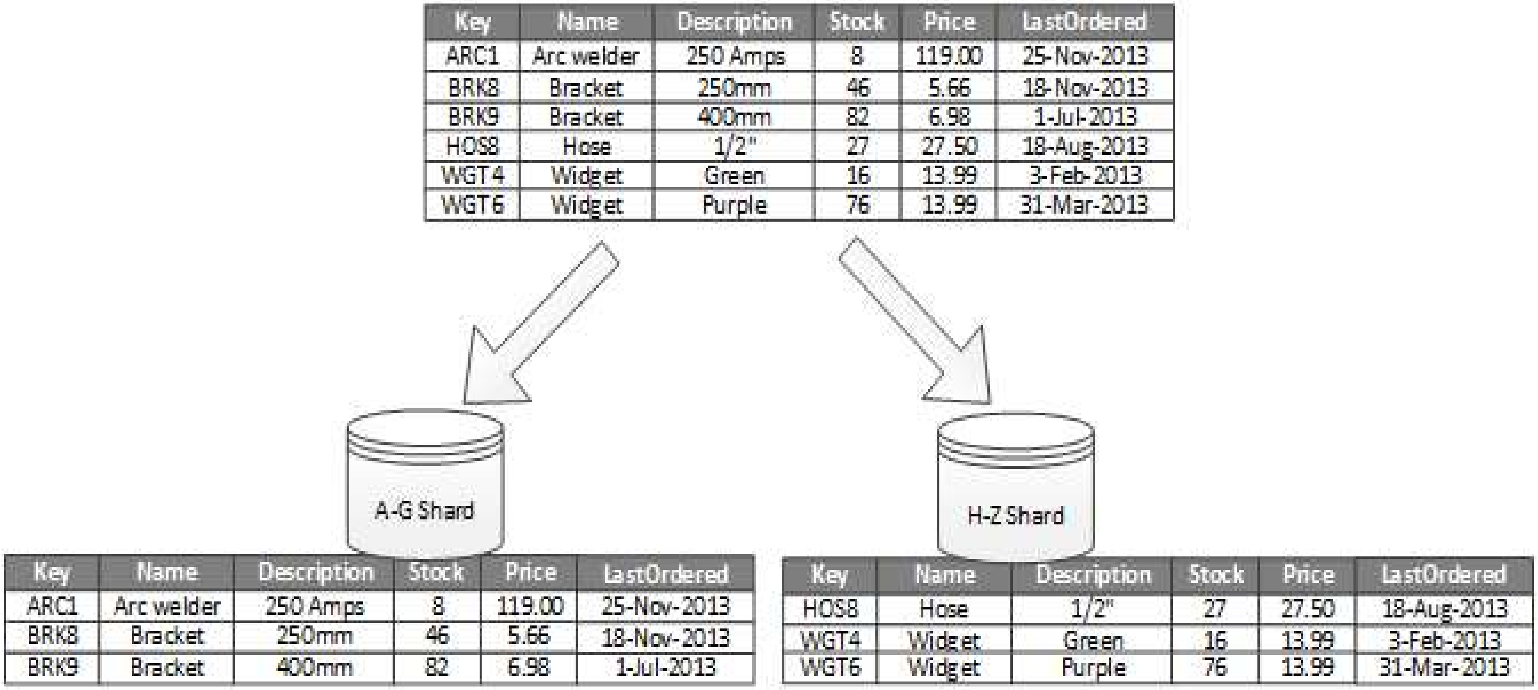
Why partition data?

- Improve scalability
- Improve performance
- Improve security
- Provide operational flexibility
- Improve availability

Types of Partitioning

- Horizontal partitioning
- Vertical partitioning
- Functional partitioning

Horizontal partitioning (Sharding)

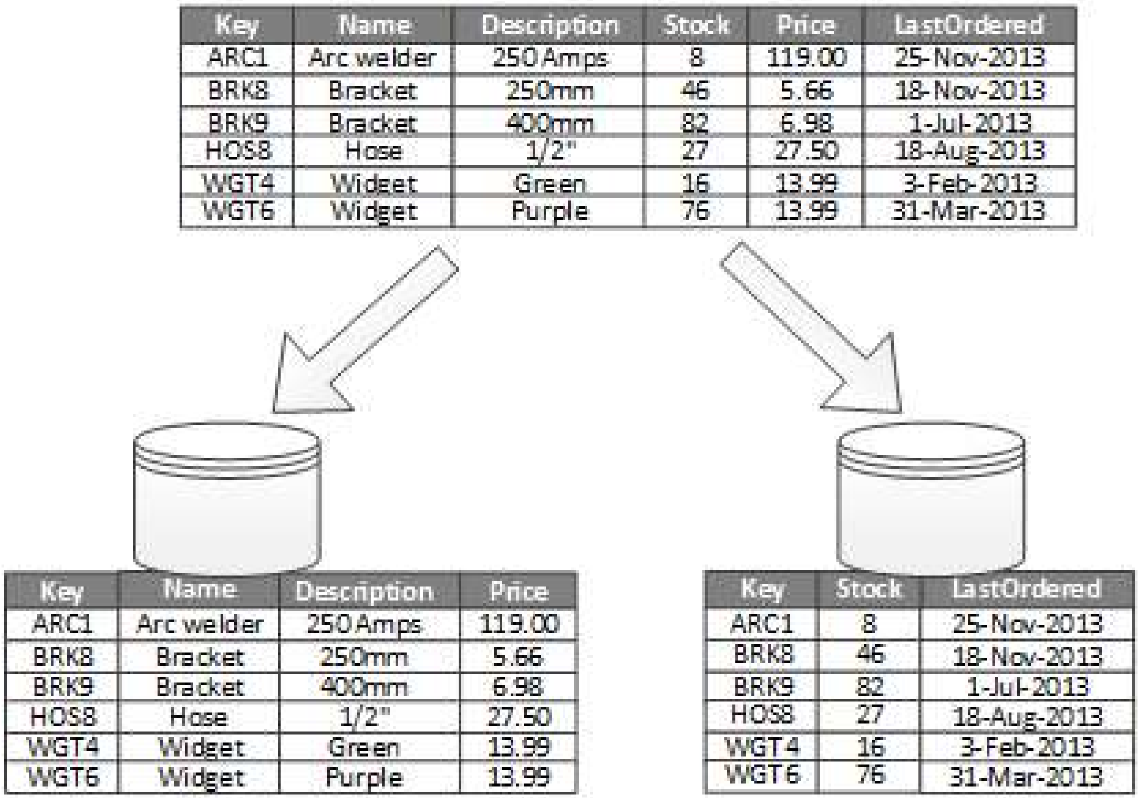


Key	Name	Description	Stock	Price	LastOrdered
ARC1	Arc welder	250 Amps	8	119.00	25-Nov-2013
BRK8	Bracket	250mm	46	5.66	18-Nov-2013
BRK9	Bracket	400mm	82	6.98	1-Jul-2013
HOS8	Hose	1/2"	27	27.50	18-Aug-2013
WGT4	Widget	Green	16	13.99	3-Feb-2013
WGT6	Widget	Purple	76	13.99	31-Mar-2013

Key	Name	Description	Stock	Price	LastOrdered
ARC1	Arc welder	250 Amps	8	119.00	25-Nov-2013
BRK8	Bracket	250mm	46	5.66	18-Nov-2013
BRK9	Bracket	400mm	82	6.98	1-Jul-2013

Key	Name	Description	Stock	Price	LastOrdered
HOS8	Hose	1/2"	27	27.50	18-Aug-2013
WGT4	Widget	Green	16	13.99	3-Feb-2013
WGT6	Widget	Purple	76	13.99	31-Mar-2013

Vertical partitioning

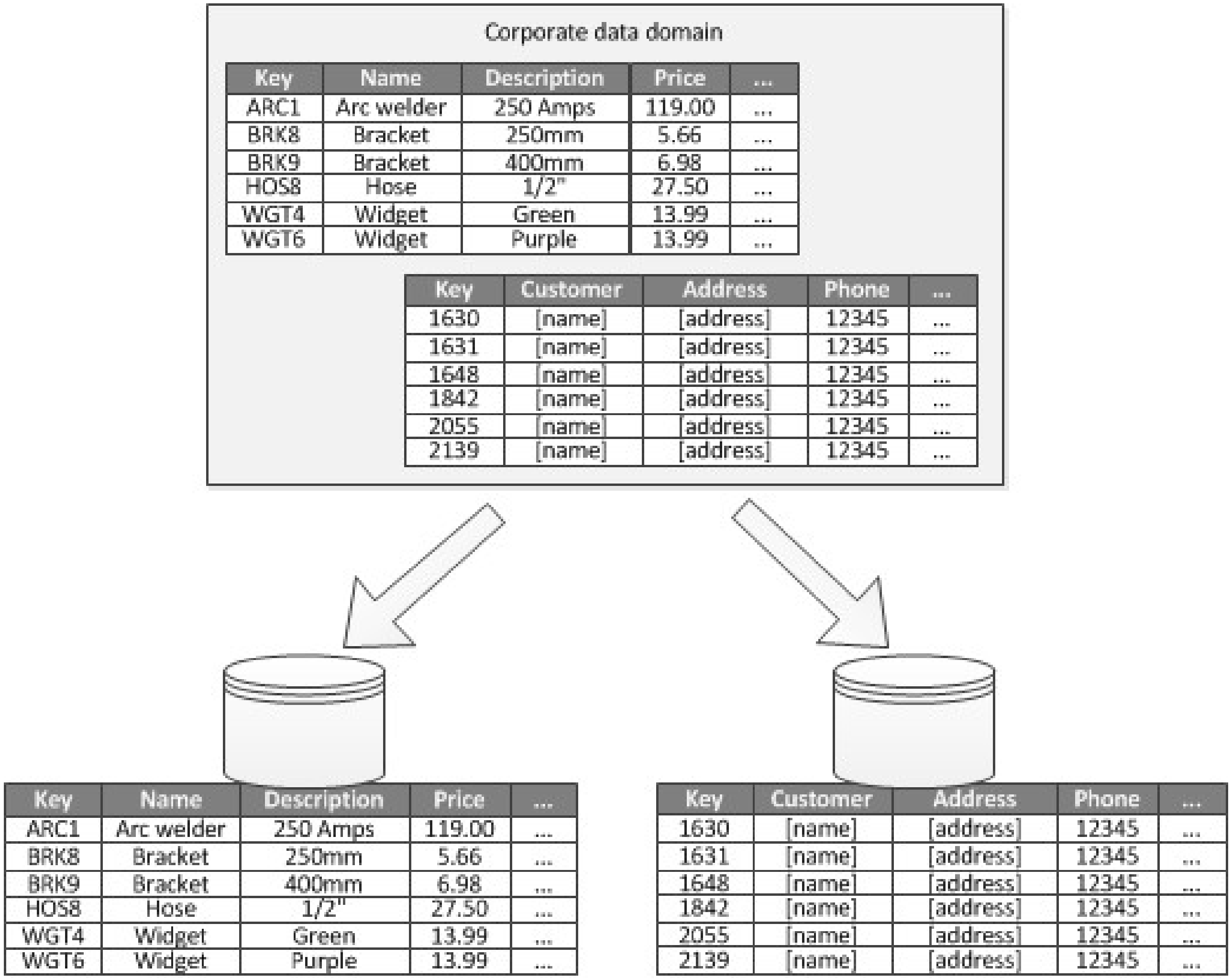


Key	Name	Description	Stock	Price	LastOrdered
ARC1	Arc welder	250 Amps	8	119.00	25-Nov-2013
BRK8	Bracket	250mm	46	5.66	18-Nov-2013
BRK9	Bracket	400mm	82	6.98	1-Jul-2013
HOS8	Hose	1/2"	27	27.50	18-Aug-2013
WGT4	Widget	Green	16	13.99	3-Feb-2013
WGT6	Widget	Purple	76	13.99	31-Mar-2013

Key	Name	Description	Price
ARC1	Arc welder	250 Amps	119.00
BRK8	Bracket	250mm	5.66
BRK9	Bracket	400mm	6.98
HOS8	Hose	1/2"	27.50
WGT4	Widget	Green	13.99
WGT6	Widget	Purple	13.99

Key	Stock	LastOrdered
ARC1	8	25-Nov-2013
BRK8	46	18-Nov-2013
BRK9	82	1-Jul-2013
HOS8	27	18-Aug-2013
WGT4	16	3-Feb-2013
WGT6	76	31-Mar-2013

Functional partitioning



Corporate data domain

Key	Name	Description	Price	...
ARC1	Arc welder	250 Amps	119.00	...
BRK8	Bracket	250mm	5.66	...
BRK9	Bracket	400mm	6.98	...
HOS8	Hose	1/2"	27.50	...
WGT4	Widget	Green	13.99	...
WGT6	Widget	Purple	13.99	...

Key	Customer	Address	Phone	...
1630	[name]	[address]	12345	...
1631	[name]	[address]	12345	...
1648	[name]	[address]	12345	...
1842	[name]	[address]	12345	...
2055	[name]	[address]	12345	...
2139	[name]	[address]	12345	...

Key	Name	Description	Price	...
ARC1	Arc welder	250 Amps	119.00	...
BRK8	Bracket	250mm	5.66	...
BRK9	Bracket	400mm	6.98	...
HOS8	Hose	1/2"	27.50	...
WGT4	Widget	Green	13.99	...
WGT6	Widget	Purple	13.99	...

Key	Customer	Address	Phone	...
1630	[name]	[address]	12345	...
1631	[name]	[address]	12345	...
1648	[name]	[address]	12345	...
1842	[name]	[address]	12345	...
2055	[name]	[address]	12345	...
2139	[name]	[address]	12345	...

innovate achieve lead

BITS Pilani
Pilani Campus

Scalability using NoSQL and HDFS

NoSQL and Scalability

NoSQL databases (e.g., MongoDB, Cassandra, DynamoDB) are specifically built for high scalability, especially in scenarios requiring distributed systems, real-time data access, and schema flexibility.

Scalability Features in NoSQL:

- Horizontal Scalability (with Sharding)
- Dynamic Schema
- Replication
- High Throughput
- Decentralized Architecture

Use Cases for NoSQL Scalability

- Social media platforms managing large volumes of user interactions.
- E-commerce systems requiring fast, scalable databases for product catalogs and transactions.
- Real-time analytics on streaming data.

HDFS and Scalability

HDFS is a distributed file system designed for storing and processing massive datasets (big data) in a scalable and fault-tolerant manner. It forms the foundation of the Hadoop ecosystem.

Scalability Features in HDFS:

- Horizontal Scalability
- Replication for Fault Tolerance
- Write Once, Read Many (WORM) Design
- MapReduce Integration
- Block Storage
- Cluster Management

Use Cases for HDFS Scalability

- Storing and analyzing massive datasets (e.g., petabytes of log data).
- Batch processing for data warehouses and ETL pipelines.
- Data lakes for structured and unstructured data.

innovate achieve lead

BITS Pilani
Pilani Campus

Managing high velocity data streams

Key Challenges

- High Throughput
- Low Latency
- Scalability
- Fault Tolerance
- Storage Efficiency

innovate achieve lead

BITS Pilani
Pilani Campus

Strategies for Managing High-Velocity Video Data

1. Content Delivery Network

- CDN is nothing more than a bunch of globally distributed computers that are directly connected and move data from one end to another.
- CDN stores multiple copies of video content at edge locations.
- We can use CDNs to cache and distribute video content closer to end-users, reducing latency and bandwidth usage.
- Examples: Akamai, Cloudflare, AWS CloudFront.

CDN Case Study: YouTube

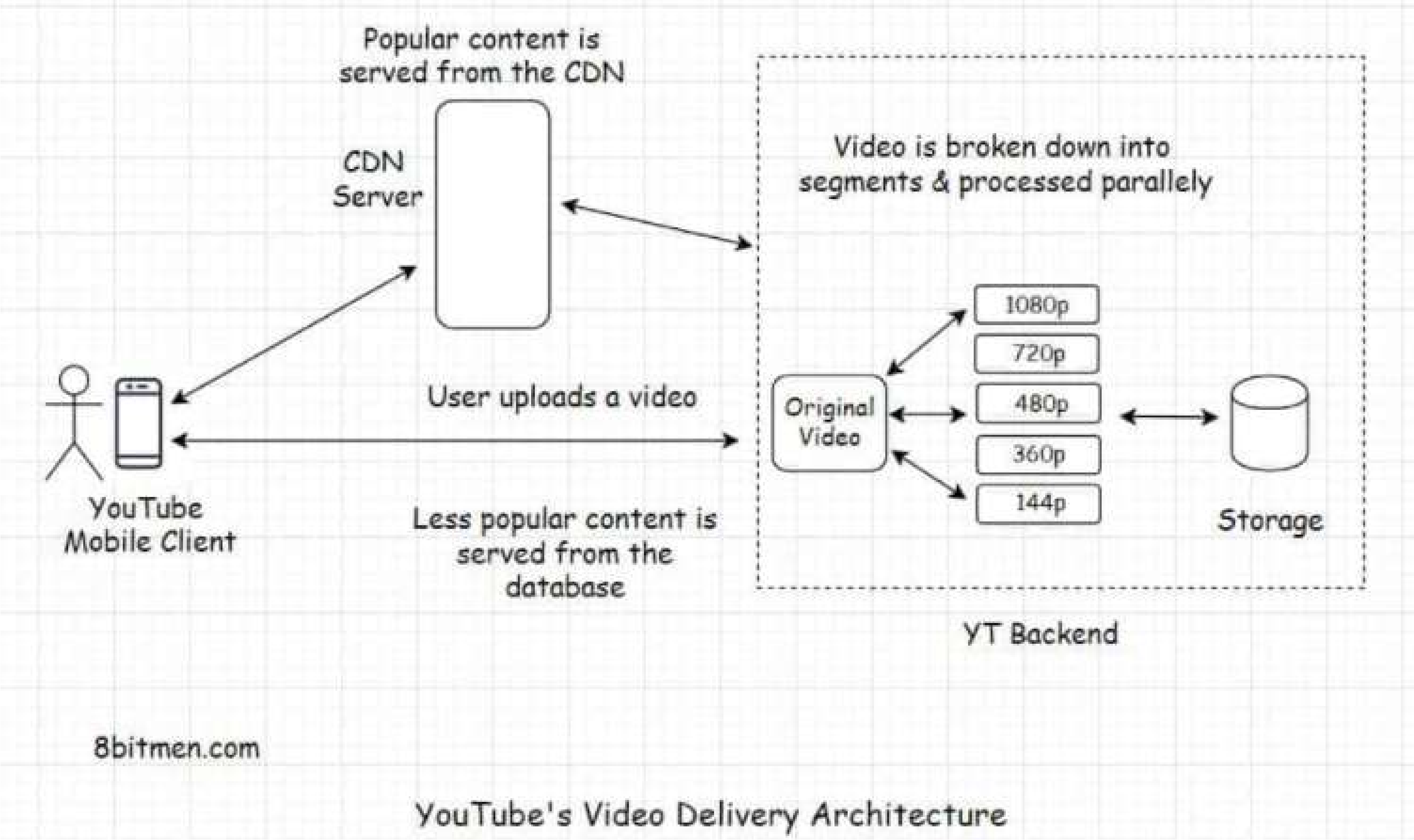

Popular content is served from the CDN
CDN Server
Video is broken down into segments & processed parallely
1080p
720p
480p
360p
144p
Original Video
User uploads a video
YouTube Mobile Client
Less popular content is served from the database
Storage
YT Backend
8bitmen.com
YouTube's Video Delivery Architecture

Image: google

2. Adaptive Bitrate Streaming (ABR)

- Deliver video streams in different quality levels (bitrates) and dynamically switch between them based on the user's network speed and device capabilities.
- Protocols That Support ABR: HLS (HTTP Live Streaming), DASH (Dynamic Adaptive Streaming over HTTP).

3. Efficient Video Compression

- Use advanced codecs to reduce video file size while maintaining quality.

- **Example for video Streaming**: To make the videos viewable on different devices Netflix performs transcoding or encoding which involves finding errors and converting the original video into different formats and resolutions.

4. Edge Computing

- Edge computing is a distributed information technology (IT) architecture in which client data is processed at the periphery of the network, as close to the originating source as possible.

- We can process and cache video content at edge locations to reduce latency and offload computation from the central servers.

- Example: Transcode video streams at the edge for faster delivery.

Case Study: Smart Security Camera

- **Device with Edge Capability**: A smart security camera equipped with an edge AI chip processes video feeds locally.
- **Edge Processing**: The camera runs motion detection algorithms (e.g., using TensorFlow Lite) directly on the device to identify unusual activity, like someone entering the camera's field of view.
- **Action at the Edge:** If motion is detected: Capture the event (e.g., save a short video clip). Send an alert to the homeowner’s mobile device (e.g., "Motion detected at the front door"). If no motion is detected, discard the video feed to save bandwidth.
- **Cloud Involvement:** Only the processed, important data (e.g., the detected motion event) is sent to the cloud for storage or additional analysis.

5. Real-Time Data Processing

- Process telemetry data (e.g., playback statistics, user behavior) in real time to optimize streaming.

- Use stream processing frameworks like Apache Flink or Kafka Streams to analyze and react to user interactions instantly.

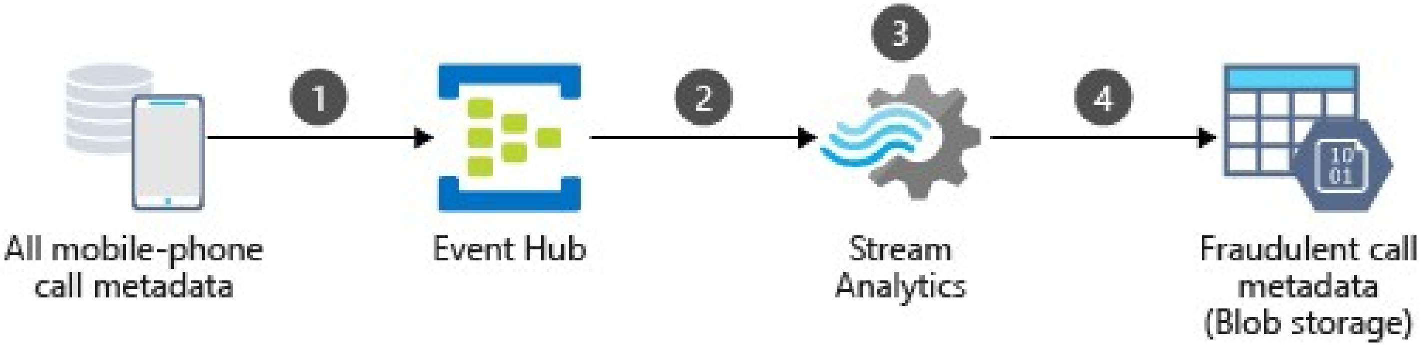
innovate achieve lead

BITS Pilani
Pilani Campus

Real Time Analytics

Case Study: Real Time Fraud Detection

Architecture


1
2
3
4
All mobile-phone call metadata
Event Hub
Stream Analytics
Fraudulent call metadata (Blob storage)

- Mobile phone call metadata is sent from the source system to an Azure Event Hubs instance.
- A Stream Analytics job is started, which receives data via the event hub source.
- The Stream Analytics job runs a predefined query to transform the input stream and analyze it based on a fraudulent-transaction algorithm.
- The Stream Analytics job writes the transformed stream representing detected fraudulent calls to an output sink in Azure Blob storage.

Web Conferencing

- Web conferencing involves real-time audio, video, and data sharing, requiring efficient processing of high-velocity data streams to ensure a seamless user experience.

Steps in Processing Streaming Data for Web Conferencing

1. Data Capture
2. Data Compression
3. Data Transmission
4. Stream Processing at the Server
5. Adaptive Bitrate Streaming
6. Content Delivery
7. Client-Side Rendering
8. Data Analytics and Feedback

Real-World Example: Zoom

- **Data Capture:** Captures audio, video, and chat data from participants.

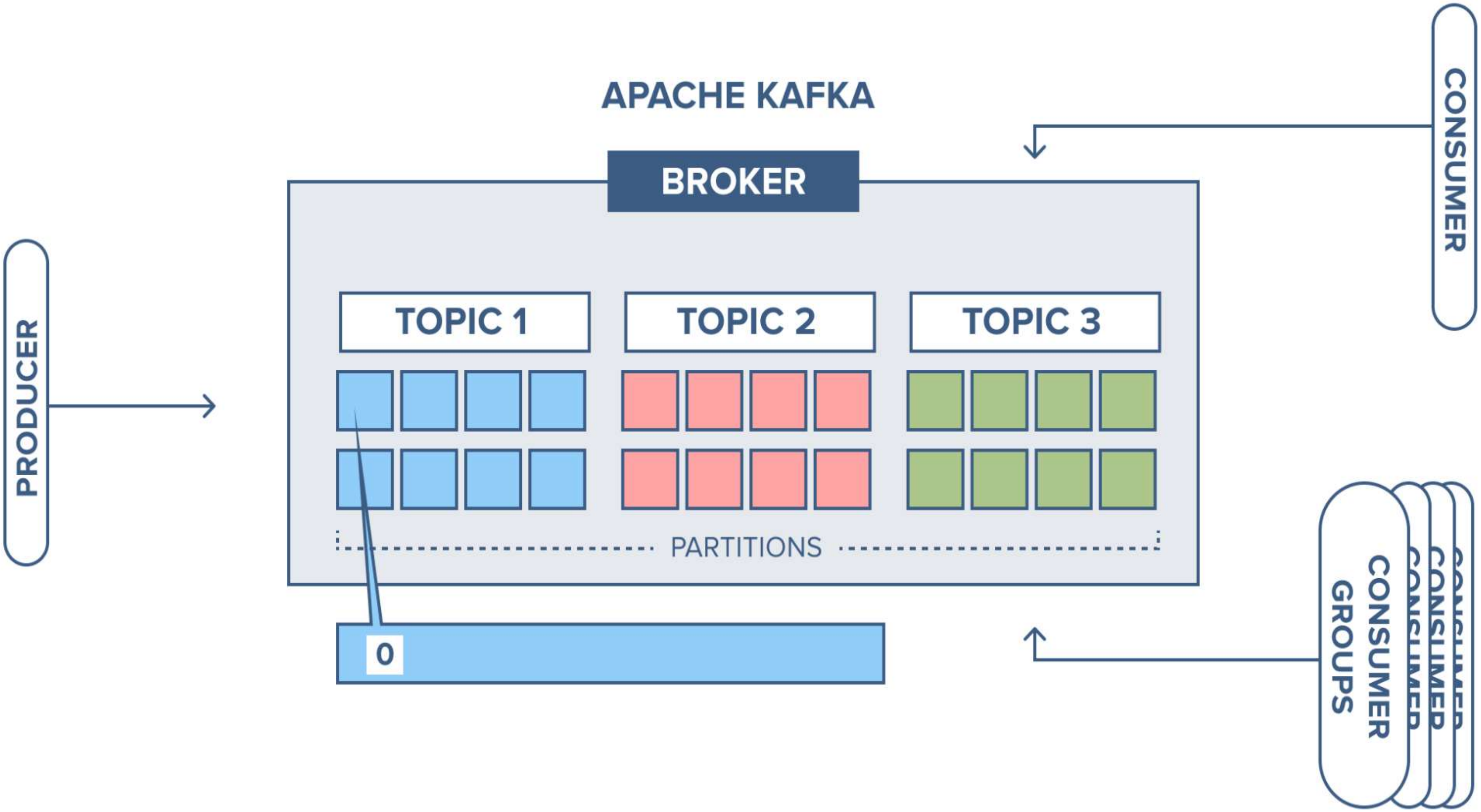
- **Compression:** Compresses streams to optimize for bandwidth.

- **Streaming Servers:** Use SFU architecture to minimize latency.

- **Scaling:** Scales dynamically using cloud servers to accommodate user load.

Managing high velocity Data Streams using Kafka

- **Kafka** is a powerful, distributed streaming platform designed to handle massive amounts of data in real-time. It's widely used for building real-time data pipelines and applications.


APACHE KAFKA
BROKER
TOPIC 1
TOPIC 2
TOPIC 3
PARTITIONS
0
PRODUCER
CONSUMER
CONSUMER GROUPS
CONSUMER
CONSUMER
CONSUMER

Kafka for Stream Processing and Analytics

- **Use Case:** Processing and analyzing continuous streams of data for real-time insights, anomaly detection, and predictive analytics.

- **Example:** Financial institutions use Kafka to process market data feeds, detect trading anomalies, and make real-time trading decisions. Retailers analyze customer behavior and preferences in real-time to offer personalized recommendations and promotions.

Ref: https://medium.com/@psnavya90/

Self Study

Article: https://netflixtechblog.com/behind-the-streams-live-at-netflix-part-1-d23f917c2f40

References

- https://docs.microsoft.com/en-us/azure/architecture/best-practices/data-partitioning
- https://www.mongodb.com/nosql-explained
- https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html
- https://netflixtechblog.com/
- http://highscalability.com/youtube-architecture
- https://docs.microsoft.com/en-us/azure/architecture/example-scenario/data/fraud-detection
- https://kafka.apache.org/11/documentation/streams/architecture
- https://www.gsma.com/iot/wp-content/uploads/2018/11/IoT-Edge-Opportunities-c.pdf
- https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7696529/