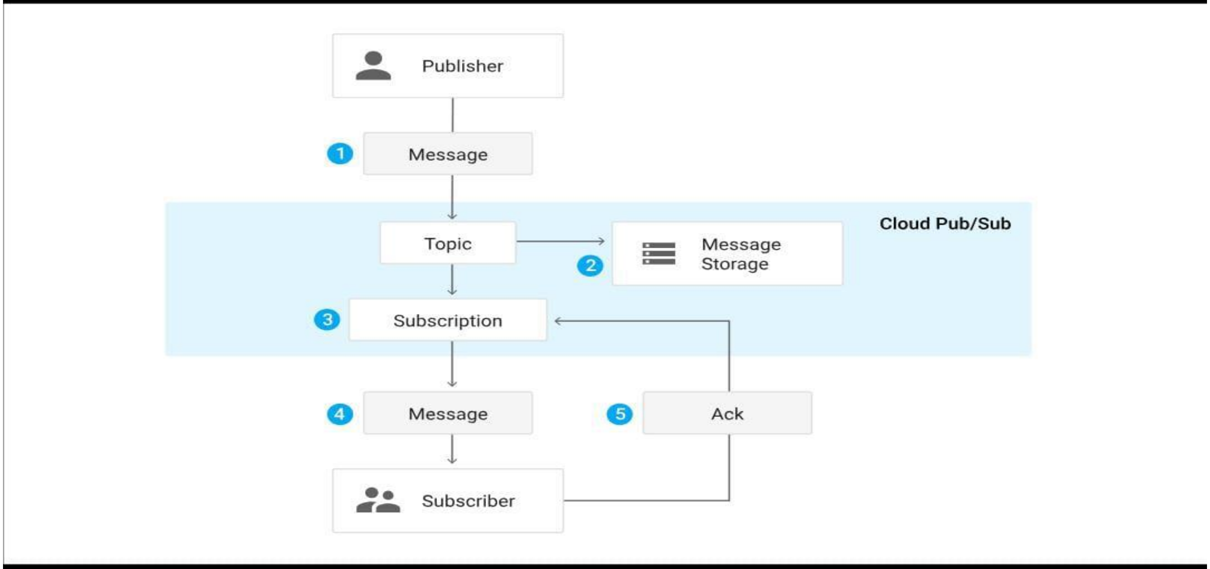
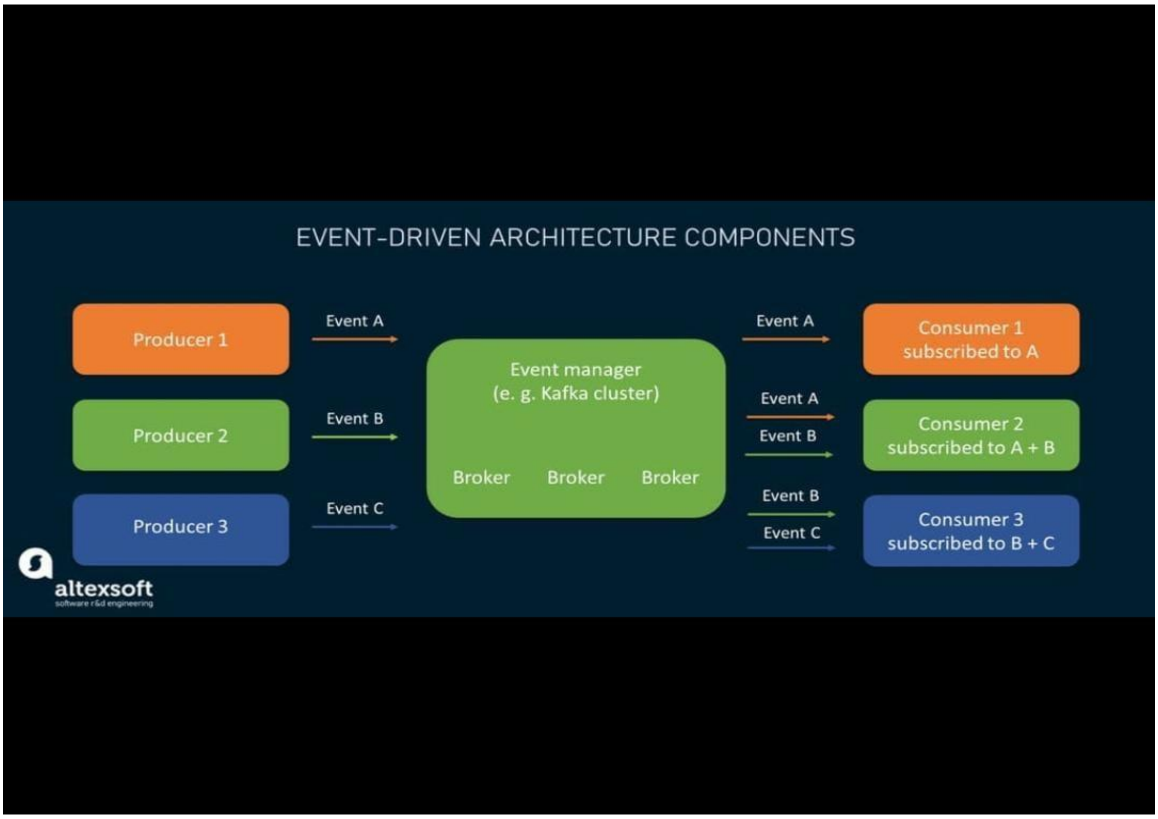
SYNOPSIS ON PUB SUB

Google Cloud Pub/Sub – Fully managed, real-time pub/sub messaging for asynchronous communication.



Architecture

- Publishers → Topics (message storage) → Subscriptions → Subscribers (pull/push delivery).
- Control Plane (routers assign clients); Data Plane (forwarders handle traffic with replication across clusters/disks).
- Message Flow: Publish → Persist (N clusters, M disks) → Ack publisher → Deliver to subscribers → Ack/delete.



Core Components List

- Topics: Message feeds with optional schemas.
- Subscriptions: Pull (fetch) or push (webhook) endpoints.
- Messages: Payload + attributes; lifecycle (unacked → redeliver → dead-letter).

Key Features Table

Feature	**Benefit**
At-least-once delivery	Reliable with retries
Global scalability	Millions msg/sec, low latency
Dead-letter queues	Handles failures gracefully
Integrations	Eventarc, Dataflow, BigQuery

Use Cases & Demo

- Event streaming (GCS events via Event arc), IoT, analytics.
- Quick Setup: gcloud pubsub topics create my-topic;
- gcloud pubsub subscriptions create my-sub --topic=my-topic.

Steps:

- gcloud services enable pubsub.googleapis.com
- gcloud services list --enabled | grep pubsub
- gcloud pubsub topics create user-events
- gcloud pubsub topics list
- gcloud pubsub subscriptions create user-event-sub --topic=user-events
- gcloud pubsub subscriptions list
- gcloud pubsub topics publish user-events --message='{"event": "User_Created", "user": "Aaryan"}'
- gcloud pubsub subscriptions pull user-event-sub --limit=5 --auto-ack