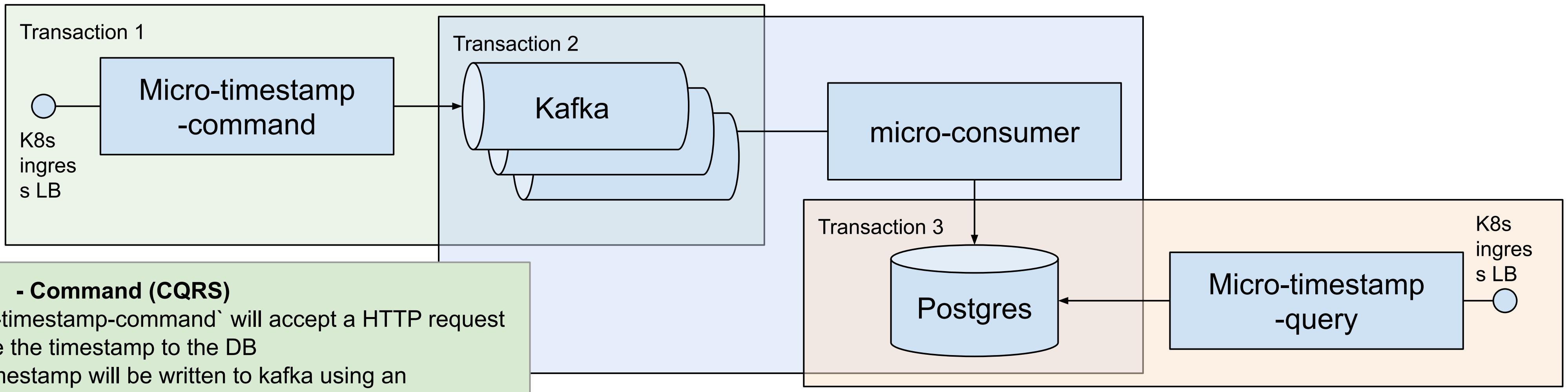
CQRS Implementation



Transaction 1 - Command (CQRS)

1) `micro-timestamp-command` will accept a HTTP request to write the timestamp to the DB
2) The timestamp will be written to kafka using an idempotent producer (the message key will be a UUID v4)
3) The response will return the UUIDv4 for the message key back to the client
4) As this is a command-service, it would not retrieve records from postgres, a separate query service would be built for that purpose (time permitting)

Non-functional Requirements

- The kafka log is the golden-record, records in the log are immutable and the postgres view is purely a “temporary” representation of the data
- No retry/Deadlettering will be implemented given time constraints, if there’s a database lock or outage the micro-consumer will not be able to aggregate the records and effectively block downstream processing but will recover upon DB recovery
- The DR process for this will be to re-deploy the consumer and re-point the configuration to another DB, the consumer will rebuild the database dynamically from the golden record in kafka.
- Multiple consumers could build multiple DBs from the single kafka log
- The postgres isn’t HA in this POC as it can be dynamically rebuilt on-the-fly it’s an aggregation of the events in Kafka

Transaction 2 - “View” creation

1) `micro-consumer` will use a consumer-group to consume the timestamp log records from Kafka
2) Upon consumption of those messages will be written to postgres using a DB transaction
    a) Open message
    b) Parse contents
    c) Open DB transaction
        i) Write message key (UUID) as the PK on the table
        ii) Write the timestamp as the value
    d) Commit transaction
        i) Upon Success
            (1) Acknowledge kafka message
        ii) Upon Failure
            (1) Do not acknowledge (BLOCK)

Implementation Details

- Where prudent, all services and supporting-services will be deployed as containers (A managed k8s i.e. GKE being the preference)
- In "production" a managed-service will be used to operate kafka e.g. Amazon MSK and provide backup/HA

Transaction 3 - Query (CQRS)

1) Micro-timestamp-query will query the database based on UUID to ensure a record has been created for the requested query

Note this may be omitted given time constraints (as it’s not explicitly required given the design)