

The ODA service

The Observable Data Access service (ODA), depicted in Figure 1, is fed by Data Generators that push data acquired by sensors, digital twins, or services, through a streaming broker (Apache Kafka). The ODA stores data in the short term (e.g. a week, a month) and provides data access to Data Consumers in two ways:

- a DC can subscribe to certain topics to get data streamed by generators about such topics, or
- a DC can pull data by invoking a REST API provided by ODA.

Conceptually, the data flow in a system exploiting an ODA service is bottom-up, from the DGs to the DCs.

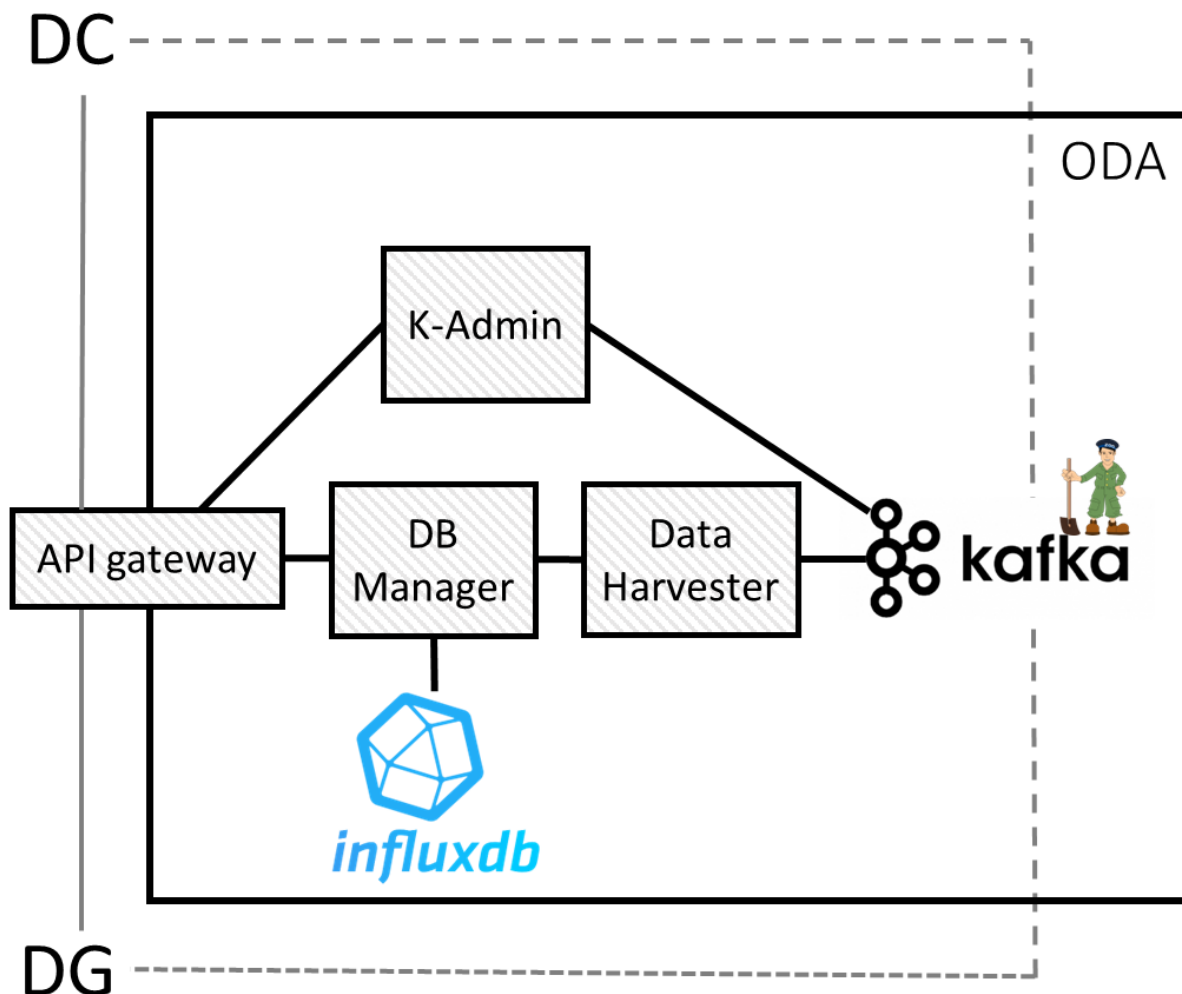


Figure 1: The ODA service

The database of the ODA is a time series database, implemented by exploiting InfluxDB.

Apache Kafka is used as a broker to implement the queue where DGs push data and DCs receive subscribed data. To manage the Kafka broker is used Apache Zookeeper.

The Data Harvester acts as a consumer of the Kafka broker subscribed to all the topics to store the data in the database and retrieve them when requested.

The K-Admin create in Kafka the topics registered in ODA by the DGs and provides the list of topics to the DCs during the registration process.

Interacting with ODA

Both DCs and DGs must register to the ODA service through the API gateway, which delivers the endpoint to reach the Kafka broker. Figure 2 shows how DGs must act to send data to the ODA service, starting with the registration of the topics it will use and then streaming the data to the Kafka broker directly.

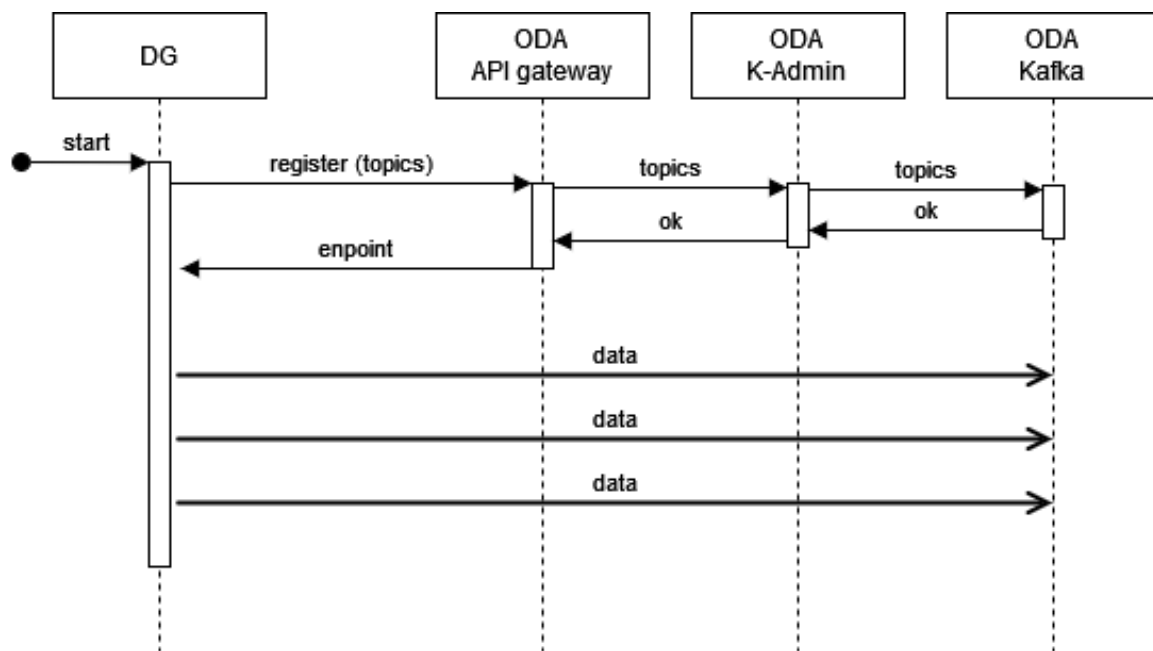


Figure 2: DG streaming data

Figure 3 shows how DCs interested in data streaming must act to receive data from the ODA service. After the registration through the API gateway, DCs receives the list of topics of ODA and the Kafka endpoint. They must subscribe to topics they are interested in contacting the Kafka broker. From that point, the data streamed to Kafka having such topics will be sent to the DC.

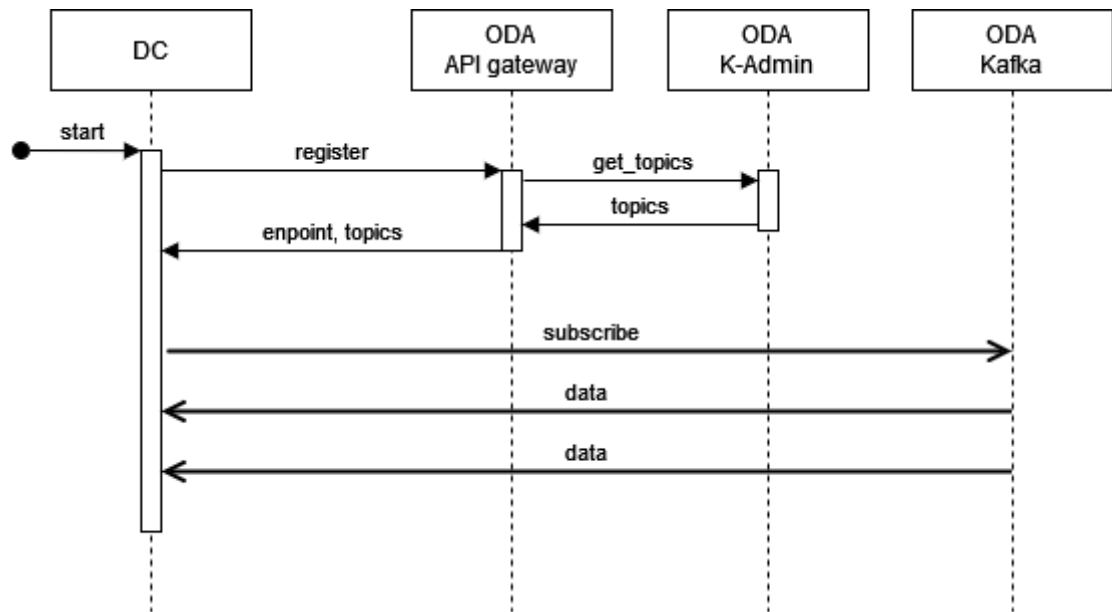


Figure 3: DC receiving streamed data

Figure 4 shows how DCs can pull data stored in the DB. They send a query to the API gateway, which propagates the query to the Data Service component. The data retrieved from the database is sent back to the DC along the reverse path.

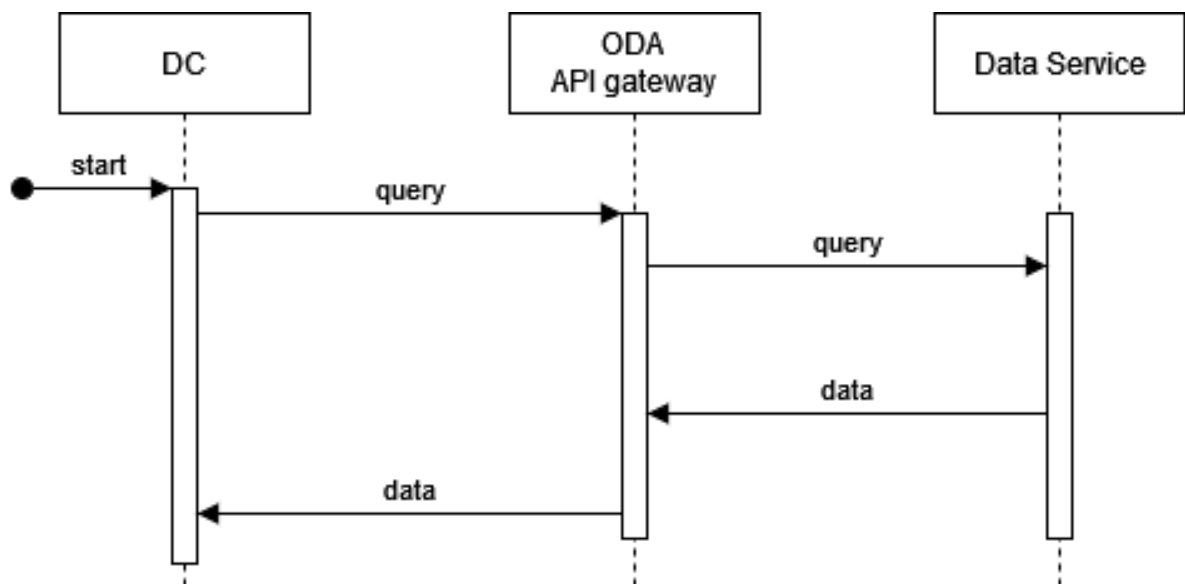


Figure 5: DC queries data stored in the DB

Data format

The format of data sent to the ODA is extensible to make it easily usable by different DGs. This is obtained by prescribing that all data must include the following mandatory fields:

```
{  
  "timestamp": <string>  
  "generator_id": <string>  
  "topic": <string>  
  "data": <string>  
}
```

where:

- `timestamp` is the data timestamp (in the ISO 8601 format),
- `generator_id` is a unique identifier string for each ODA representing the generator of the data,
- the `topic` chosen by the DG to label the data, and
- the payload data (as a string).

Further structuring of the data can be done by adding further structure within the data field.