Data Storage Implementation: KV + relational

Kafka Connect
- Framework
- Connects external systems/databases w/ Kafka

Sink and Source connectors.
- Sink connector
    - Ability to export data from topic(s) to a relational database
    - Don't need to know where the data comes from
        - Loose coupling
    - Gets generic representation of the data and then sink connector plugin writes it to the target system

- Source connector
    - Ability to import data/information from a relational database to a topic
    - Source doesn't need to know where the data is being synced to
        - Loose coupling
    - Interfaces with the source API and extracts the schema and then creates an external object (connect record = an object within the API) and passes it on

The applications advantages of using Kafka Connectors with data storage.
- Advantages include:
    - Flexibility
        - Decouples source and target
        - Changes in source/sink can be done without impacting the other
    - Scalability
        - Buffer for the data
        - When there is too much happening it provides a basic queuing functionality
    - Fault tolerance
        - Connection to sink/source may go down, with knowing that you are still producing data
            - Data is stored into kafka
    - Building pipelines
        - With the data in some exterior location and you want to through kafka to another location → transactional db to an object store

How do Kafka connectors maintain availability?
- Kafka connectors maintain availability by:
    - Kafka Connect → takes in large amounts of data from databases into a topic
        - Data then would become available towards stream processing

List the popular Kafka converters for values and the properties/advantages of each

- Converters receive connect records and turns them into bytes
    - Writes it as key and value into a kafka cluster
- Serializer in a regular kafka producer

- Popular converters
    - Json Schema
        - Advantages include that it generates clear and readable documentation
    - Avro
        - Advantages:
            - It's binary format
            - Fast → doesn't require code generation
            - Flexibility → has wide variety of programming languages
    - ProtoBuf
        - Advantages include
            - that it is faster, simpler and smaller
            - Has RPC support
            - Structure validation → predefined + larger structure

What's a Key-Value (KV) database?
- Stores messages as key and value pairs
    - Ability to store, retrieve and update data
- Non relational database

What are KV databases' advantages and disadvantages?
- Advantages
    - Scalable
        - Increase on database load/data has no negative impact on performance
        - Infinitely scalable horizontally
    - Speed/responsiveness
    - Reliability
    - Flexibility
        - DB can be easily relocated w/o change in structure
- Disadvantages
    - Single key value
    - No Query language
        - May not be able to import data into a different KV database
        - Not optimized for lookup
    - Values cannot be filtered

List some popular KV databases.
- Amazon DynamoDB
- Aerospike
- Redis

Resources

https://www.techtarget.com/searchdatamanagement/tip/NoSQL-database-types-explained-Key-value-store
https://www.freecodecamp.org/news/the-pros-and-cons-of-different-data-formats-key-values-vs-tuples-f526ad3fa964/
https://sakshichahal53.medium.com/json-vs-protocol-buffer-simplified-dbd6b69ca528