

# Architecture Decision Record: Choosing a Database Technology

---

## Status

Accepted

## Context

We are designing a new application that requires storing and retrieving data in a scalable and performant manner. We have identified three types of database technologies that are commonly used: relational databases, document databases, and event databases.

Relational databases store data in tables with fixed schemas and enforce strict data integrity constraints. They are suitable for applications that require complex data relationships and transactions. Examples include MySQL, PostgreSQL, and Oracle.

Document databases store data in JSON-like documents and are schema-less. They are well-suited for applications that require flexible data models and scaling horizontally. Examples include MongoDB, Couchbase, and Amazon DynamoDB.

Event databases store data as a series of events, capturing every change to the data. They are suitable for applications that require auditing, event sourcing, and complex data processing. Examples include Apache Kafka, Apache Pulsar, and AWS Kinesis. Decision

After carefully evaluating the requirements and constraints of our application, we have decided to use a document database.

## Rationale

We have chosen a document database because:

1. Our application requires a flexible data model that can evolve over time. Document databases allow us to store data in a schema-less format, which means we can add new fields or change the structure of existing documents without having to modify the database schema.
2. Our application needs to scale horizontally to handle large volumes of data and traffic. Document databases provide built-in support for sharding and replication, which enables us to distribute data across multiple servers and handle high read and write throughput.
3. Our application requires fast and efficient data retrieval. Document databases provide powerful indexing and querying capabilities that enable us to retrieve data quickly and efficiently.
4. Our application does not require complex transactions or data relationships. While relational databases excel at enforcing data integrity constraints and handling complex transactions, our application does not have such requirements. Document databases can provide sufficient consistency and durability guarantees for our use case.

## Consequences

By choosing a document database, we will need to invest in learning and understanding the specific technology we choose to use. Additionally, we will need to ensure that our application's data model fits well with the document database's data model to maximize performance and scalability.

However, we believe that the benefits of using a document database outweigh the costs, and that it is the best fit for our application's requirements and constraints.