

CouchDB Report

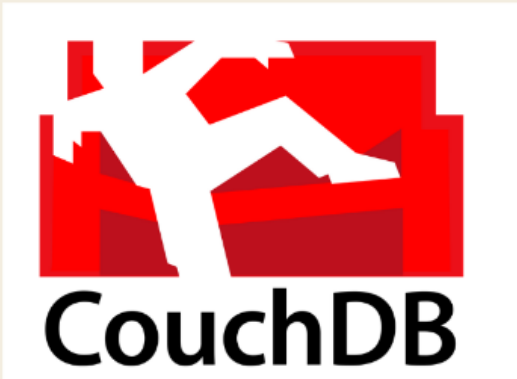
Group members

Dilip Patel

Shubham Golwal

Rupin Malik

Through our presentation, we covered multiple aspects of couchDB.



What is CouchDB?

Apache CouchDB™ is a database that uses JSON for documents, JavaScript for MapReduce indexes, and regular HTTP for its API

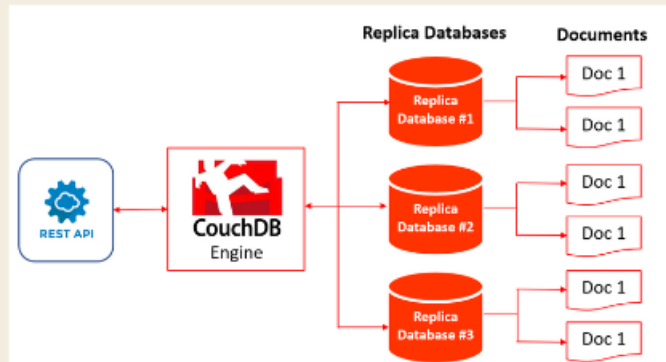
- Document database
- Embraces web approach
- Simple to use
- Written in Erlang [high concurrency & fault tolerance]
- Well documented

Introduction:

CouchDB is a document-oriented NoSQL database that uses a flexible JSON-based data model to store data. It allows users to store, retrieve, and manage semi-structured data, making it ideal for applications that require flexible data modeling. Its built-in replication feature allows data to be synchronized across multiple instances, enabling high availability and scalability. It is open-source and can be used on a variety of platforms, making it a popular choice for developers looking for a flexible and scalable database solution.

Architecture

1. A client makes an HTTP request to the server, which passes it through a set of modules that perform various functions such as authentication, request parsing, and query processing.
2. The server retrieves the data from the database, which is made up of a set of JSON documents. CouchDB automatically updates the views that have been defined for that database whenever a document is added, removed, or updated.
3. In a cluster environment, peer-to-peer replication is used to keep the data synchronized across all nodes in the cluster, ensuring that the client can retrieve the data from any server in the cluster if needed.



Architecture of CouchDB:

CouchDB follows a distributed architecture where data is stored in self-contained, self-describing documents. The database consists of multiple nodes, with each node being able to act as a master or a replica. CouchDB uses a replication protocol to synchronize data between nodes, ensuring high availability and fault tolerance. The database is accessed via HTTP, and supports RESTful API calls that can be used to perform CRUD operations and to create, modify, and delete MapReduce views. Finally, CouchDB has a pluggable storage architecture, which allows for different storage engines to be used depending on the application requirements.

Apache CouchDB Development Features



HTTP API

All items have a unique URI that gets exposed via HTTP. It uses the HTTP methods like POST, GET, PUT, and DELETE.



Built for Offline

It can replicate to devices like smartphones that have a feature to go offline and handle data sync for you when the device is back online.



Eventual Consistency

CouchDB guarantees Final consistency to be able to provide both availability and partition tolerance.



Document Storage

It is a NoSQL database that follows document storage where each field is uniquely named and contains values of various data types such as text, number, Boolean, lists, etc.



Reduce Views and Indexes

Each view in CouchDB is constructed by a JavaScript function that acts reduce operation and index views and keep those indexes updated as documents are added, removed, or updated.



ACID Semantics

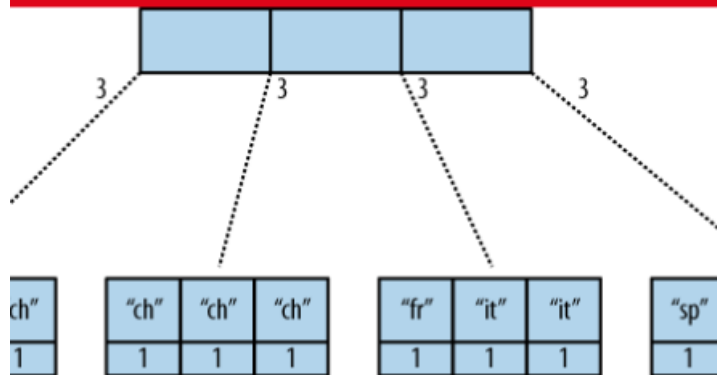
Implementing a form of Multi-Version Concurrency Control, meaning that CouchDB can handle a very high volume of concurrent writers and readers without conflict.

Features of CouchDB:

Features of CloudDB include:

1. RESTful HTTP API calls that can be used to perform CRUD operations on the database.
2. Built-in replication protocol that allows data to be synchronized across multiple instances, ensuring high availability and fault tolerance.
3. It offers Eventual Consistency in the Replicated DataBases.
4. Document-oriented data model based on JSON format that supports flexible and dynamic data schemas.
5. MapReduce views for querying and indexing data, enabling fast and efficient data analysis.
6. ACID-compliant transactions (from CouchDB version 3.0) that provide data consistency and integrity guarantees for critical applications.

Views in CouchDB



- Adds a structure to semi-structured data, Map Indices -> data
- With views you can
 - Filter documents
 - Extract data
 - Build efficient indices
- Views are built on-demand
- Multiple view allowed for one document

Views in CouchDB:

Views in CouchDB are used to query and index data, and are implemented using MapReduce. They allow users to define custom indexes over the data stored in the database and retrieve results in a highly efficient manner. Views can be used to perform complex queries, sorting and filtering operations, and to group and aggregate data based on specified criteria.

Literature Survey



Title: CouchDB: The Definitive Guide

By J. Anderson and N. Slater, O'Reilly Media Publication, 2010

- The paper is a guide to learning about CouchDB, published by O'Reilly Media in 2010.
- It covers CouchDB's unique features, including its flexible data model and querying capabilities using MapReduce views.
- The authors provide many examples of how to use MapReduce views to query and analyze data in CouchDB.
- The paper gives in-depth knowledge on CouchDB and its capabilities

CouchDB: The Reference guide:

This Research paper was published by J. Anderson and N. Slater in the year 2010. This research paper briefs about the characteristics and capabilities of CouchDB. It also has information about its data model. It also includes CouchDB's querying capabilities using MapReduce views to analyze the data in the DataBase.

Literature Survey



Title: Master-Master Replication with CouchDB

Authors: D. Boniface and R. McFarlane, 2012

- The paper describes the master-master replication feature in Apache CouchDB, a NoSQL database, and its benefits for distributed data management.
- It contains a case study of using CouchDB's master-master replication in a cloud computing environment, showing its effectiveness in handling large volumes of data with low latency and high throughput.
- The paper compares master-master replication with other replication techniques used in distributed databases.
- The authors provide experimental results that show the effectiveness of master-master replication in CouchDB for fault tolerance, scalability, and data consistency.

Master-Master Replication with CouchDB:

This Research paper was written by D. Boniface and R. McFarlane in the year 2012. This paper describes how the master-master replication is implemented in CouchDB. It also talks about its effectiveness in handling large amounts of data, with low-latency and high throughput. It also talks about the fault tolerance, scalability and data consistency in Master-Master replication.



The market share of CloudDB is about 1.87%. As we see the chart, most of the market is captured by MySQL and PostgreSQL.

Pros of Couch DB

Couch DB has a number of great features such as

- Data Replication**
Couch DB can easily handle data replication and synchronization across multiple servers
- Scalability**
Couch DB is scalable and can handle large amounts of data and can be scaled horizontally by adding more nodes to the cluster
- Offline Sync**
CouchDB has a built-in feature for offline data sync, which allows users to work with data even when they are not connected to the internet.

Cons of Couch DB

Couch DB also has some disadvantages, namely:



Steep Learning Curve

It may take some time for developers to get used to working with a schema-less database and using the MapReduce function for querying data.



Limited Queries

CouchDB has limited query capabilities since it does not support advanced query features like JOIN, GROUP BY, and HAVING.



Slow Execution

Couch DB is slow as other databases, especially when dealing with large datasets and complex queries. It is slower when performing operations on a large number of documents

Pros and Cons CouchDB:

CouchDB has several advantages, such as its flexible data model, which allows for dynamic schemas and easy scaling. It also has built-in replication and synchronization capabilities, making it a great choice for applications that require high availability and fault tolerance. The MapReduce views feature enables fast and efficient querying and analysis of data. Additionally, CouchDB has a RESTful API, which makes it easy to integrate with other applications and tools. On the downside, CouchDB has limited support for transactions and can consume significant disk space due to its append-only file storage model. It also has a steeper learning curve compared to traditional relational databases, and may require additional development effort to implement certain functionality.



SCALABILITY AND SECURITY OF COUCH DB

SCALABILITY

- Apache Couch DB is designed to be highly scalable, with support for distributed databases and replication.
- Apache Couch DB supports incremental replication, which allows data to be synchronized between different devices and servers in real-time.

SECURITY

- Apache Couch DB provides several security features like role-based access control, SSL encryption, and validation functions.
- It allows developers to control access to specific documents based on user roles

Scalability and Security of CouchDB:

CouchDB has good scalability features, allowing it to handle large amounts of data through its partitioning and replication capabilities. It also supports horizontal scaling by allowing new nodes to be added to a cluster dynamically. As for security, CouchDB provides several built-in security mechanisms, such as user authentication, data encryption, and role-based access control.

Backend Comparlson			
NoSQL Databases			
	MONGO DB	COUCH DB	FIREBASE
Data Model	Document - Oriented (BSON)	Document - Oriented (JSON)	JSON and BSON
Interface	Custom protocol over TCP/IP	HTTP/REST	HTTP/REST
Object Storage	Database contains Collections which contain Documents	Database contains Documents	Database contains Collections which contain Documents
Query Method	Map/Reduce, Object based	Map/Reduce, Range Queries	Object based and Range based
Replication	Master - Slave	Master - Master	Master - Master
Concurrency	In place	Multi-version Concurrency control	Multi-version Concurrency control

Comparison with MongoDB and Firebase:

CouchDB, Firebase, and MongoDB are all NoSQL databases with different strengths. CouchDB has a flexible data model and is well-suited for distributed environments. Firebase is a cloud-hosted platform with real-time data synchronization and built-in authentication, making it easy to develop and deploy mobile and web applications. MongoDB has a rich query language and strong community support, making it a popular choice for complex applications.

Conclusion:

- CouchDB has a RESTful HTTP API, which makes it easy to integrate with web applications and other services.
- While CouchDB may have a steeper learning curve compared to some other databases, it offers a lot of flexibility and customization options for developers.

- CouchDB is a NoSQL document-oriented database with a flexible data model and powerful querying capabilities using MapReduce views.
- It is designed for distributed environments and offers built-in partitioning and replication for easy scaling.
- CouchDB also has strong security features, including user authentication, data encryption, and role-based access control.
- Compared to other NoSQL databases like Firebase and MongoDB, CouchDB has unique strengths in its distributed architecture and query performance.