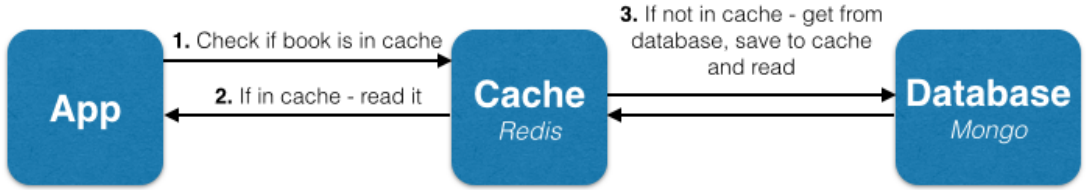
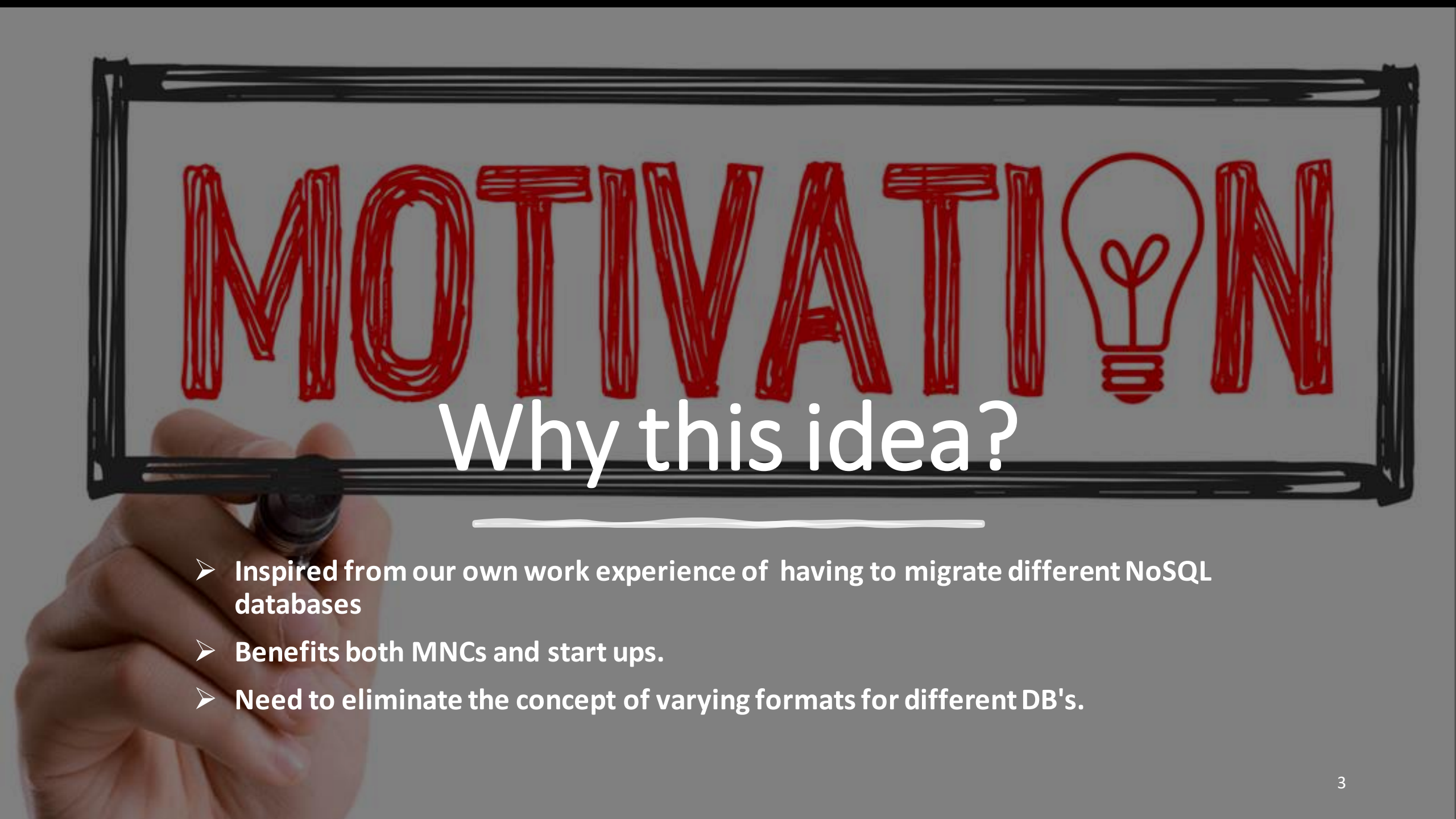

redis
CouchDB
relax
RAVENDB
Neo4j
Cassandra
riak
mongoDB
APACHE
HBASE
membase
amazon
DynamoDB

NoSQLite

Problem Statement

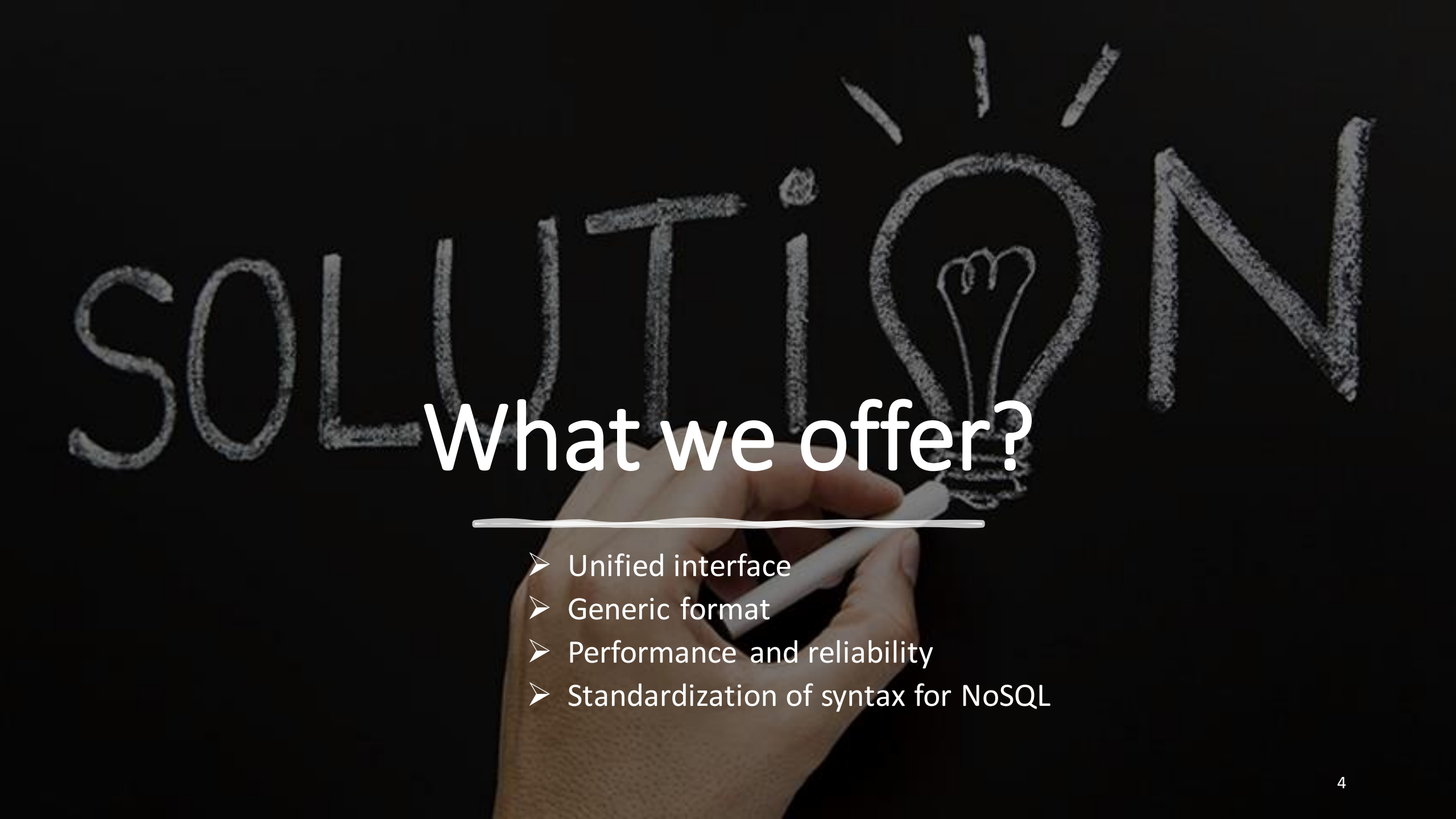


The goal is to create a database access library that provides the illusion of embedding NoSQL in regular C++ code. Unlike SQL, there is no specific defined format for NoSQL. The library will provide users a common format, unified interface, and a single point of access to multiple NoSQL databases.



Why this idea?

- Inspired from our own work experience of having to migrate different NoSQL databases
- Benefits both MNCs and start ups.
- Need to eliminate the concept of varying formats for different DB's.



What we offer?

- Unified interface
- Generic format
- Performance and reliability
- Standardization of syntax for NoSQL

Initial Plan

- Design a simple yet powerful interface
- Create a generic format and software design
- Integrate MongoDB and Redis
- Use existing performance-trained APIs for connections
- Target C++ users, with future scope to extend
- Analyze benefits and limitations of existing solutions

Market Analysis

S. No.	Software	Solution	Problem	Active	What we offer?
1.	UnQL	➢ Unified interface to all NoSQL DBs ➢ It focused on providing a very SQL DML-like interface.	Software failed due to their idea of providing a solution that “fits all situations.”	No	➢ Fit most of the situations while handling others smartly. ➢ Provide a generic format for similar commands and shall handle different ones efficiently as well.
2.	Eclipse JNoSQL	➢ Defines a set of APIs and provides a standard implementation for most NoSQL databases ➢ Bifurcates databases into key-value, column, document, and graph type	Implementations are specific to type of NoSQL DB. Solution is for JAVA programmers.	Yes	➢ Library with C++ compatibility ➢ Provide a unified interface without differentiating the DB types.

THANK YOU !