SQL Layer over a KV store

SQL layer is an interface that allows for SQL like querying on Key-Value data stores. It has several components, each of which will be developed independently, and will be totally pluggable

Parser

Convert raw SQL query into Abstract Syntax Tree (AST) representation
Start AST with three fields: type(operation types such as select, insert), columns(columns to be returned), values, from(table from which to fetch data), where and into

AST Traversal

Traverse AST nodes and extract relevant information, such as the required columns, filtering conditions, and sorting order.

Query Executor

Execute query based on the extracted information from AST.

In-memory KV Store

Store data in kv format, support fast lookups, inserts, and updates.

Indexing

To enable efficient querying, implement indexing on specific columns by creating additional data structures (e.g., B-trees or hash indexes) that map column values to corresponding keys in the KV store.

Query Optimizer

Analyze query and determine most efficient way to execute it, involves reordering JOIN operations, applying filtering conditions early, or utilizing indexes.

Result Set Formatter

Once the query has been executed, this component formats the results according to the requested output (e.g., JSON, CSV, or tabular format)