Summary On Facebook DBMS System TAO

* TAO(**“The Associations and Objects”**) is a heavily read-optimized data bank that skillfully provides fast and quick access to the social media graph to handle Facebook's workload applying least amount of queries possible.
* TAO uses a social graph to overcome some of the major drawbacks of using lookaside cache architecture (**Memcached**). TAO has better read latency, with the ability to process approximately a billion reads and millions of writes each second.
* Unlike lookaside cache key-value storage system, TAO is based completely on a huge social graph of nodes and edges. And all of it is stored in TAO’s directed social graph-cache.
* TAO acts as an efficient write-through cache. It aggressively caches the objects and associations to reduce latency and load on the database system.
* Before TAO, the application servers heavily relied on lookaside cache for the data being queried, and accessed the My SQL database for copying data that were absent in Memcached server storage. But since TAO, the mediation between the social graph and the main database has become easier.

Image for post

* TAO is the implementation of a huge graph that consists of nodes and edges. The Nodes represents objects such as users, places, posts, comments etc. and the associations between the nodes represent the user’s relationships, content origination and creation, the overall inter-relation between check-in to a location, the comments and reviews.
* TAO’s social graph that captures the activities of billion users, is a FB-specific **NoSQL graph API built to run on sharded MySQL**. It operates on the basis of two database tables, an OBJECT table and an ASSOCIATION table.
* The OBJECT table has a very simple schema consisting of **3 columns**. They are the **id column**( int type), **otype**[ object type (string)] and the **data column** which stores a serialized list of key-value pairs( of each object). The ASSOCIATION table has **4 columns** -------- **id1 and id2 columns** respectively represent the origin and target of an edge, the **atype column** represents the edge data type and lastly **data column** represents an optional list of possible key-value pairs in correspondance to an edge.

Eg. Object table data - *632 | “Post” | {“text”: “It was a great party! @ Bob”}*

Association table data - *632 | 731 | Comment | null*