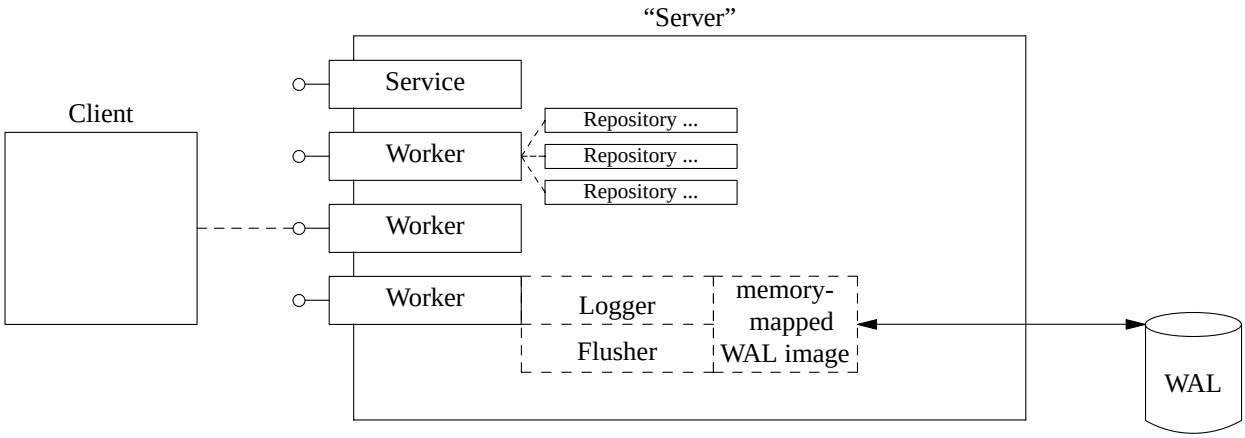
Logical	Physical
Topology	Hive
Host	Service
Pod	Worker
Column	Repository



Notes

Each worker is responsible for N column stores, each cached in a Repository.

Each worker uses a Wal object to capture transactions to a log. The Wal object updates the WAL as a memory-mapped file. On construction the Wal object creates a separate thread of control, known as the Flusher, and a pipe to communicate with it. The Flusher blocks against a read to the pipe.

When the client sends a Change set to the worker, the work calls Wal::Write, which reacts as follows:

1. Update the memory-mapped WAL image
2. Return OK to the worker

When the client sends a Flush request to the worker, the worker calls Wal::Write(), parks the connection, and returns. Wal::Write acts as follows:

1. Write a *sync* request to the Flusher pipe
2. Return OK to the worker
3. Flusher reads sync request
4. Flusher calls OS sync function e.g. **msync(2)** and blocks
5. Flusher sends Flush_OK to the worker's socket

On receipt of Flush_OK, the worker unparks the connection and returns OK to the Client.

The Flusher is required to sync the memory-mapped image periodically, perhaps every 500 milliseconds. (Tunable. Another option might be every N pages or 30 seconds, whichever came first.) The is effected by a timeout on reading the pipe using e.g. **poll(2)**. On timeout, the Flusher reacts as to a Flush request, except that no notification is sent to the worker.