

What is Bookkeeper?

Bookkeeper is a log replication service which can be used to build replicated state machines. A log contains a sequence of events which can be applied to a state machine. Bookkeeper guarantees that each replica state machine will see all the same entries, in the same order.

Eh? What good is that to me?

Imagine for example that you have a database that you want to be able access even if the database server goes down. You'll need to replicate it to multiple servers. You need to ensure that if one database sees an update, all databases see the update. But what happens if one database server is cut off from the network for a time? Or if two clients try to update the same field at exactly the same instance? This is where log replication comes in.

A database can be seen as a state machine. It is the sum of all the updates which has applied since its initial state. Therefore, if you consider your replicated database as a replicated statemachine, you can do the replication using log replication service. If all updates are written to the log replication service before being applied to the database, then the database will continue to be available and consistent even if some of the replicas fail.

This approach can be applied to many types of distributed systems, such as messaging systems, coordination systems, filesystems, etc.

What bookkeeper is not?

Bookkeeper has nothing to do with application/error/trace logging. There are already many projects (link to log4j, slf4j, logback) dedicated to that problem.

How about Hedwig?

Hedwig is a distributed publish and subscribe system, which uses bookkeeper to replicate its messages.

