

Lecture 17 - Hash Based Databases v.s. SQLite

SQLite3 v.s. Hash Store - <https://youtu.be/0GG2s9uz8BY>

get-issue-detail walk through - <https://youtu.be/lfCNVEJEOxQ>

From Amazon S3 - for download (same as youtube videos)

SQLite3 v.s. Hash Store

get-issue-detail walk through

The stated reason for using a "hash" database like "bitcoin" uses is that you don't have an external database that requires setup and maintenance.

Bitcoin uses LevelDB a hash-file store.

SQLite is a SQL database that stores its data in a single file. All of you are already users of SQLite - it is builtin to Google Chrome, Firefox, Microsoft Edge and Apple Safari. This is how your bookmarks are stored (And a lot of other stuff).

SQLite is designed to be compiled right into an application. So if you need to ship a "demo" of a SQL tool with the data and you don't want to force the demo-user to setup an entire database - this is the tool for you.

It is reasonably fast - I have run websites with 100's of requests a minute on it.

It works at scale for data. You can load gigabyte size data. Inside a single application it is ACID compliant - even with multiple threads.

Let's take a look at performance.

Database/Rows	10	100	1000	10000
SQLite3	8	47	391	3141
LevelDb	5	52	431	5210
PostgreSQL	19	61	532	4328

Known "issue" or problems that I have encountered.

1. Inserting a NULL row will not do an insert.
2. Porting data files between big-ndian and little-ndian systems is "iffy".
3. Performance is not the same as 50,000 TPS from PostgreSQL - probably more on the 500-1000 TPS range.