

Tokyo Cabinet

Very fast database

Simple library
(process embedded)

Three flavors:

- Key-value (hash/btree)
- Fixed-length (array)
- Table (schema-free)

Key-value (hash/btree)

- ✦ Hash table or B+ tree
- ✦ Hash tables = fast writes, good over-all performance
- ✦ B+ tree = slow writes, really fast chunk reads

Fixed-length (array)

- ✦ Values have a fixed width (maximum length)
- ✦ Keys are serial (natural) numbers
- ✦ The fastest possible key-value store
- ✦ Only usable for certain types of applications

Table (schema-free)

- ✦ Records must have primary keys
- ✦ Schema free = versioning is somewhat painless
- ✦ Extensible
- ✦ Slower than key-value and array engines

Two layers:

- Tokyo Cabinet (engine)
- Tokyo Tyrant (server)

(Actually three components, if you count the Dystopia search engine)

Learn more

- ✦ <http://tokyocabinet.sourceforge.net/index.html>
- ✦ <http://blog.hunch.se/archive/2009/02/28/tokyo-cabinet>
- ✦ <http://blog.hunch.se/archive/2009/03/02/tokyo-cabinet-python-bindings>