Per.10: Rely on the static type system

Дмитрий Шебордаев, Ведущий специалист

OTUS

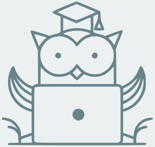
ОНЛАЙН-ОБРАЗОВАНИЕ

Хранилища

Key Value

Дмитрий Шебордаев

Ведущий специалист



Сегодня

- Репликация
- Курсоры
- NoSQL
- MongoDB
- Aerospike/Redis
- MongoDB

MongoDB / Create a Document

```
auto builder = bsoncxx::builder::stream::document{};
bsoncxx::document::value doc_value = builder
  << ”name” << ”MongoDB”
  << ”type” << ”database”
  << ”count” << 1
  << ”versions” << bsoncxx::builder::stream::open_array
    << ”v3.2” << ”v3.0” << ”v2.6”
  << close_array
  << ”info” << bsoncxx::builder::stream::open_document
    << ”x” << 203
    << ”y” << 102
  << bsoncxx::builder::stream::close_document
  << bsoncxx::builder::stream::finalize;

bsoncxx::stdx::optional<mongocxx::result::insert_one> result =
 restaurants.insert_one(value);
```

MongoDB / Query

```
mongocxx::cursor cursor = collection.find(
  document{} << ”i” << open_document <<
    ”$gt” << 50 <<
    ”$lte” << 100
  << close_document << finalize);
for(auto doc : cursor) {
  std::cout << bsoncxx::to_json(doc) << ”\n”;
}
```

Redis / Commads

- Geo
- Hashes
- HyperLogLog
- Lists
- Pub/Sub
- Scripting
- Sets
- Sorted Sets
- Strings
- Transactions

Redis / Simple pipeline

```
redisReply *reply;
redisAppendCommand(context,”SET␣foo␣bar”);
redisAppendCommand(context,”GET␣foo”);
redisGetReply(context,&reply); // reply for SET
freeReplyObject(reply);
redisGetReply(context,&reply); // reply for GET
freeReplyObject(reply);
```

Aerospike / C client

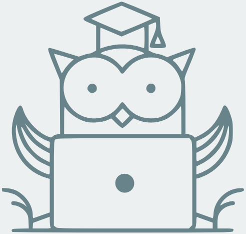
```
as_config_init(&config);

config.hosts[0] = { .addr = ”127.0.0.1”, .port = 3000 };
aerospike_init(&as, &config);
aerospike_connect(&as, &err);

as_key_init(&key, ”test”, ”demo_set”, ”test_key”);

as_record_inita(&rec, 2);
as_record_set_int64(&rec, ”test−bin−1”, 1234);
as_record_set_str(&rec, ”test−bin−2”, ”test−bin−2−data”);

aerospike_key_put(&as, &err, NULL, &key, &rec);

aerospike_close(&as);
aerospike_destroy(&as);
```

LevelDB / Atomic Updates

```
#include ”leveldb/write_batch.h”
...
std::string value;
leveldb::Status s = db->Get(leveldb::ReadOptions(), key1, &valu
if (s.ok()) {
  leveldb::WriteBatch batch;
  batch.Delete(key1);
  batch.Put(key2, value);
  s = db->Write(leveldb::WriteOptions(), &batch);
}
```

Полистать

- https://mongodb.github.io/mongo-cxx-driver/mongocxx-v3/tutorial/
- https://www.aerospike.com/docs/client/c/index.html
- https://github.com/redis/hiredis
- https://github.com/google/leveldb/blob/master/doc/index.md



Спасибо за внимание!