

RocksDB - BlobDB

BlobDB là một mô-đun bổ sung cho RocksDB, được thiết kế đặc biệt để lưu trữ và quản lý dữ liệu lớn không cấu trúc như văn bản, hình ảnh, âm thanh, video và các đối tượng nhị phân khác. Nó cung cấp khả năng lưu trữ hiệu quả dữ liệu lớn và quản lý metadata liên quan đến dữ liệu.

BlobDB sử dụng một cấu trúc dữ liệu gọi là "blob" để lưu trữ dữ liệu không cấu trúc. Mỗi blob là một đơn vị lưu trữ độc lập, có thể chứa một hoặc nhiều đối tượng nhị phân. BlobDB sử dụng RocksDB để lưu trữ metadata của các blob, bao gồm thông tin như kích thước, thời gian tạo và các thuộc tính tùy chỉnh khác.

BlobDB cung cấp các API để thêm, truy xuất và xóa blob, cho phép bạn làm việc với dữ liệu không cấu trúc một cách dễ dàng và hiệu quả. Dưới đây là một ví dụ về việc sử dụng BlobDB trong RocksDB bằng Java:

```
import org.rocksdb.*;

public class RocksDBBlobDBExample {

    public static void main(String[] args) {
        RocksDB.loadLibrary();
        try (final Options options = new Options().setCreateIfMissing(true);
            final RocksDB db = RocksDB.open(options, "/path/to/database")) {

            // Khởi tạo BlobDB
            try (final BlobDB blobDB = BlobDB.open(db)) {
                // Thêm dữ liệu vào BlobDB
                byte[] blobId = blobDB.put("Hello, BlobDB!".getBytes());
                System.out.println("Blob ID: " + new String(blobId));

                // Truy xuất dữ liệu từ BlobDB
                byte[] data = blobDB.get(blobId);
                System.out.println("Data: " + new String(data));

                // Xóa dữ liệu từ BlobDB
                blobDB.delete(blobId);
            } catch (RocksDBException e) {
                // Xử lý lỗi khi thao tác với BlobDB không thành công
            }
        }
    }
}
```

```

    } catch (RocksDBException e) {
        // Xử lý lỗi khi thao tác với cơ sở dữ liệu RocksDB không thành công
    }
}
}

```

Trong ví dụ trên, chúng ta sử dụng `BlobDB.open()` để khởi tạo một phiên làm việc với BlobDB. Sau đó, chúng ta có thể sử dụng `put()` để thêm dữ liệu vào BlobDB, `get()` để truy xuất dữ liệu từ BlobDB và `delete()` để xóa dữ liệu từ BlobDB.

Đây là một ví dụ về việc sử dụng BlobDB trong RocksDB bằng C++:

```

#include "rocksdb/db.h"
#include "rocksdb/blob_db.h"

int main() {
    rocksdb::DB* db;
    rocksdb::Options options;
    options.create_if_missing = true;
    rocksdb::Status status = rocksdb::DB::Open(options, "/path/to/database", &db);
    if (status.ok()) {
        // Khởi tạo BlobDB
        rocksdb::BlobDB* blobDB;
        rocksdb::BlobDBOptions blobDBOptions;
        rocksdb::Status blobDBStatus = rocksdb::BlobDB::Open(blobDBOptions, db, &blobDB);
        if (blobDBStatus.ok()) {
            // Thêm dữ liệu vào BlobDB
            rocksdb::BlobDB::PutOptions putOptions;
            rocksdb::Slice data = "Hello, BlobDB!";
            rocksdb::BlobIndex blobIndex;
            rocksdb::Status putStatus = blobDB->Put(putOptions, data, &blobIndex);
            if (putStatus.ok()) {
                std::cout << "Blob ID: " << blobIndex.blob_handle.ToString() << std::endl;

                // Truy xuất dữ liệu từ BlobDB
                rocksdb::BlobDB::GetOptions getOptions;
                rocksdb::PinnableSlice readValue;
                rocksdb::Status getStatus = blobDB->Get(getOptions, blobIndex.blob_handle,
&readValue);

                if (getStatus.ok()) {

```

```

        std::cout << "Data: " << readValue.ToString() << std::endl;

        // Xóa dữ liệu từ BlobDB
        rocksdb::BlobDB::DeleteOptions deleteOptions;
        rocksdb::Status deleteStatus = blobDB->Delete(deleteOptions,
blobIndex.blob_handle);
        if (!deleteStatus.ok()) {
            // Xảy lỗi khi xóa dữ liệu không thành công
        }
    } else {
        // Xảy lỗi khi truy xuất dữ liệu không thành công
    }
} else {
    // Xảy lỗi khi thêm dữ liệu không thành công
}
} else {
    // Xảy lỗi khi khởi tạo BlobDB không thành công
}
} else {
    // Xảy lỗi khi khởi tạo RocksDB không thành công
}
return 0;
}

```

Revision #1

Created 23 September 2023 10:30:52 by Laptrinh.vn

Updated 23 September 2023 10:56:18 by Laptrinh.vn