

<epam>

Reverse Engineering

Relational Databases Basics



TRAINING
CENTER

— <epam> —

Database Reverse Engineering – a process of database investigation with the aim of schema creation or implementation details gathering.

When and why do we need reverse engineering?

We are catching up a project from a previous contractor

We need to improve something, but the documentation is not enough to understand exact actions

We are analyzing some old project to implement the same functionality in a new one

...

The main objective

We have to collect technical details on existing database.

Ideas to consider

Physical level

Datological level

Infological level

Settings, diagnostics, debug info, monitoring, and so on.

The same **specialized** tools we used for forward engineering.

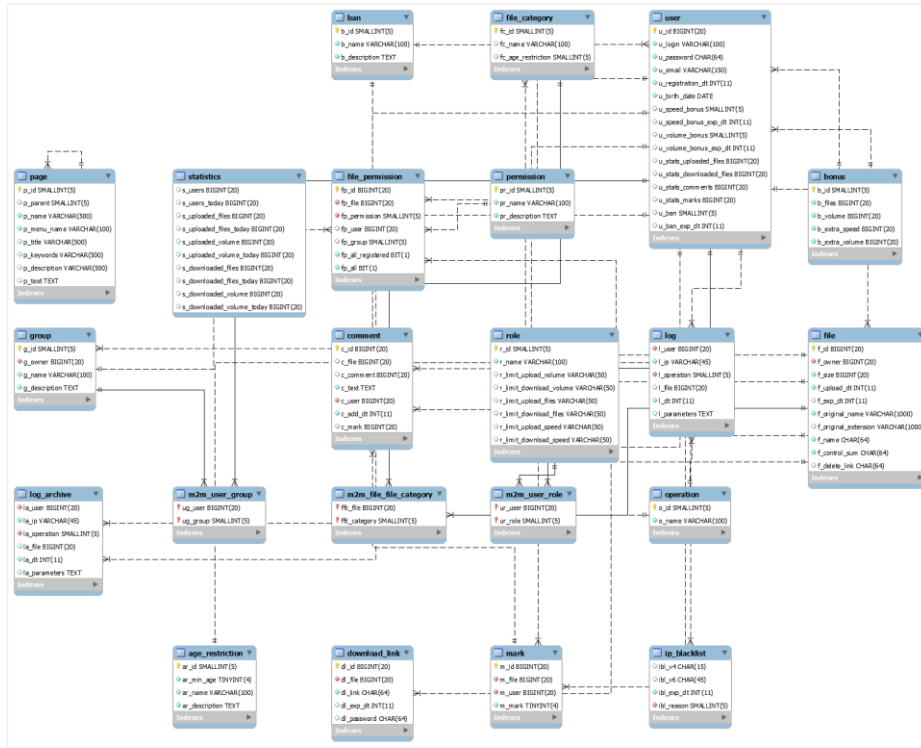
If we are here, it means that we literally have to start the modelling process from the beginning.

Just a terrifying sample...



A simple database (about 100 tables) schema (acquired using DBeaver).

A little less terrifying sample...



This is our “File Exchange” service database imported into MySQL Workbench. We can see all tables, indexes, views, stored procedures and so on...

```
Name: SET_ENCODING_AND_STORAGE_ENGINE_TO_ALL_TABLES
DDL:
1 CREATE PROCEDURE `SET_ENCODING_AND_STORAGE_ENGINE_TO_ALL_TABLES`
2     IN collation_name VARCHAR(150),
3     IN storage_engine VARCHAR(150))
4 BEGIN
5     DECLARE done INT DEFAULT 0;
6     DECLARE tbl_name VARCHAR(200) DEFAULT '';
7     DECLARE all_tables_cursor CURSOR FOR
8     SELECT `table_name`
9     FROM `information_schema`.`tables`
```

Quick live demo...

This is how to perform the reverse engineering of the whole database...

How to use SQL for reverse engineering

```
-- Get stored procedures list:
SHOW PROCEDURE STATUS WHERE `db` = DATABASE()

-- Get tables list:
SELECT `table_name`
  FROM `information_schema`.`tables`
 WHERE `table_schema` = DATABASE()
       AND `table_type` = 'BASE TABLE'

-- Get stored procedure source code:
SHOW CREATE PROCEDURE `SET_ENCODING_AND_STORAGE_ENGINE_TO_ALL_TABLES`

-- Get table source code:
SHOW CREATE TABLE `file`

-- Get table indexes:
SHOW INDEX FROM `file`
```

P.S.

Unlike the forward engineering, the reverse one usually has too specific goals to achieve them with “general approaches”. So, read manuals/books/etc carefully.

<epam>

Reverse Engineering

Relational Databases Basics



TRAINING
CENTER

— <epam> —