MySQL View Processing Algorithms

Summary: in this tutorial, you will learn about MySQL view processing algorithms including `MERGE`, `TEMPTABLE`, and `UNDEFINED`.

The `CREATE VIEW` and `ALTER VIEW` statements have an optional clause: `ALGORITHM`. The algorithm determines how MySQL process a view and can take one of three values `MERGE`, `TEMPTABLE`, and `UNDEFINE`.

Here is the `CREATE VIEW` statement with the `ALGORITHM` clause:

```
CREATE [OR REPLACE][ALGORITHM = {MERGE | TEMPTABLE | UNDEFINED}] VIEW
   view_name[(column_list)]
AS
   select-statement;
```

And this is the `ALTER VIEW` statement with the `ALGORITHM` clause:

```
CREATE [ALGORITHM = {MERGE | TEMPTABLE | UNDEFINED}] VIEW
   view_name[(column_list)]
AS
   select-statement;
```

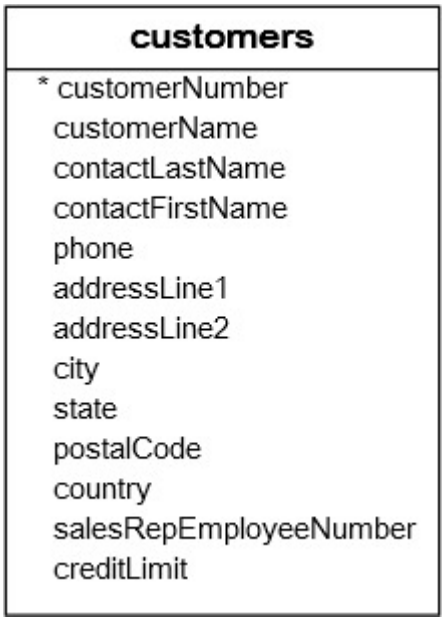
MERGE

When you query from a `MERGE` view, MySQL processes the following steps:

- First, merge the input query with the `SELECT` statement in the view definition into a single query.
- Then, execute the combined query to return the result set.

Note that the combination of input query and the `SELECT` statement of the view definition into a single query is referred to as *view resolution*.

See the following `customers` from the sample database:



The following statement creates a view based on the `customers` table with the name `contactPersons` with the `MERGE` algorithm:

```
CREATE ALGORITHM=MERGE VIEW contactPersons(
    customerName,
    firstName,
    lastName,
    phone
) AS
SELECT
    customerName,
    contactFirstName,
    contactLastName,
```

```
    phone
FROM customers;
```

Suppose that you issue the following statement:

```
SELECT * FROM contactPersons
WHERE customerName LIKE '%Co%';
```

MySQL performs these steps:

- Convert view name `contactPersons` to table name `customers`.
- Convert askterisk (*) to a list column names `customerName`, `firstName`, `lastName`, `phone`, which corresponds to `customerName`, `contactFirstName`, `contactLastName`, `phone`.
- Add the `WHERE` clause.

The resulting statement is:

```
SELECT
    customerName,
    contactFirstName,
    contactLastName,
    phone
FROM
    customers
WHERE
    customerName LIKE '%Co%';
```

TEMPTABLE

When you issue a query to a `TEMPTABLE` view, MySQL performs these steps:

- First, create a temporary table to store the result of the `SELECT` in the view definition.
- Then, execute the input query against the temporary table.

Because MySQL has to create the temporary table to store the result set and moves the data from the base tables to the temporary table, the algorithm `TEMPTABLE` is less efficient than the

`MERGE` algorithm.

Note that `TEMPTABLE` views cannot be updatable.

UNDEFINED

The `UNDEFINED` is the default algorithm when you create a view without specifying the `ALGORITHM` clause or you explicitly specify `ALGORITHM=UNDEFINED`.

In addition, when you create a view with `ALGORITHM = MERGE` and MySQL can only process the view with a temporary table, MySQL automatically sets the algorithm to `UNDEFINED` and generates a warning.

The `UNDEFINED` allows MySQL to choose either `MERGE` or `TEMPTABLE`. And MySQL prefers `MERGE` over `TEMPTABLE` if possible because `MERGE` is often more efficient than `TEMPTABLE`.

In this tutorial, you have learned about the MySQL view processing algorithms including `MERGE`, `TEMPTABLE`, and `UNDEFINED`.

PREVIOUSLY

MySQL CREATE VIEW

UP NEXT

MySQL ALTER VIEW

Search ...

MYSQL VIEWS

MySQL CREATE VIEW

MySQL View Processing Algorithms

Create MySQL Updatable Views

Rename View

Show View

SHOW CREATE VIEW

DROP VIEW

WITH CHECK OPTION

LOCAL & CASCADED

ABOUT MYSQL TUTORIAL WEBSITE

MySQLTutorial.org helps you master MySQL quickly, easily, and with enjoyment. Our tutorials make learning MySQL a breeze.

All MySQL tutorials are clear, practical and easy-to-follow. More About Us

LATEST TUTORIALS

MySQL Port

MySQL Commands

innodb_dedicated_server: Configure InnoDB Dedicated Server

innodb_flush_method: Configure InnoDB Flush Method

innodb_log_buffer_size:

SITE LINKS

Donation ❤️

Contact Us

About

Privacy Policy

OTHERS

MySQL Cheat Sheet

MySQL Resources

Configure InnoDB Log Buffer Size

innodb_buffer_pool_chunk_size: Configure Buffer Pool Chunk Size

innodb_buffer_pool_instances: Configuring Multiple Buffer Pool Instances for Improved Concurrency in MySQL

innodb_buffer_pool_size: Configure InnoDB Buffer Pool Size

MySQL InnoDB Architecture

How to Kill a Process in MySQL

MySQL Books