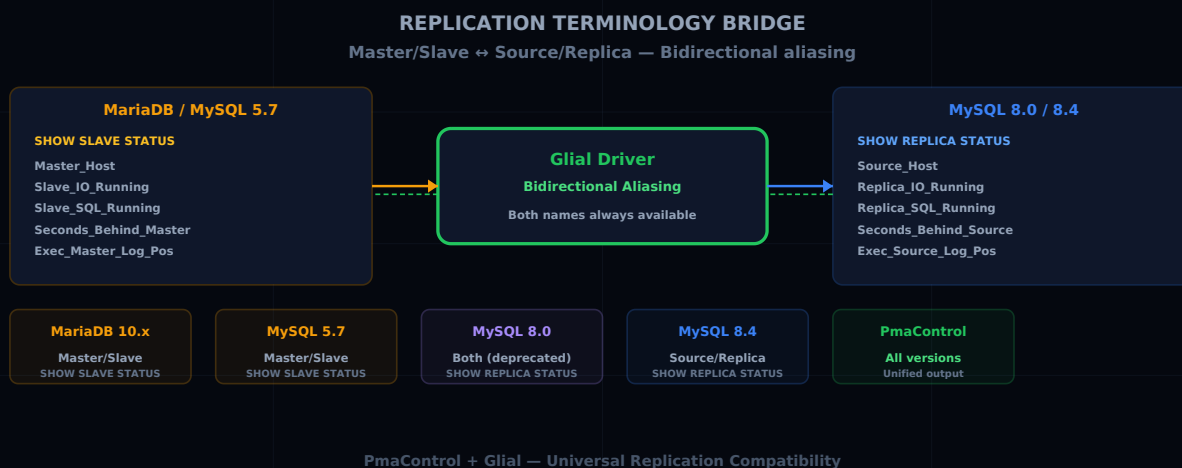


MySQL 8.0 репликация: совместимость Source/Replica с Master/Slave

Aurélien LEQUOY · April 8, 2026

MYSQL REPLICATION COMPATIBILITY MYSQL-8.0 MIGRATION



Переименование

MySQL 8.0.22 представил крупное терминологическое изменение: термины **Master** и **Slave** в командах и столбцах вывода репликации были заменены на **Source** и **Replica**.

Это не просто косметическое изменение. Конкретно:

- `SHOW SLAVE STATUS` становится `SHOW REPLICATION STATUS`
- `CHANGE MASTER TO` становится `CHANGE REPLICATION SOURCE TO`
- Столбец `Master_Host` в выводе становится `Source_Host`
- `Slave_IO_Running` становится `Replica_IO_Running`
- `Slave_SQL_Running` становится `Replica_SQL_Running`
- `Seconds_Behind_Master` становится `Seconds_Behind_Source`

И так далее для примерно тридцати столбцов.

MySQL 8.0 поддерживает совместимость: старые команды по-прежнему работают (с предупреждением об устаревании). Но MySQL 8.4 начинает удалять старые алиасы. А

MariaDB, со своей стороны, сохраняет историческую терминологию.

Проблема для инструментов мониторинга

Любой инструмент, который парсит вывод `SHOW SLAVE STATUS` / `SHOW REPLICATION STATUS`, теперь должен обрабатывать **два набора имён столбцов** в зависимости от версии сервера:

Старое (MariaDB, MySQL 5.7)	Новое (MySQL 8.0+)
Master_Host	Source_Host
Master_User	Source_User
Master_Port	Source_Port
Master_Log_File	Source_Log_File
Read_Master_Log_Pos	Read_Source_Log_Pos
Slave_IO_Running	Replica_IO_Running
Slave_SQL_Running	Replica_SQL_Running
Slave_IO_State	Replica_IO_State
Seconds_Behind_Master	Seconds_Behind_Source
Last_IO_Error	Last_IO_Error
Last_SQL_Error	Last_SQL_Error
Exec_Master_Log_Pos	Exec_Source_Log_Pos

Проблема двойная:

1. **Существующий код** использует старые имена столбцов повсюду. Сотни ссылок на `$row['Master_Host']` или `$row['Slave_IO_Running']`.
2. **Инфраструктура смешанная**: MariaDB 10.6 и 10.11, MySQL 5.7 на исходе жизни, MySQL 8.0, только что развёрнутый MySQL 8.4. Все в одном PmaControl.

Наивное решение (и почему оно не работает)

Первая идея: определять версию и использовать нужное имя столбца.

```
// Не делайте так
if ($version >= '8.0.22') {
    $host = $row['Source_Host'];
    $io_running = $row['Replica_IO_Running'];
} else {
    $host = $row['Master_Host'];
    $io_running = $row['Slave_IO_Running'];
}
```

Этот паттерн катастрофичен в обслуживании. Каждое место в коде, читающее поле репликации, должно быть продублировано. С 30 столбцами и десятками мест в коде получаются сотни условий.

А когда MySQL 9.0 добавит новые алиасы? Утроить код?

Подход Glial: двунаправленный алиасинг на уровне драйвера

Решение, реализованное во фреймворке Glial (обеспечивающем слой доступа к MySQL в PmaControl), более элегантно: **двунаправленный алиасинг на уровне драйвера**.

Когда драйвер Glial выполняет `SHOW REPLICA STATUS` (или `SHOW SLAVE STATUS`), он обогащает результат, добавляя **оба имени** для каждого столбца:

```
// В MySQL-драйвере Glial
$replication_aliases = [
    'Master_Host'      => 'Source_Host',
    'Master_User'     => 'Source_User',
    'Master_Port'     => 'Source_Port',
    'Master_Log_File' => 'Source_Log_File',
    'Read_Master_Log_Pos' => 'Read_Source_Log_Pos',
    'Slave_IO_Running' => 'Replica_IO_Running',
    'Slave_SQL_Running' => 'Replica_SQL_Running',
    'Slave_IO_State'  => 'Replica_IO_State',
    'Seconds_Behind_Master' => 'Seconds_Behind_Source',
    'Exec_Master_Log_Pos' => 'Exec_Source_Log_Pos',
    // ... все пары
];
```

```
// После получения результата
foreach ($replication_aliases as $old => $new) {
    if (isset($row[$old]) && !isset($row[$new])) {
        $row[$new] = $row[$old];
    }
    if (isset($row[$new]) && !isset($row[$old])) {
        $row[$old] = $row[$new];
    }
}
```

Результат: **потребляющий код может использовать любое имя — старое или новое.** Независимо от того, является ли сервер MariaDB 10.6, MySQL 5.7, MySQL 8.0 или MySQL 8.4, оба имени всегда присутствуют в результате.

Преимущества этого подхода

1. Нулевые изменения существующего кода

Сотни ссылок на `$row['Master_Host']` в PmaControl продолжают работать. Никакой миграции кода не требуется.

2. Новый код может использовать новую терминологию

Разработчики, пишущие новый код, могут использовать `$row['Source_Host']` прямо сейчас. Когда старый код будет постепенно рефакторирован, переход будет прозрачным.

3. Единая точка обслуживания

Алиасинг определяется один раз, в драйвере. Если MySQL 9.0 добавит новые поля или переименует другие столбцы, достаточно расширить таблицу алиасов.

4. Совместимость с SHOW SLAVE STATUS и SHOW REPLICA STATUS

Драйвер сначала пробует `SHOW REPLICA STATUS`. Если команда не выполняется (MySQL 5.7, MariaDB), он делает fallback на `SHOW SLAVE STATUS`. В обоих случаях результат содержит оба набора имён.

```
try {
    $result = $db->query('SHOW REPLICA STATUS');
```

```
} catch (Exception $e) {
    $result = $db->query('SHOW SLAVE STATUS');
}
// В любом случае $result содержит Master_Host И Source_Host
```

Случай multi-source

MySQL 8.0 поддерживает multi-source репликацию (несколько каналов). Команда `SHOW REPLICA STATUS` тогда возвращает **несколько строк**, по одной на канал. Алиасинг применяется к каждой строке индивидуально.

MariaDB тоже поддерживает multi-source репликацию, но с другим синтаксисом (`SHOW SLAVE 'channel_name' STATUS` или `SHOW ALL SLAVES STATUS`). Драйвер Glial нормализует эти различия, чтобы PmaControl получал единый формат.

Влияние на PmaControl

Благодаря алиасингу в драйвере Glial, PmaControl прозрачно поддерживает:

- **MariaDB 10.6 / 10.11:** терминология Master/Slave, `SHOW SLAVE STATUS`
- **MySQL 5.7:** терминология Master/Slave, `SHOW SLAVE STATUS`
- **MySQL 8.0:** терминология Source/Replica, `SHOW REPLICA STATUS` (со старым fallback)
- **MySQL 8.4:** только терминология Source/Replica, `SHOW REPLICA STATUS`

Интерфейс PmaControl отображает информацию о репликации единообразно, независимо от версии сервера. Дашборд репликации показывает `Source_Host` в столбцах, но внутренние запросы работают с обоими именами.

Команды тоже

Переименование касается не только столбцов вывода. Сами SQL-команды тоже были переименованы:

Старое	Новое
<code>CHANGE MASTER TO</code>	<code>CHANGE REPLICATION SOURCE TO</code>
<code>START SLAVE</code>	<code>START REPLICA</code>

Старое	Новое
<code>STOP SLAVE</code>	<code>STOP REPLICATION</code>
<code>RESET SLAVE</code>	<code>RESET REPLICATION</code>
<code>SHOW SLAVE HOSTS</code>	<code>SHOW REPLICAS</code>

PmaControl использует тот же подход: пытается новую команду, fallback на старую.

Рекомендации для ваших собственных инструментов

Если вы поддерживаете скрипты или инструменты, парсящие вывод репликации:

1. **Не используйте условия на версию** — это хрупко и не масштабируется
2. **Реализуйте двунаправленный алиасинг** на максимально низком уровне (драйвер, слой абстракции)
3. **Пишите новый код с новой терминологией** — Source/Replica
4. **Тестируйте на 4 основных версиях:** MariaDB 10.x, MySQL 5.7, MySQL 8.0, MySQL 8.4
5. **Готовьтесь к MySQL 9.x** — старые алиасы могут полностью исчезнуть

Заключение

Переименование Master/Slave в Source/Replica в MySQL 8.0 — это изменение, простое на вид, но потенциально ломающее любой инструмент мониторинга, резервного копирования или оркестрации, который парсит вывод репликации.

Чистое решение — двунаправленный алиасинг на уровне драйвера: одна реализация, ноль изменений в потребляющем коде, полная совместимость от MariaDB 10.x до MySQL 8.4.

Именно так работает фреймворк Glial, и именно это позволяет PmaControl контролировать смешанные инфраструктуры MariaDB / MySQL без специальной конфигурации для каждой версии.