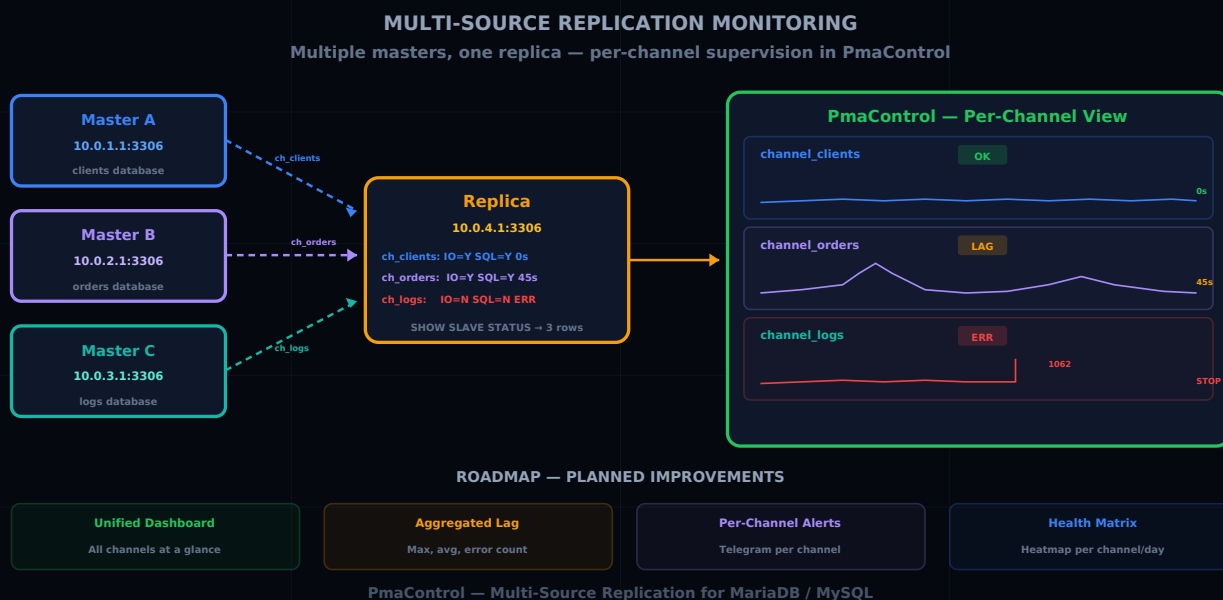


Мониторинг multi-source репликации в PmaControl

Aurélien LEQUOY · April 13, 2026

MARIADB MYSQL MULTI-SOURCE REPLICATION MONITORING PMACONTROL



Multi-source: несколько мастеров, одна реплика

Multi-source репликация позволяет единственному серверу-реплике получать данные от **нескольких мастеров одновременно**. Каждое подключение к мастеру — это отдельный **канал** (channel), со своим IO-поток, своим SQL-поток и своим состоянием репликации.

Это мощный паттерн для:

- **Консолидации**: агрегирование данных нескольких бизнес-баз на аналитической реплике
- **Миграции**: получение данных из старой и новой системы во время перехода
- **Data warehouse**: наполнение хранилища из нескольких источников MariaDB / MySQL

В нашей предыдущей статье о multi-source репликации MySQL 8.4 мы рассмотрели конфигурацию. Здесь мы сосредоточимся на мониторинге: как PmaControl обнаруживает, отображает и контролирует эти множественные каналы.

Быстрая настройка (напоминание)

MariaDB

MariaDB нативно поддерживает multi-source начиная с версии 10.0:

```
-- Канал 1: база клиентов
CHANGE MASTER 'channel_clients' TO
  MASTER_HOST = '10.0.1.1',
  MASTER_USER = 'repl',
  MASTER_PASSWORD = 'secret',
  MASTER_USE_GTID = slave_pos;
START SLAVE 'channel_clients';

-- Канал 2: база заказов
CHANGE MASTER 'channel_orders' TO
  MASTER_HOST = '10.0.2.1',
  MASTER_USER = 'repl',
  MASTER_PASSWORD = 'secret',
  MASTER_USE_GTID = slave_pos;
START SLAVE 'channel_orders';
```

MySQL 8.0+

MySQL использует синтаксис `FOR CHANNEL` :

```
-- Канал 1
CHANGE REPLICATION SOURCE TO
  SOURCE_HOST = '10.0.1.1',
  SOURCE_USER = 'repl',
  SOURCE_PASSWORD = 'secret',
  SOURCE_AUTO_POSITION = 1
  FOR CHANNEL 'channel_clients';
START REPLICATION FOR CHANNEL 'channel_clients';

-- Канал 2
CHANGE REPLICATION SOURCE TO
  SOURCE_HOST = '10.0.2.1',
  SOURCE_USER = 'repl',
  SOURCE_PASSWORD = 'secret',
```

```
SOURCE_AUTO_POSITION = 1
FOR CHANNEL 'channel_orders';
START REPLICA FOR CHANNEL 'channel_orders';
```

Как PmaControl обнаруживает multi-source

Запрос обнаружения

Когда Аспиратор собирает данные репликации, он выполняет:

```
SHOW SLAVE STATUS;
-- или в MySQL 8.0+:
SHOW REPLICA STATUS;
```

При классической репликации (один мастер) этот запрос возвращает **одну строку**. При multi-source он возвращает **по одной строке на канал**.

PmaControl автоматически обнаруживает multi-source, когда `SHOW SLAVE STATUS` возвращает более одной строки. Никакой специальной конфигурации со стороны PmaControl не требуется.

Внутреннее хранение

Каждый канал хранится как независимая запись репликации в таблицах временных рядов.

Имя канала сохраняется и используется как дискриминатор:

```
ts_value_general_json:
  server_id = 42
  variable = slave_status
  channel   = 'channel_clients'
  value     = { "Slave_IO_Running": "Yes", "Seconds_Behind_Master": 3, ... }

server_id = 42
variable = slave_status
channel   = 'channel_orders'
value     = { "Slave_IO_Running": "Yes", "Seconds_Behind_Master": 0, ... }
```

Страница slave в режиме multi-source

Когда PmaControl обнаруживает несколько каналов на сервере, страница slave отображает **каждый канал независимо**. Конкретно:

Блок на каждый канал

Каждый канал имеет свой блок с:

- **Имя канала** в заголовке (например: `channel_clients`)
- **Состояние IO/SQL**: индикаторы `Slave_IO_Running` и `Slave_SQL_Running` , специфичные для канала
- **Лаг**: `Seconds_Behind_Master` , специфичный для канала
- **GTID**: состояние GTID, специфичное для канала
- **Последняя ошибка**: `Last_SQL_Error` канала

График лага по каналам

Каждый канал имеет свой график Chart.js с историей лага за 5 дней. Графики расположены вертикально на странице.

Это позволяет визуально сравнить: если `channel_clients` имеет стабильный лаг 0 секунд, а `channel_orders` показывает периодические всплески, проблема явно локализована на мастере заказов.

Полоса здоровья по каналам

Каждый канал имеет свою полосу здоровья. Один канал зелёный, другой красный — сразу ясно: проблема на одном канале, а не на самой реплике.

Действия по каналам

Действия (START, STOP, SKIP) доступны **по каналам**:

```
-- Остановить один канал
STOP SLAVE 'channel_orders';

-- Пропустить ошибку на канале
SET GLOBAL sql_slave_skip_counter = 1;
START SLAVE 'channel_orders';
```

```
-- MariaDB: остановить конкретный канал
STOP SLAVE 'channel_clients';
```

REPLICATE_REWRITE_DB по каналам

В multi-source часто используется `REPLICATE_REWRITE_DB` для перенаправления имён баз данных:

```
CHANGE REPLICATION SOURCE TO
SOURCE_HOST = '10.0.1.1',
...
FOR CHANNEL 'channel_clients';

-- Перенаправить 'clients' в 'dw_clients' на реплике
CHANGE REPLICATION FILTER
REPLICATE_REWRITE_DB = ((clients, dw_clients))
FOR CHANNEL 'channel_clients';
```

PmaControl обнаруживает и отображает эти фильтры в деталях канала. Это важно для отладки: если таблица не появляется на реплике, проверка фильтров перенаправления — первый рефлекс.

Текущие ограничения

PmaControl хорошо справляется с multi-source на уровне каждого отдельного канала. Однако некоторых функций не хватает для действительно унифицированного мониторинга.

Нет представления «все каналы здоровы»

В настоящее время нет консолидированной страницы, показывающей состояние здоровья всех каналов реплики одним взглядом. Нужно прокручивать страницу slave для проверки каждого канала.

Чего не хватает: сводная таблица в верхней части страницы с зелёным/красным индикатором на канал, как:

```
channel_clients  □ I0: Yes  SQL: Yes  Lag: 0s
channel_orders   ▲□ I0: Yes  SQL: Yes  Lag: 45s
channel_logs     □ I0: No   SQL: No   Error: 1062
```

Нет агрегированного лага

PmaControl отслеживает лаг каждого канала отдельно, но не вычисляет агрегированную метрику. Например:

- Максимальный лаг среди всех каналов
- Средний лаг
- Количество каналов с ошибками от общего числа

Эти агрегированные метрики были бы полезны для оповещений: «хотя бы один канал имеет лаг более 60 секунд» более уместно, чем поканальные оповещения при 10 каналах.

Нет графа зависимостей между каналами

В multi-source каналы независимы. Но в некоторых архитектурах существуют логические зависимости: канал А должен быть применён до канала В (например, справочные таблицы до транзакционных).

PmaControl не моделирует эти зависимости. DBA должен вручную управлять порядком применения при необходимости.

Нет алертинга по каналам

Текущие оповещения — на уровне сервера, а не канала. Если канал `channel_clients` останавливается, но `channel_orders` работает, оповещение просто указывает «репликация с ошибкой на сервере X» без уточнения канала.

Практические случаи

Аналитическая консолидация

Типичная архитектура:

```
Master A (clients)  — channel_clients —>
                                     Аналитическая реплика
```

```
Master B (orders)  — channel_orders  —> (MariaDB / MySQL)
Master C (logs)   — channel_logs    —>
```

Реплика консолидирует три базы. PmaControl мониторит три канала. Если Master C (logs) медленный, лаг канала `channel_logs` растёт, не затрагивая два других.

В PmaControl вы увидите три блока на странице slave реплики, с тремя независимыми графиками лага. Канал `channel_logs` будет иметь свою полосу здоровья в жёлтом или красном цвете, тогда как два других останутся зелёными.

Миграция через каналы

Во время миграции временно может быть:

```
Старый мастер — channel_legacy —>
                                     Новая реплика
Новый мастер  — channel_new    —>
```

Канал `channel_legacy` будет удалён после завершения миграции. Во время перехода PmaControl мониторит оба канала и позволяет убедиться, что новый канал догнал, прежде чем отключить старый.

Фильтрация по базам

С `REPLICATE_D0_DB` или `REPLICATE_REWRITE_DB` каждый канал может быть фильтрован для репликации только определённых баз:

```
CHANGE REPLICATION FILTER
  REPLICATE_D0_DB = (clients, clients_archive)
  FOR CHANNEL 'channel_clients';

CHANGE REPLICATION FILTER
  REPLICATE_D0_DB = (orders, order_items)
  FOR CHANNEL 'channel_orders';
```

PmaControl отображает активные фильтры для каждого канала, что облегчает диагностику, когда ожидаемая таблица не появляется на реплике.

Дорожная карта

Планируемые улучшения для поддержки multi-source в PmaControl:

Унифицированный дашборд multi-source

Дашборд, посвящённый multi-source, с:

- Матричное представление: реплика в строке, каналы в столбцах
- Глобальный индикатор здоровья (зелёный если все каналы ОК, красный если хотя бы один КО)
- Агрегированный лаг (максимум, среднее, количество ошибок)

Матрица здоровья каналов

Компактная визуализация типа «тепловой карты»:

	Пн	Вт	Ср	Чт	Пт	Сб	Вс
ch_clients	🟢	🟢	🟢	🟢	🟢	🟢	🟢
ch_orders	🟢	🟢	🟢	🟢	🟢	🟢	🟢
ch_logs	🟢	🟢	🟢	🟢	🟢	🟢	🟢

Алертинг по каналам

Telegram-оповещения, специфичные для каналов:

```
⚠️ Replication Warning
Server: replica-analytics (10.0.3.1:3306)
Channel: channel_orders
Lag: 120s (threshold: 60s)
Status: IO=Yes, SQL=Yes
```

Автоматическое обнаружение зависимостей

Анализ внешних ключей между каналами для обнаружения потенциальных зависимостей и оповещения при отставании зависимого канала.

Лучшие практики multi-source

1. Давать каналам понятные имена

Используйте описательные имена: `channel_clients`, `channel_orders`, не `ch1`, `ch2`. PmaControl отображает имена как есть — понятные имена облегчают диагностику.

2. Один канал = одна база (или логическая группа)

Избегайте смешивания несвязанных баз в одном канале. Если канал останавливается на ошибке, всё его содержимое блокируется.

3. Мониторить пространство relay log

С несколькими каналами дисковое пространство relay log может взорваться. Каждый канал имеет свой relay log. Следите за `Relay_Log_Space` по каналам в PmaControl.

4. Тестировать остановку каналов по отдельности

Убедитесь, что остановка одного канала не влияет на другие. PmaControl позволяет STOP/START каждый канал независимо — используйте эту возможность для валидации изоляции.

5. Документировать архитектуру

Multi-source добавляет сложность. Документируйте, какой канал несёт какую базу, от какого мастера, с какими фильтрами. PmaControl отображает эту информацию, но внешняя документация остаётся полезной для онбординга.

Заключение

Multi-source репликация — мощный инструмент для консолидации данных MariaDB / MySQL. PmaControl автоматически обнаруживает множественные каналы и мониторит их индивидуально с графиками лага, полосами здоровья и корректирующими действиями по каналам.

Текущие ограничения — отсутствие унифицированного представления, нет алертинга по каналам, нет агрегированного лага — выявлены и входят в дорожную карту. Пока же поканальный мониторинг покрывает основные потребности: знать, у какого канала лаг, какой канал в ошибке и иметь возможность действовать индивидуально.

Для DBA, управляющих multi-source архитектурами в продакшне, PmaControl остаётся наиболее подходящим инструментом — пусть и несовершенным — потому что ни один конкурент не предлагает такую видимость по каналам из коробки.