
Яндекс

Яндекс

Как сохранить статистику при мажорном обновлении

и что за это бывает

Дмитрий Сарафанников
Разработчик

Процесс обновления

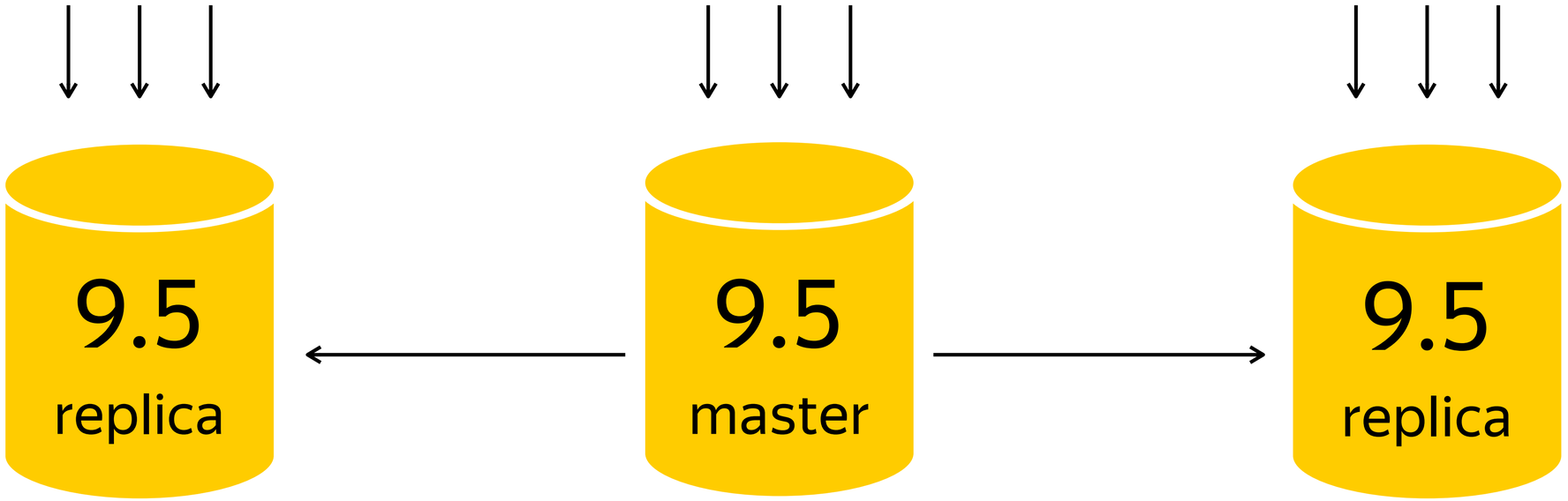
Требования

- Минимальный даунтайм на запись
- 100% доступность на чтение
- Обновление реплик без полного копирования с мастера

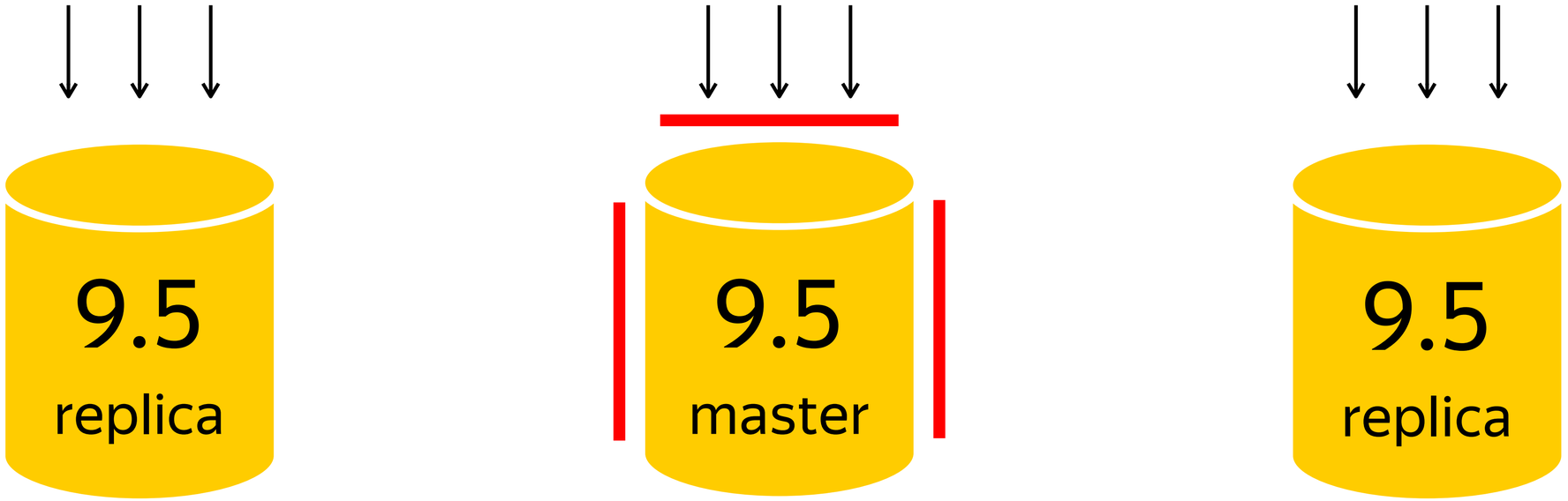
План

- Закрываем мастер от нагрузки и реплик
- Обновляем мастер: `pg_upgrade -k ...`
- Обновляем реплики по одной (остановка, обновление, запуск):
  `rsync --archive --delete --hard-links --size-only --no-inc-recursive old_data new_data remote_dir`
- Открываем мастер для нагрузки

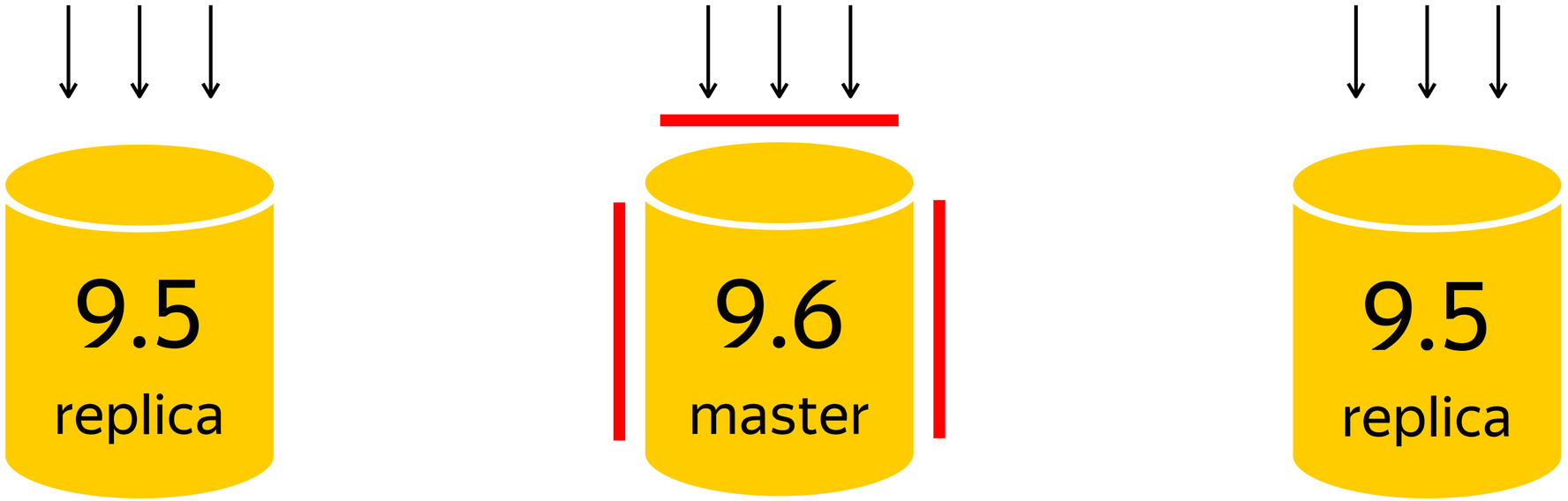
Шаг 0



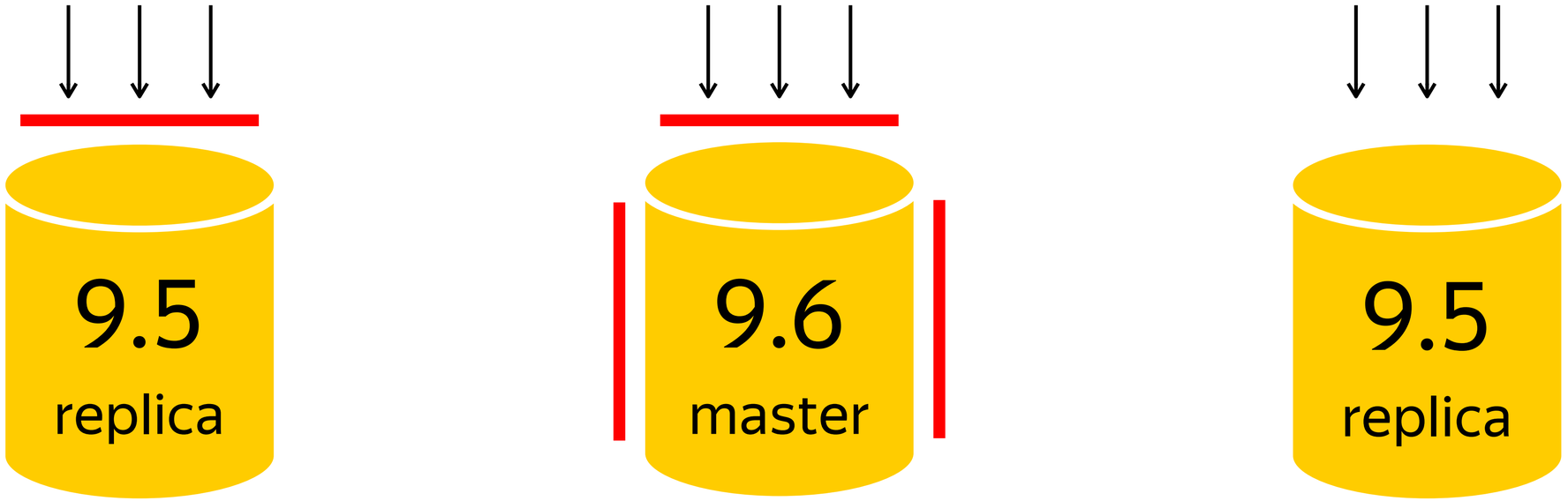
Шаг 1 – закрываем мастер



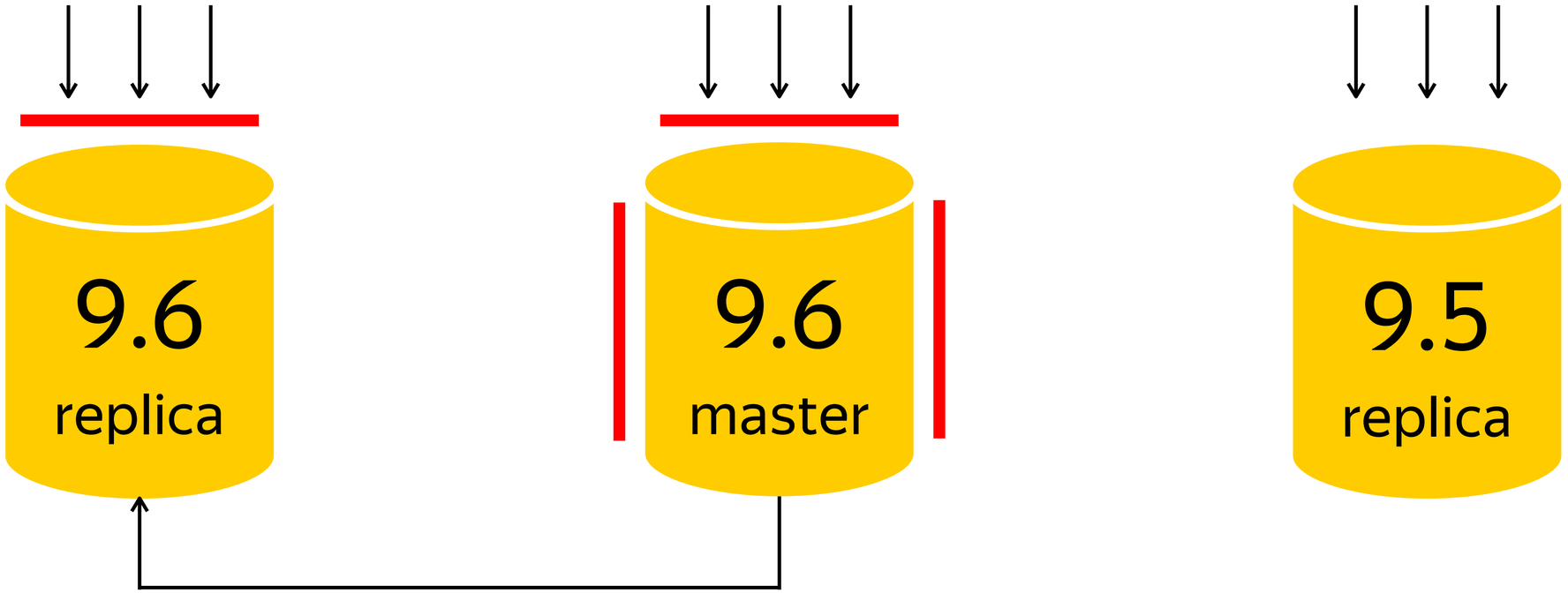
Шаг 2 – обновляем мастер



Шаг 3 – закрываем 1 реплику

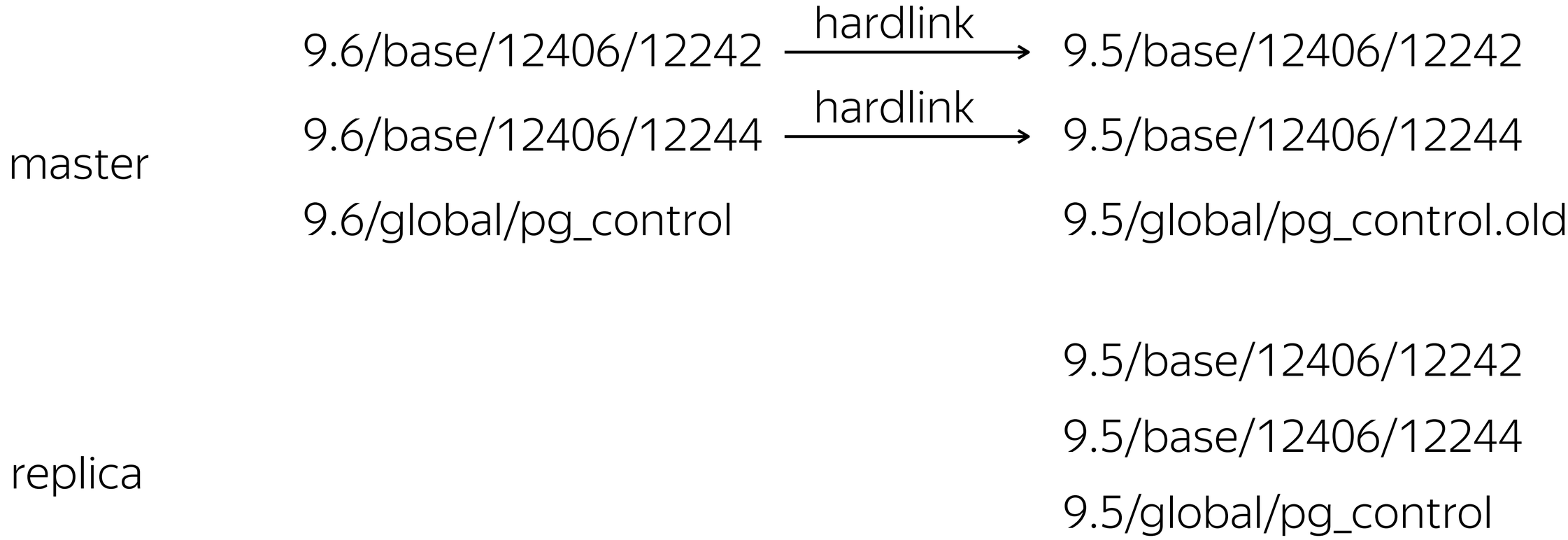


Шаг 4 – обновляем 1 реплику




```
rsync --hard-links --size-only ...
```

Шаг 4 – магия rsync



Шаг 4 – магия rsync

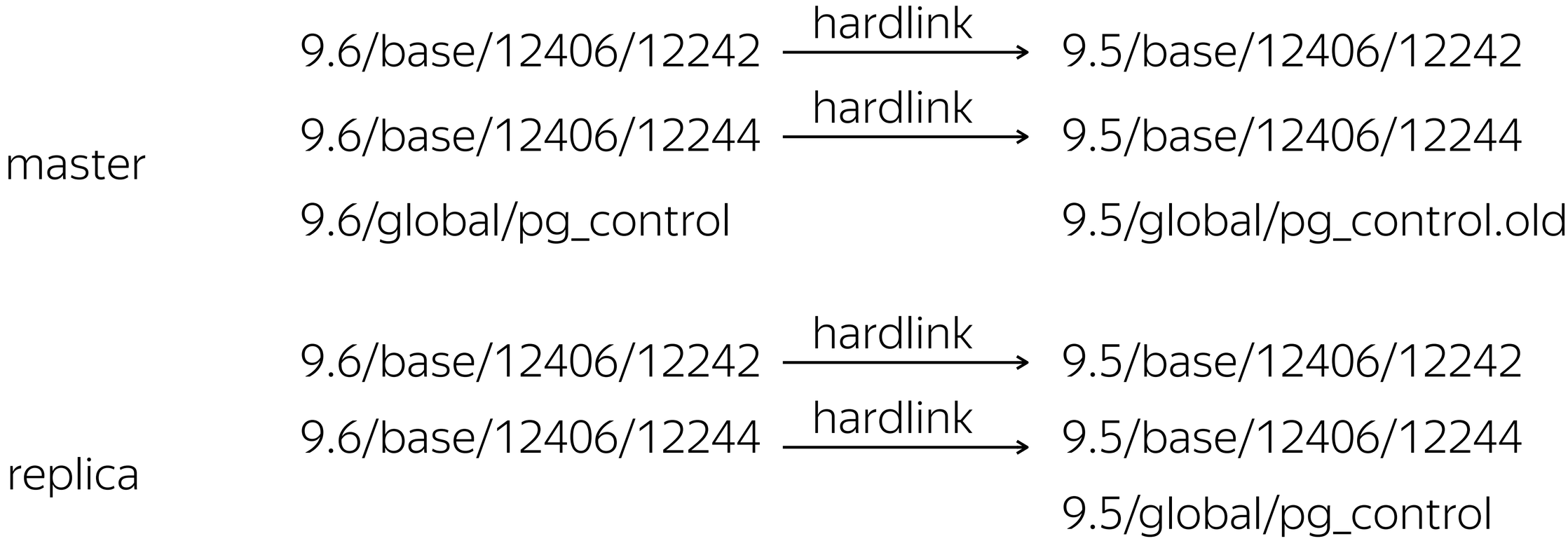
master

9.6/base/12406/12242	hardlink →	9.5/base/12406/12242
9.6/base/12406/12244	hardlink →	9.5/base/12406/12244
9.6/global/pg_control		9.5/global/pg_control.old

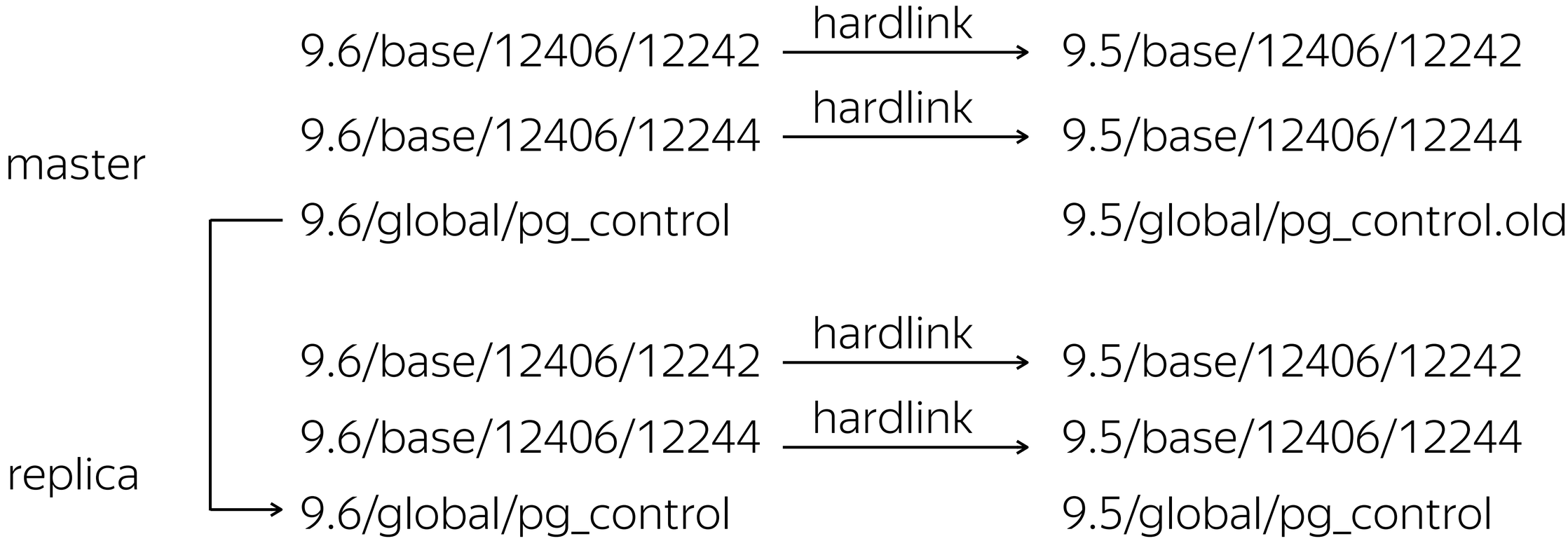
replica

9.6/base/12406/12242	hardlink →	9.5/base/12406/12242
		9.5/base/12406/12244
		9.5/global/pg_control

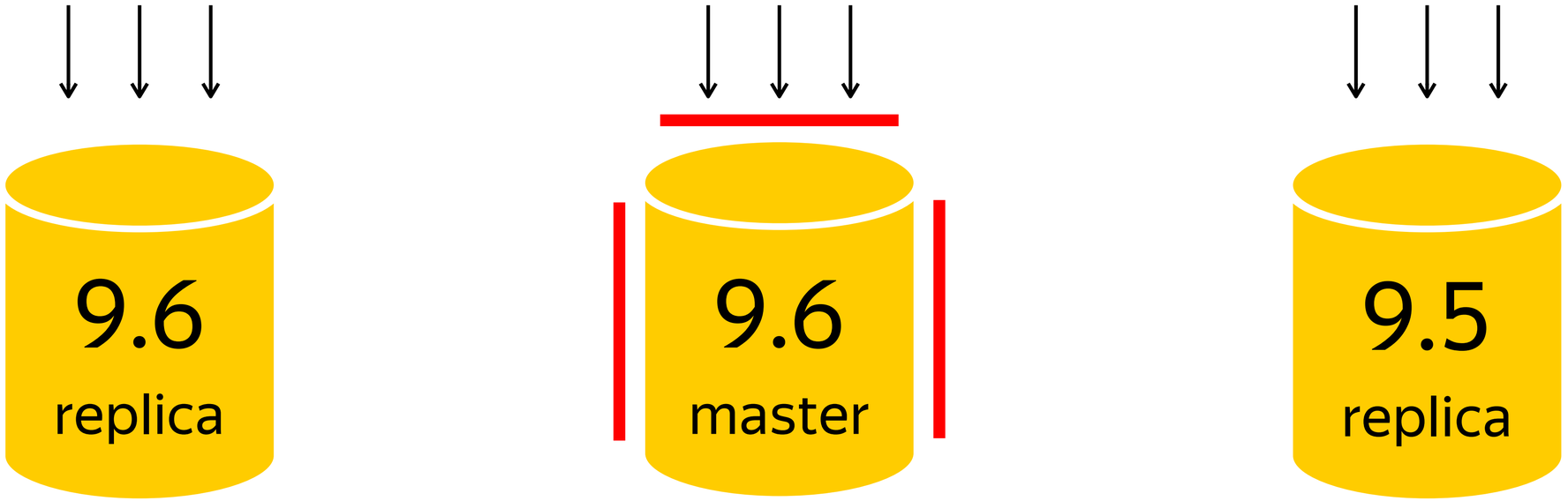
Шаг 4 – магия rsync



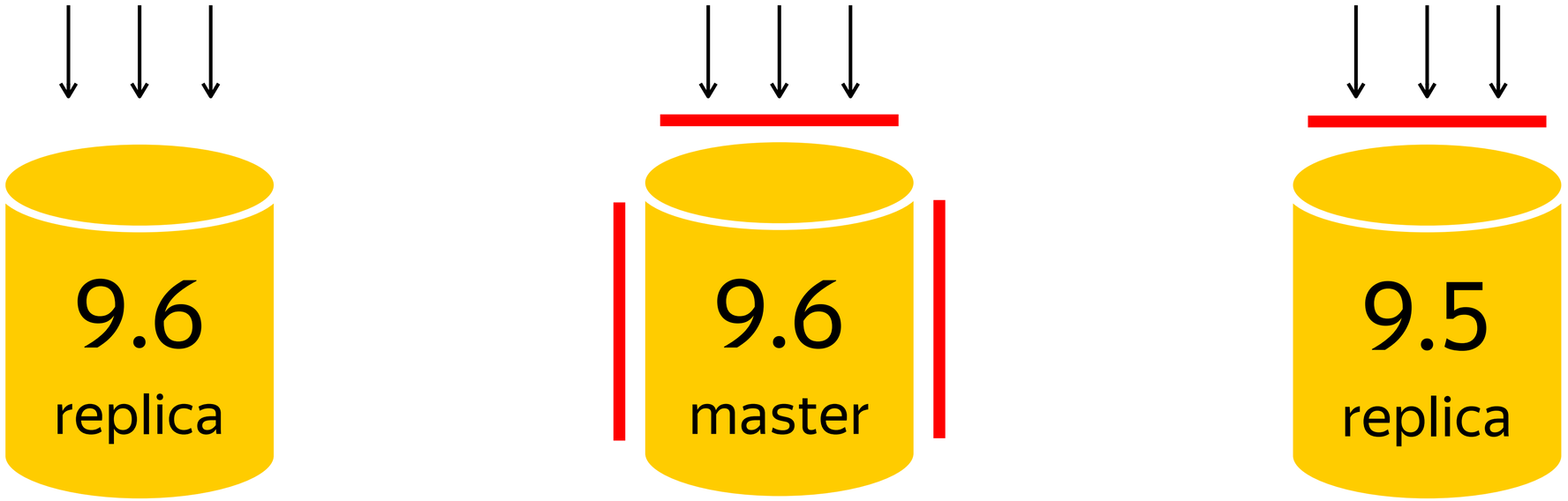
Шаг 4 – магия rsync



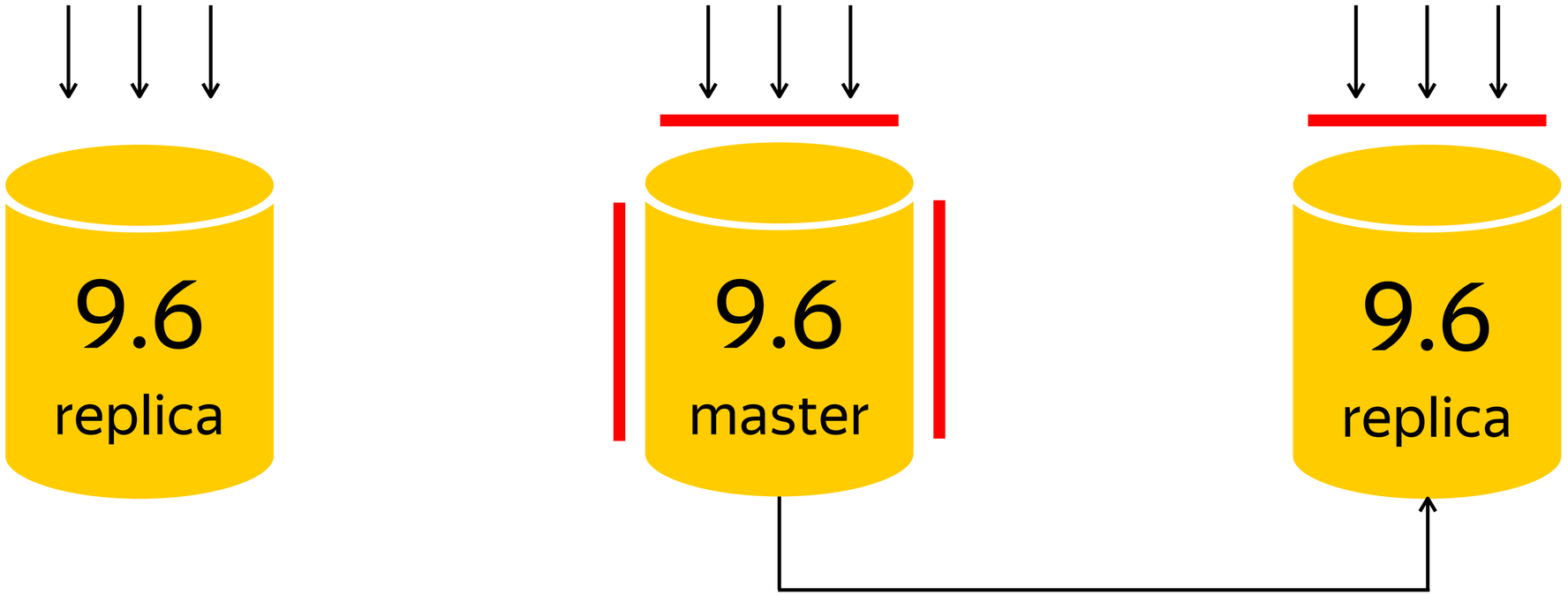
Шаг 5 – открываем 1 реплику



Шаг 6 – закрываем 2 реплику

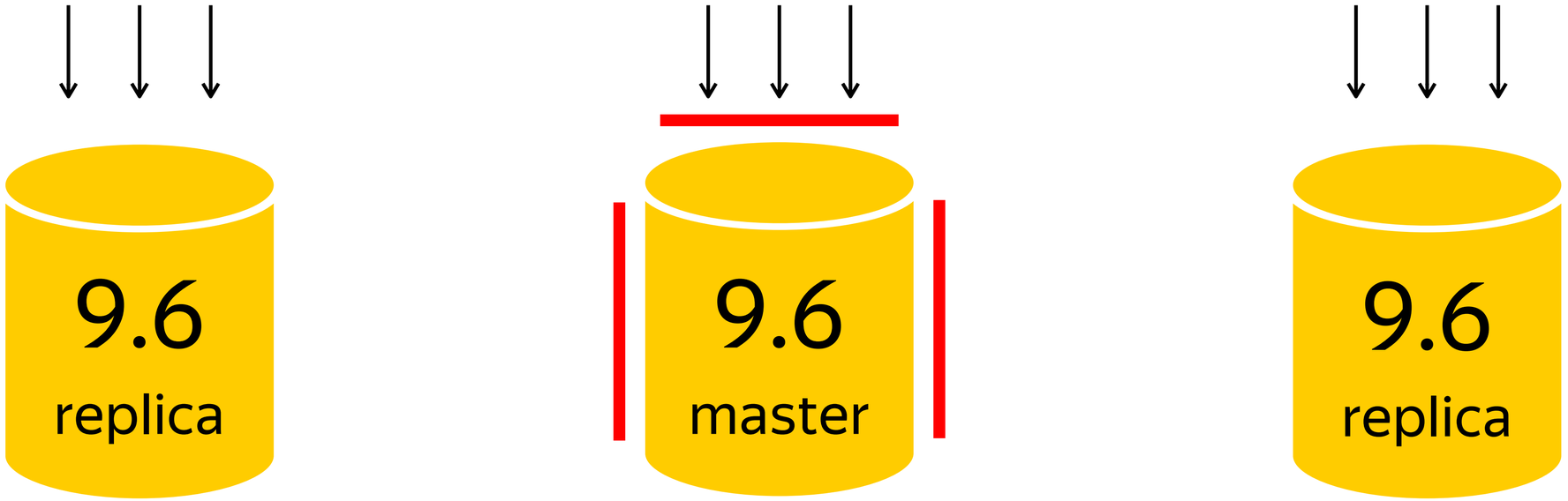


Шаг 6 – обновляем 2 реплику

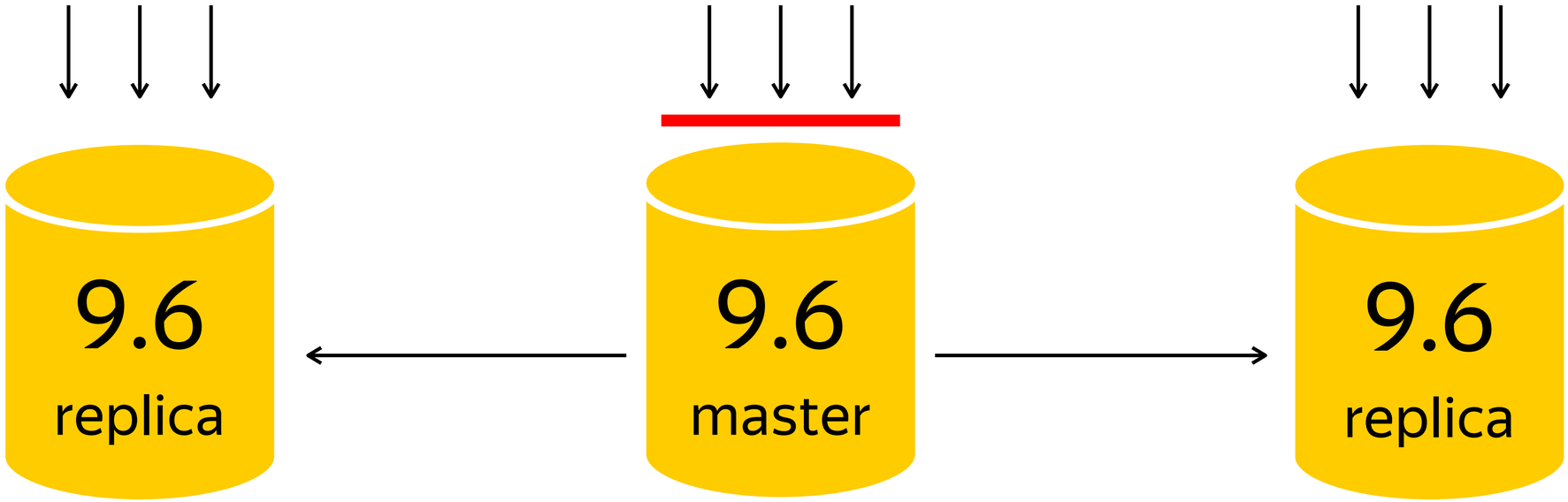



```
rsync --hard-links --size-only ...
```

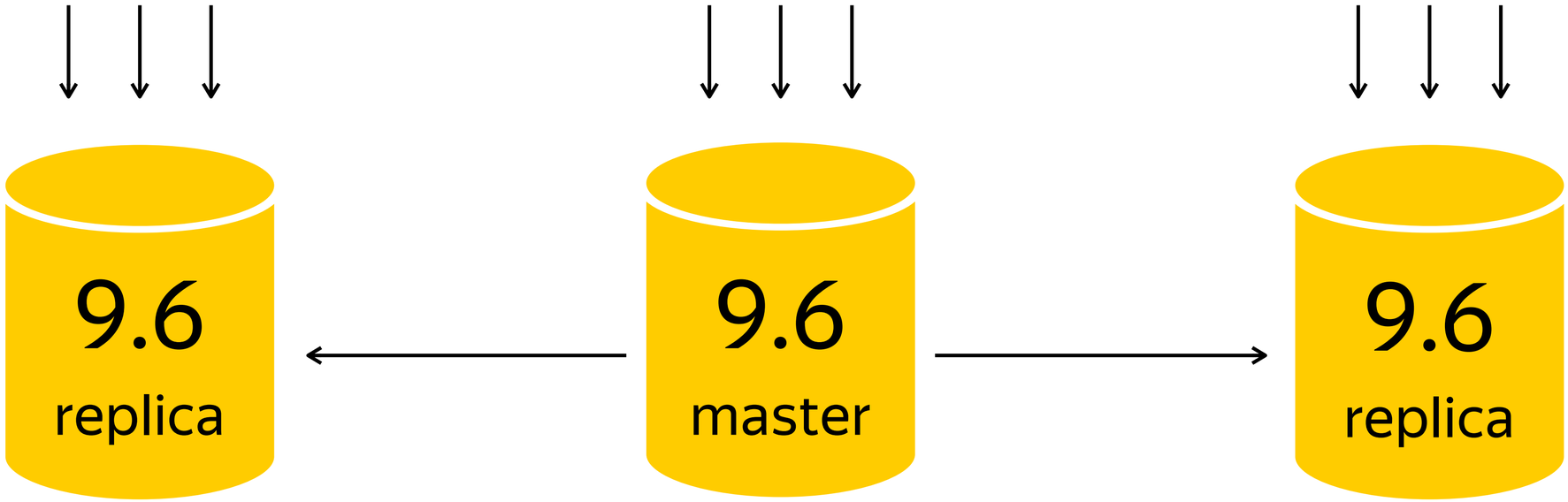
Шаг 7 – открываем 2 реплику



Шаг 8 – репликация

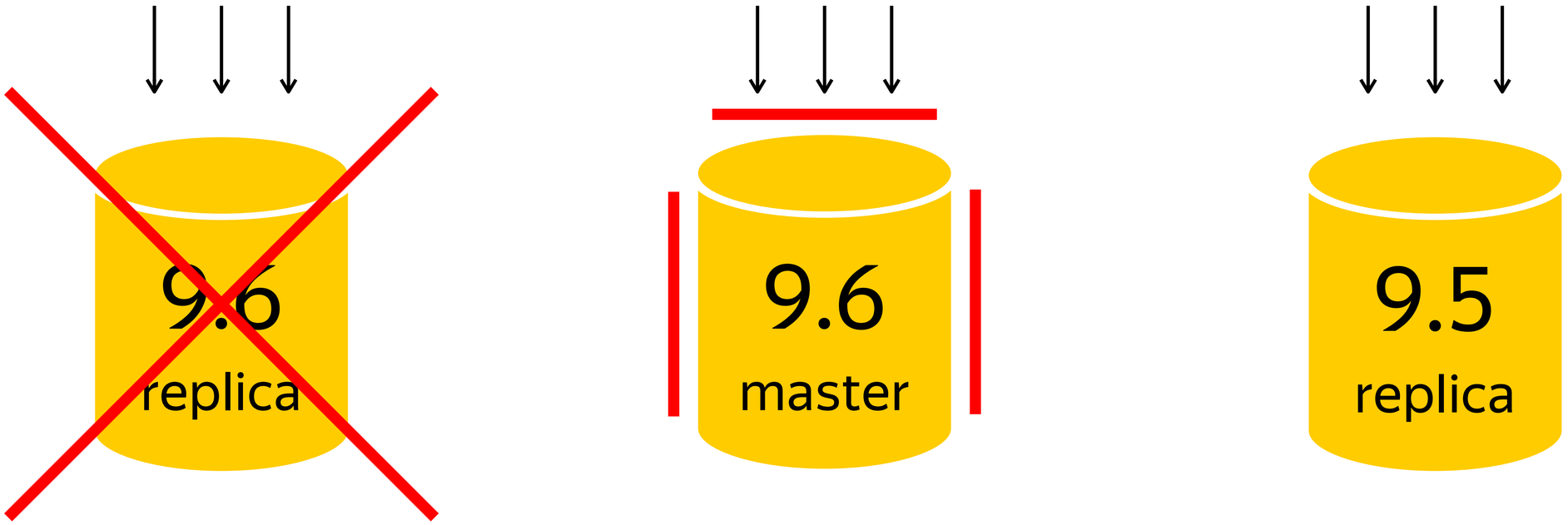


Шаг 8 – открываем мастер



Проблемы

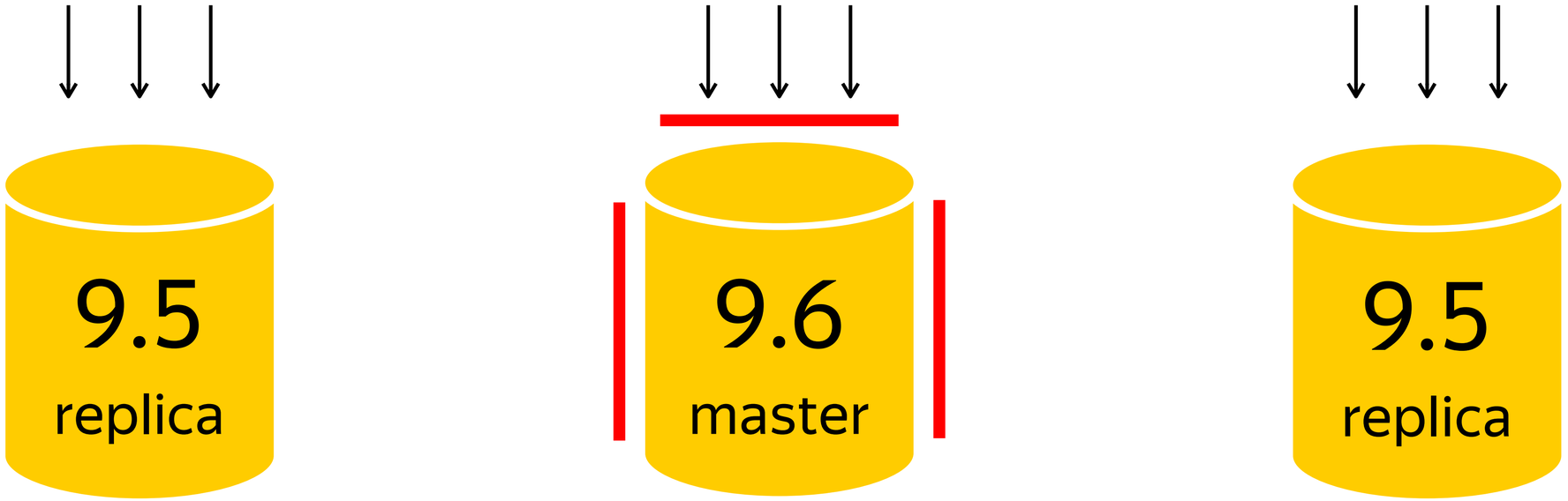
Шаг 5 – реплика не справляется с нагрузкой



Нет статистики => `load average: 1000.90, 820.83, 497.11`

Решение

Шаг 2.5 – собираем статистику

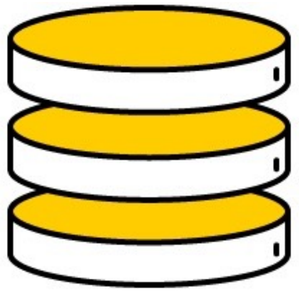



```
vacuumdb -d mydb -j 8 --analyze-in-stages
```

Последствия

Последствия

Регламентные работы
в ДЦ



Мастера
переключились

Последствия

```
ERROR:  58P01: could not access status of transaction
1951521353

DETAIL:  Could not open file "pg_clog/0745": No such file
or directory.

LOCATION:  SlruReportIOError, slru.c:896
```

Последствия

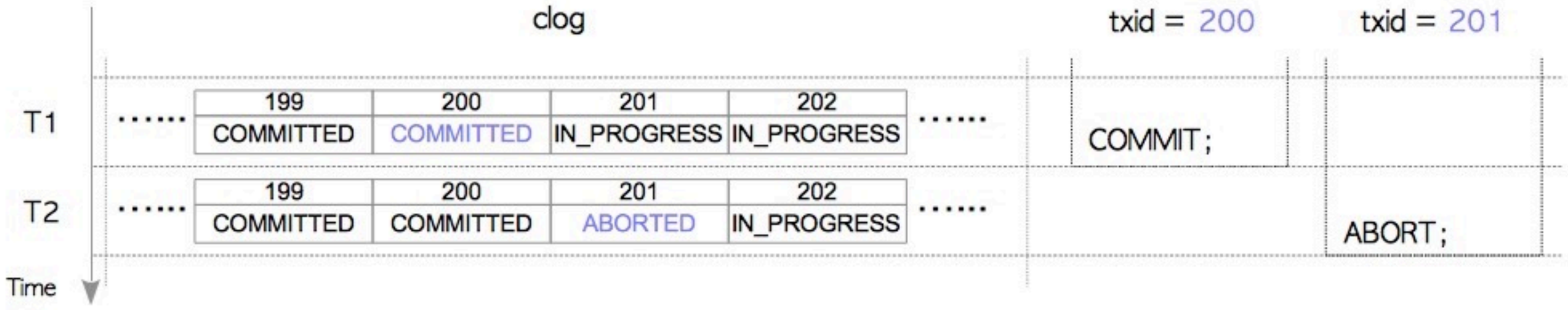
- Запросы падают с ошибками
- VACUUM тоже падает с ошибкой

pg_clog / pg_xact

pg_clog / pg_xact

```
 ~ # ls -lh /var/lib/postgresql/9.6/data/pg_clog/
total 9.2M
-rw------- 1 postgres postgres 256K Nov 30 15:20 0000
-rw------- 1 postgres postgres 256K Nov 30 15:20 0001
-rw------- 1 postgres postgres 256K Nov 30 15:20 0002
-rw------- 1 postgres postgres 256K Nov 30 15:20 0003
-rw------- 1 postgres postgres 256K Nov 30 15:20 0004
-rw------- 1 postgres postgres 256K Nov 30 15:20 0005
-rw------- 1 postgres postgres 256K Nov 30 15:20 0006
-rw------- 1 postgres postgres 256K Nov 30 15:20 0007
-rw------- 1 postgres postgres 256K Nov 30 15:20 0008
-rw------- 1 postgres postgres 256K Nov 30 15:20 0009
-rw------- 1 postgres postgres 256K Nov 30 15:20 000A
```

pg_clog / pg_xact - commit log



http://www.interdb.jp/pg/pgsql05.html

pg_clog / pg_xact

```
#define CLOG_BITS_PER_XACT 2

#define CLOG_XACTS_PER_BYTE 4

#define CLOG_XACTS_PER_PAGE (BLCKSZ * CLOG_XACTS_PER_BYTE)

#define CLOG_XACT_BITMASK  ((1 << CLOG_BITS_PER_XACT) - 1)
```

pg_clog / pg_xact

```
#define TransactionIdToPage(xid)  ((xid) / (TransactionId)
                                        CLOG_XACTS_PER_PAGE)
#define TransactionIdToPgIndex(xid) ((xid) %
                                (TransactionId) CLOG_XACTS_PER_PAGE)
#define TransactionIdToByte(xid)
      (TransactionIdToPgIndex(xid) / CLOG_XACTS_PER_BYTE)
#define TransactionIdToBIndex(xid)  ((xid) %
                        (TransactionId) CLOG_XACTS_PER_BYTE)
```

pg_clog / pg_xact

```
pageno = TransactionIdToPage(xid);
byteno = TransactionIdToByte(xid);
bshift = TransactionIdToBIndex(xid) * CLOG_BITS_PER_XACT;
slotno = SimpleLruReadPage_ReadOnly(ClogCtl, pageno, xid);
byteptr = ClogCtl->shared->page_buffer[slotno] + byteno;
status = (*byteptr >> bshift) & CLOG_XACT_BITMASK;

#define TRANSACTION_STATUS_IN_PROGRESS    0x00
#define TRANSACTION_STATUS_COMMITTED      0x01
#define TRANSACTION STATUS ABORTED        0x02
```

Что делать?

Варианты

- `dd if=/dev/zero of=pg_clog/0726 bs=256k count=1`
  потеря тех строк, на которых возникает ошибка

- Найти и обезвредить

Найти и обезвредить

```
xdb311g(master)=# SELECT * FROM mytable
WHERE ctid = '(4,21)';
ERROR:  58P01: could not access status of transaction
1951521353
DETAIL:  Could not open file "pg_clog/0745": No such file
or directory.
LOCATION:  SlruReportIOError, slru.c:896
```

Найти и обезвредить

```
xdb311g(master)=# SELECT lsn FROM
page_header(get_raw_page(‘mytable',4));
   lsn
---------------
8092/6A26DD08

xdb311e(replica)=# SELECT lsn FROM
page_header(get_raw_page(‘mytable',4));
   lsn
---------------
838D/C4A0D280
```

Найти и обезвредить

```
xdb311g(master)=# SELECT t_xmin, t_infomask::bit(32) &
X'0300'::int::bit(32) FROM
heap_page_items(get_raw_page('mytable',4)) WHERE lp=21;
-[ RECORD 1 ]--------------------------
t_xmin   | 1951521353
?column? | 00000000000000000000000000000000
```

Найти и обезвредить

```
xdb311e(replica)=# SELECT t_xmin, t_infomask::bit(32) &
X'0300'::int::bit(32) FROM
heap_page_items(get_raw_page('mytable',4)) WHERE lp=21;
-[ RECORD 1 ]--------------------------
t_xmin   | 1951521353
?column? | 00000000000000000000001100000000
```

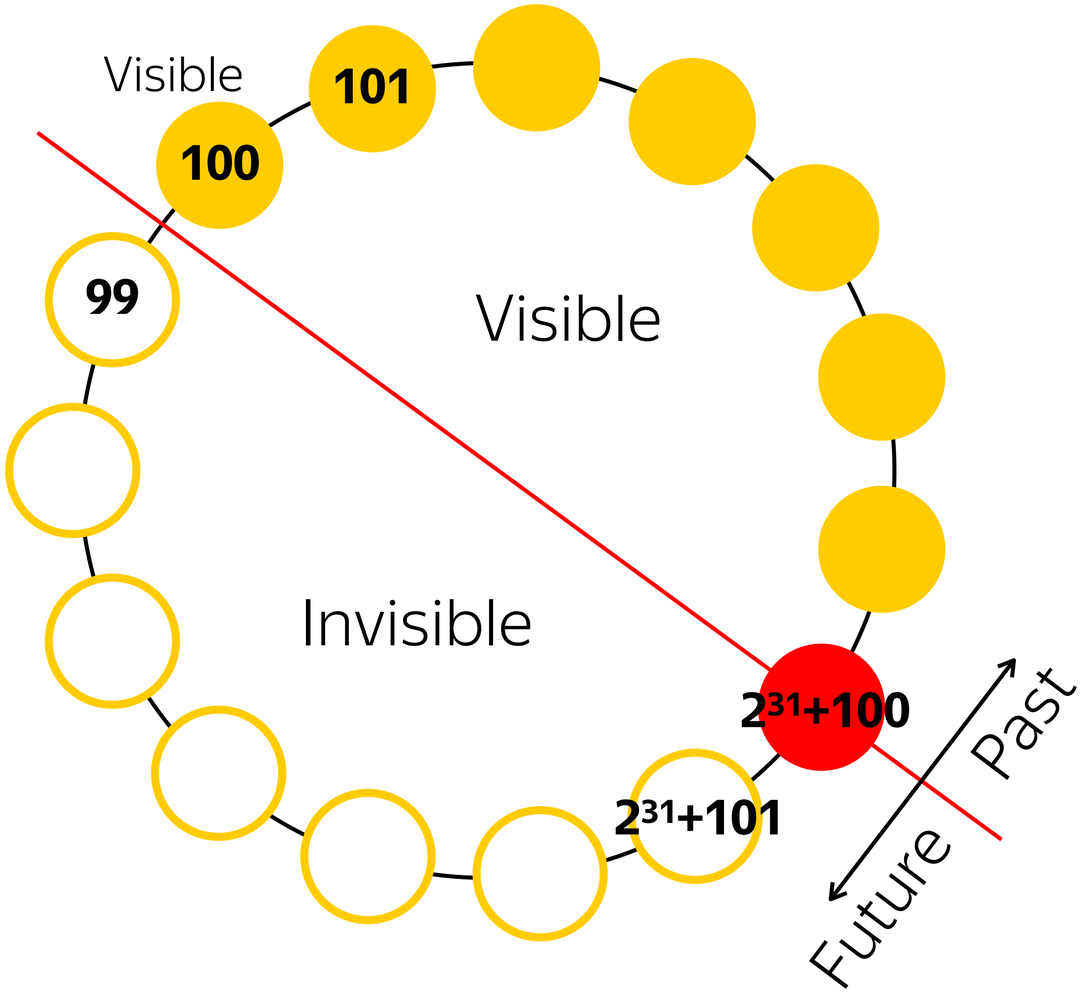
Найти и обезвредить

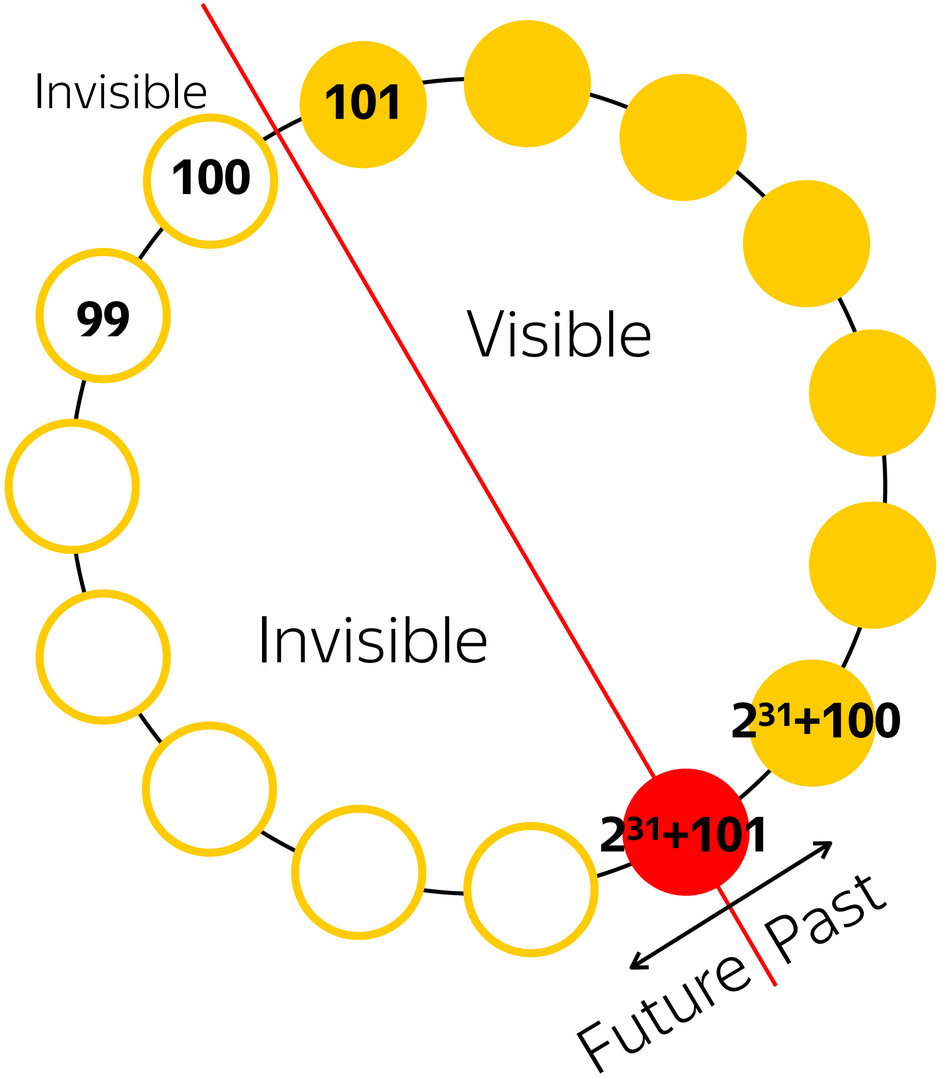
```
#define HEAP_XMIN_COMMITTED     0x0100

#define HEAP_XMIN_INVALID       0x0200

#define HEAP_XMIN_FROZEN        (HEAP_XMIN_COMMITTED|
                                 HEAP_XMIN_INVALID)
```

В любой непонятной ситуации виноват autovacuum

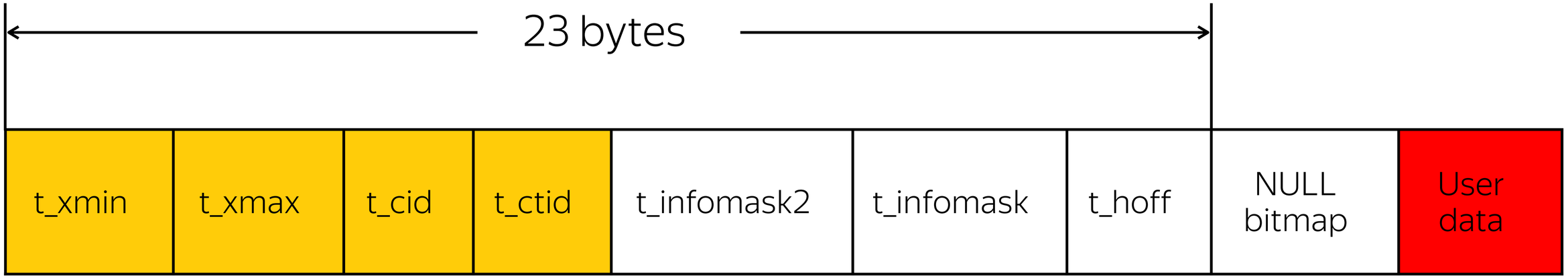
VACUUM FREEZE

Xid wraparound





HeapTuple


23 bytes
t_xmin
t_xmax
t_cid
t_ctid
t_infomask2
t_infomask
t_hoff
NULL bitmap
User data

VACUUM FREEZE

	t_xmin	t_xmax	t_infomask	user data
Tuple 1	99			'A'
Tuple 2	100			'B'
Tuple 3	200000			'C'
Tuple 4	2000000			'D'

VACUUM FREEZE до 9.4

	t_xmin	t_xmax	t_infomask	user data
Tuple 1	2			'A'
Tuple 2	2			'B'
Tuple 3	200000			'C'
Tuple 4	2000000			'D'

VACUUM FREEZE с 9.4

	t_xmin	t_xmax	t_infomask	user data
Tuple 1	99		XMIN_FROZEN	'A'
Tuple 2	100		XMIN_FROZEN	'B'
Tuple 3	200000			'C'
Tuple 4	2000000			'D'

Найти и обезвредить

```
#define HEAP_XMIN_COMMITTED     0x0100

#define HEAP_XMIN_INVALID       0x0200

#define HEAP_XMIN_FROZEN        (HEAP_XMIN_COMMITTED|
                                 HEAP_XMIN_INVALID)
```

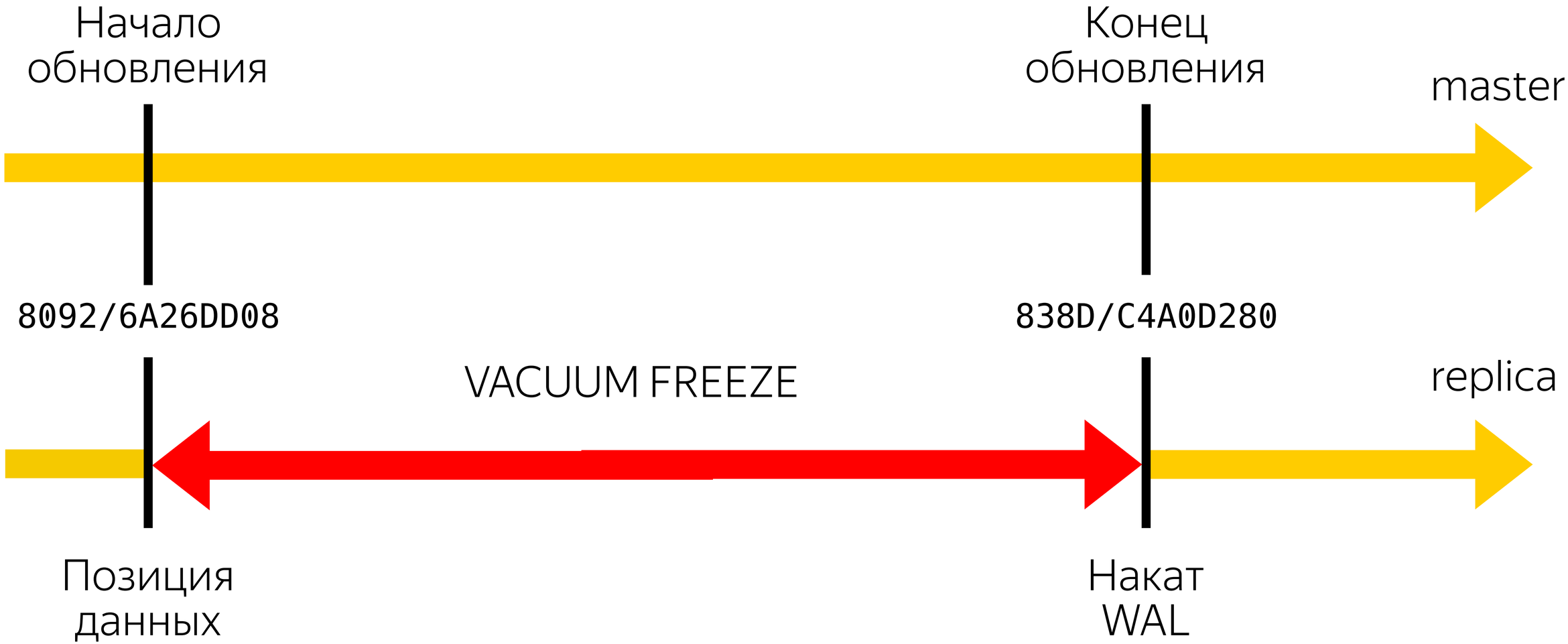
Что произошло?

Что произошло?

https://clck.ru/CbuNS - тред в pgsql-hackers

- Во время сбора статистики autovacuum начал FREEZE
- CHECKPOINT при остановке мастера
- rsync скопировал pg_control с новой позицией на реплику
- rsync --size-only не скопировал измененные файлы
- реплика не проиграла WAL, который сгенерировал autovacuum

Что произошло?



Что произошло?

```
commit 0f33a719fdbb5d8c43839ea0d2c90cd03e2af2d2
Author: Bruce Momjian <bruce@momjian.us>
Date:   Thu Jun 15 12:30:02 2017 -0400

    docs:  Fix pg_upgrade standby server upgrade docs
```

Что произошло?

```
commit b710248dd3f90c46bd4208e6bf1290048b9d76cd
Author: Bruce Momjian <bruce@momjian.us>
Date:   Tue Jun 20 13:20:02 2017 -0400

    pg_upgrade:  start/stop new server after pg_resetwal
```

Как избежать?

Если не сразу остановить мастер, можно наступить на те же грабли

- запускать мастер в режиме single user и ANALYZE в один поток
- отключить autovacuum
- собирать статистику после обновление реплик

Как чинить?

pg_dirty_hands

```
corruption=# CREATE EXTENSION pg_dirty_hands ;
CREATE EXTENSION
corruption=# \dx+ pg_dirty_hands
    Objects in extension "pg_dirty_hands"
              Object description
-----------------------------------------------
 function freeze_tuple(regclass,tid,boolean)
 function freeze_tuple_unlogged(regclass,tid)
(2 rows)
```

pg_dirty_hands – пример

```
corruption=# CREATE TABLE mytable AS (SELECT 1);
SELECT 1

corruption=# SELECT t_infomask::bit(32) FROM
heap_page_items(get_raw_page('mytable', 0));
             t_infomask
------------------------------------
 00000000000000000010000000000000
```

pg_dirty_hands – пример

```
corruption=# SELECT ctid, * FROM mytable ;
 ctid  | ?column?
-------+----------
 (0,1) |        1

corruption=# SELECT t_infomask::bit(32) FROM
heap_page_items(get_raw_page('mytable', 0));
            t_infomask
----------------------------------
 00000000000000000001001100000000
```

pg_dirty_hands – пример

```
corruption=# SELECT freeze_tuple('mytable', '(0,1)');
 freeze_tuple
--------------
 t

corruption=# SELECT t_infomask::bit(32) FROM
heap_page_items(get_raw_page('mytable', 0));
             t_infomask
-----------------------------------
 00000000000000000001011000000000
```

pg_dirty_hands – пример

```
$ pg_xlogdump --path=pg_xlog/ --start 0/8FAABF30 \
      --limit=1 00000001000000000000008F

rmgr: Heap2       len (rec/tot):     57/   117, tx:
0, lsn: 0/8FAABF30, prev 0/8FAABEF8, desc: FREEZE_PAGE
cutoff xid 1698 ntuples 1, blkref #0: rel 1663/31145/31179
blk 0 FPW
```

pg_dirty_hands

Pull requests are welcome!

Ссылки

- https://www.postgresql.org/message-id/DA18C5E1-A115-4C1C-9F7C-E7B9A5F3EBC5%40yandex.ru
- https://git.postgresql.org/gitweb/?p=postgresql.git;a=commit;h=0f33a719fdbb5d8c43839ea0d2c90cd03e2af2d2
- https://git.postgresql.org/gitweb/?p=postgresql.git;a=commit;h=b710248dd3f90c46bd4208e6bf1290048b9d76cd
- https://github.com/dsarafan/pg_dirty_hands

Спасибо за внимание!

Дмитрий Сарафанников
Разработчик

 dsarafan@yandex-team.ru