Вся правда о мультимастере

PGConf.Сибирь 2022

Рутман Михаил

PostgresPro

PGConf.Сибирь 2022

PostgresPro

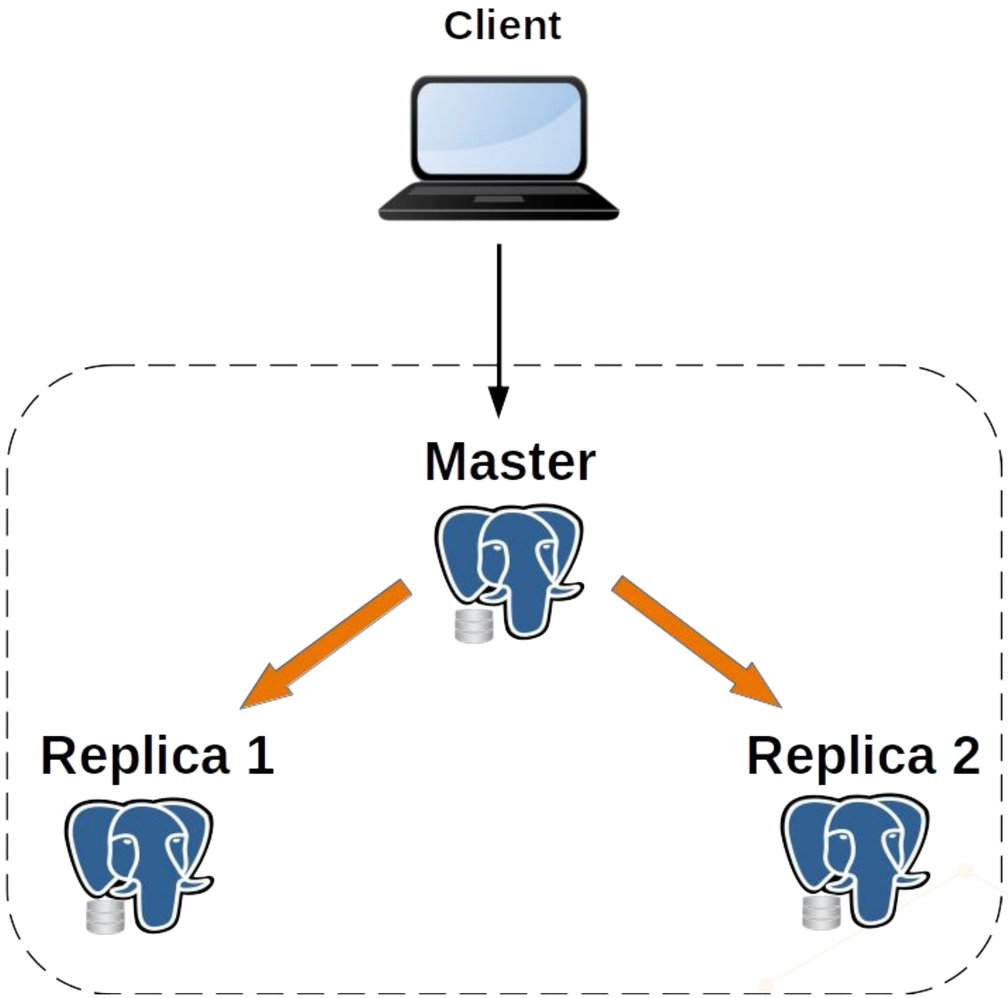
О докладчике:

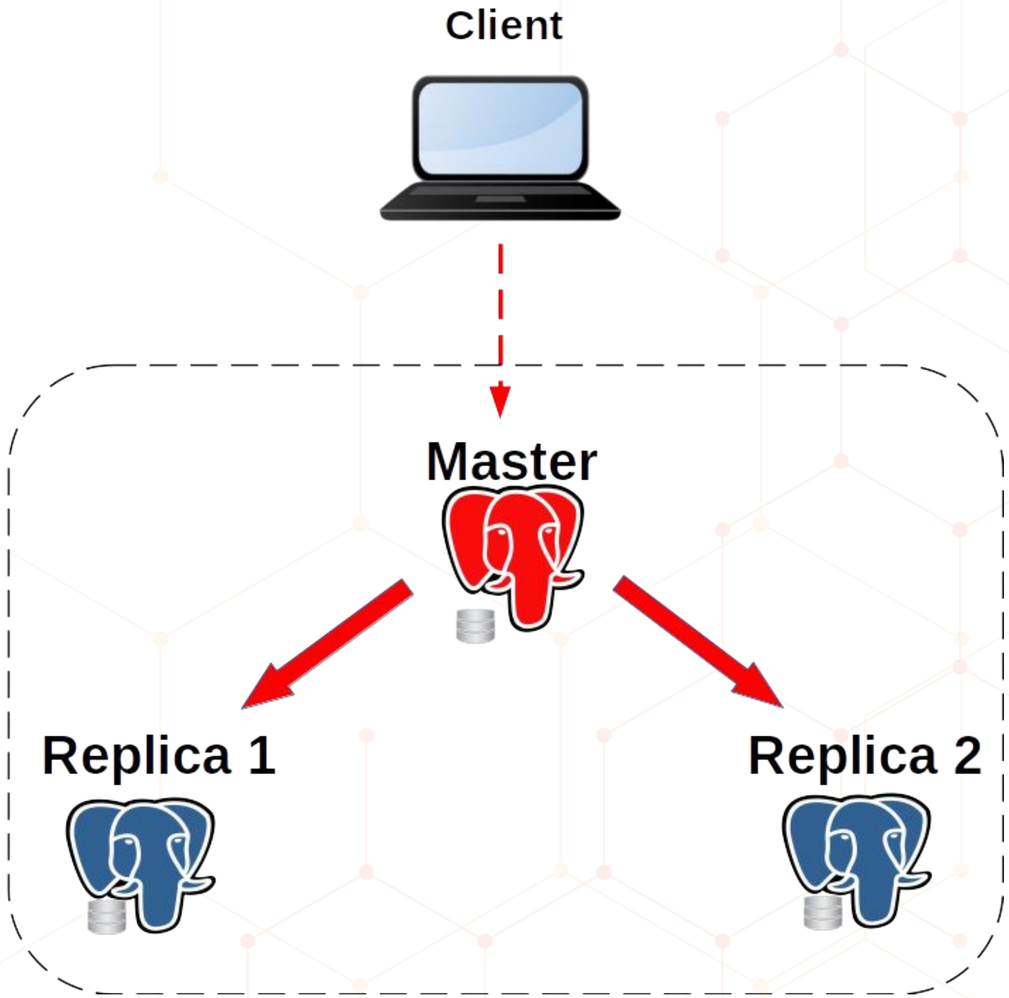
Рутман
Михаил

- Работаю в PostgresPro с 2021 года в проекте Multimaster.
- Более 20 лет в ИТ (Сигнатек, Eyeline, Navitel, Parallels, Plesk, Kublr).
- Преподаю в Новосибирском гос. университете курс “Операционные системы.”

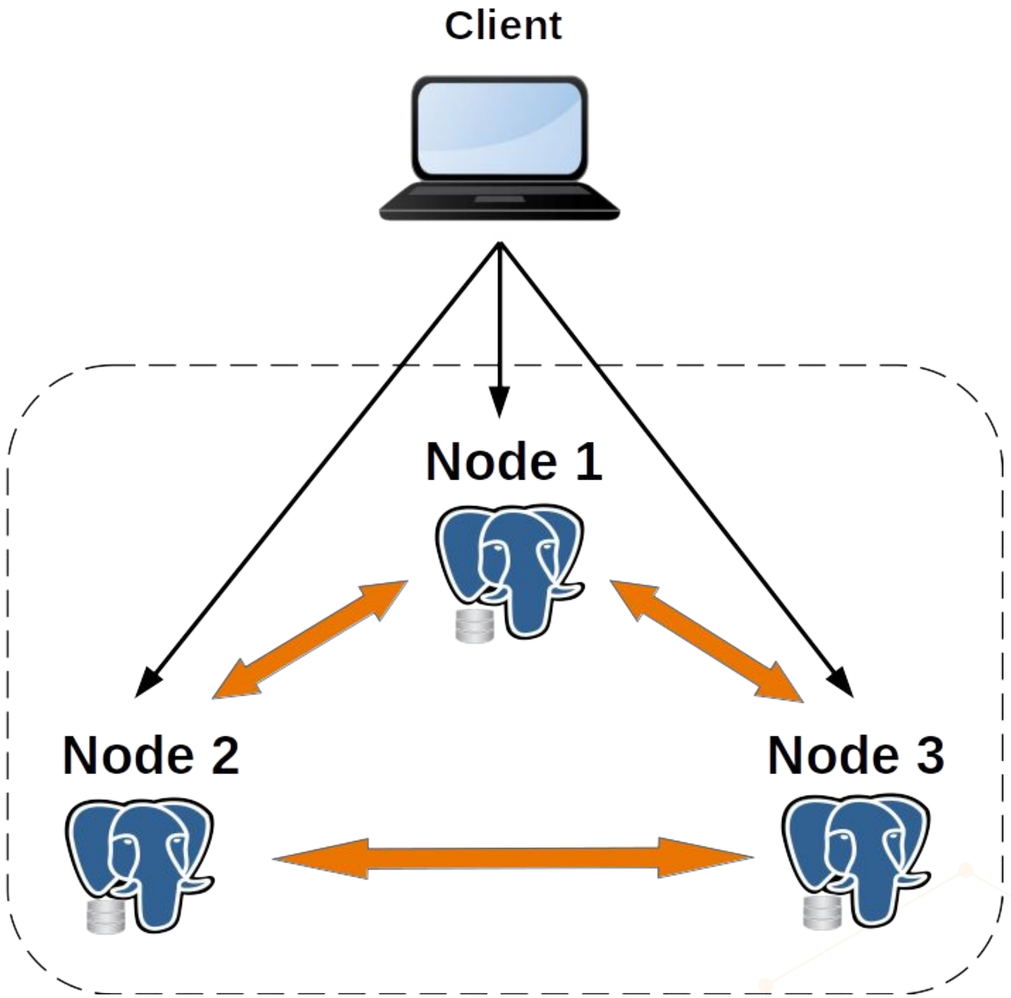
PGConf.Сибирь 2022

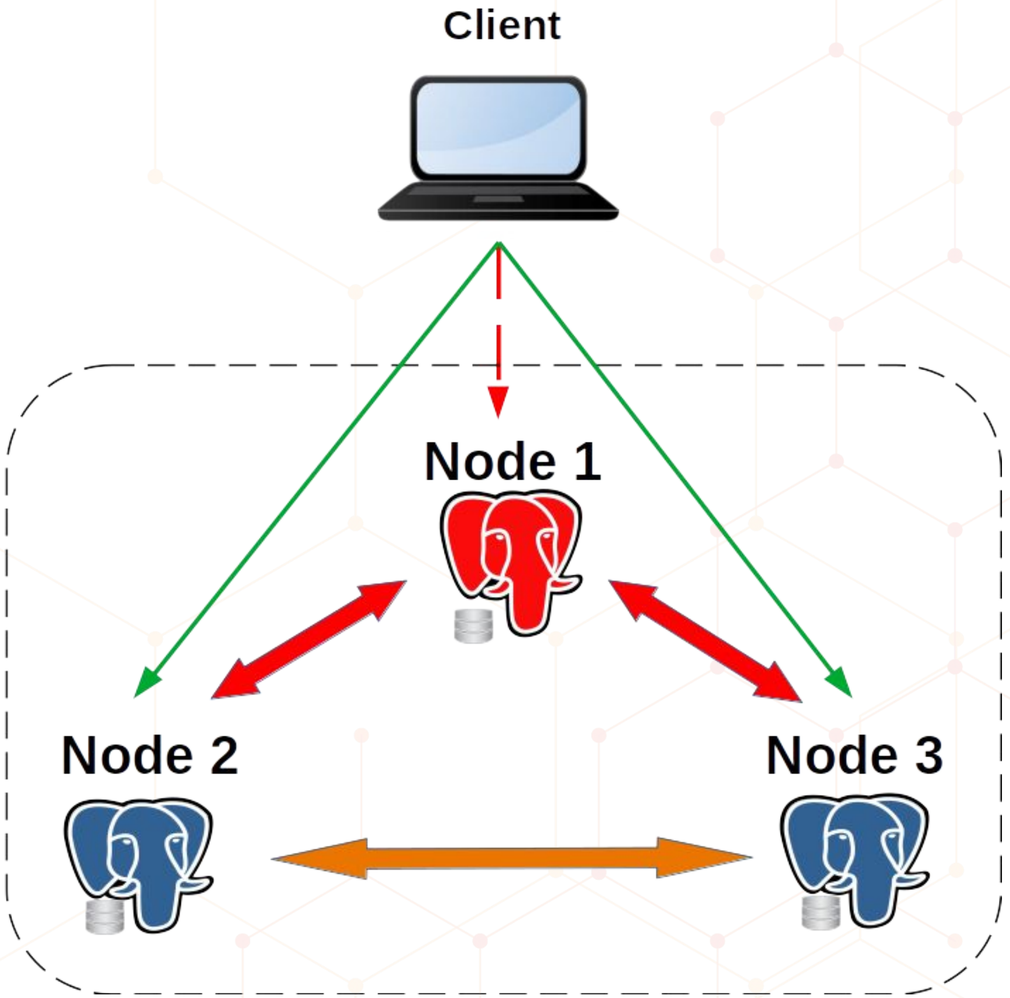
PostgresPro





PostgresPro





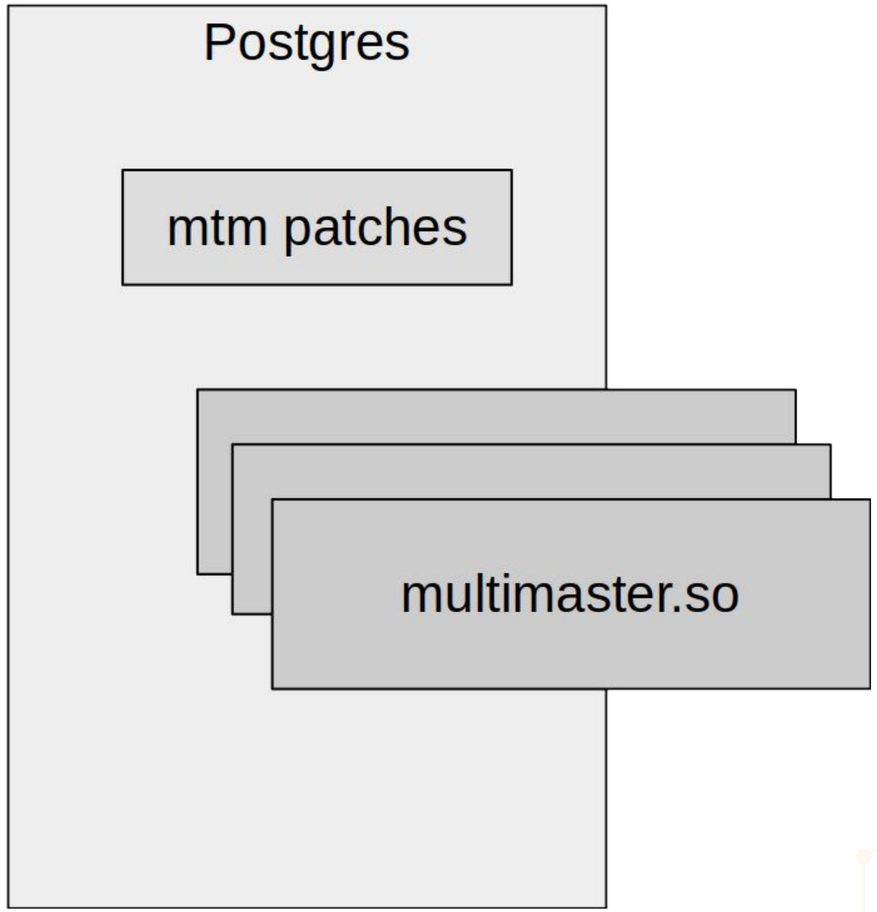
PostgresPro

Мультимастер это….

- высокая доступность
- отказоустойчивость
- симметричный кластер

PostgresPro

Мультимастер это….



- расширение Postgres
- набор патчей в ядро Postgres

PostgresPro

Расширение Postgres

- протокол логической репликации
- поддержка 3PC
- параллельный Executor
- PAXOS (Synod)
- failover

PostgresPro

Расширение Postgres

```
/home/demo/postgrespro/postgres -D /home/demo/node1/pgdata --cluster-name=node1
 \_ ............
 \_ postgres: node1: mtm-dmq-sender
 \_ postgres: node1: mtm-monitor
 \_ postgres: node1: user demodb 127.0.0.1(47206) mtm-dmq-receiver node3
 \_ postgres: node1: mtm-logrep-receiver-1-3
 \_ postgres: node1: mtm-logrep-receiver-1-2
 \_ postgres: node1: mtm-replier
 \_ postgres: node1: mtm-campaigner
 \_ postgres: node1: walsender user 127.0.0.1(47214) idle
 \_ postgres: node1: walsender user 127.0.0.1(47298) idle
 \_ postgres: node1: user demodb 127.0.0.1(47230) mtm-dmq-receiver node2
```

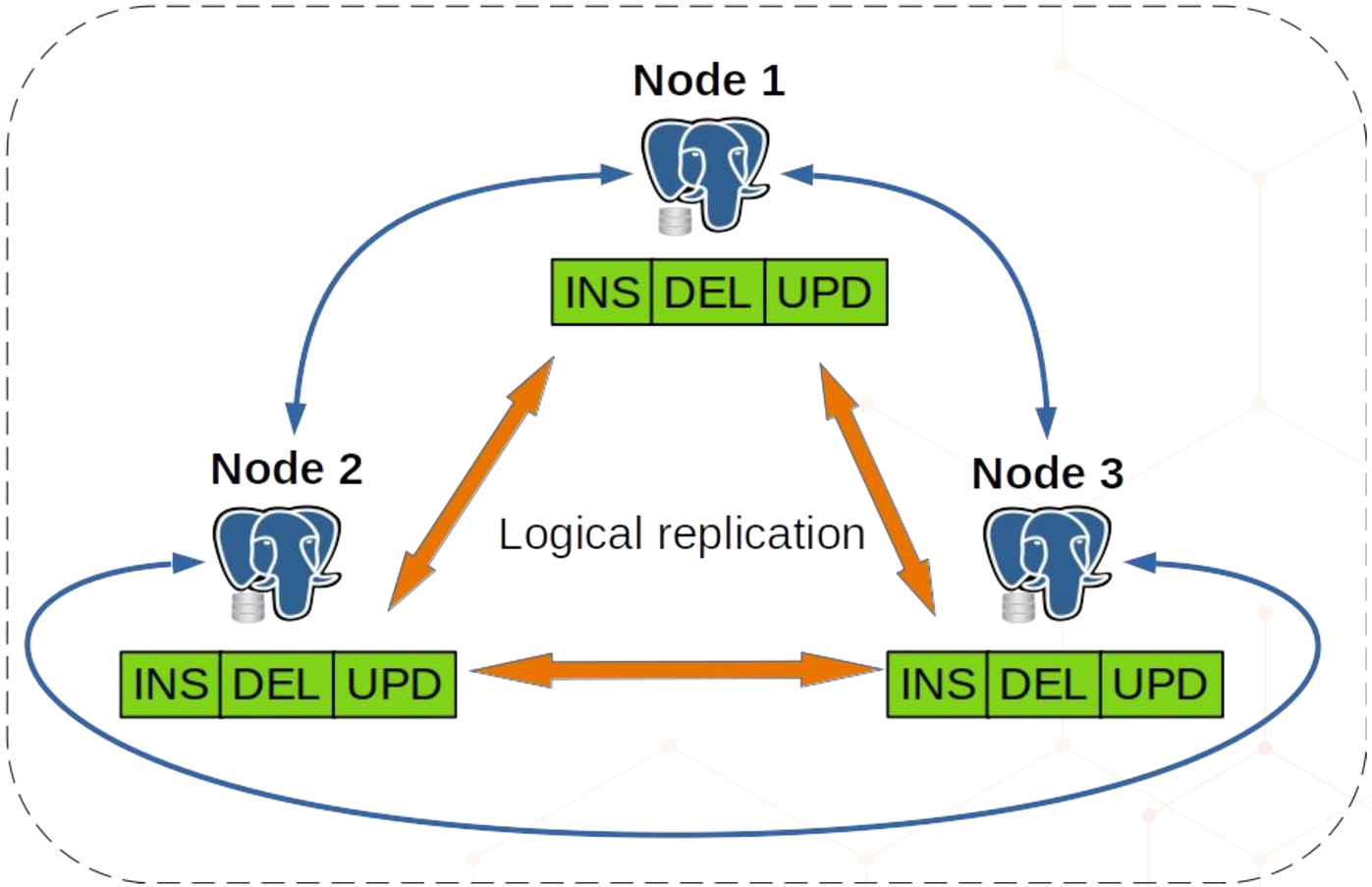
PostgresPro

Набор патчей в ядро

- поддержка 3PC в logical decoding
- API для обнаружения deadlock
- ….

PostgresPro

Мультимастер



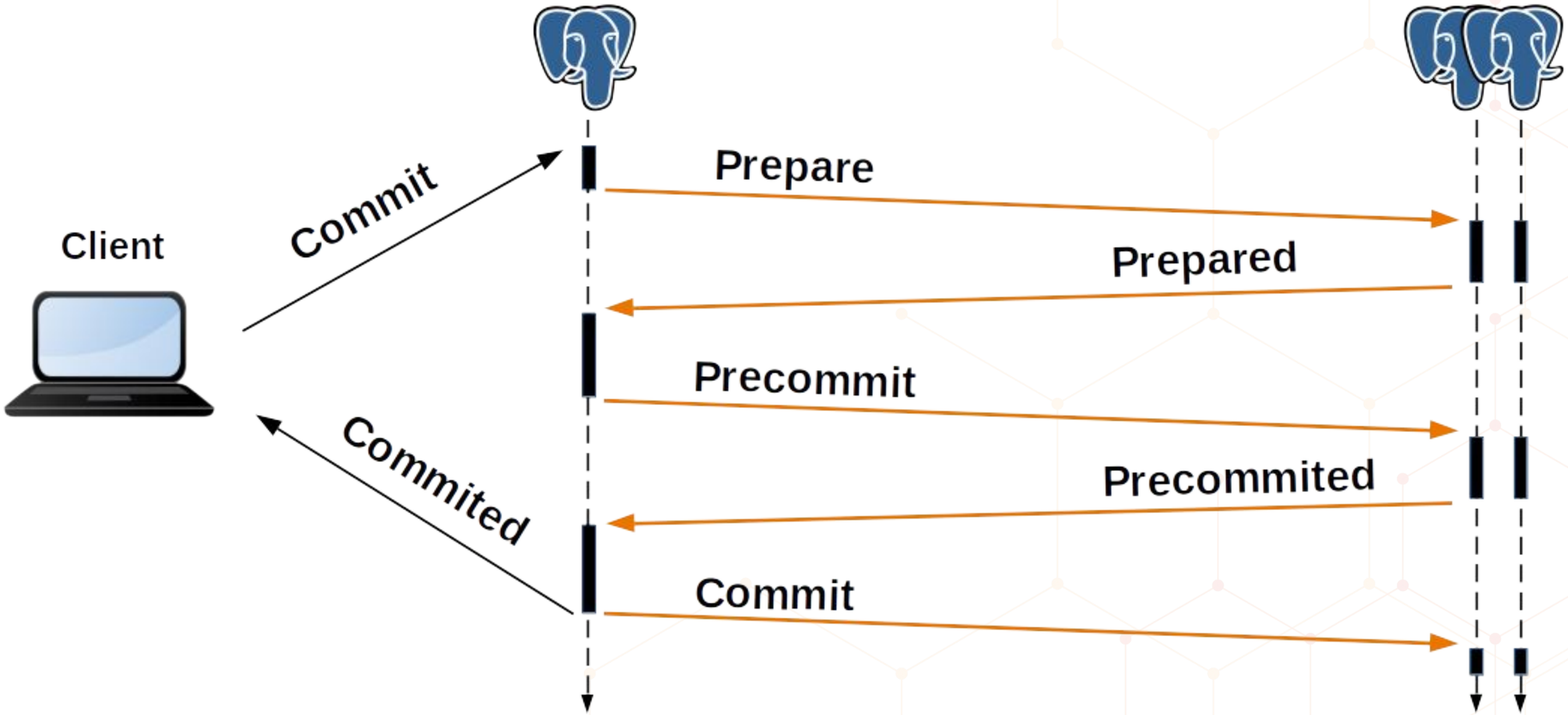
PostgresPro

Согласованность данных

- ACP (Atomic Commit Protocol). Каждая транзакция подтверждается консенсусом нод
- Поколения нод

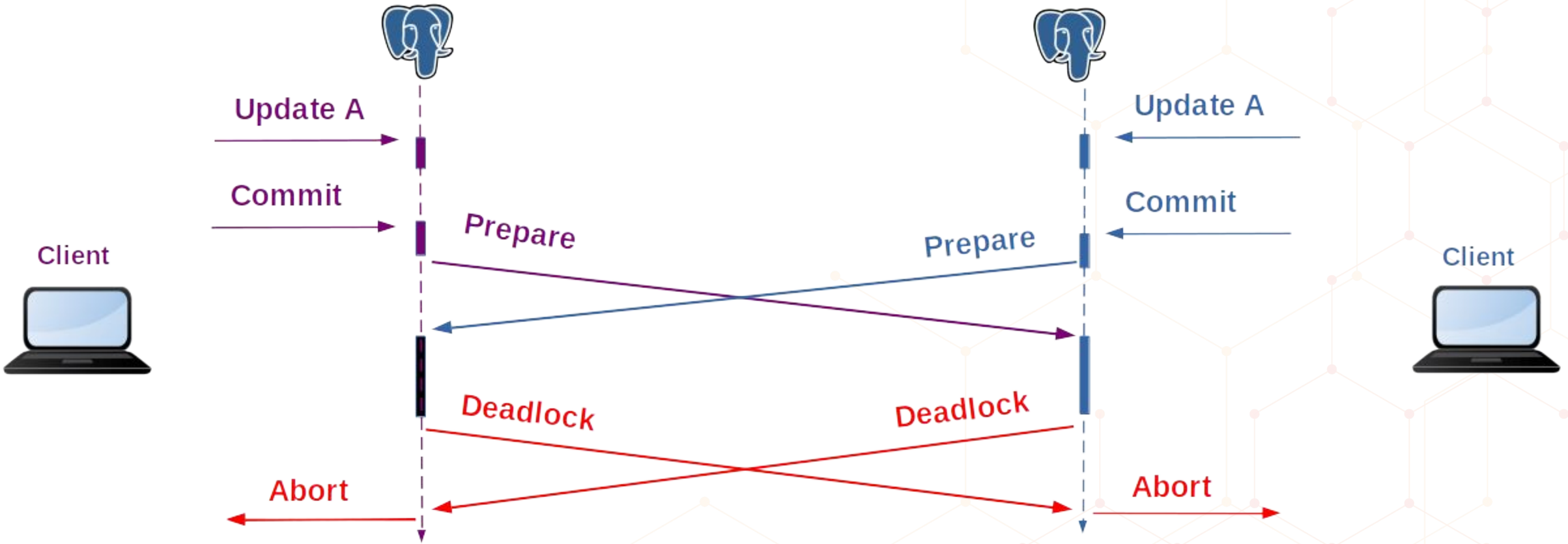
PostgresPro

Commit в Мультимастере



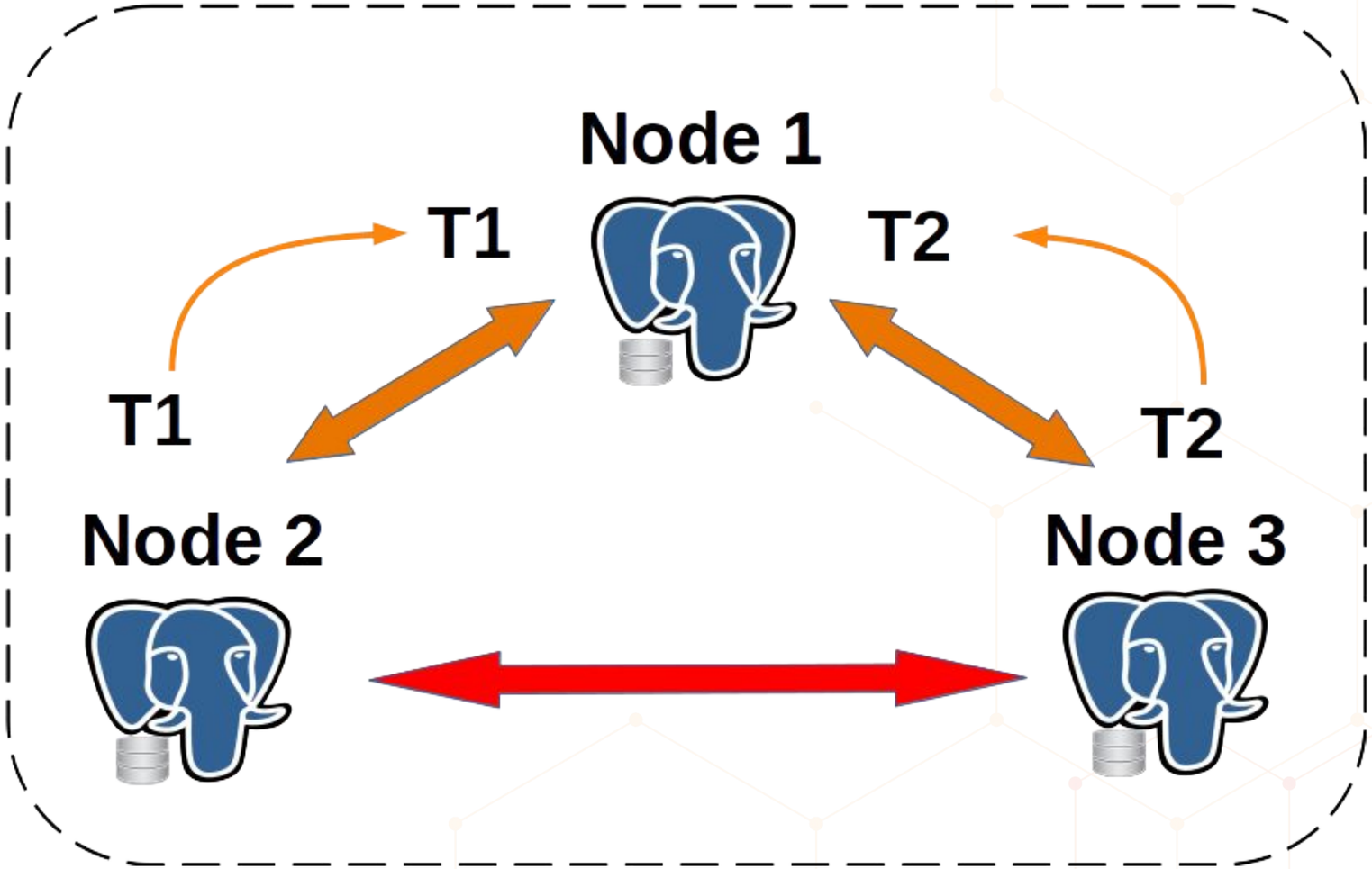
PostgresPro

Конфликт при обновлении



PostgresPro

Нарушение порядка транзакций



PostgresPro

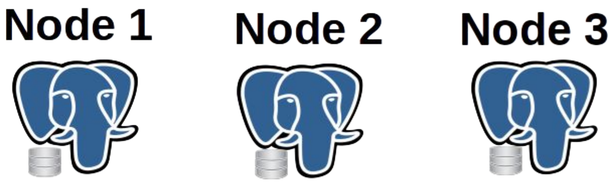
Поколения

- номер поколения - это пара <gen_number, members>
- только эти ноды могут делать транзакции
- нода стремится переключиться в поколение с большим номером
- нода в поколении может быть в нескольких состояниях:
  - ONLINE - нормальная работа
  - RECOVERY - нода вошла в поколение и должна применить недостающие транзакции
  - DEAD - не является членом текущего поколения

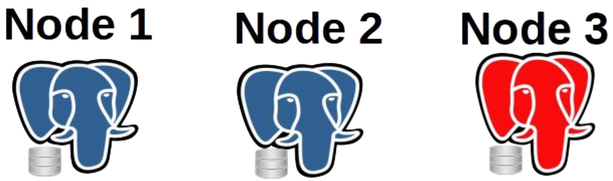
PGConf.Сибирь 2022

PostgresPro

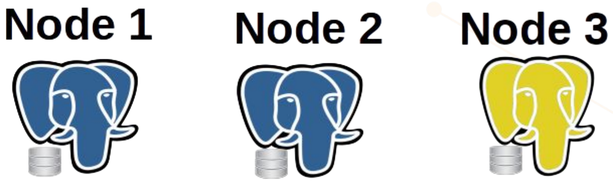
Поколение <1, {1, 2, 3}>



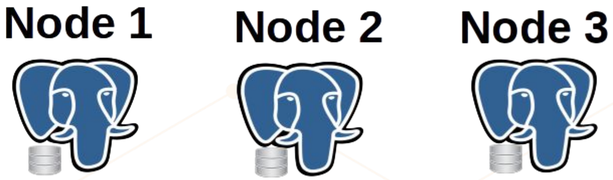
Поколение <2, {1, 2}>



Поколение <3, {1, 2, 3}>



Поколение <3, {1, 2, 3}>



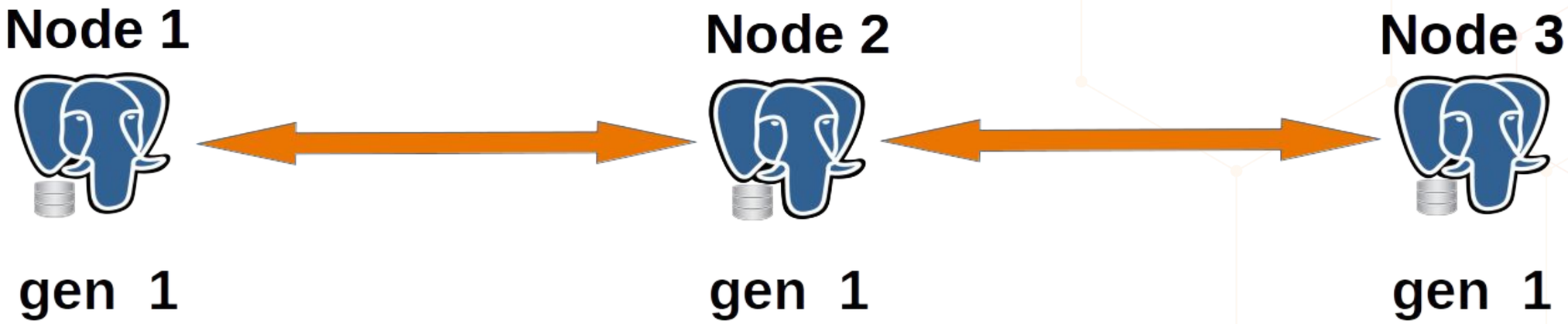
PostgresPro

Поколения

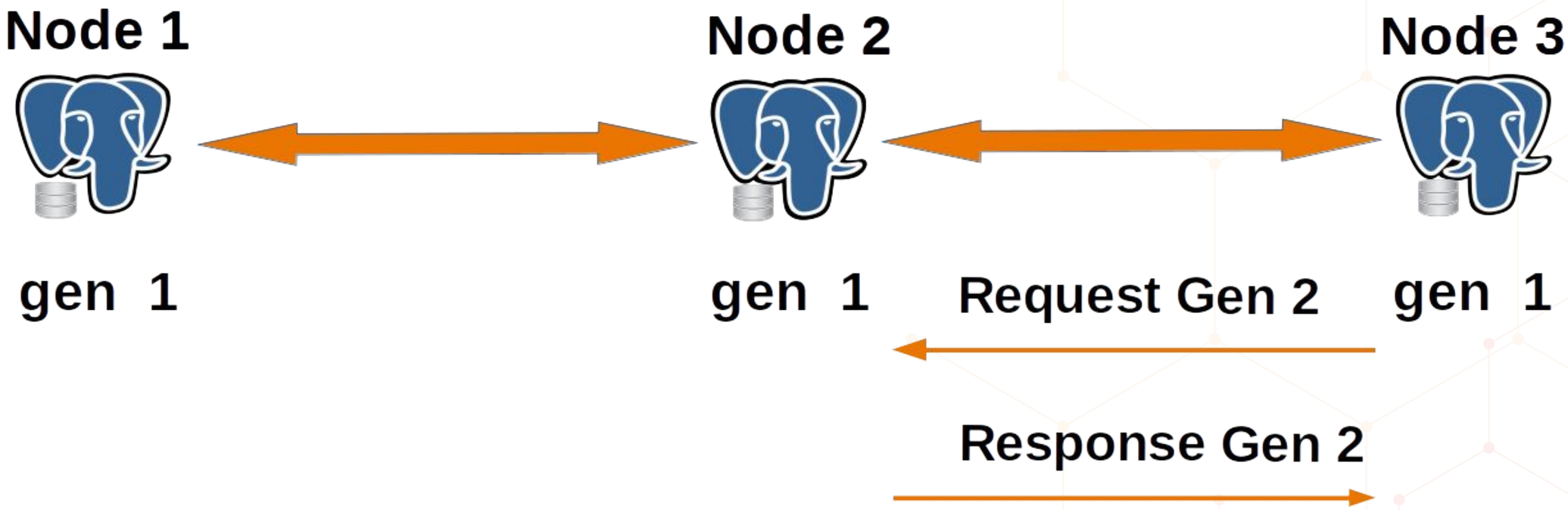
- Нода из поколения N не применяет транзакции поколения N-1
- Поколения состоят из большинства нод

- нода ONLINE в поколении N содержит все транзакции предыдущих поколений
- все новые транзакции будет в поколениях >= N
- все транзакции поколения N будут применены до транзакций поколения M (N < M)
- порядок транзакций внутри поколения обеспечен ACP

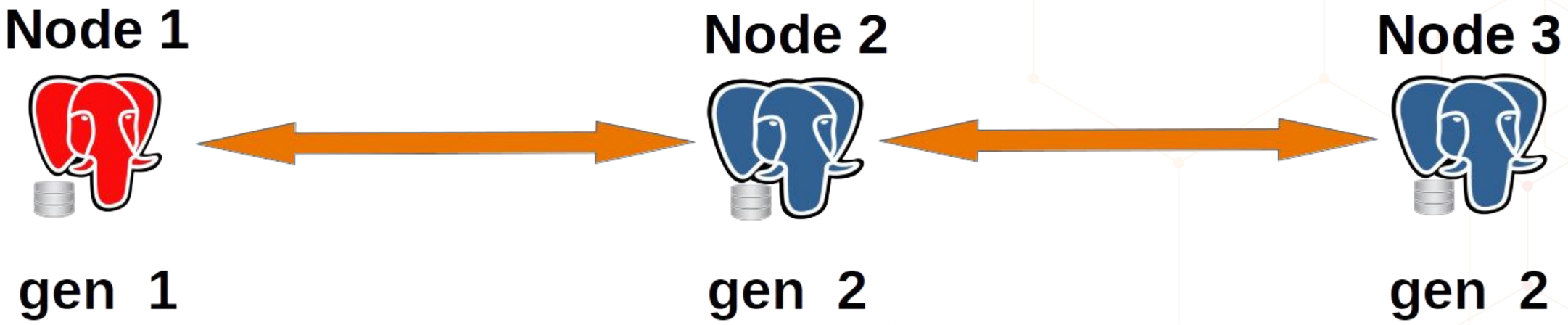
PostgresPro



PostgresPro



PostgresPro



PostgresPro

DDL

- передаются с WAL-ом как текст
- есть нюансы (temp tables, …)

PostgresPro

Где взять мультимастер

- Postgres Pro Enterprise
- Postgresql

https://github.com/postgrespro/mmts

PostgresPro

Подготовка к запуску мультимастера

- подготовить рабочие ноды
- добавить в конфиг postgresql.conf:

```
shared_preload_libraries = 'multimaster'

wal_level = logical
max_connections = 100
max_prepared_transactions = 300
max_wal_senders = 10
max_replication_slots = 10
wal_sender_timeout = 0
```

PostgresPro

Запуск мультимастера

на одной из них выполнить:

```
> create extension multimaster;
> select mtm.init_cluster(
      'host=node1',
        '{"host=node2", host3=node3"}');
```

PostgresPro

Мониторинг мультимастера

Функция **mtm.ping()** - определение работоспособности кластера:

```
demodb=# select * from mtm.ping();
 ping
------
 t
(1 row)
```

PostgresPro

Мониторинг мультимастера

Функция **mtm.status()** - запрос статуса нод:

```
demodb=# select * from mtm.status();
 my_node_id | status | connected | gen_num | gen_members | gen_members_online | gen_configured
------------+--------+-----------+---------+-------------+--------------------+---------------
          1 | online | {1,2,3}   |       3 | {1,2,3}     | {1,2,3}            | {1,2,3}
(1 row)
```

PostgresPro

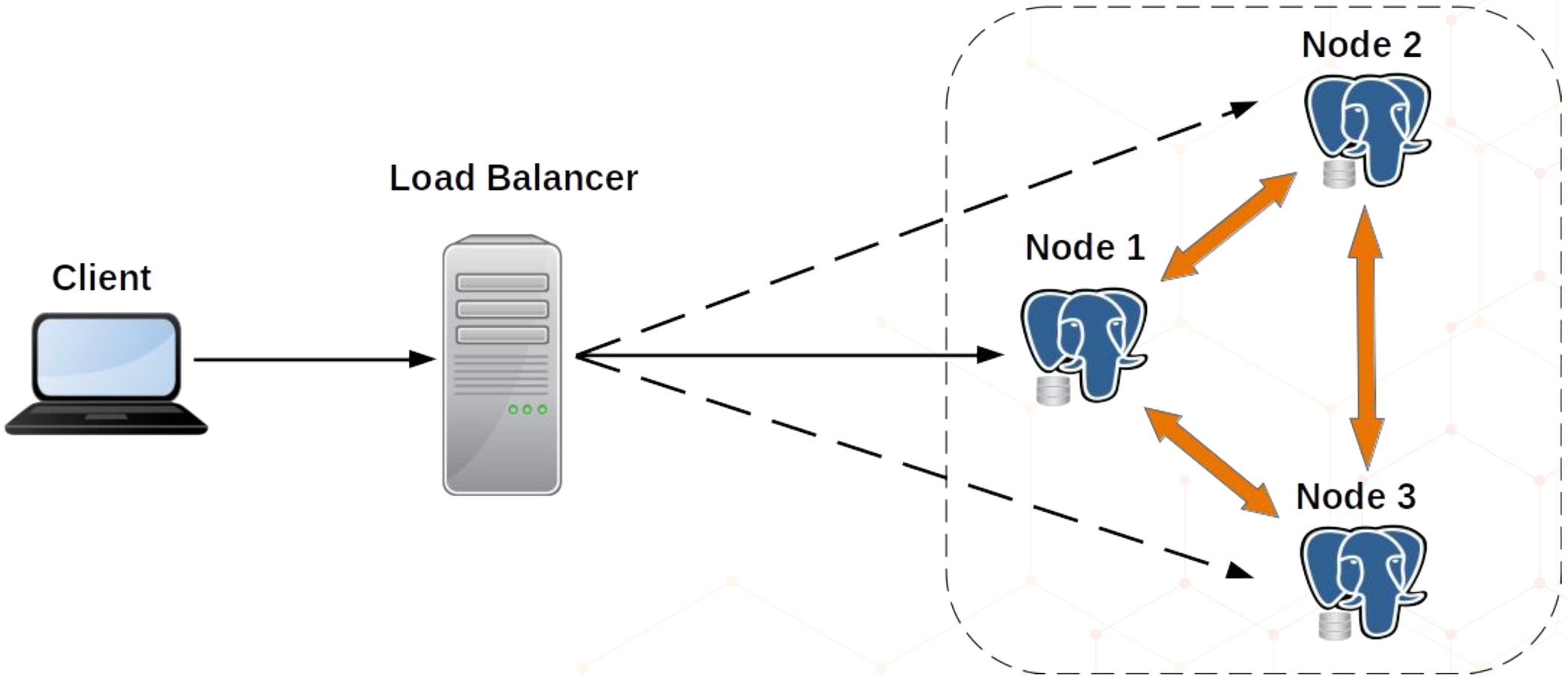
Мониторинг мультимастера

Функция **mtm.nodes()** - запрос списка нод кластера их состояния:

```
demodb=# select * from mtm.nodes();
 id |                  conninfo                  | is_self | enabled | connected | sender_pid | receiver_pid | n_workers | receiver_mode
----+--------------------------------------------+---------+---------+-----------+------------+--------------+-----------+---------------
  1 | port=65432 host=127.0.0.1 dbname='demodb' | t       | t       | t         |            |              |           |
  2 | port=65433 host=127.0.0.1 dbname='demodb' | f       | t       | t         |    1939895 |      1939795 |         0 | normal
  3 | port=65434 host=127.0.0.1 dbname='demodb' | f       | t       | t         |    1939797 |      1939794 |         0 | normal
```

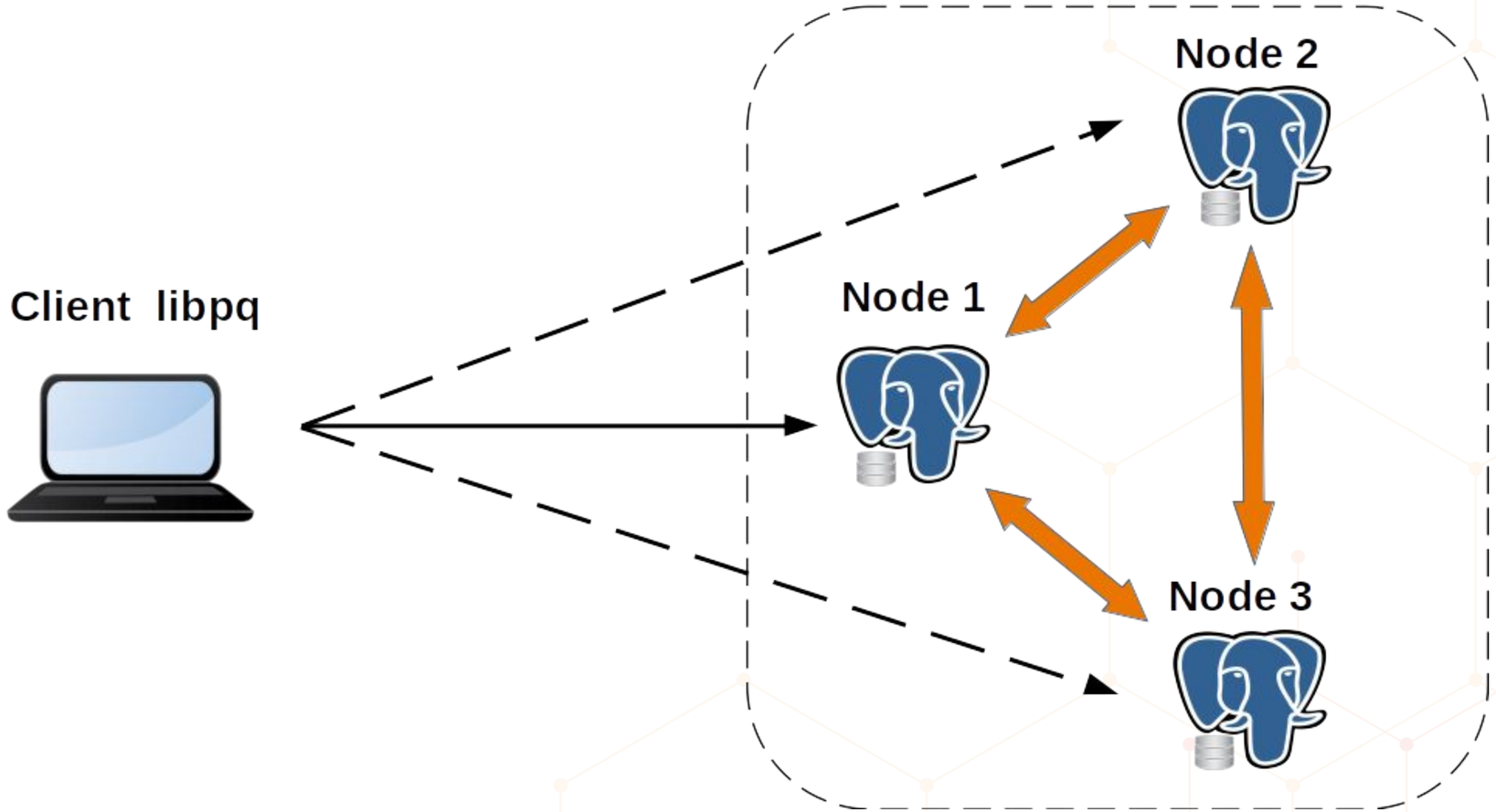
PostgresPro

Схема включения. 3-х нодовый кластер



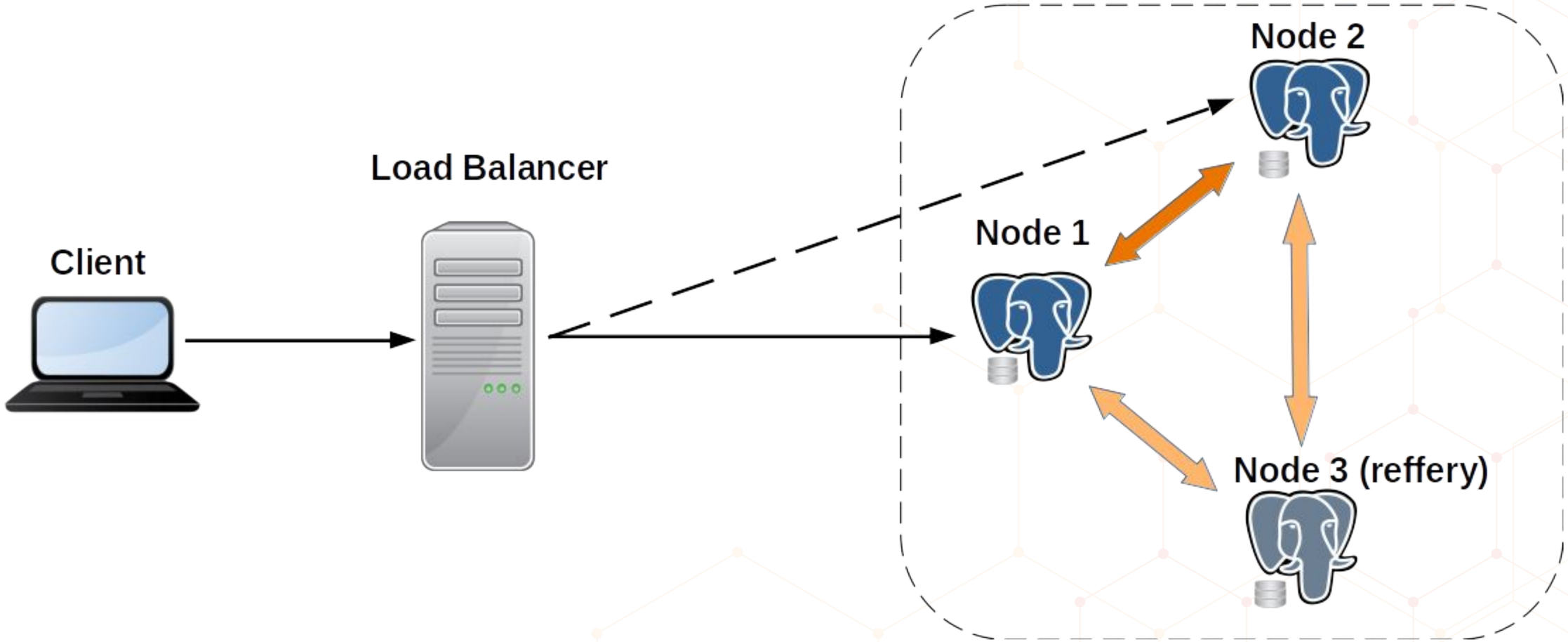
PostgresPro

Схема включения. 3-х нодовый кластер



PostgresPro

Схема включения. 2 ноды + рефери



PostgresPro

Применение мультимастера

Отказоустойчивый кластер	да
Катастрофоустойчивый кластер	нет
Виртуальные среды	да
Облачные сервисы	да
Непрерывность сервиса	да

PostgresPro

Заключение. Особенности мультимастера

- быстрый failover
- не требуется дополнительного ПО
- запись на все ноды
- кластер доступен пока живо большинство нод
- можно добавить ноду “на лету”.
- автоматическое восстановление ноды после сбоя
- совместимсть с Postgres

PGConf.Сибирь 2022

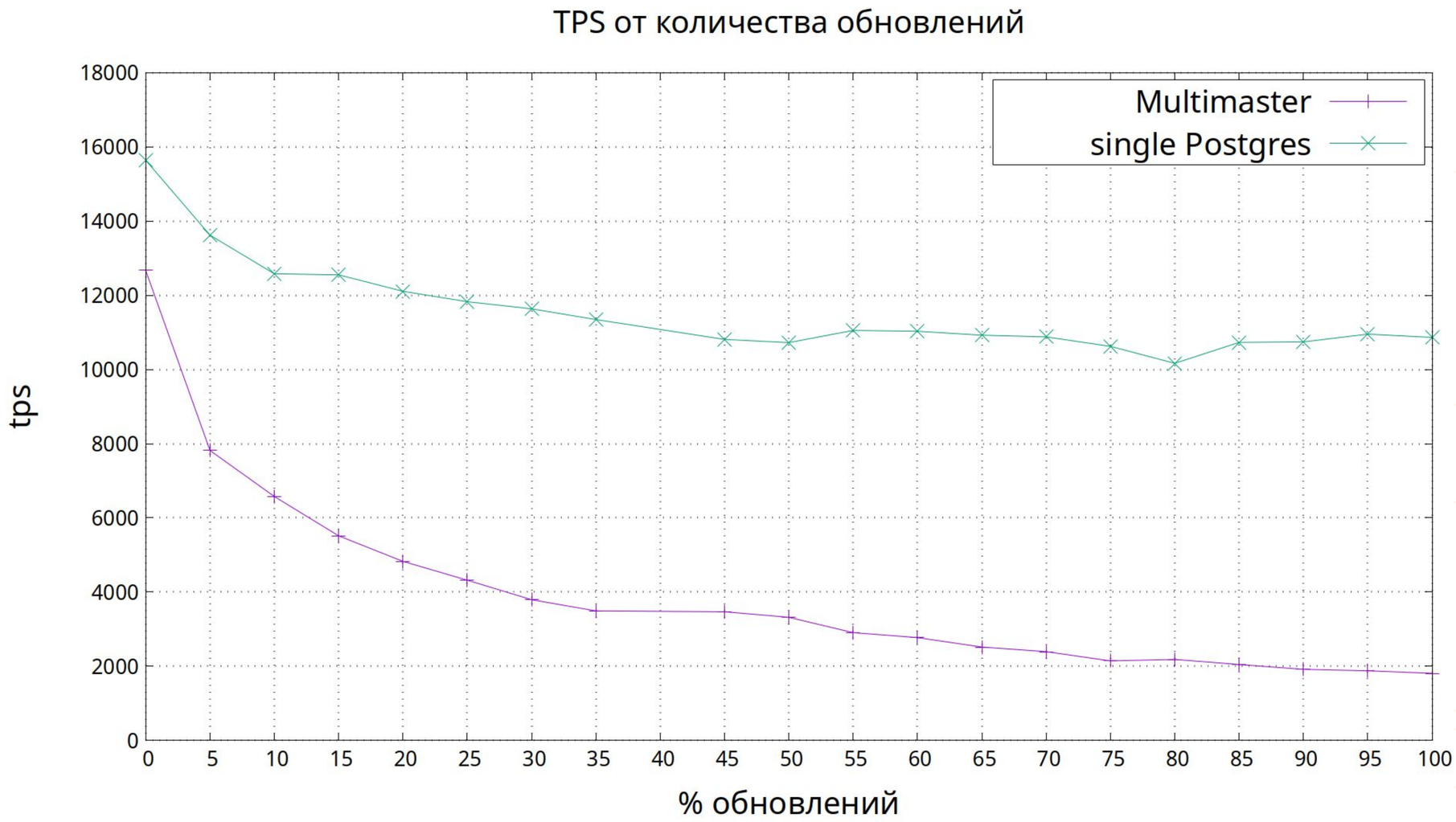
СПАСИБО!

Михаил В. Рутман:
m.rutman@postgrespro.ru

postgrespro.ru

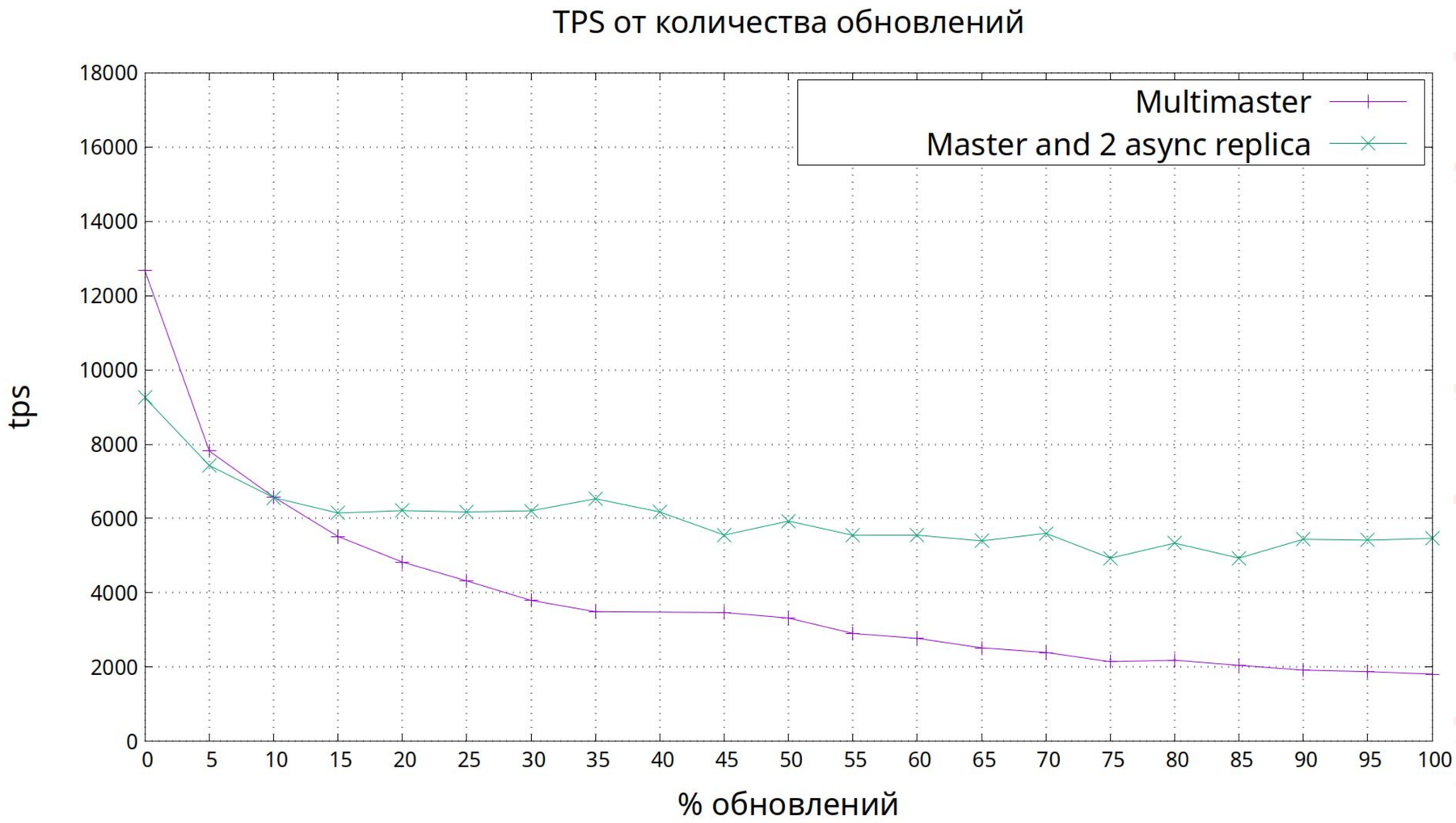
PostgresPro

Замеры



PostgresPro

Замеры



PostgresPro

Замеры