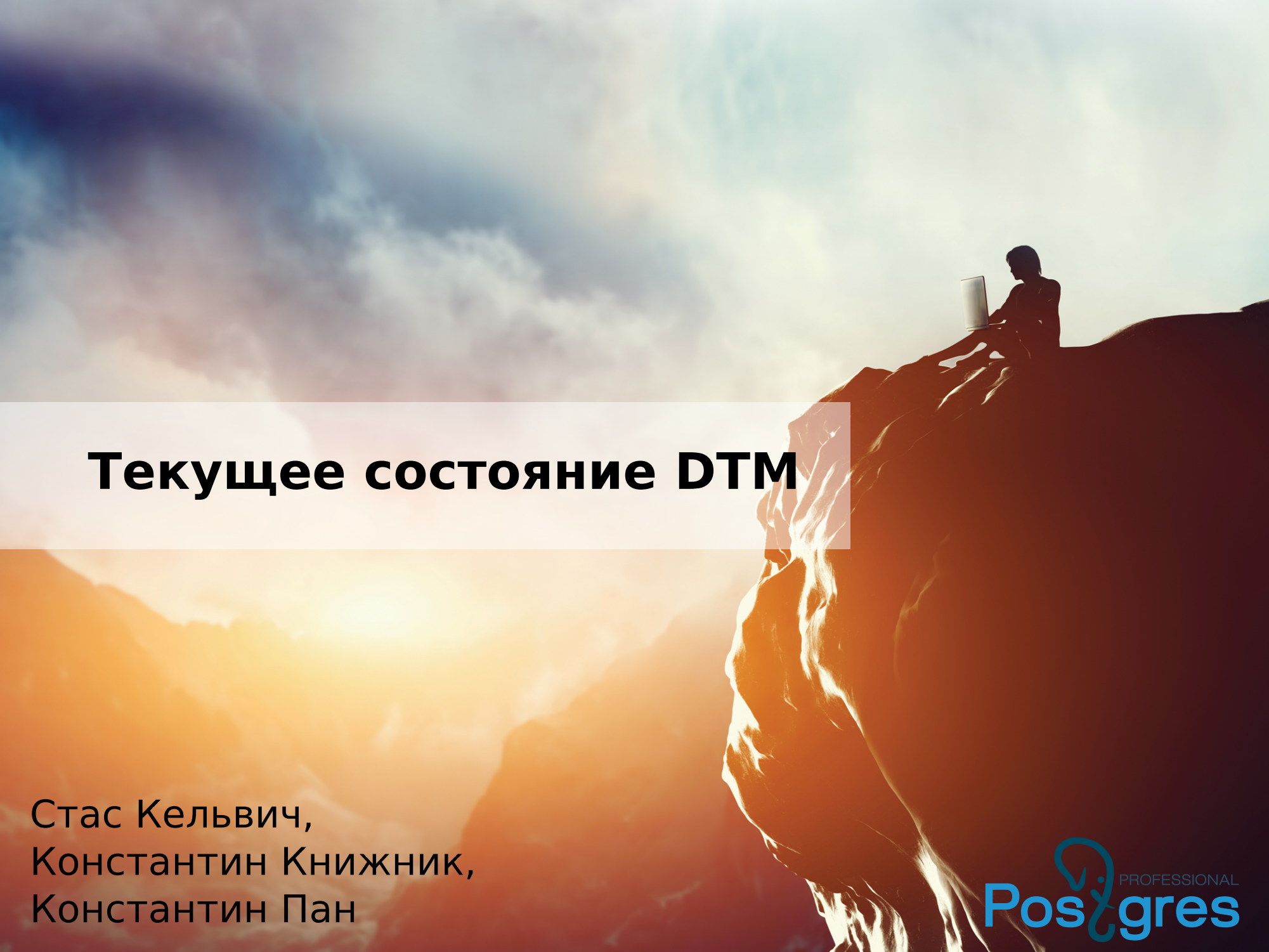


Текущее состояние DTM

Стас Кельвич,
Константин Книжник,
Константин Пан

PROFESSIONAL Postgres

Задача

- ACID-кластер

Пример

- таблица accounts (id, amount)
- разбита по id на три сервера

Переводим деньги с одного счёта на другой.
Возможно, с одного сервера на другой.

Атомарность

COMMIT происходит либо на всех участвующих узлах, либо ни на одном.

Согласованность и изоляция

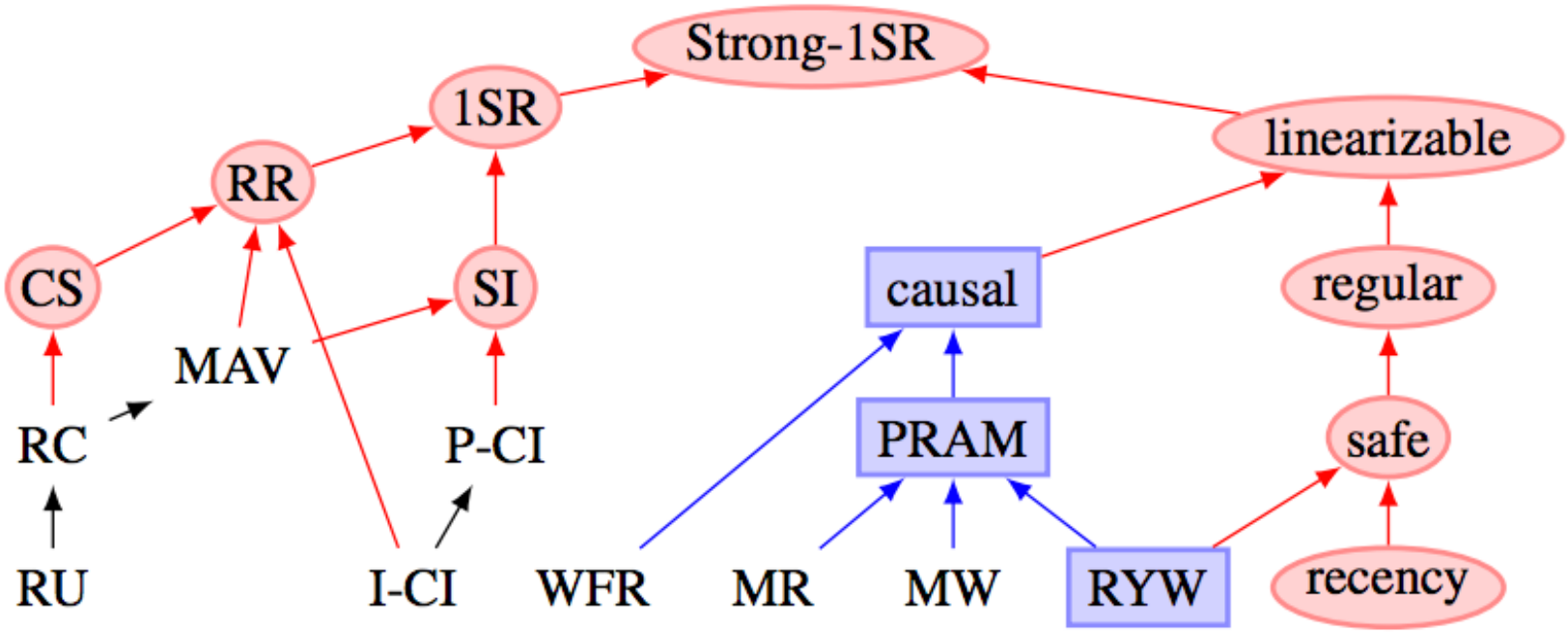
```
begin;
update acc set a = a -
100 where id = 1;

update acc set a = a +
100 where id = 2;
commit;
```

```
select sum(a) from acc;
```

Уровни изоляции

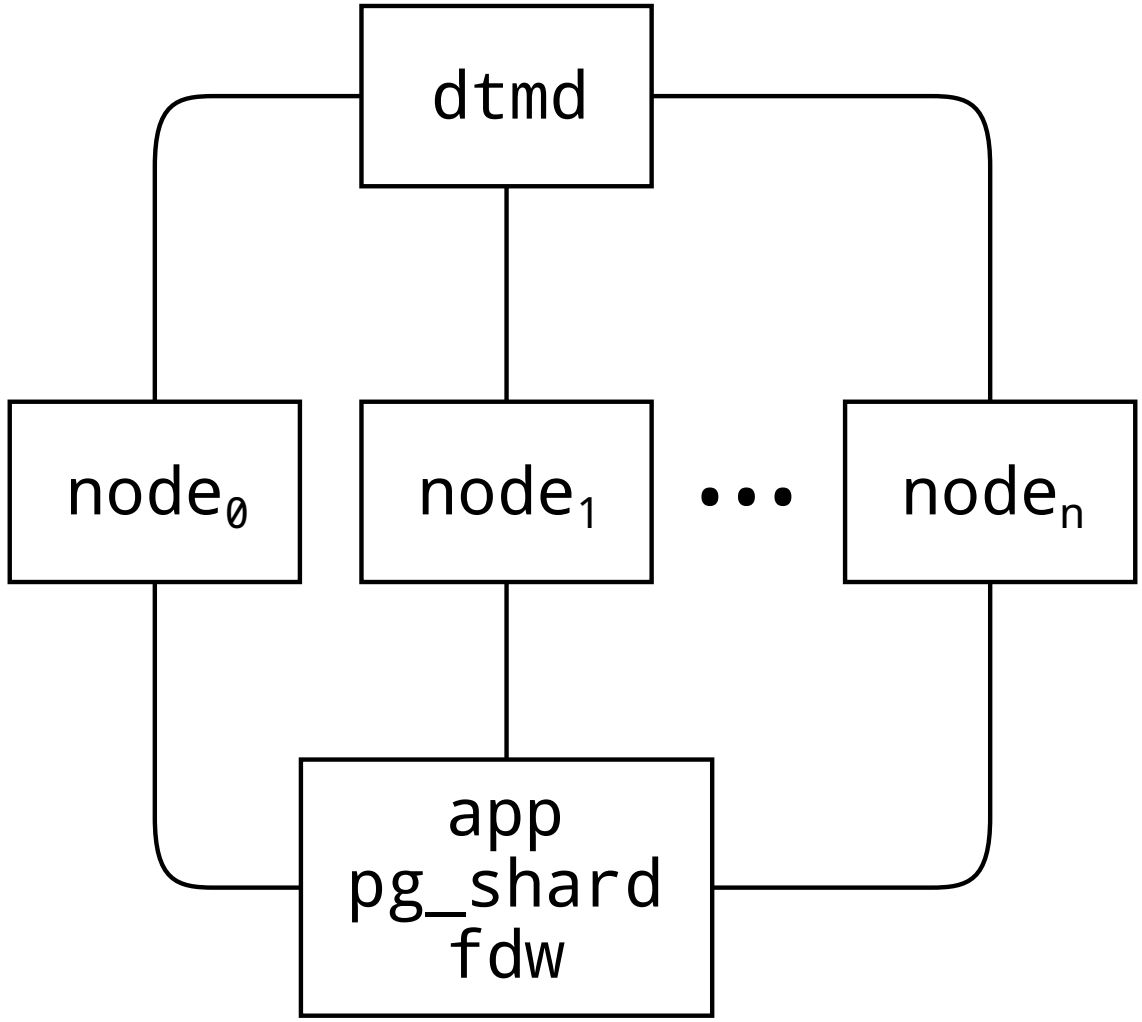
Isolation Level	Dirty Read	Nonrepeatable Read	Phantom Read	Serialization Anomaly
Read uncommitted	Allowed, but not in PG	Possible	Possible	Possible
Read committed	Not possible	Possible	Possible	Possible
Repeatable read	Not possible	Not possible	Allowed, but not in PG	Possible
Serializable	Not possible	Not possible	Not possible	Not possible

Уровни изоляции



(Peter Bailis, VLDB, 2013)

Структура кластера

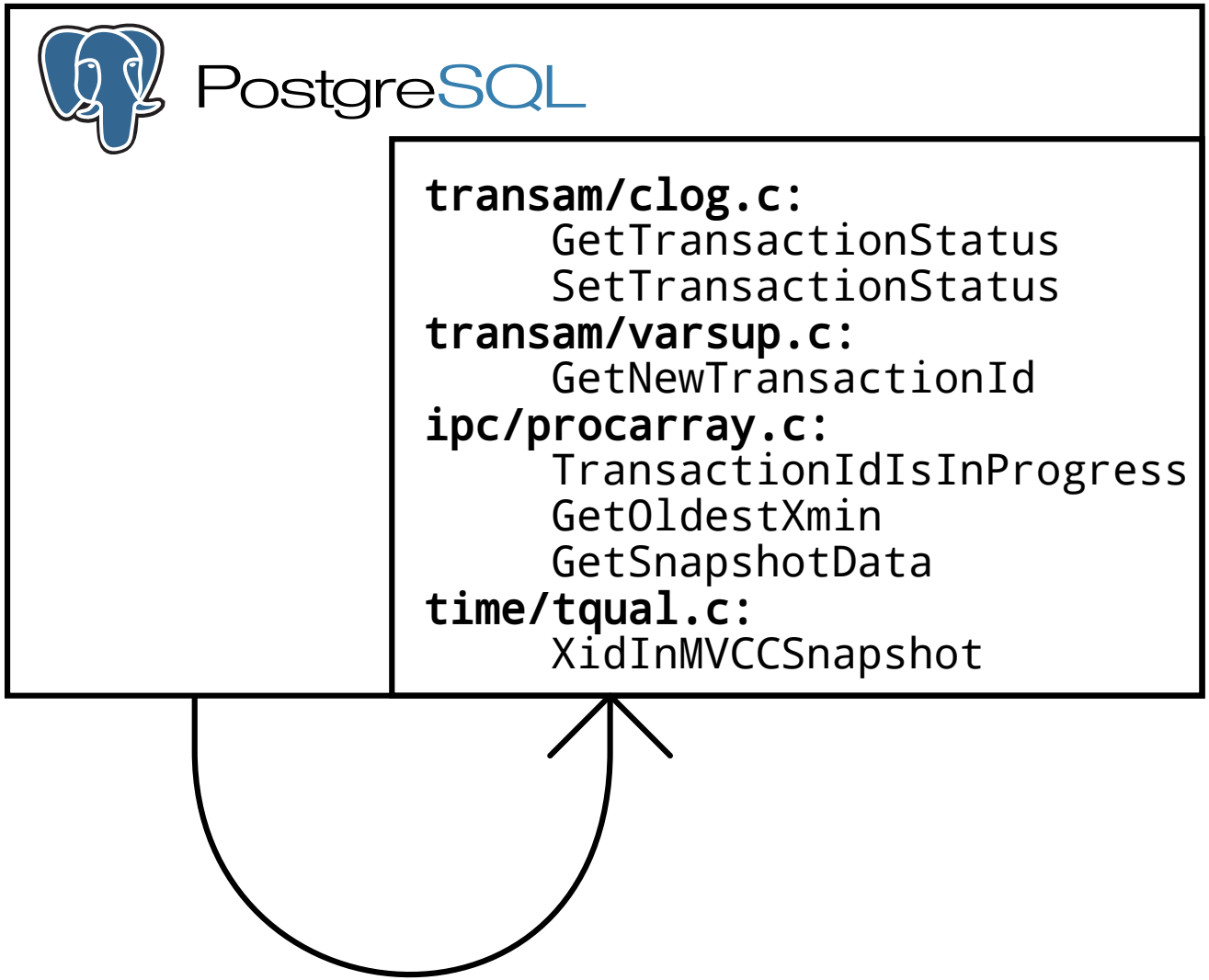


С точки зрения клиента

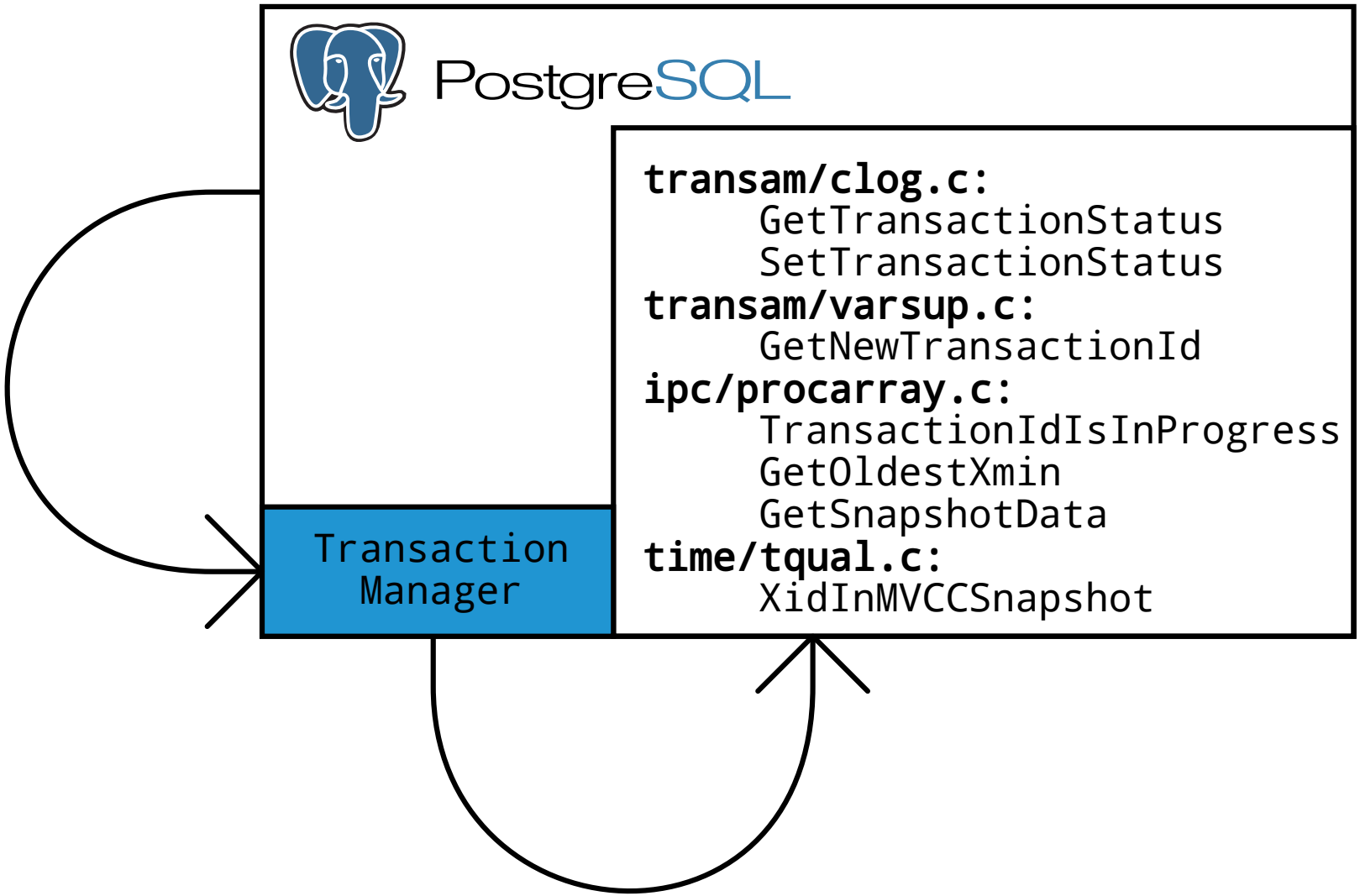
src server	dst server
`create extension pg_dtm;`	`create extension pg_dtm;`

`select begin_glob...();`
 	
`begin`
 	
`update ...;`
 	
`commit;`
	`select join_glob...(xid);`
	`begin;`
	
	`update ...;`
	`commit;`

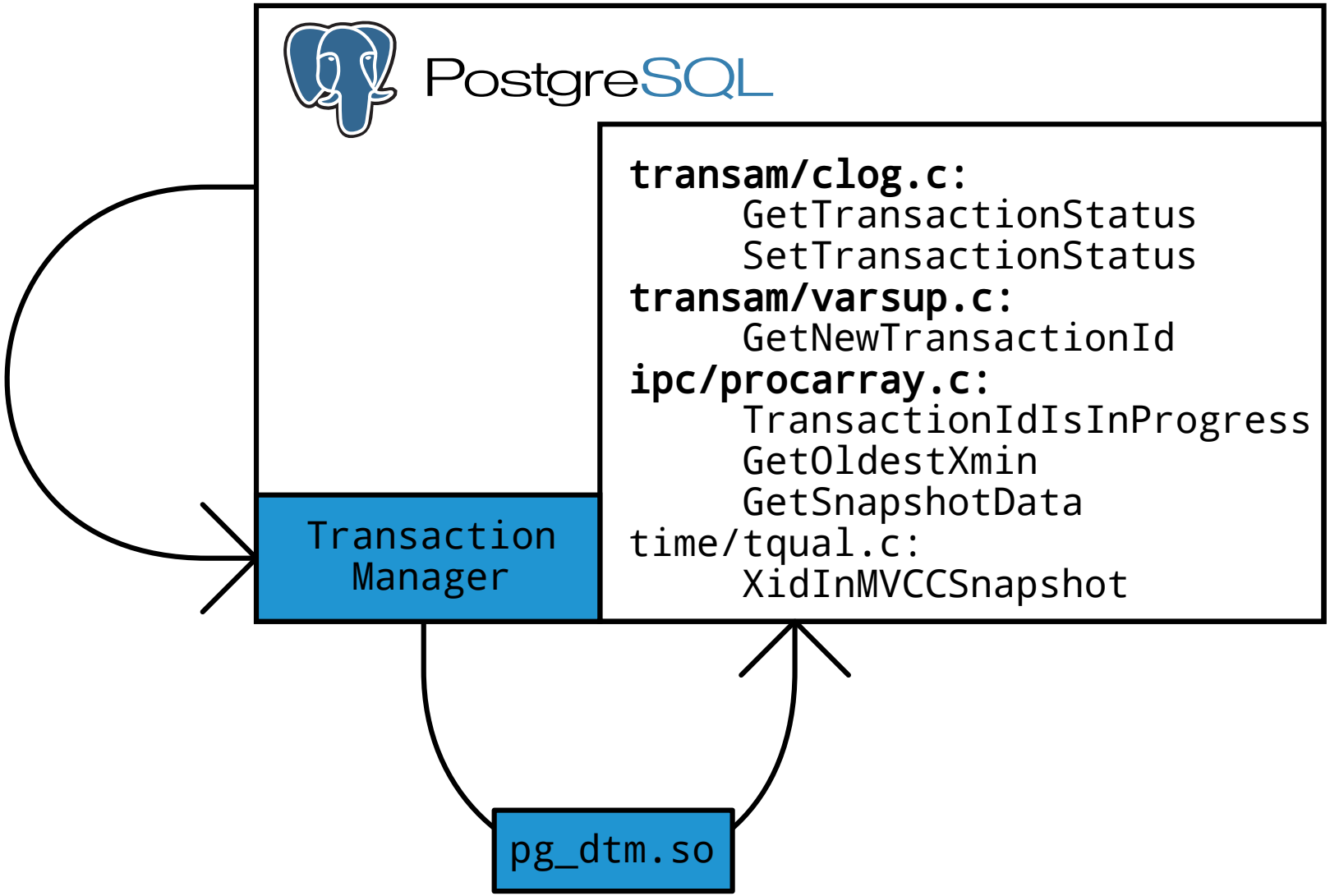
С точки зрения кода: vanilla



С точки зрения кода: после patch



С точки зрения кода: после preload .so

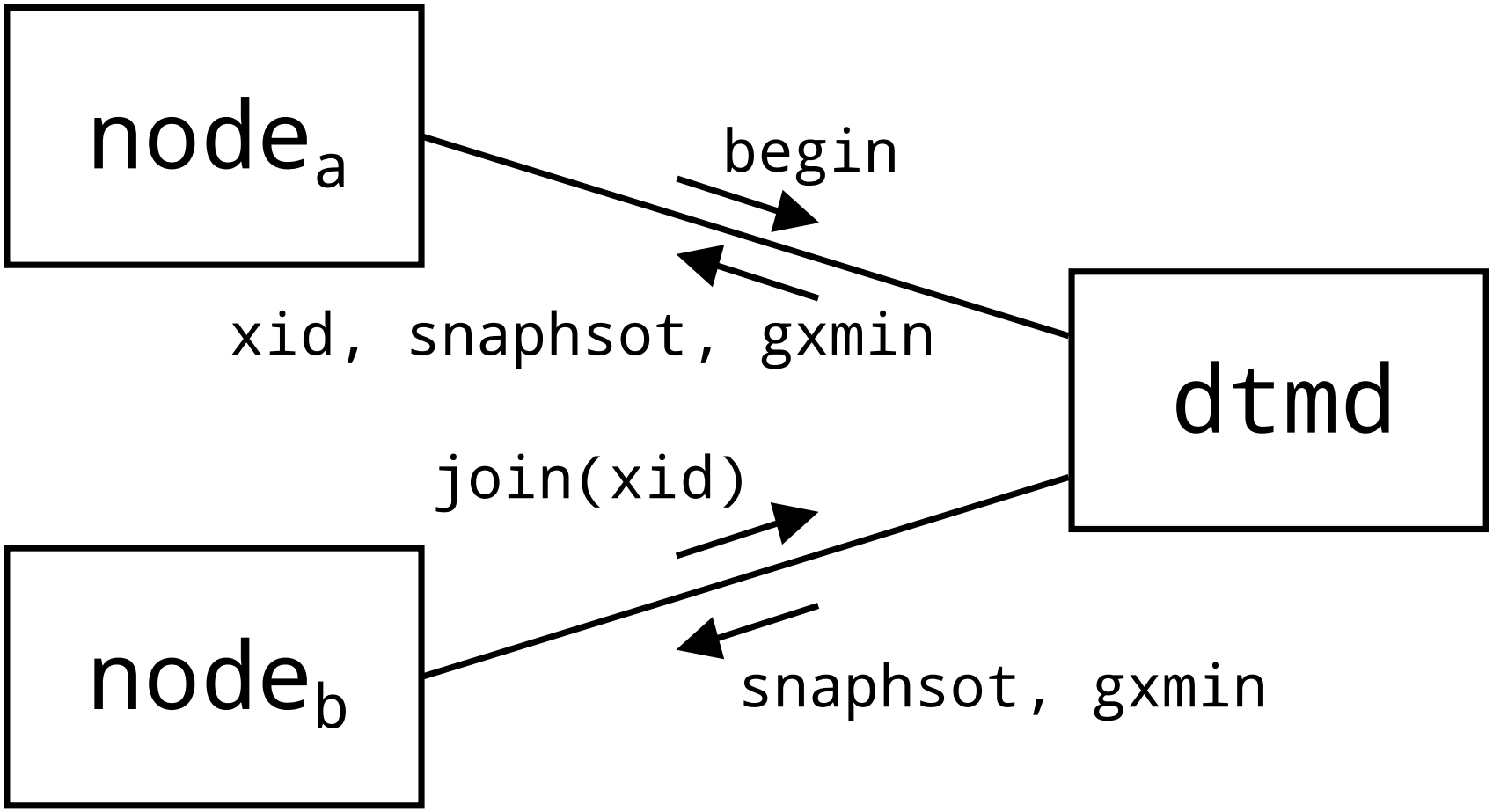


С точки зрения кода: реализации транзакций

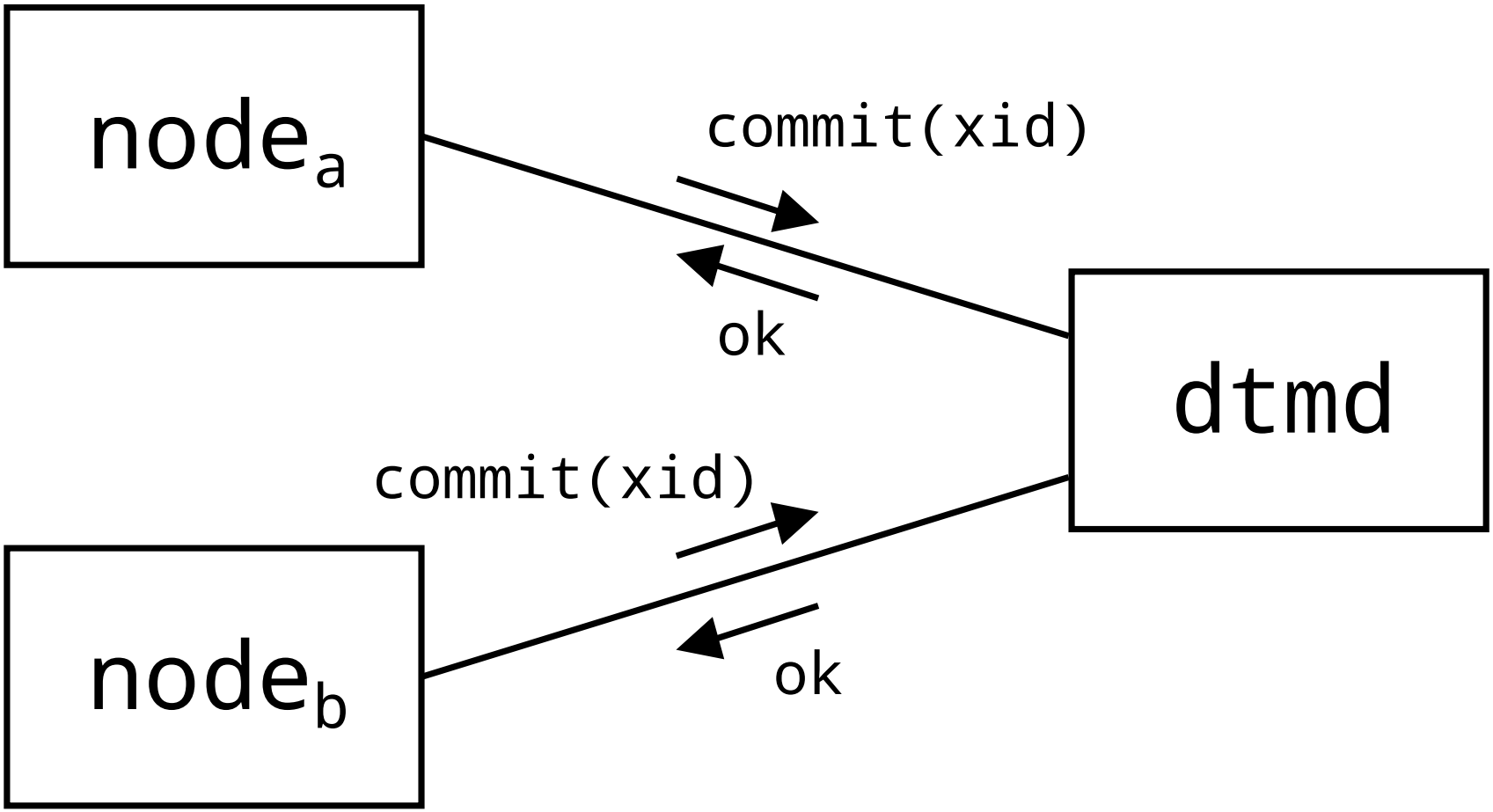
В качестве TM можно прикрутить разные реализации:

- **Snapshot Sharing** (XL, DTM)
  no 2pc, no local, daemon
- **Timestamp** (Spanner, Cockroach)
  2pc, local, no daemon
- **Incremental** (SAP HANA)
  no 2pc, locals, increments

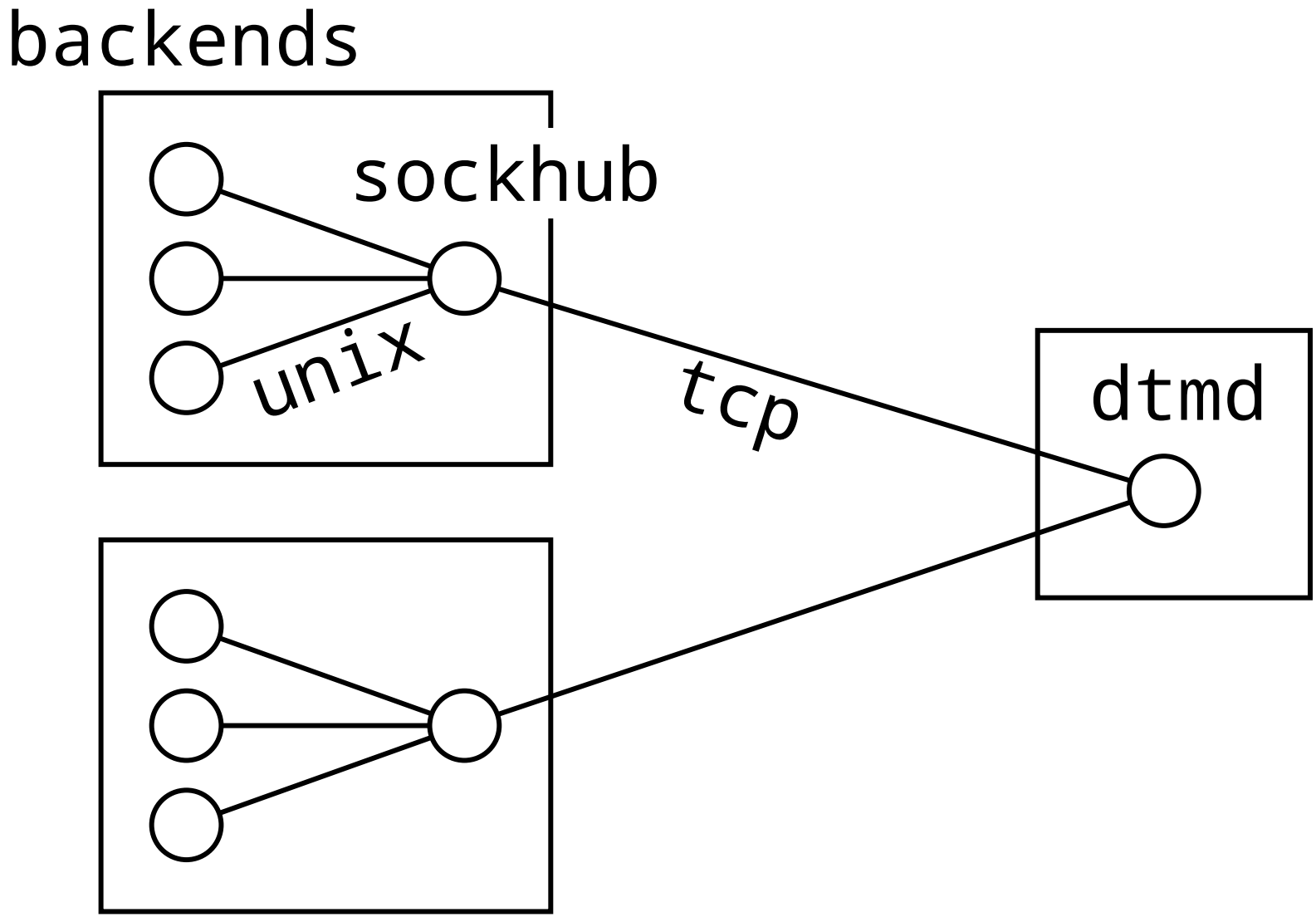
Протокол (начало)



Протокол (конец)

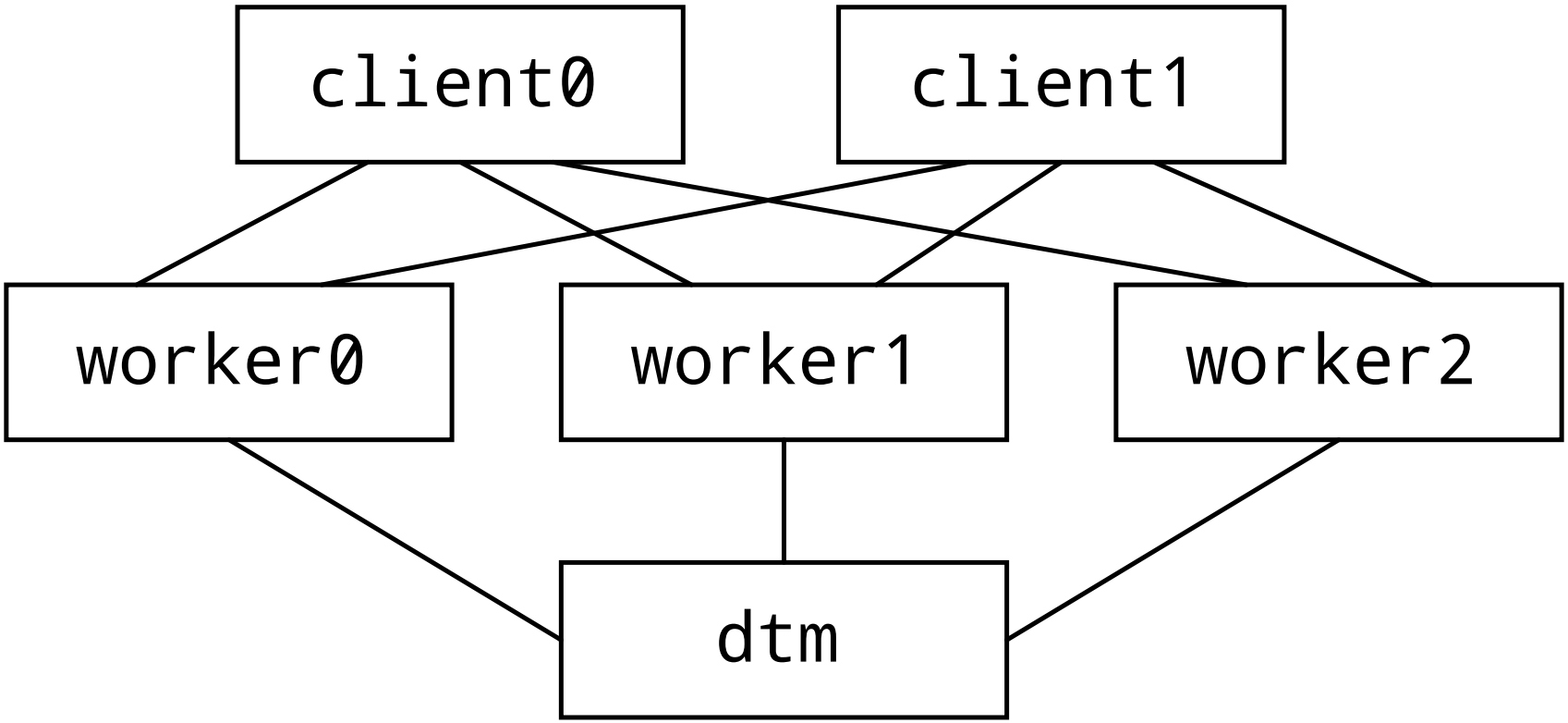


Мультиплексирование

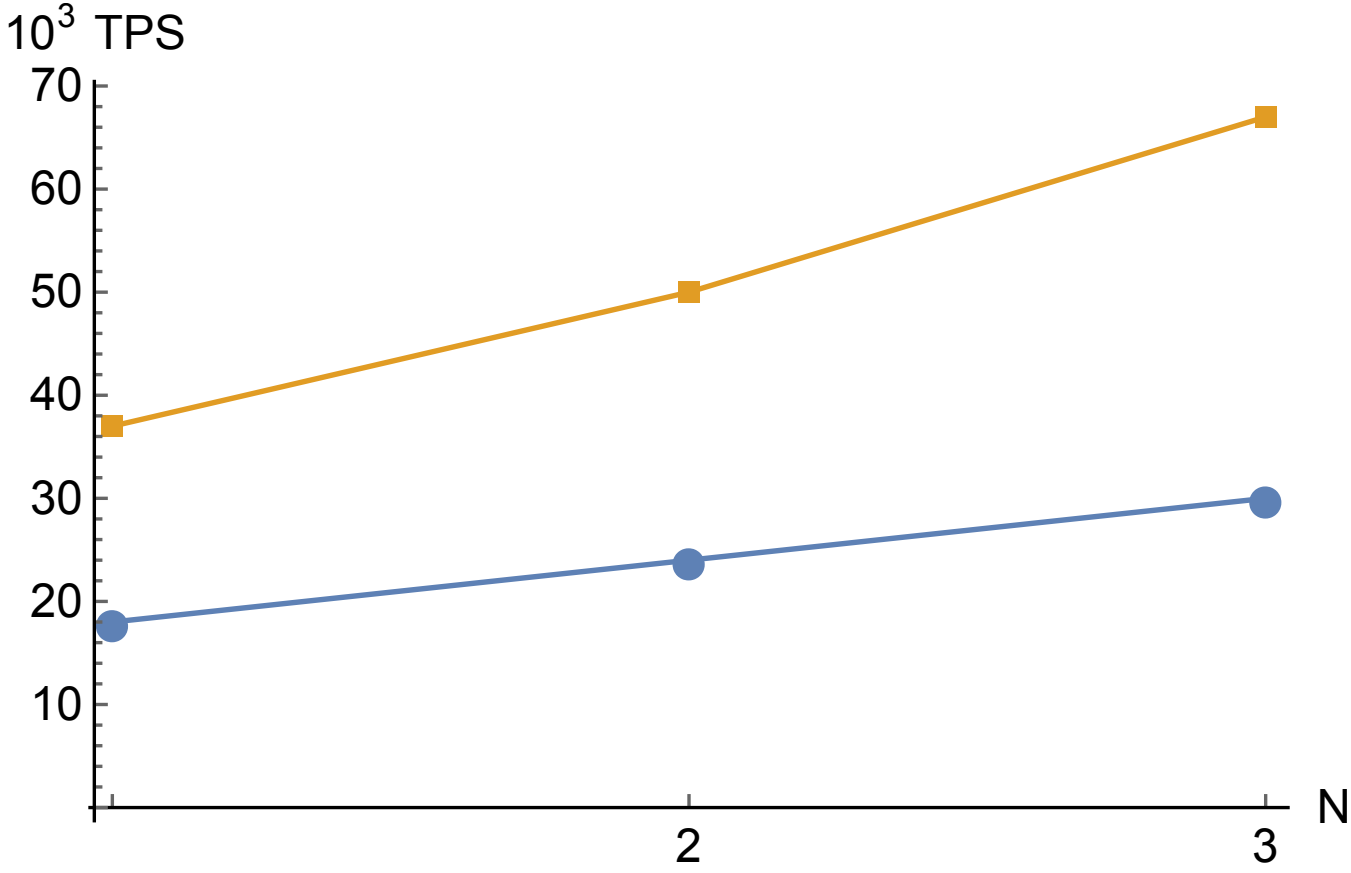


Тесты

Конфигурация



Результаты



желтый — без DTM
синий — с DTM

Результаты

- pg_shard: 17 k TPS
- fdw: 15 k TPS

Ограничения

Переключение контекста

На этом тесте без DTM упираемся в Context Switches. Включение DTM = x2 переключений контекста на транзакцию. Т.е. в 2 раза меньше tps с сервера.

Что можно сделать? Писать в очередь в общей памяти и будет быстрее.

CPU

DTM однопоточный.

Пусть нужно 3k ops на транзакцию (оптимистично).

3 GHz / 3 k.ops = 1M TPS

Сеть

1 транзакция: 112 + 8*N bytes
N ≈ 400
около 1700 байт на транзакцию

- 1Gb: 300k tps
- 10Gb: 3M tps

Репликация

...