

SMARTDATA 2026 · 24 СЕНТЯБРЯ

Читать быстрее, чем отдаёт Серh

Как мы построили S3-шардирование
без дополнительной инфраструктуры

Дмитрий Листвин

Big Data Engineer, AvitoTech

dwh-support

2195 участников

A

Аналитик 09:41

Запрос в Trino бежит уже час, @dwh
посмотрите, пожалуйста

E

ETL-инженер 09:43

Мне в этом спринте нужно перелить витрину, а
места в S3 нет

M

Менеджер 09:47

У нас дашборды до сих пор не обновились,
посмотрите **ASAP**

A

Аналитик 09:52

Как `select * limit 100` может выполняться
10 минут?

ML

ML-инженер 09:55

Не можем залить датасеты, пишет, что нет
места, что делать?

Написать в #dwh-support...

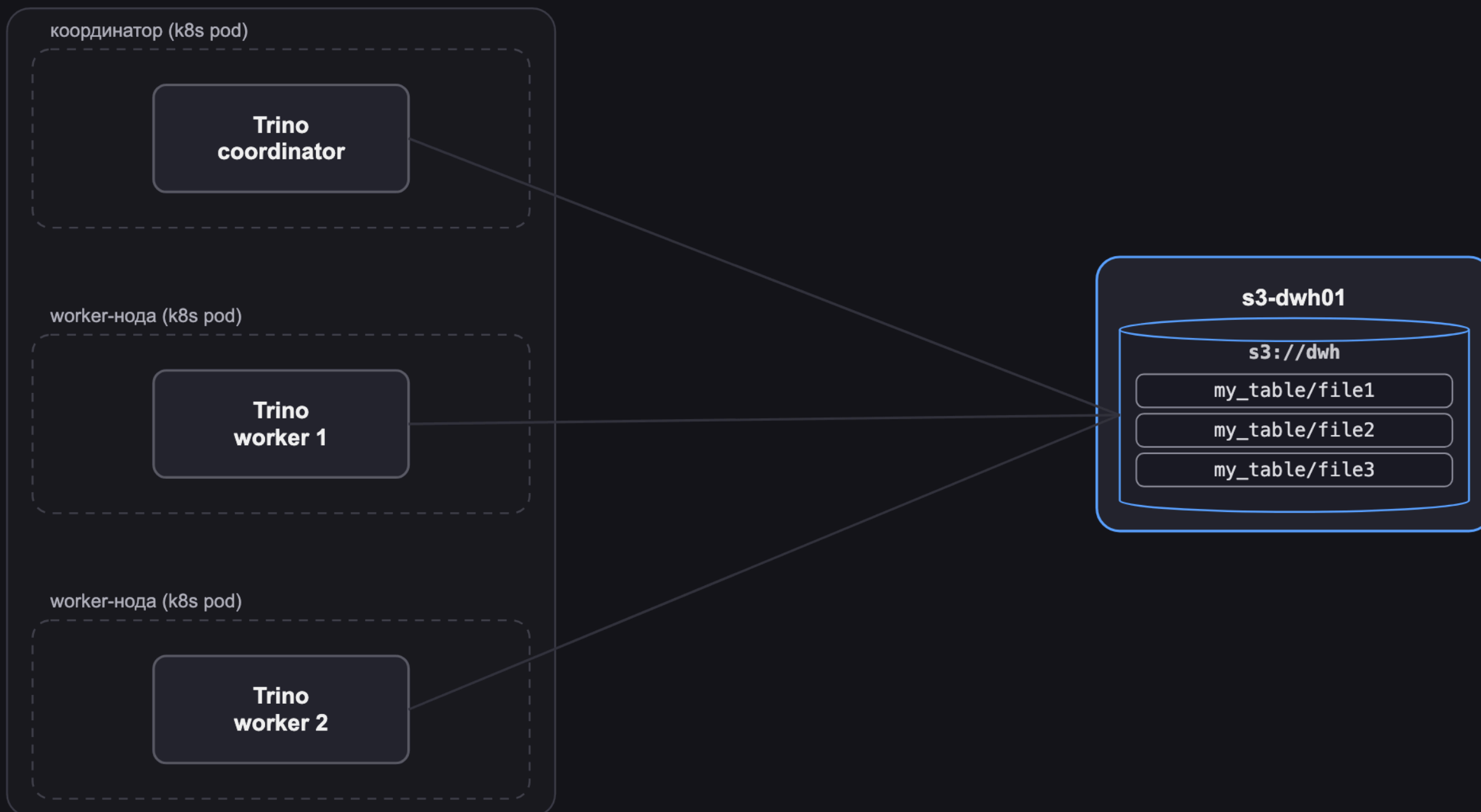
МЕСТО

Не хватает места в
хранилище

СКОРОСТЬ

Не хватает скорости, с
которой хранилище отдаёт
данные

кластер Trino



Воркеры качают файлы напрямую из Сепх

✓ Модель
Trino + Сепх✓ Planning
листинг таблицыExecution
воркеры читают данные

1

Хранить лучше

форматы, компрессия, компакшн мелких файлов

ДЕЛАЕМ

2

Читать меньше

партиционирование, статистики, pruning

ДЕЛАЕМ

3

Кэшировать

кэш на воркерах, Alluxio

ДЕЛАЕМ

4

Холодное хранилище

tiering по возрасту данных

ДЕЛАЕМ

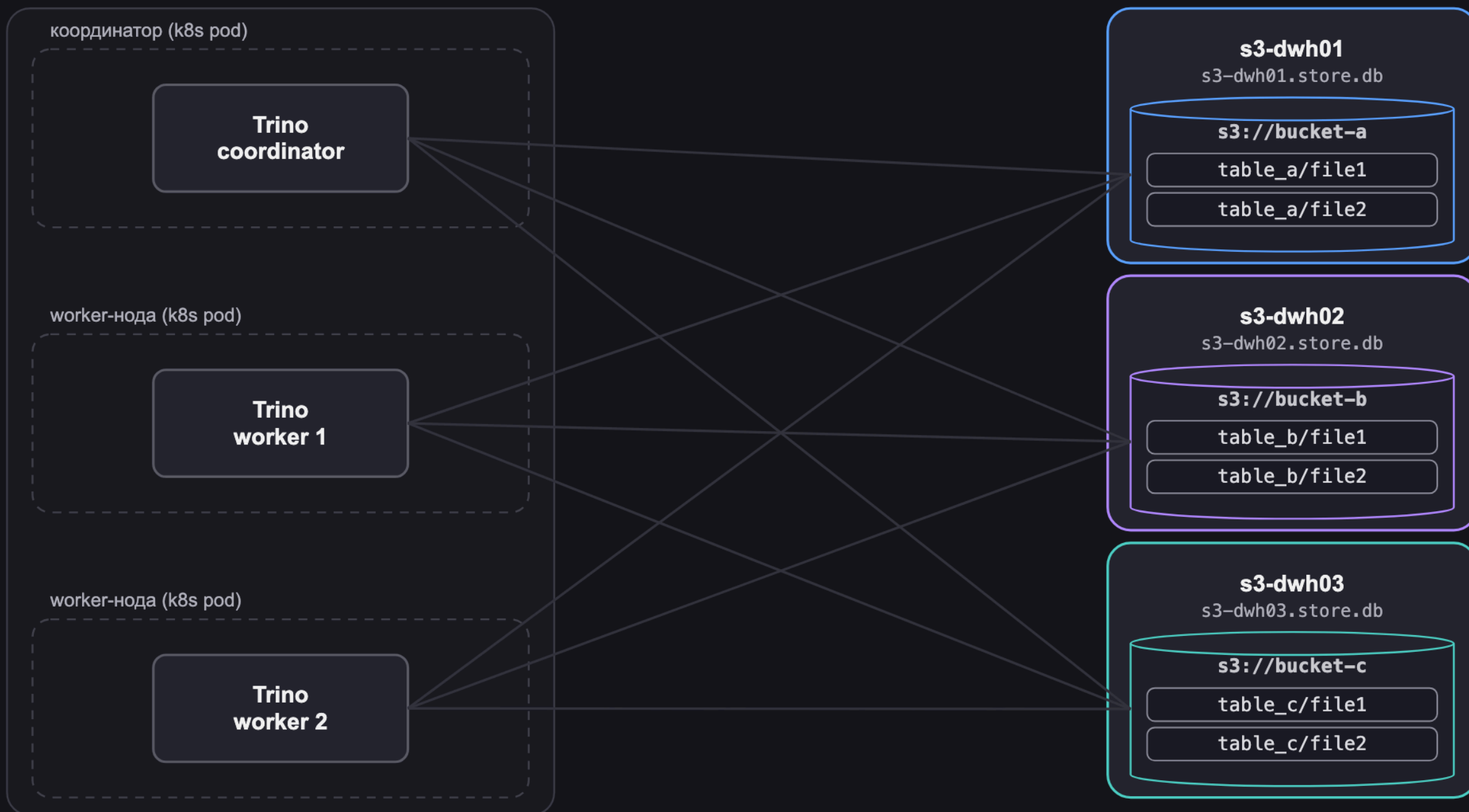
5

Масштабировать S3 горизонтально

несколько Серв за одним именем

ЭТОТ ДОКЛАД

кластер Trino



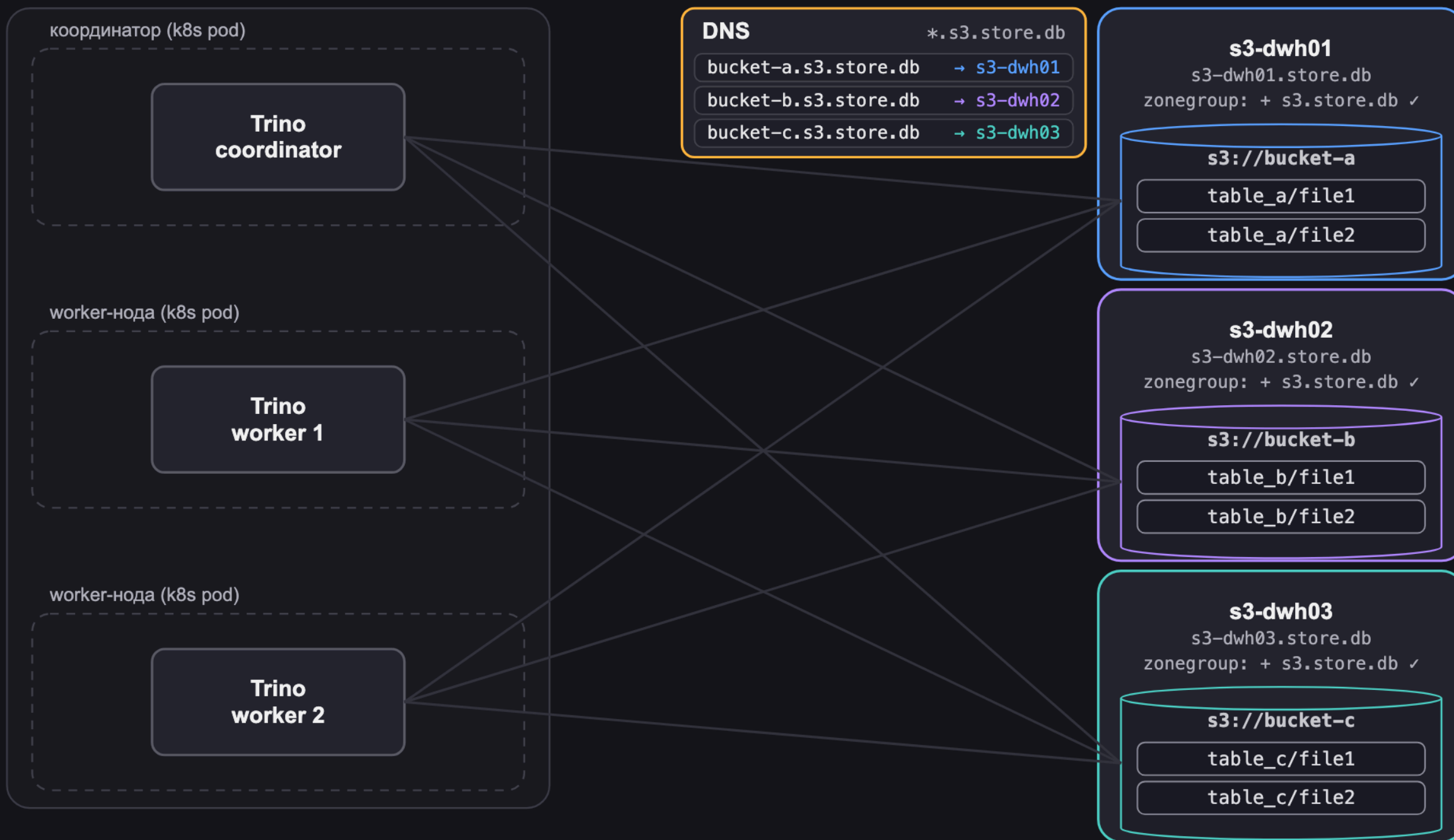
Третий Серв — третий каталог: место и суммарная скорость чтения растут кластерами

✓ **SELECT ... FROM dwh01.dma.table_a**
каталог dwh01 → s3-dwh01

✓ **SELECT ... FROM dwh02.dma.table_b**
каталог dwh02 → s3-dwh02

SELECT ... FROM dwh03.dma.table_c
каталог dwh03 → s3-dwh03

кластер Trino



table_b лежит в bucket-b: другая запись DNS, другой кластер — и никакого сервера посередине

✓ Zonegroup + DNS

одно имя, бакет → кластер

✓ SELECT ... table_a

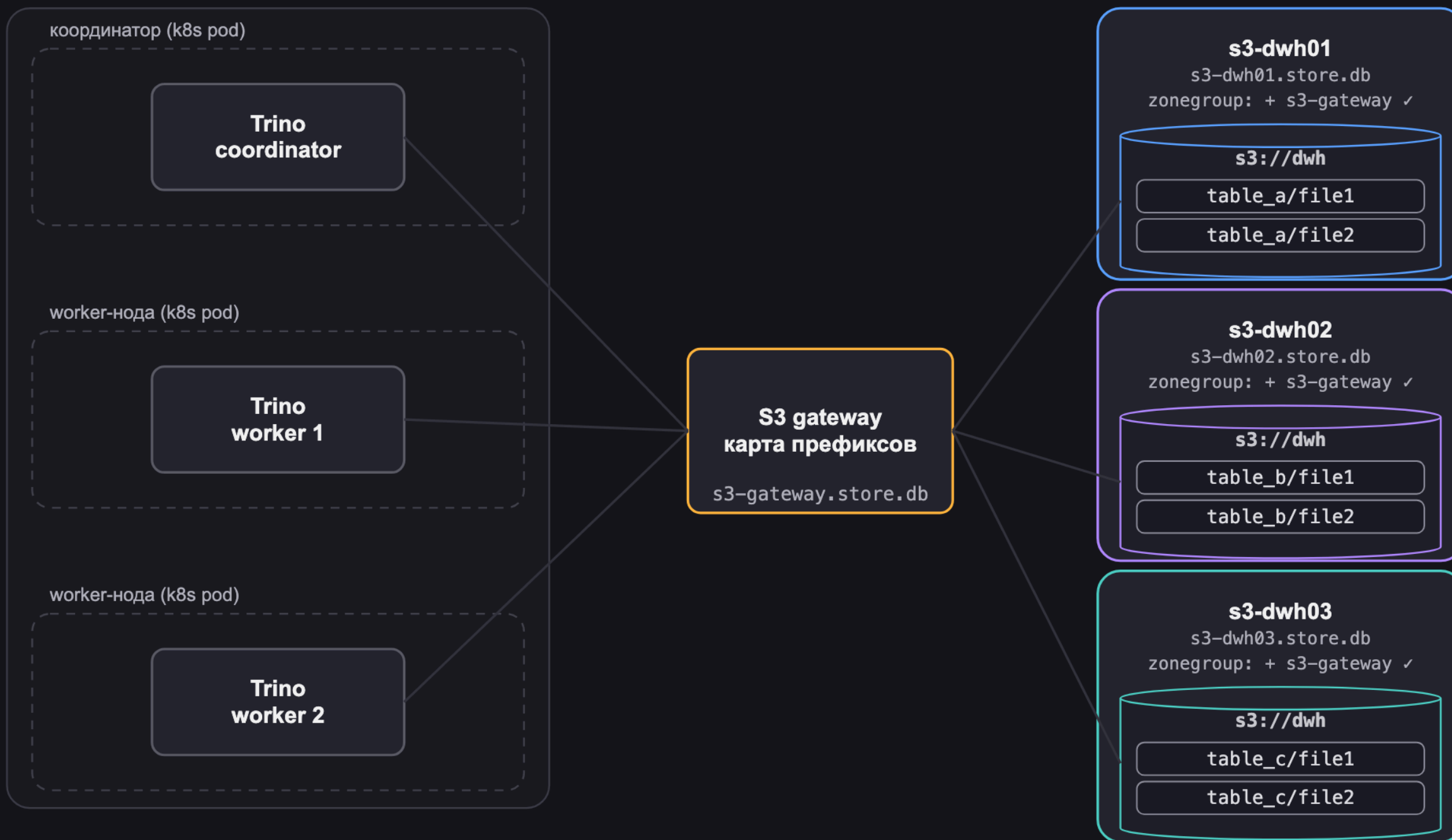
bucket-a.s3.store.db → dwh01

SELECT ... table_b

bucket-b.s3.store.db → dwh02

Прокси режет бакет по префиксам

кластер Trino



Можно раскладывать схемы и таблицы, но таблица всё ещё целиком в одном Serp

✓ **SELECT ... table_a**
карта: table_a/ → s3-dwh01

✓ **SELECT ... table_b**
карта: table_b/ → s3-dwh02

SELECT ... table_c
карта: table_c/ → s3-dwh03

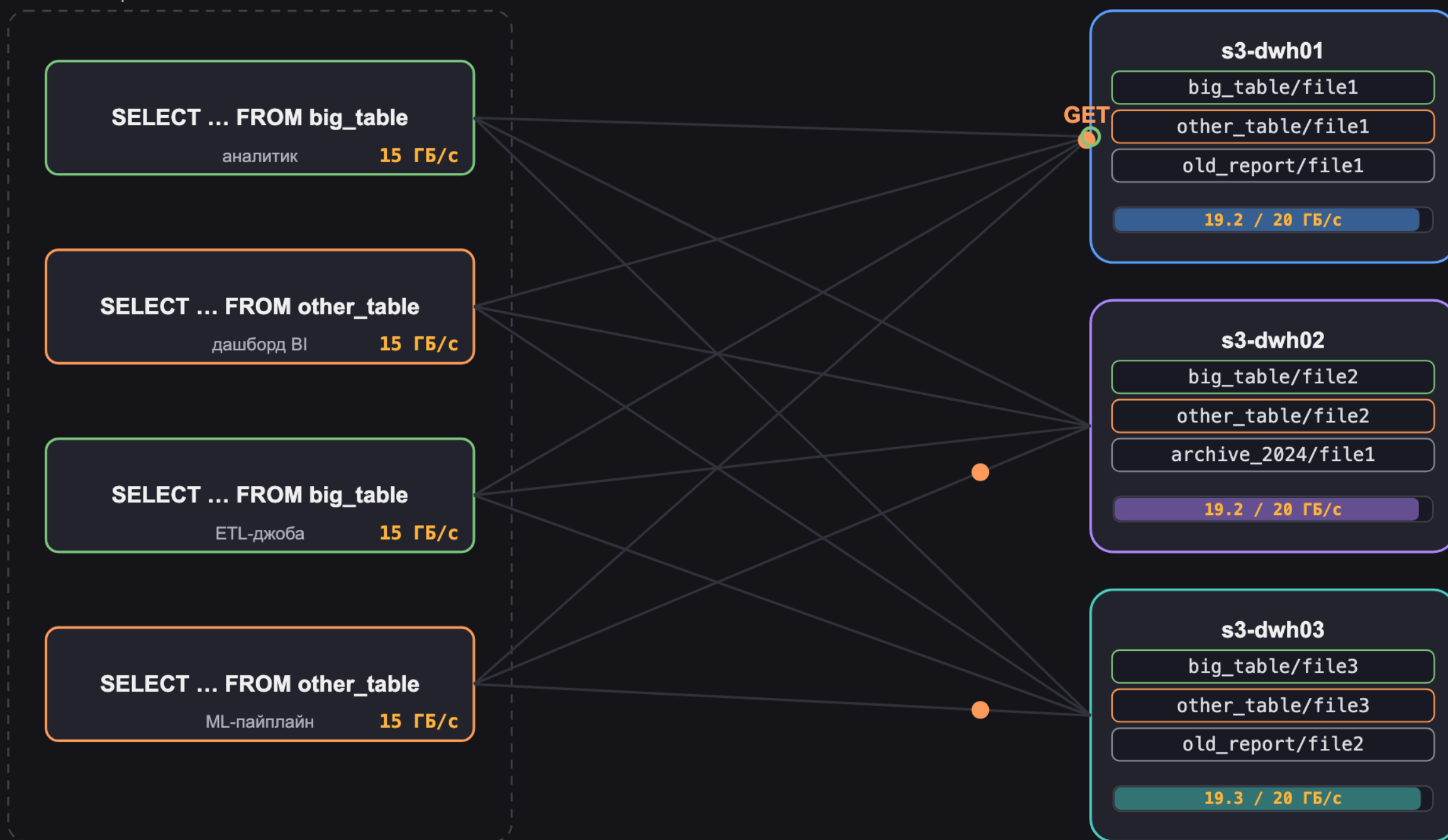
1 Сложно равномерно заполнить место

2 Сложно равномерно нагружать хранилища

3 Одна таблица остаётся заперта на одном Серв



Trino: четыре читателя

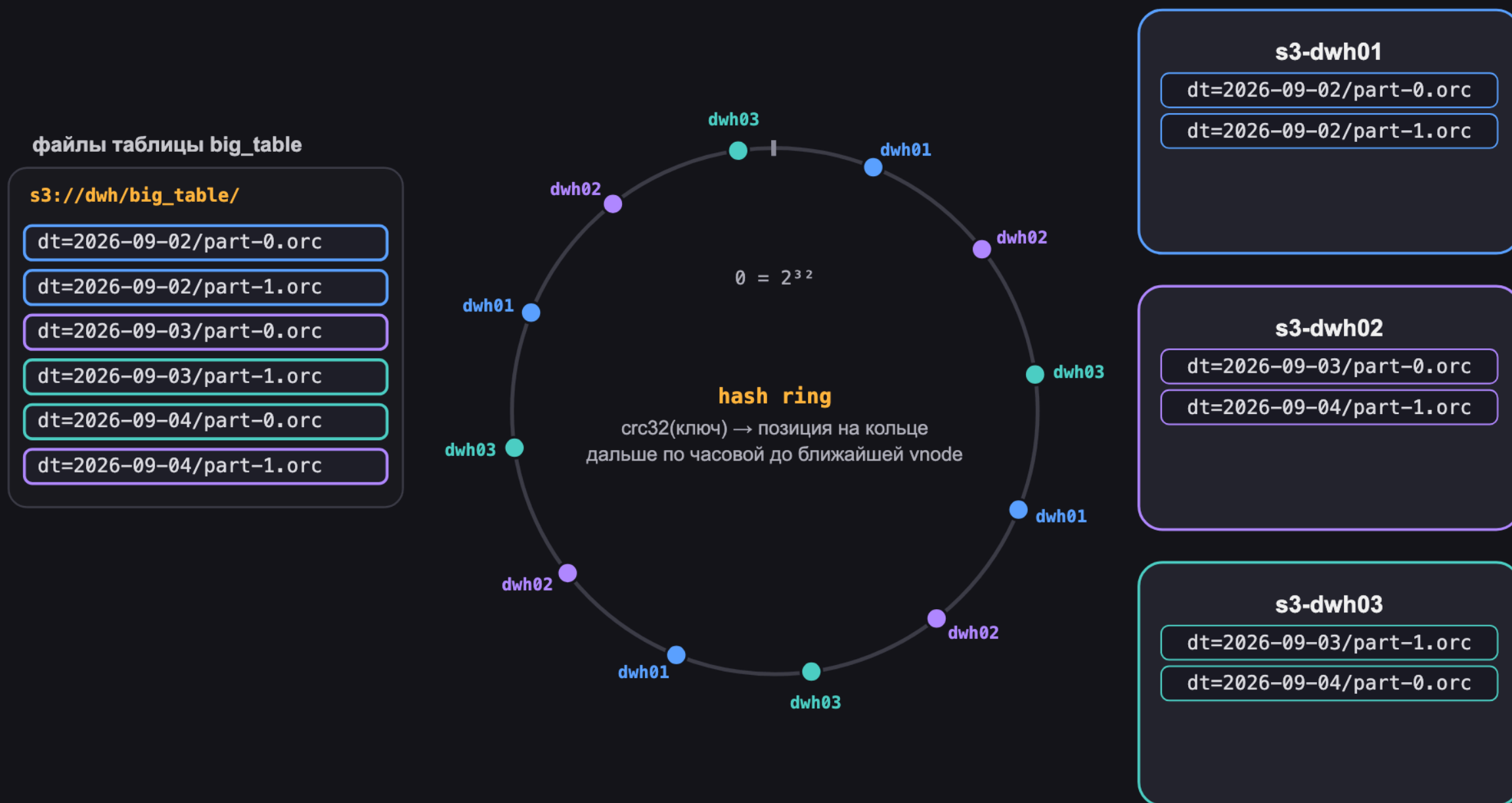


Файлы каждой таблицы на всех трёх Серв: те же читатели получают 60 ГБ/с, по 15 вместо 5

✓ Таблица на одном Серв
читатели делят один канал

Таблица на всех Серв
каналы складываются

Consistent hashing: имя файла → точка на кольце → шард



Одна таблица лежит на трёх Серв сразу: и место, и скорость чтения складываются

✓ Кольцо

виртуальные ноды шардов

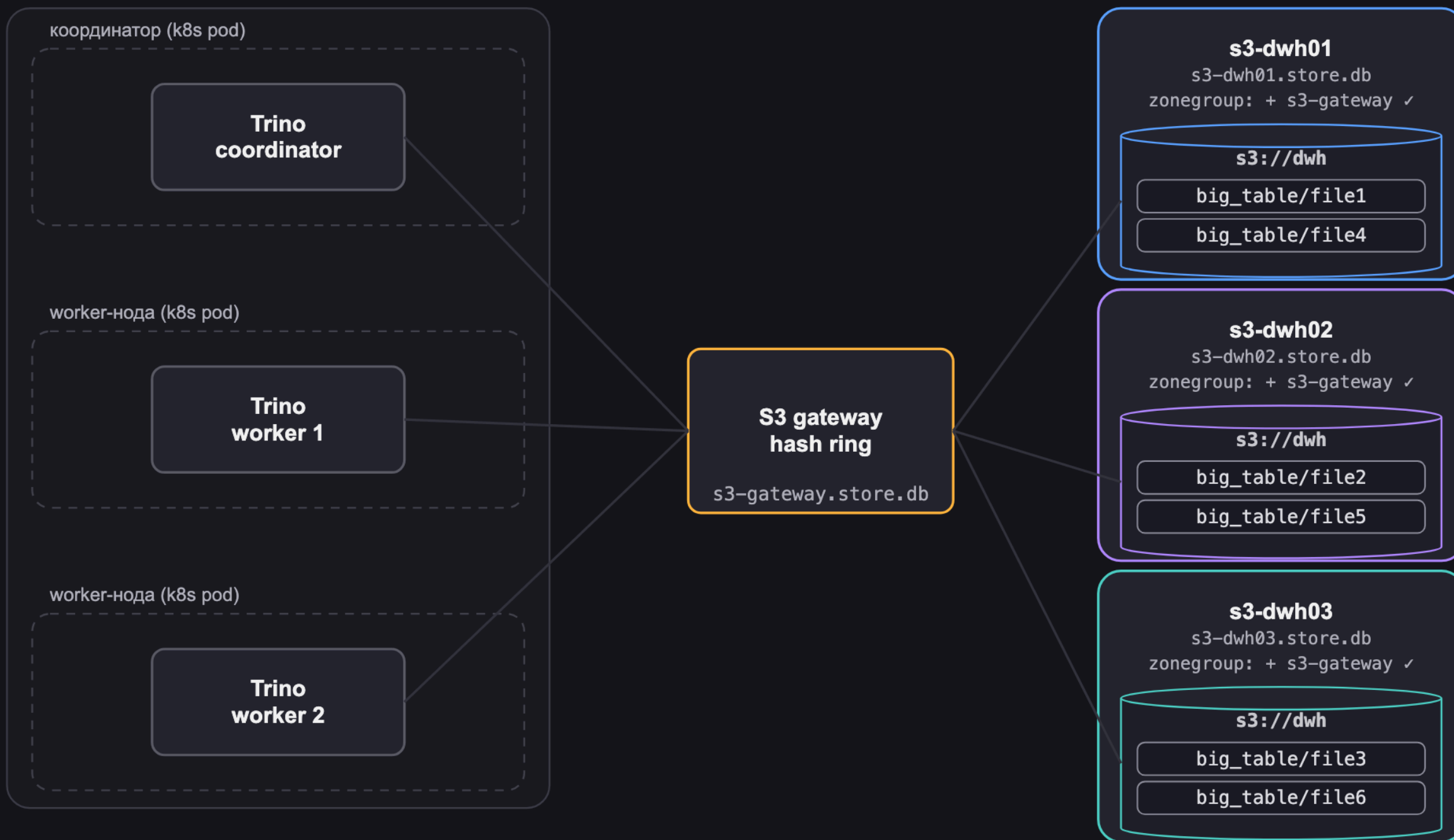
✓ Раскладка

сrc32 → точка → ближайшая vnode по часовой

Одна таблица

файлы разъехались по трём Серв

кластер Trino

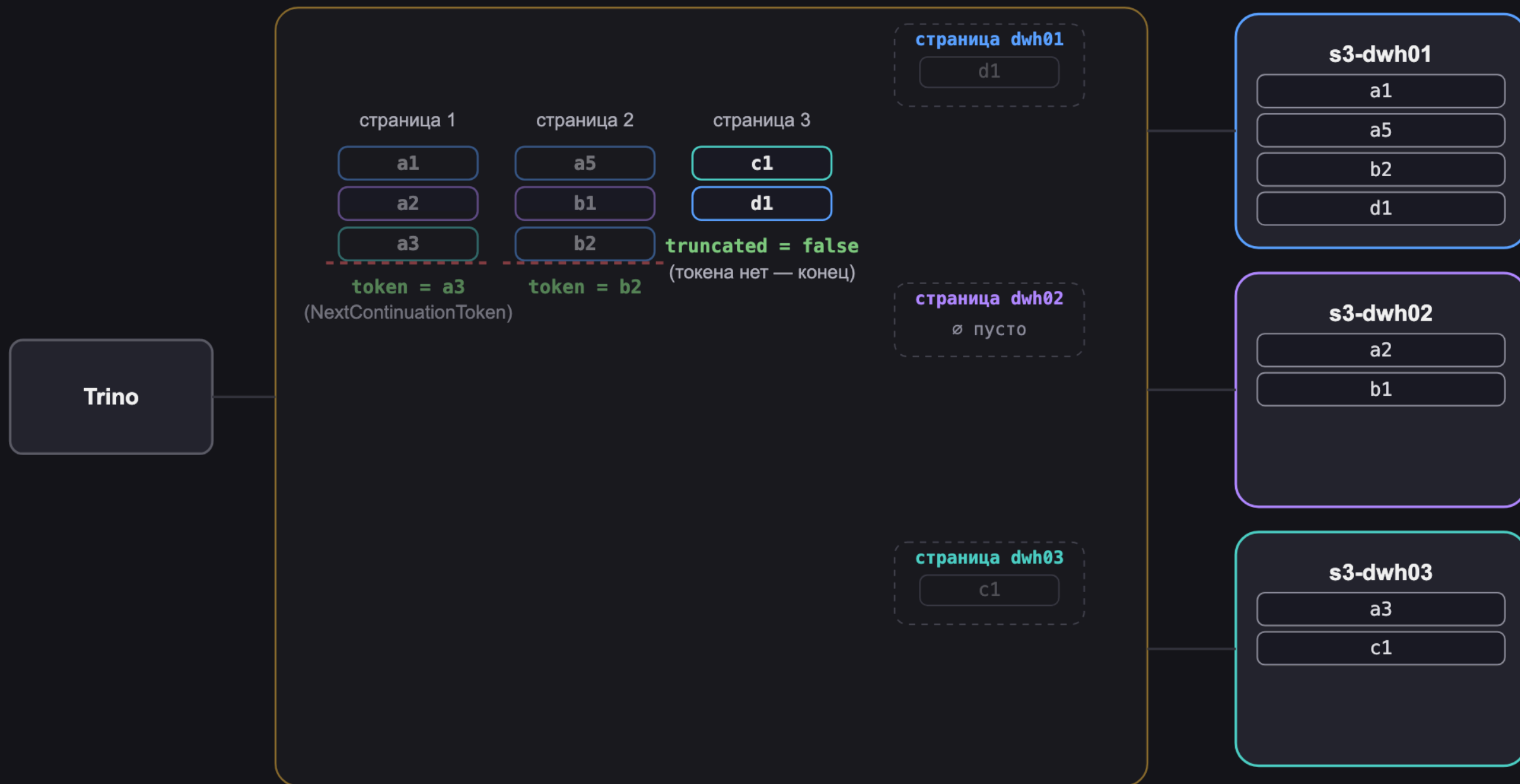


GET: hash(путь) детерминированно указывает шард — fan-out не нужен

✓ LIST big_table/
fan-out на все шарды + k-way merge

GET по ключу
consistent hashing: hash(путь) → шард

S3 gateway — сборка листинга



Страницы исчерпаны, но все ответы были полными — резать нечего

✓ Страницы с шардов
max-keys уходит шардам

✓ k-way merge
потокково, стейта нет

✓ max-keys → token
страница набрана — режим

✓ Вторая страница
start-after=token

truncated=false
данные кончились — токена нет

S3 gateway — сборка листинга



У dwh01 за краем ещё a7 и d1; $a7 < b1$ — отдав сейчас b1, мы бы его потеряли

✓ Страницы с шардов

шард может вернуть меньше, чем попросили

✓ k-way merge

потоково, пока живы все головы

Конец страницы шарда

режем, не дождавшись max-keys

1

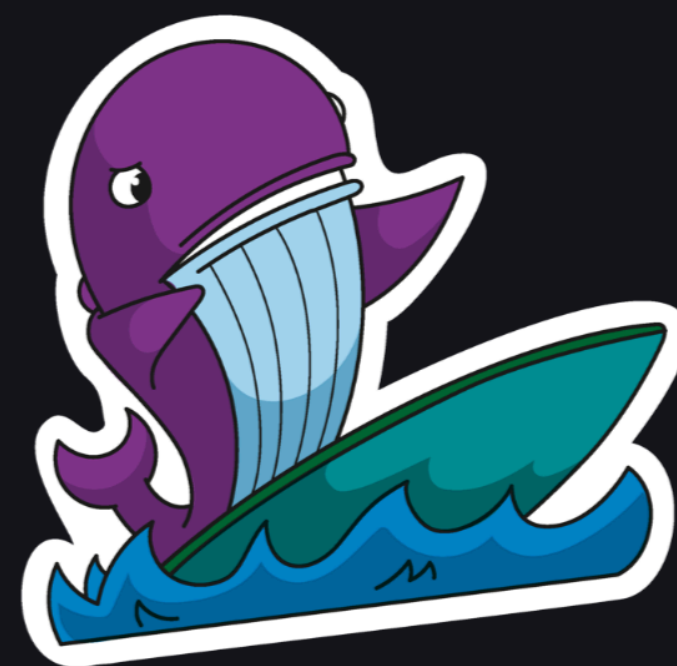
Можем читать таблицу в 3 раза быстрее

2

Листинг стал тяжелее

3

Гейтвей должен мочь пропустить через себя много трафика

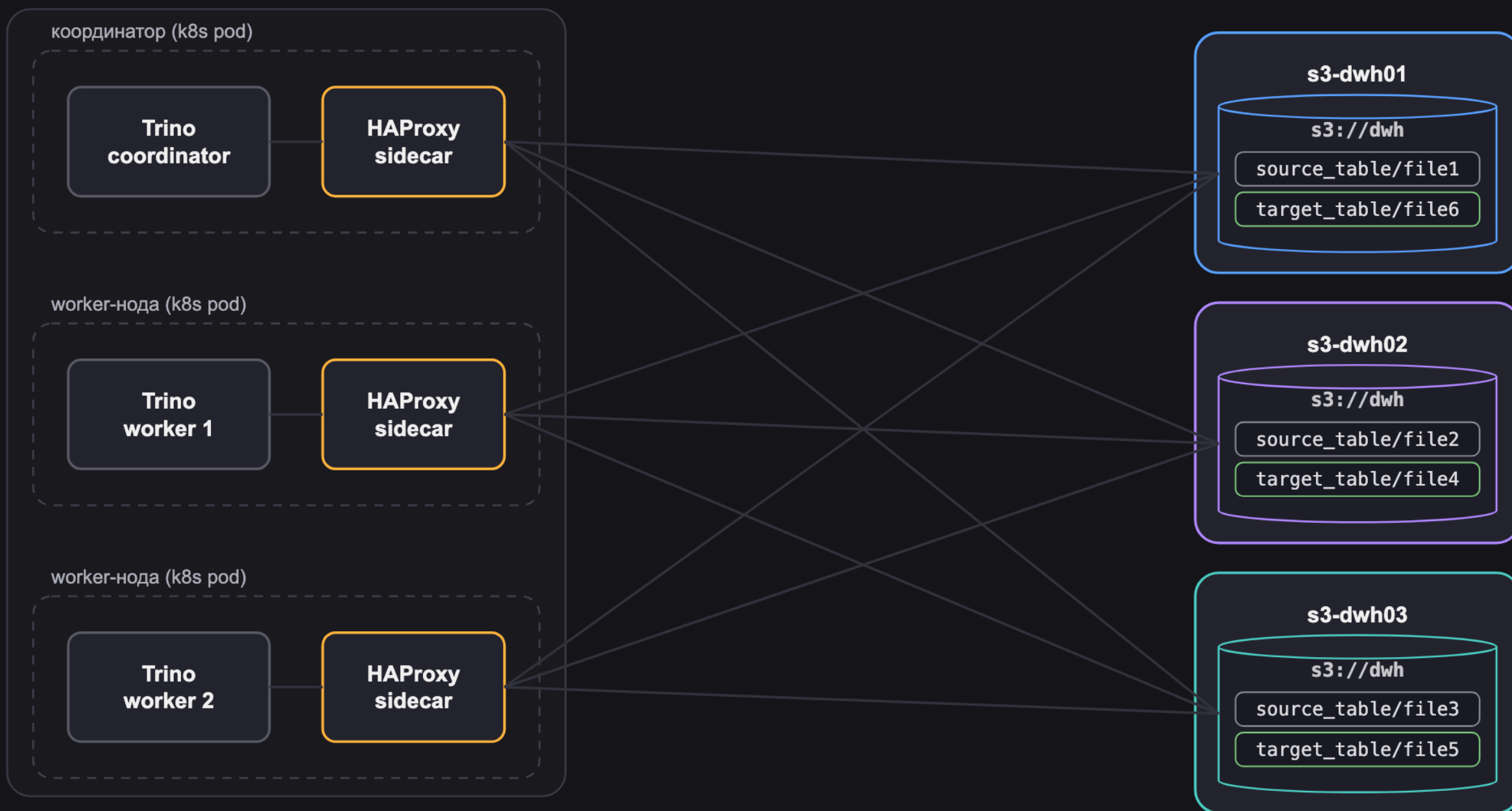


The diagram illustrates the S3 Gateway architecture. On the left, a Trino cluster is shown with a coordinator (labeled 'координатор (k8s pod)') and two workers (labeled 'worker-нода (k8s pod)'). The coordinator is connected to the S3 gateway hash ring. The workers are also connected to the hash ring. In the center is the 'S3 gateway hash ring' (labeled 's3-gateway.store.db'). On the right, there are three S3 data warehouse instances (s3-dwh01, s3-dwh02, s3-dwh03). Each instance is connected to the hash ring and contains a source table and a target table. The source tables are 'source_table/file1', 'source_table/file2', and 'source_table/file3'. The target tables are 'target_table/new3', 'target_table/new1', and 'target_table/new2'.

PUT — точно по хешу: новые файлы могут лечь на другие шарды

PUT — точно по хешу

кластер Trino



PUT — новые файлы ложатся каждый на свой шард

✓ Planning

листинг партиции

✓ Execution

воркеры читают данные

✓ Cleanup

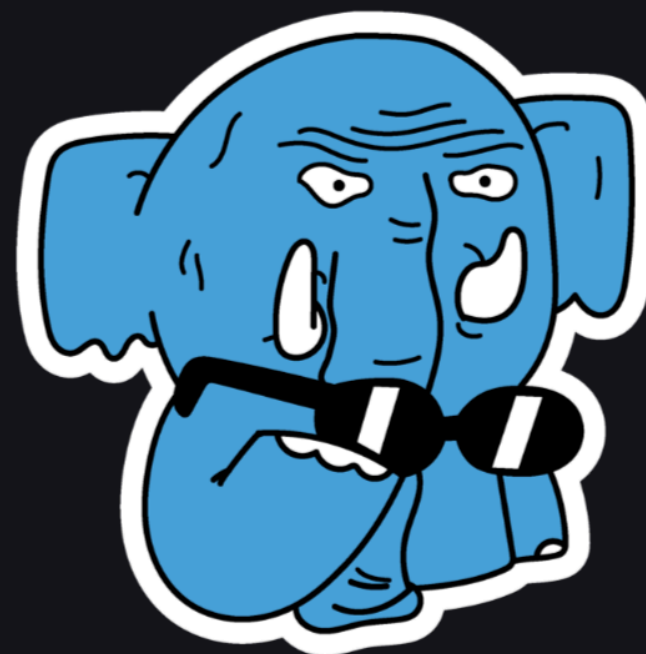
удаление старой партиции

Write

запись новых файлов

1 Мы можем читать x3 без лишнего Gateway

2 Но что-то мы упускаем...



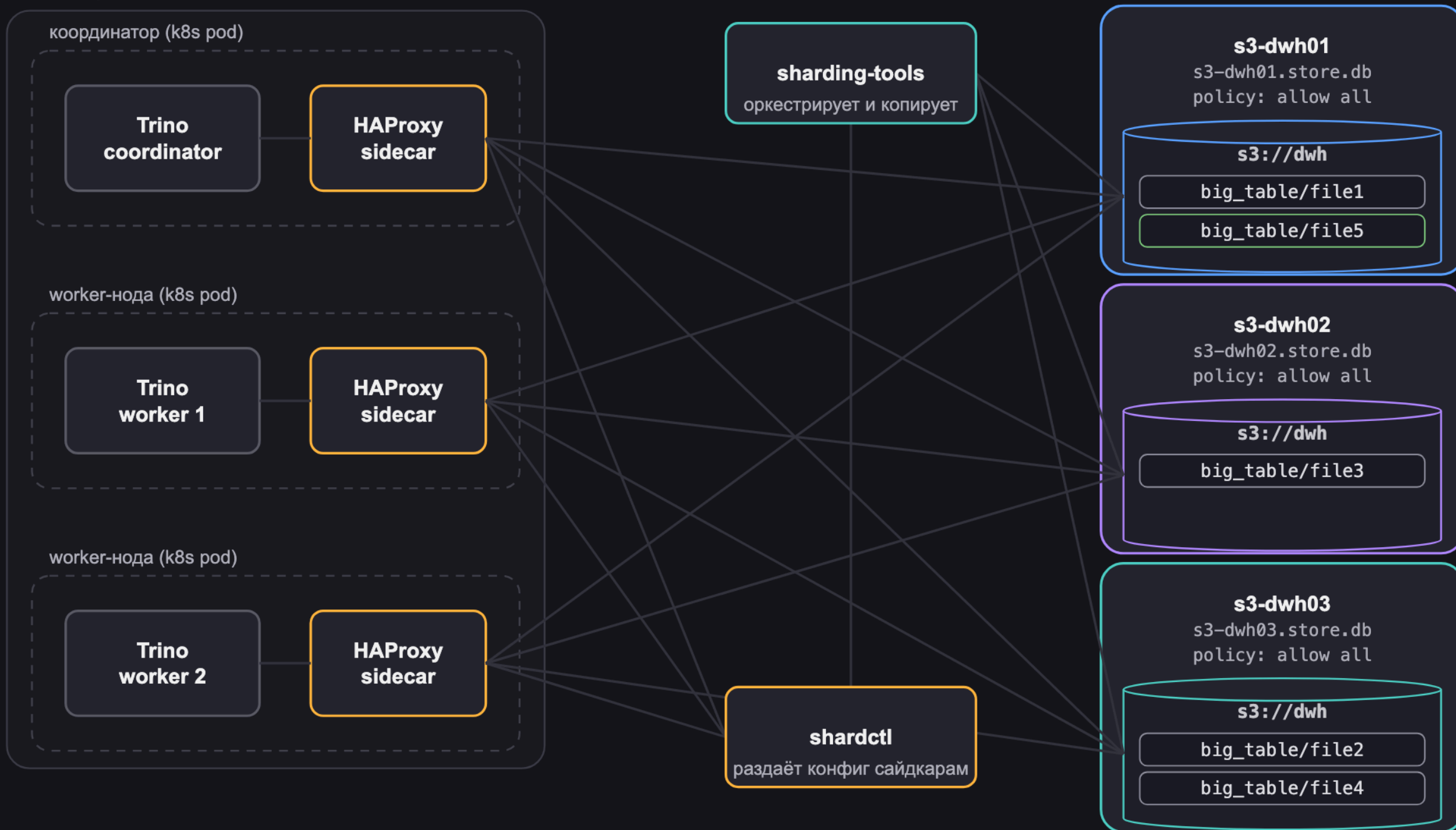


Как менять конфигурацию
на множестве сайдкаров?

Как гарантировать
консистентность данных?

Как минимизировать
даунтайм?

кластер Trino



Открываем бакет: GET file2 идёт на dwh03, PUT file5 ложится по новому кольцу

✓ Инструменты
sharding-tools, shardctl

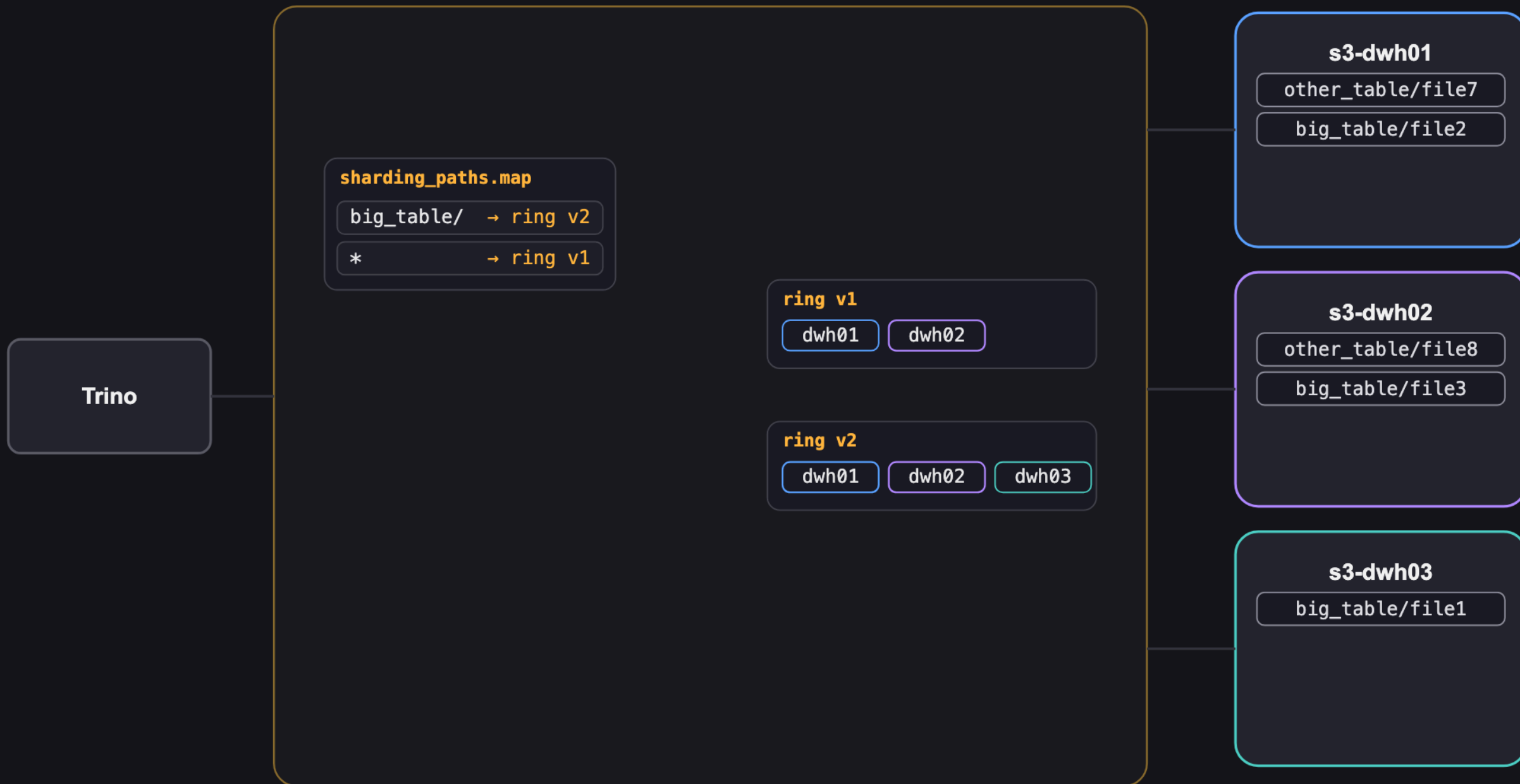
✓ Freeze
setpolicy: deny all

✓ Перенос
file2, file4 → dwh03

✓ Новое кольцо
shardctl → сайдкары

Unfreeze
setpolicy: allow

HAProxy sidecar — маршрутизация S3-запросов



Резолв в два шага: префикс → кольцо по карте, потом crc32 → шард по кольцу

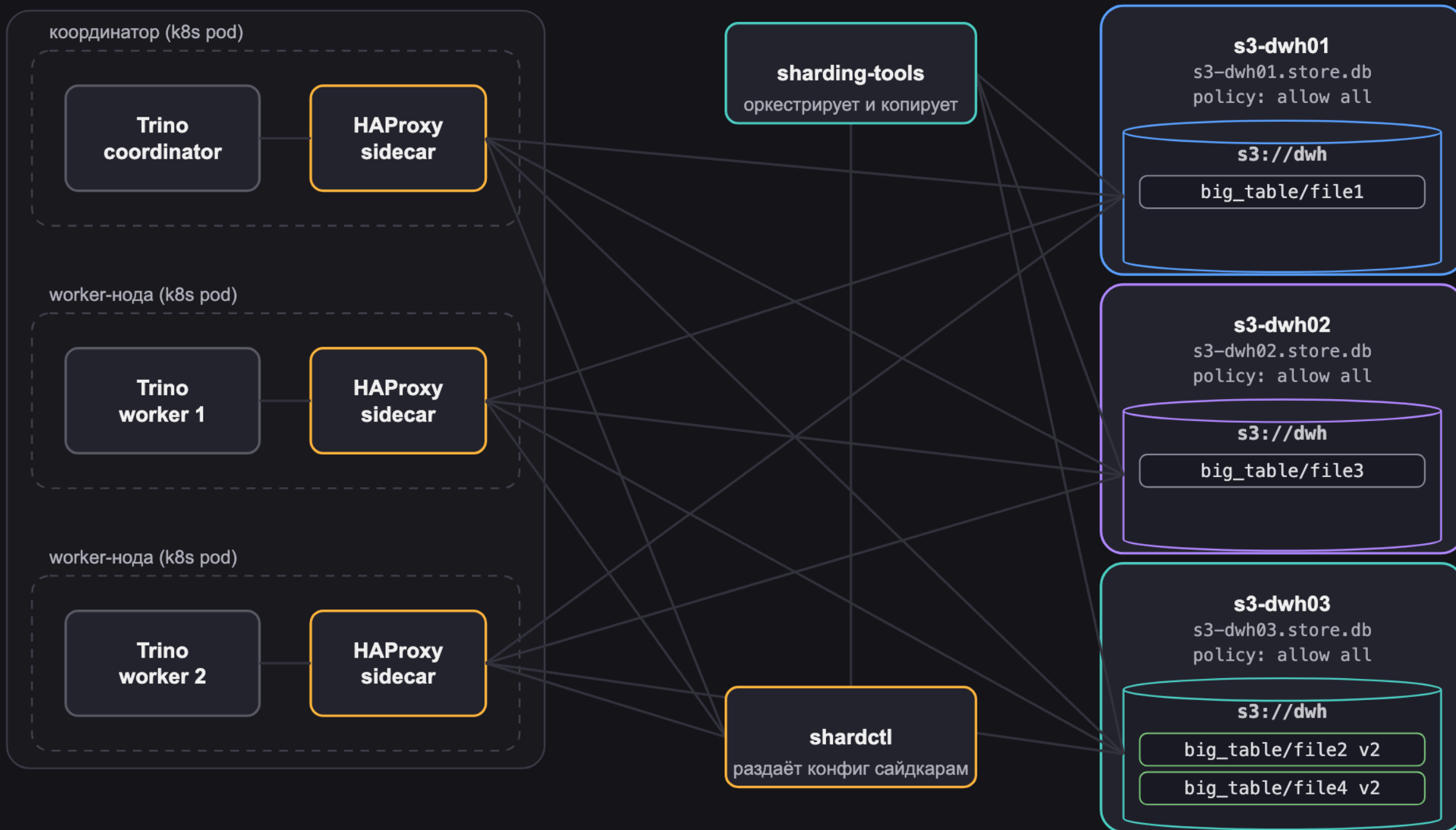
✓ Карта префиксов

несколько колец; префикс → кольцо

Резолв запроса

сначала карта, потом хеш

кластер Trino



Старые копии никому не видны по новому кольцу — удаляем file2 с dwh01 и file4 с dwh02

✓ Pre-copy
копии на dwh03

✓ PUT
новая версия

✓ LIST + GET
дедупликация

✓ Freeze
только запись

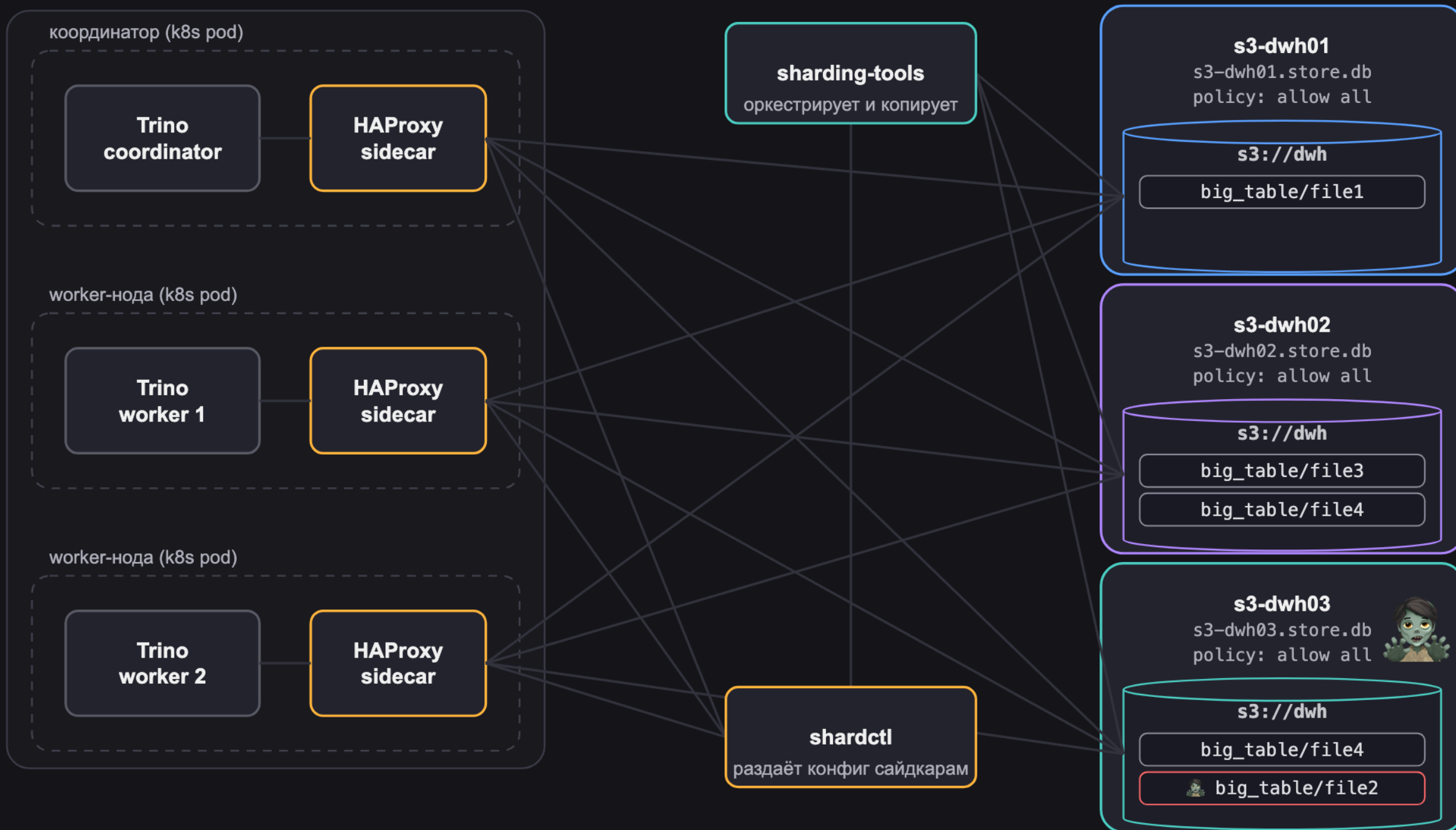
✓ Дельта
синк изменений

✓ Flip
новое кольцо

✓ Доступ
запись открыта

Cleanup
старые копии

кластер Trino



Листинг видит воскресший file2 — и воркер забирает зомби себе

✓ Копирование

а если совсем без setpolicy?

✓ GET

file2 в буфере копировальщика

✓ DELETE

Trino удаляет file2 везде

✓ PUT

из буфера — на новый шард

LIST + GET

воркер забирает зомби

1 Добавили карту префиксов — можем шардировать всё по-разному

2 Можем копировать и чистить в любое время — «неправильные» файлы игнорируются

3 Небольшой фриз на запись — не проблема, если знаем, когда запись



Сайдкаров на
вычислительных нодах

635

новый конфиг
доезжает до всех за <
1 минуты

Читается в день

3 ПБ

через сайдкары

LIST — fan-out на все
шарды, GET —
точечно

Серв за одним именем

5

4 HDD-кластера по 45 нод
1 SSD-кластер на 17 нод

HDD под витрины,
SSD под DDS и tmp

Пользователей / день

900

аналитики, ETL,
ML-инженеры

решард они не
замечают

shardctl хранит кольца и карту и раздает их сайдкарам; sharding-tools оркестрирует решард префикса за префиксом.

HAProxy sidecar — маршрутизация S3-запросов



DDS и временные таблицы тем же механизмом уходят на SSD: dds/ и tmp/ → ring dds из одного hot01

✓ Витрины на HDD

* → ring v2 → три HDD-шарда

DDS и tmp на SSD

dds/, tmp/ → ring dds → hot01

Хранилище

4 ПБ

×4 · было 1 ПБ

один Серв → 4 HDD-кластера
+ 1 SSD за одним именем

Чтение из S3

90 ГБ/с

×4,5 · было 20 ГБ/с

потолок одного кластера →
сумма кластеров

Новых серверов под прокси

0

**сайдкары — в подах
клиентов**

gateway для сервисов собран
из таких же сайдкаров

Спасибо за внимание

Дмитрий Листвин

Big Data Engineer, AvitoTech



Цикл статей про
аналитическую платформу Авито