СКОЛЬКО «ВЕШАТЬ В ГРАММАХ»?

«ВЗЯТЬ С ЗАПАСОМ» — МОЖЕМ СЕБЕ ПОЗВОЛИТЬ, НО НУЖНО ЛИ?

КАК OTP BANK ПОДОШЕЛ К ОЦЕНКЕ LLM-ИНФЕРЕНСА

ДЛЯ КОГО ЭТОТ ДОКЛАД? ВЫ ТОЧНО ЗДЕСЬ:

1.  **Если вы планируете внедрение GenAI**

2.  **Если вы платите за GPU**

3.  **Если вас спрашивают «почему столько?»**

ЧТО ВЫ СМОЖЕТЕ ПОСЛЕ ДОКЛАДА

1.  **Самостоятельно рассчитать потребность**

2.  **Защитить цифру перед финансистами**

3.  **Уточнять и калибровать расчеты**

КОРОТКО О ДОКЛАДЧИКЕ

AI Platform Center Lead
Богдан Гарбар

ПЛАН ДОКЛАДА

1.  **Предпосылки. Как мы пришли к мысли разработать свою методику**
2.  **Методика. Принципы расчета**
3.  **Проверка. Уточнение и калибровка**
4.  **Практическая польза**

01 ПРЕДПОСЫЛКИ

С ЧЕГО ВСЕ НАЧИНАЛОСЬ: РАЗГОВОР С САМИ СОБОЙ

Что мы хотим, и сколько GPU для этого нужно на 1-2 года вперед?

С ЧЕГО ВСЕ НАЧИНАЛОСЬ: РАЗГОВОР С ФИНАНСИСТАМИ

Ну что, ребят? Рассказывайте…
Чего, сколько, какое ROI?

РЕАЛЬНОСТЬ LLM-ИНФЕРЕНСА СЕГОДНЯ

1

Много и дорого

Для создания своего LLM инференс кластера нужно большое кол-во GPU. А GPU – это дорого.

2

Сложности с поставками в РФ

В текущей обстановке, наблюдаются сложности с поставками, что увеличивает сроки и повышает риски.

3

Ошибка в подсчетах - не дешево

GPU – это дорого, а поставки длительные.
Подсчет «пальцем в небо» может дать существенную ошибку.

4

Быстрое устаревание железа

Новые поколения AI моделей и чипов появляются каждый квартал. Нужно выжать максимум из оборудования за конечное время его актуальности.

“Взять с запасом” – можем конечно себе позволить, но нужно ли?

Любое инженерное капиталоемкое решение должно быть объяснимо, воспроизводимо и защищаемо.

КАКИЕ ИНСТРУМЕНТЫ ОЦЕНКИ БЫЛИ ДОСТУПНЫ И ПОЧЕМУ ИХ ОКАЗАЛОСЬ НЕДОСТАТОЧНО

В процессе планирования мы использовали:

Публичные и вендорские калькуляторы
для первичной прикидки

Эвристические оценки и бенчмарки
как ориентиры

Экспертные статьи и кейсы
как источник гипотез

Почему этого оказалось недостаточно для production:

- Эти подходы не дают воспроизводимой логики расчёта
- Не всегда есть прямая связь с SLA и пользовательской нагрузкой
- Сложно адаптировать расчёт под реальные сценарии эксплуатации
- Сложно или невозможно защищать итоговую цифру перед CFO

Всё это недостаточно прозрачно для принятия обоснованных решений

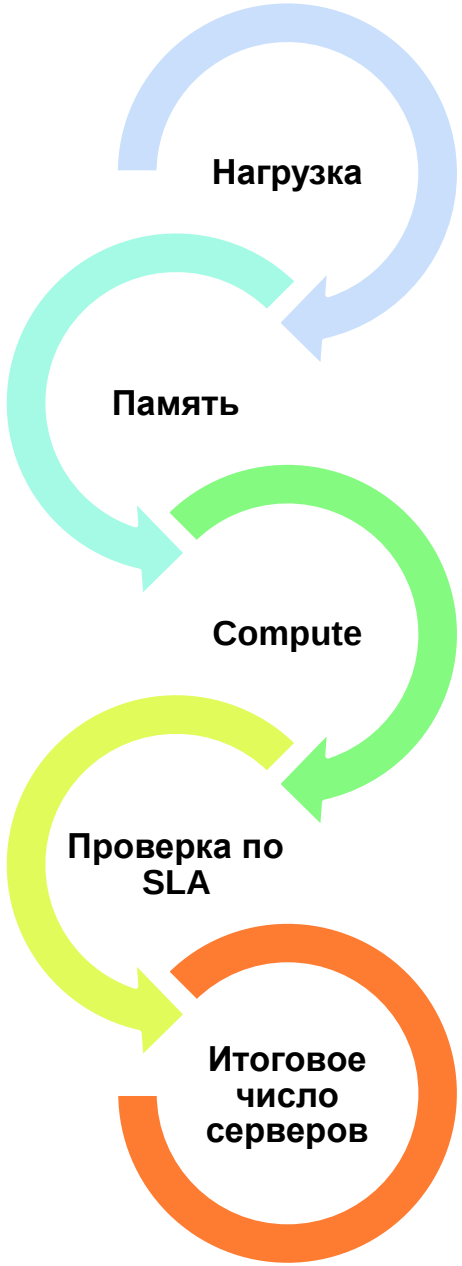
А может сделаем расчет сами?
Можно будет объяснить, повторить и уточнять по данным...

02 ПРИНЦИПЫ РАСЧЕТА

ЧТО И КАК МЫ СЧИТАЕМ?

Расчет потребления памяти на KV кэш и веса моделей

Проверка конфигураций по SLA



Расчет кол-ва пользователей и потребления токенов

Расчет пропускной способности

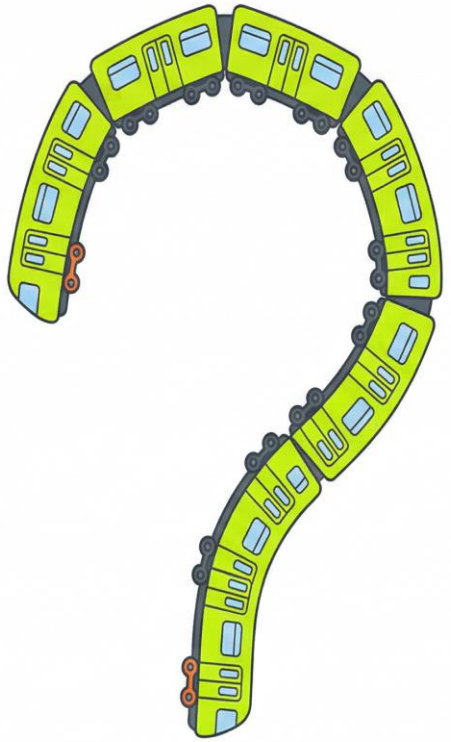
Максимальное из расчетов по Памяти и Пропускной способности

ОБЪЯСНЕНИЕ МЕТОДИКИ ЧЕРЕЗ АНАЛОГИЮ С СОСТАВОМ МЕТРО

отпбанк

АНАЛОГИЯ

У города есть **N жителей**, которых надо перевозить каждый день.



Сколько составов нужно закупить, чтобы все доехали вовремя?

ДВА МОМЕНТА

- Хватает ли мест, чтобы все сели?
- Хватает ли скорости, чтобы всех перевезти за час?

В переводе на GPU:

сколько серверов нужно, чтобы обслужить целевое число пользователей с заданным SLA?

ЧТО ПОДАЕМ В КАЧЕСТВЕ ИСХОДНЫХ ДАННЫХ

Пользовательская нагрузка

- Количество пользователей
- Конкурентность (одновременные сессии)

Характер запросов

- Средний размер контекста (tokens)
- Средняя длина ответа (tokens)

Модель

- Размер и архитектура (вес модели)

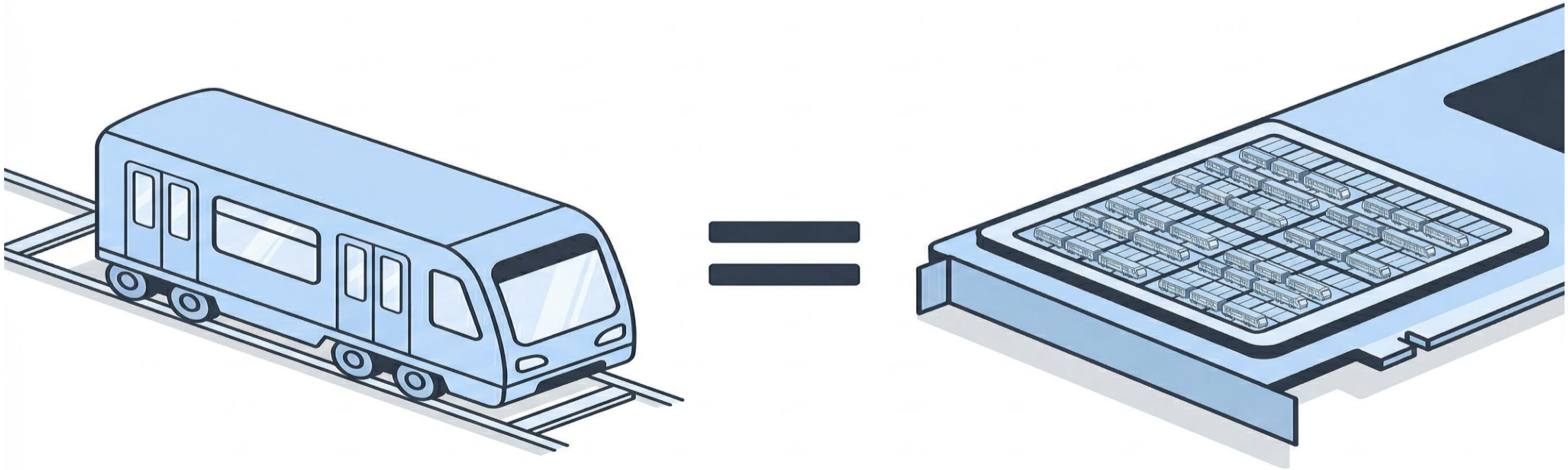
Требования

- Целевая латентность (SLA)
- Требуемый throughput (tokens/sec)

Инфраструктурные допущения

- Тип и конфигурация GPU (как входное ограничение)

ПРЕДСТАВИМ, ЧТО GPU = ВАГОН МЕТРО

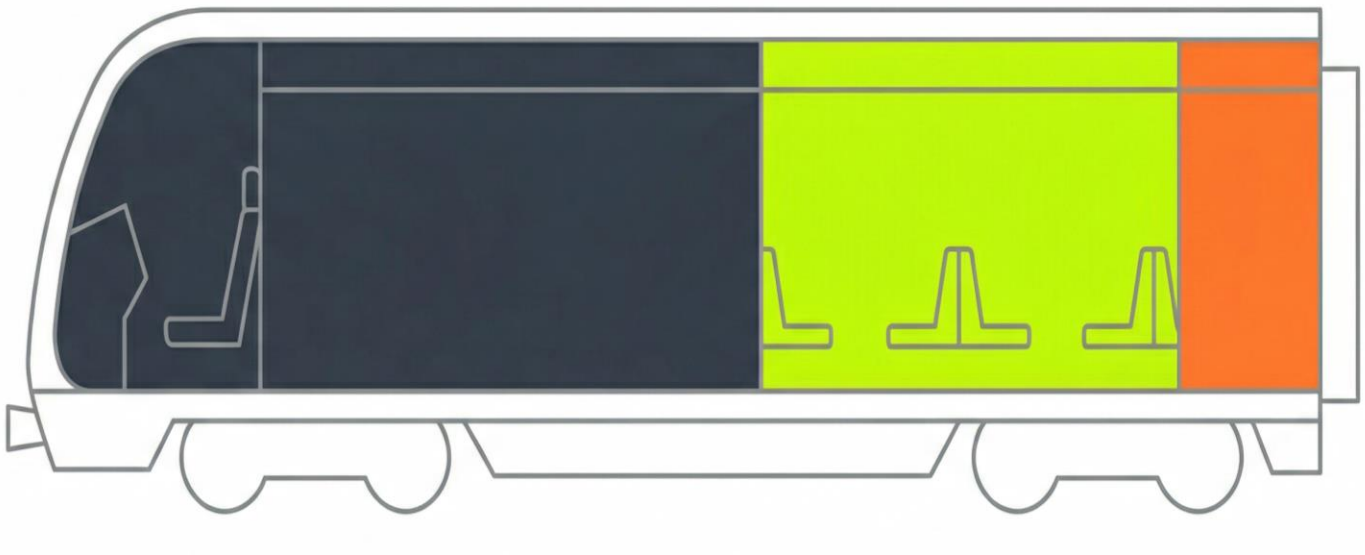


GPU-чип
= один вагон метро
Память (HBM) = общий объём внутри вагона

GPU-сервер
= состав (несколько вагонов в сцепке)

Кластер GPU
= сеть метро города (много линий, много составов)

ТОГДА, ЧТО ЗАНИМАЕТ МЕСТО В ВАГОНЕ?



Кабина + Двигатель + Оборудование
= Weights модели (параметры нейросети)

Пассажирский салон
= KV-cache (контекст каждой сессии)

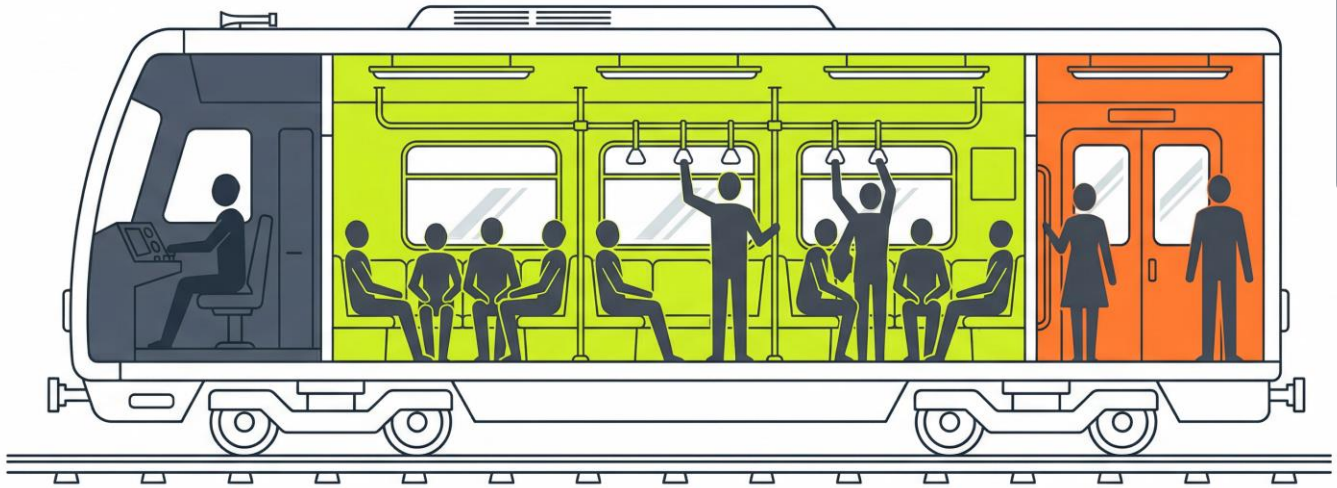
Проходы и тамбуры
= Reserve (буферы, активации, KV overhead)

Формула:

Память GPU = Кабина (Weights) + Салон (KV-cache) + Проходы (Reserve)

Чем больше кабина (модель) — тем меньше места для пассажиров (сессий)

ПАССАЖИРЫ И БАГАЖ (KV-CACHE)



Пассажир = сессия пользователя

Каждый пассажир занимает место (KV-cache) в салоне. Чем длиннее его маршрут (контекст), тем больше багажа.

ВМЕСТИМОСТЬ САЛОНА

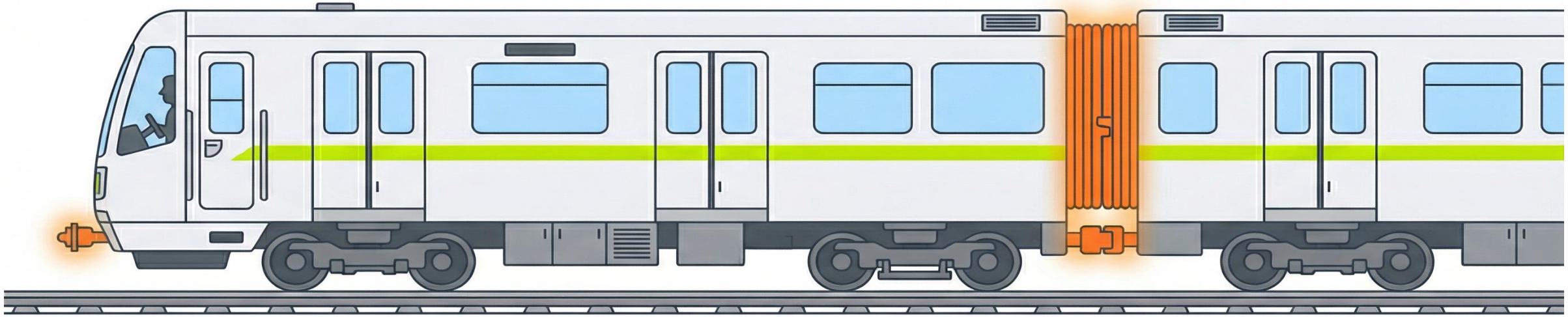
- Размер вагона — память GPU (HBM)
- Минус кабина — веса модели (weights)
- Минус проходы — резерв на пики (reserve)

Формула:

Max сессий = (HBM − Weights − Reserve) / KV на сессию

Чем больше размер вагона — тем больше помещается пассажиров

СОСТАВ ИЗ НЕСКОЛЬКИХ ВАГОНОВ (TENSOR PARALLELISM)



Один машинист на весь состав

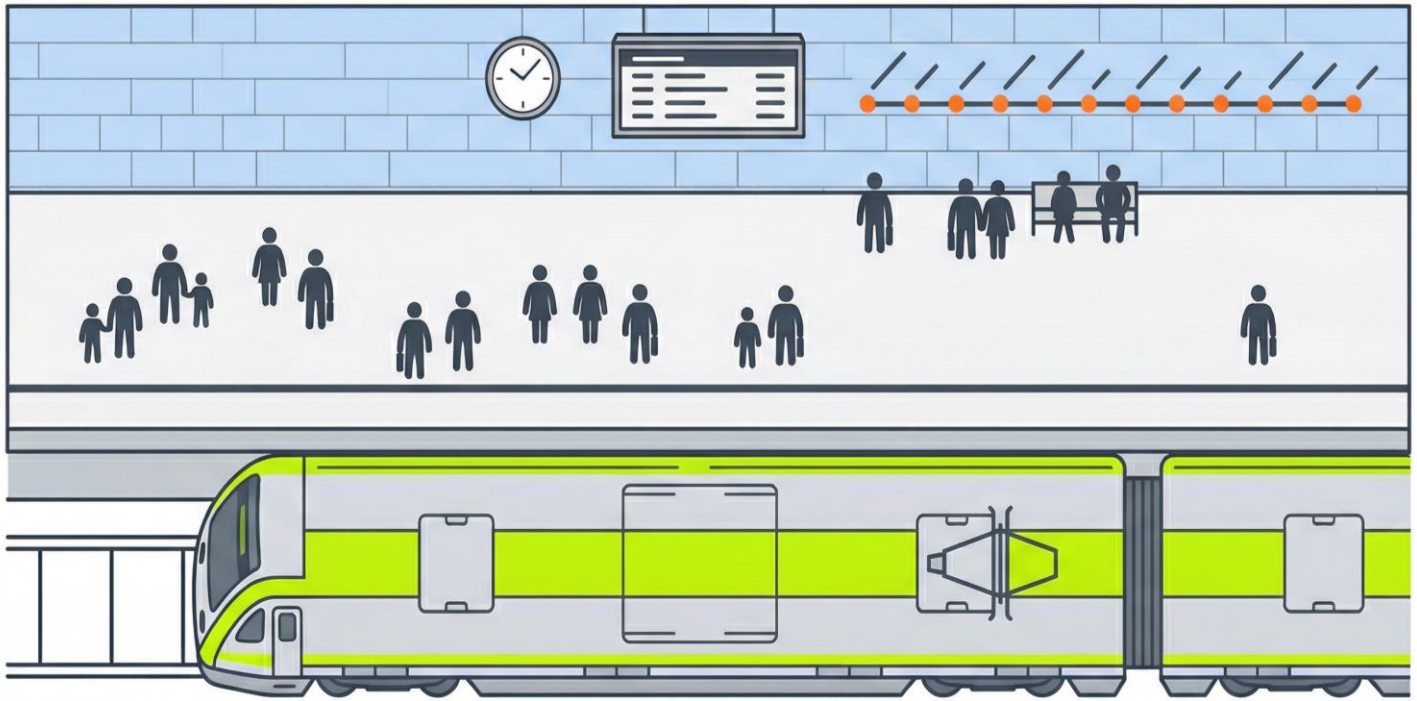
Модель (weights) распределена по всем вагонам (GPU), но работает как единое целое.

Зачем нужен состав?

Если модель не помещается в один вагон — сцепляем несколько.

Например, TP=2: два вагона на одну модель.

СКОЛЬКО ПОЕЗДОВ НАДО ЗАПУСТИТЬ?



Ограничение по местам:

Каждый поезд вмещает ограниченное число пассажиров одновременно.

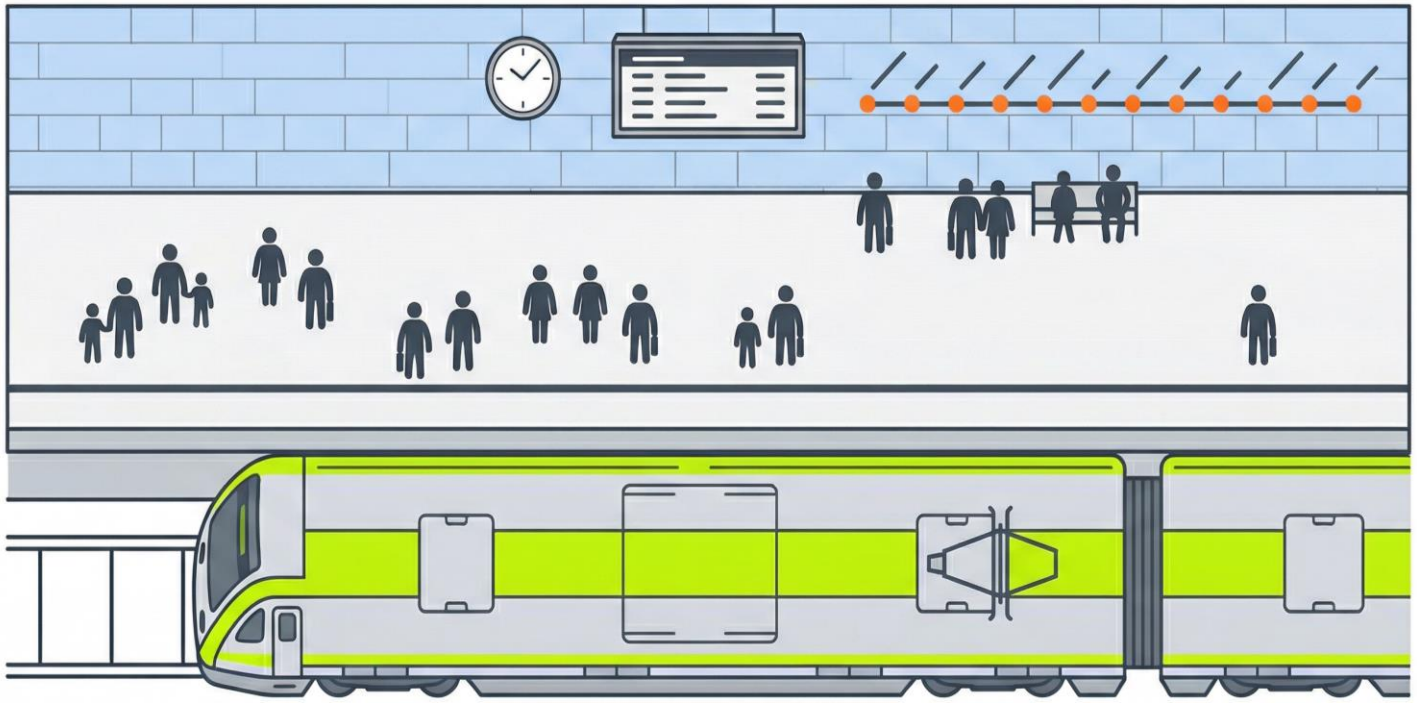
- Concurrent users — сколько пассажиров одновременно в метро
- Max sessions per train — вместимость одного состава (зависит от GPU-памяти, модели, контекста)

Формула:

N_поездов (по местам) = Concurrent_users / Max_sessions_per_train

Чем больше пассажиров в метро — тем больше требуется составов

РАСПИСАНИЕ: СКОРОСТЬ ПЕРЕВОЗКИ



TTFT

ОЖИДАНИЕ НА ПЛАТФОРМЕ

Сколько пассажир ждёт первый поезд

Time to First Token

ITL

ВРЕМЯ МЕЖДУ СТАНЦИЯМИ

Скорость, с которой генерируются следующие токены

Inter-Token Latency

THR

ПАССАЖИРОВ ЗА ЧАС

Общая пропускная способность линии

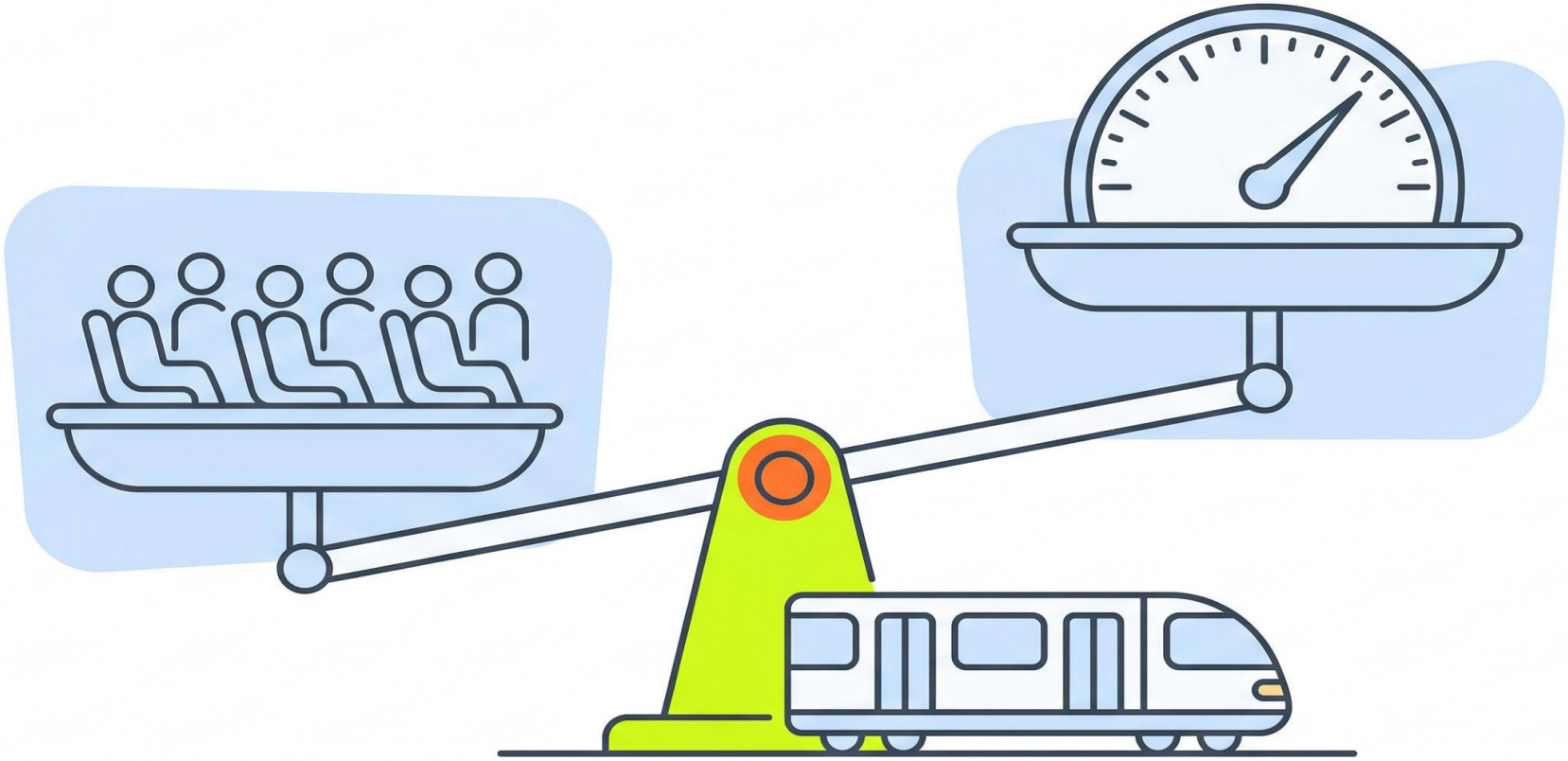
Throughput

Формула:

N_поездов (по скорости) = Target_throughput / Throughput_per_train

Чем больше пассажиров надо перевезти за единицу времени — тем больше требуется составов

ОТВЕТ НА ВОПРОС «СКОЛЬКО ВЕШАТЬ В ГРАММАХ»



Формула:

N_servers = MAX(N_по_местам, N_по_скорости)

Берём максимум из двух ограничений — вместимость и пропускная способность

А что если мы не уложились в целевой SLA? Что можно сделать, чтобы улучшить показатели?

ОПТИМИЗАЦИЯ: СЕРИИ ВАГОНОВ (H100 / H200 / B200)

H100
80 GB HBM3

Классический вагон. Достаточно для большинства моделей до 70B.

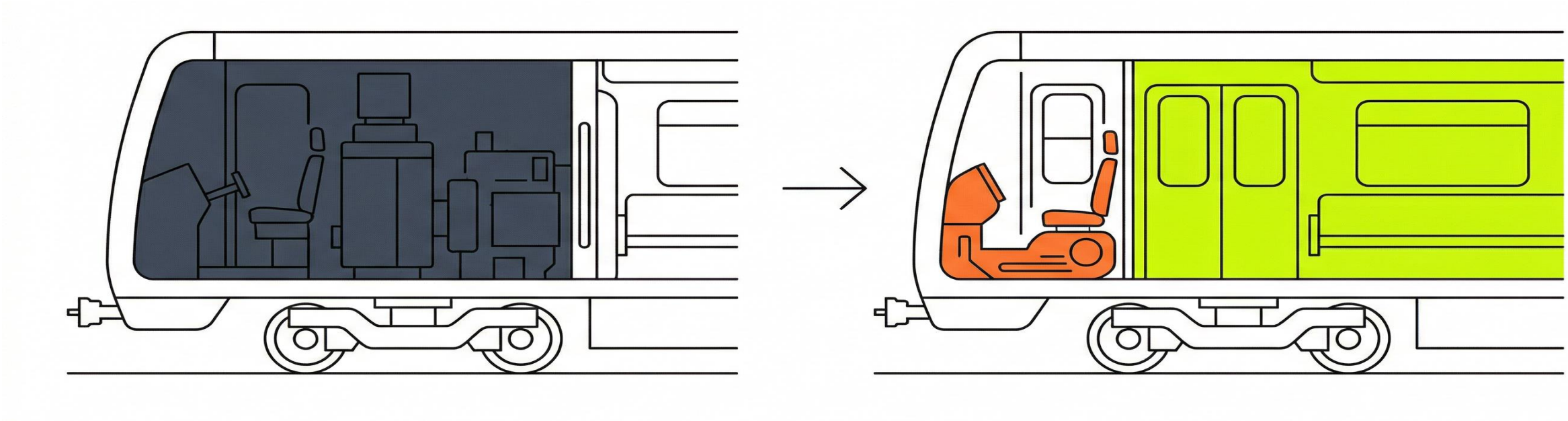
H200
141 GB HBM3e

Просторный вагон. Вмещает больше пассажиров (длинные контексты).

B200
192 GB HBM3e

Вагон нового поколения. Больше места + лучше технические условия для размещения двигателя.

ОПТИМИЗАЦИИ: КОМПАКТНОЕ ОБОРУДОВАНИЕ



КВАНТОВАНИЕ

Компактное оборудование кабины

Уменьшаем «оборудование кабины» — освобождаем место для пассажиров.

2–4×

меньше памяти на веса модели

Точность падает на единицы процентов — пропускная способность кратно растёт

ОПТИМИЗАЦИИ: КОМПАКТНОЕ ОБОРУДОВАНИЕ

CONTINUOUS BATCHING

Подбираем пассажиров на ходу

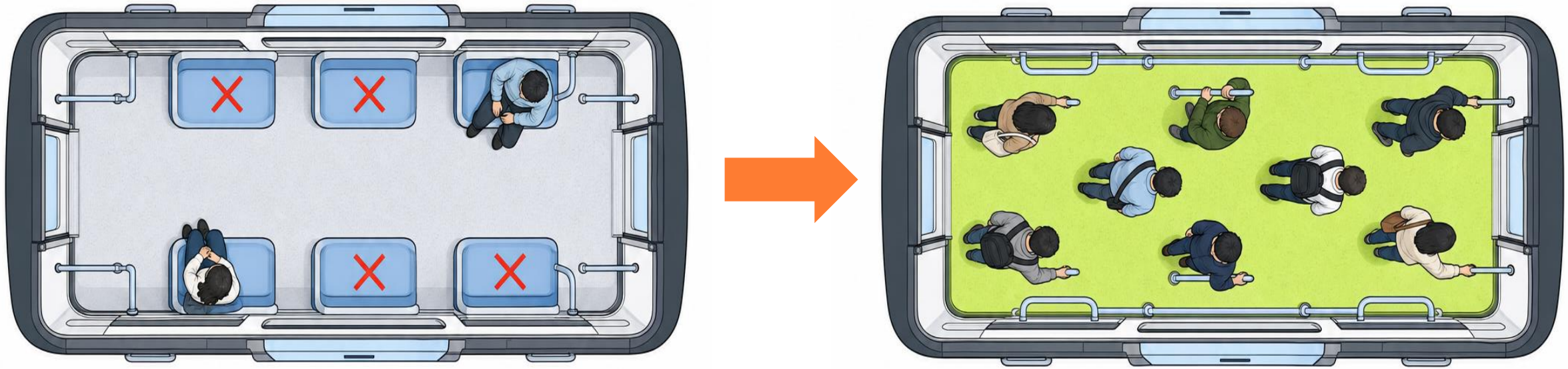
Один пассажир сошёл — на его место тут же садится следующий, не дожидаясь конечной станции.

2–3×

выше utilization GPU

Вагон всегда +/- полный.

ОПТИМИЗАЦИИ: КОМПАКТНОЕ ОБОРУДОВАНИЕ



PAGED ATTENTION

Динамическая рассадка

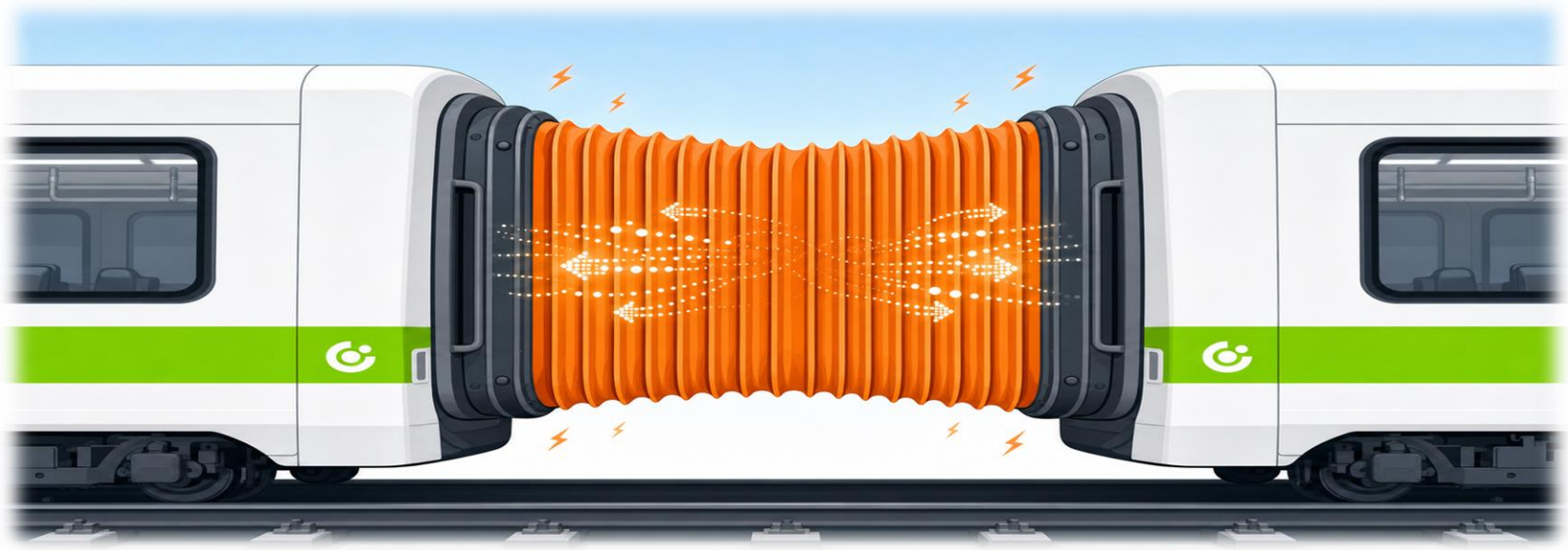
Рассаживаем пассажиров гибко — без фиксированных, зарезервированных мест.

2–3×

Больше параллельных сессий

на тот же GPU без потери качества

ОПТИМИЗАЦИИ: КОМПАКТНОЕ ОБОРУДОВАНИЕ



NVLINK

Мягкая сцепка между вагонами

Поток пассажиров внутри вагона осуществляется бесшовно и намного быстрее

5–10×

ПРОПУСКНАЯ СПОСОБНОСТЬ

выше, чем просто у PCIe

CORNER CASES: ГДЕ РАСЧЁТ МОЖЕТ НЕ СРАБОТАТЬ

отпбанк

Новая модель — другие размеры

Пересели с Llama-70B на Llama-405B?
Пересчёт всех параметров.

Длинный маршрут

Пользователь с контекстом 128K
занимает места как 10 обычных.

Вывод:

Расчёт даёт хорошую стартовую точку, но нужна калибровка.

Формула + замер + корректировка = надёжная оценка.

03 ПРОВЕРКА И УТОЧНЕНИЕ

КАЛИБРОВКА

1. Расчёт

Формула даёт теоретическую оценку N серверов.

Входные данные: модель, пользователи, SLA.

2. Замер

Бенчмарк на реальной инфраструктуре.

TTFT, ITL, throughput — сравниваем с расчётом.

3. Уточнение

Корректируем формулу по результатам.

Итеративный цикл:
расчёт → замер → уточнение.

НАГРУЗОЧНОЕ ТЕСТИРОВАНИЕ: ЧТО МЕРЯЕМ

Три параметра, которые отличают **работающий стенд от красивого слайда**

01

Latency

Укладываемся ли в SLA

TTFT и ITL — основные метрики качества для пользователя

02

Throughput

Запросов и токенов в секунду

Максимальная пропускная способность, после которой начинается деградация

03

Concurrency

Параллельные сессии

Как система ведёт себя при росте числа одновременных пользователей

ВАЖНЫЙ ПРИНЦИП:
Тестируем не одну точку, а поведение системы до предела

НАКОПЛЕНИЕ СТАТИСТИКИ ИСПОЛЬЗОВАНИЯ

Фактические токены
Реальное распределение длин запросов и ответов

Реальные пики активности
Пиковая нагрузка по времени суток

Реальная одновременность
Фактическое число параллельных сессий

Реальные профили пользователей
Поведенческие паттерны и частота обращений

ВАЖНЫЙ ПРИНЦИП:
Теория → Практика: каждый параметр уточняется на реальных данных

ЦИКЛ УТОЧНЕНИЯ

Расчёт — это не «посчитал и забыл», а **непрерывный цикл уточнения по фактическим данным**

01 **Стартовая оценка**

По методике, на бумаге

02 **Замер на стеке**

Нагрузочное тестирование

03 **Поправочные коэффициенты**

Считаем отклонения от аналитики

04 **Обновлённый расчёт**

Уточнённая модель — снова в работу

НАБЛЮДЕНИЕ:
Через месяц-два production оценка из теоретической становится эмпирической

Идеальная ли это модель?

Это не идеальная модель

Но достаточная для оценки сценариев
и принятия взвешенных решений

И мы уверены, что она будет становиться точнее

04 ПРАКТИЧЕСКАЯ ПОЛЬЗА

БЕРИ И ПОЛЬЗУЙСЯ: OPEN SOURCE

2 Definition of Simultaneously Active Users

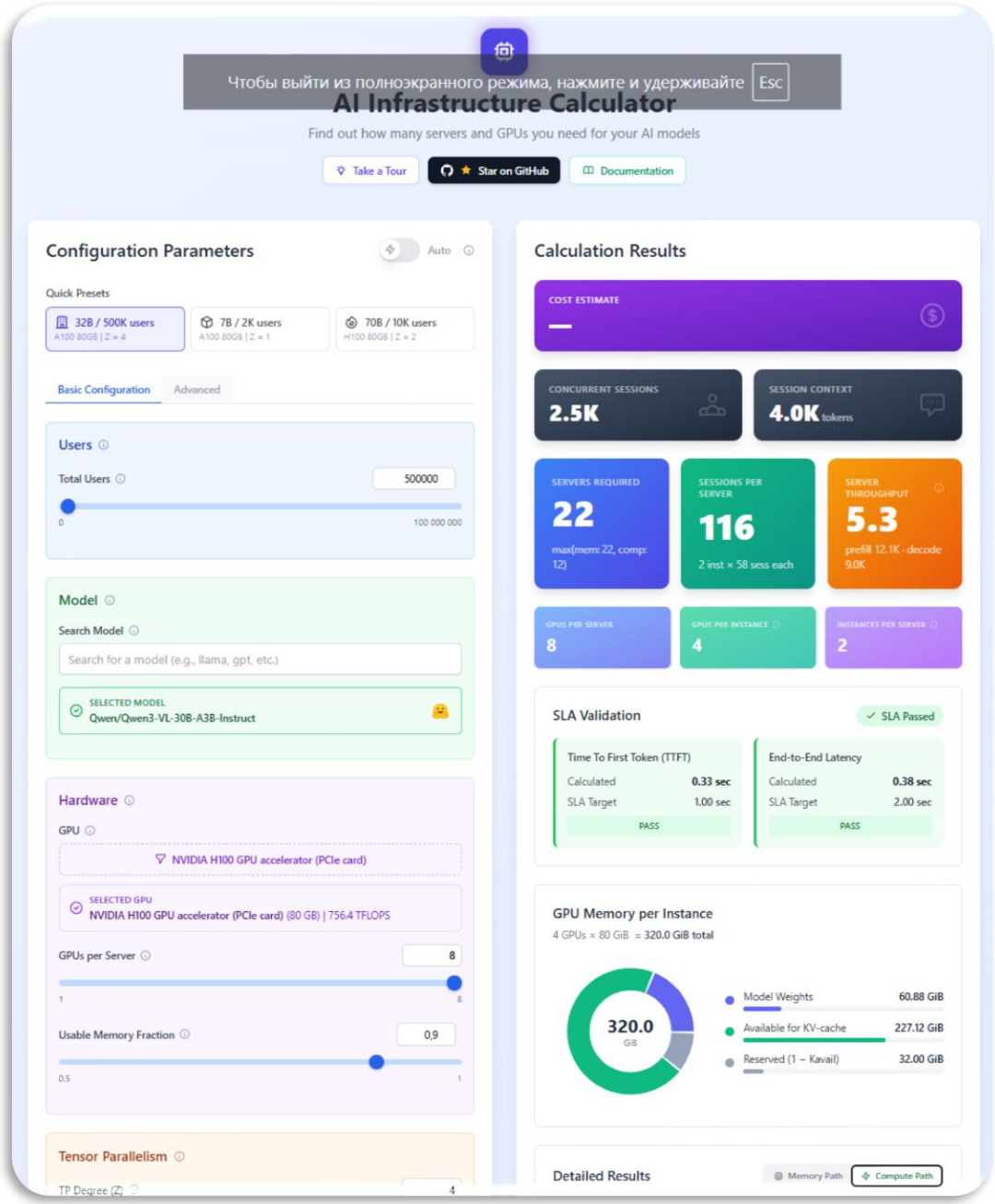
The calculation of the number of users is performed separately for each segment. In particular, if there is a division into external and internal users (for example, staff and contractors, staff specialists and vendor/customer representatives), the calculation is performed for the internal segment, then for the external segment.

$$N_{\text{sim}} = N_{\text{users}} \times K_{\text{pen}} \times K_{\text{sim}} \quad (1)$$

Used in: eq. (13), eq. (18)
where:

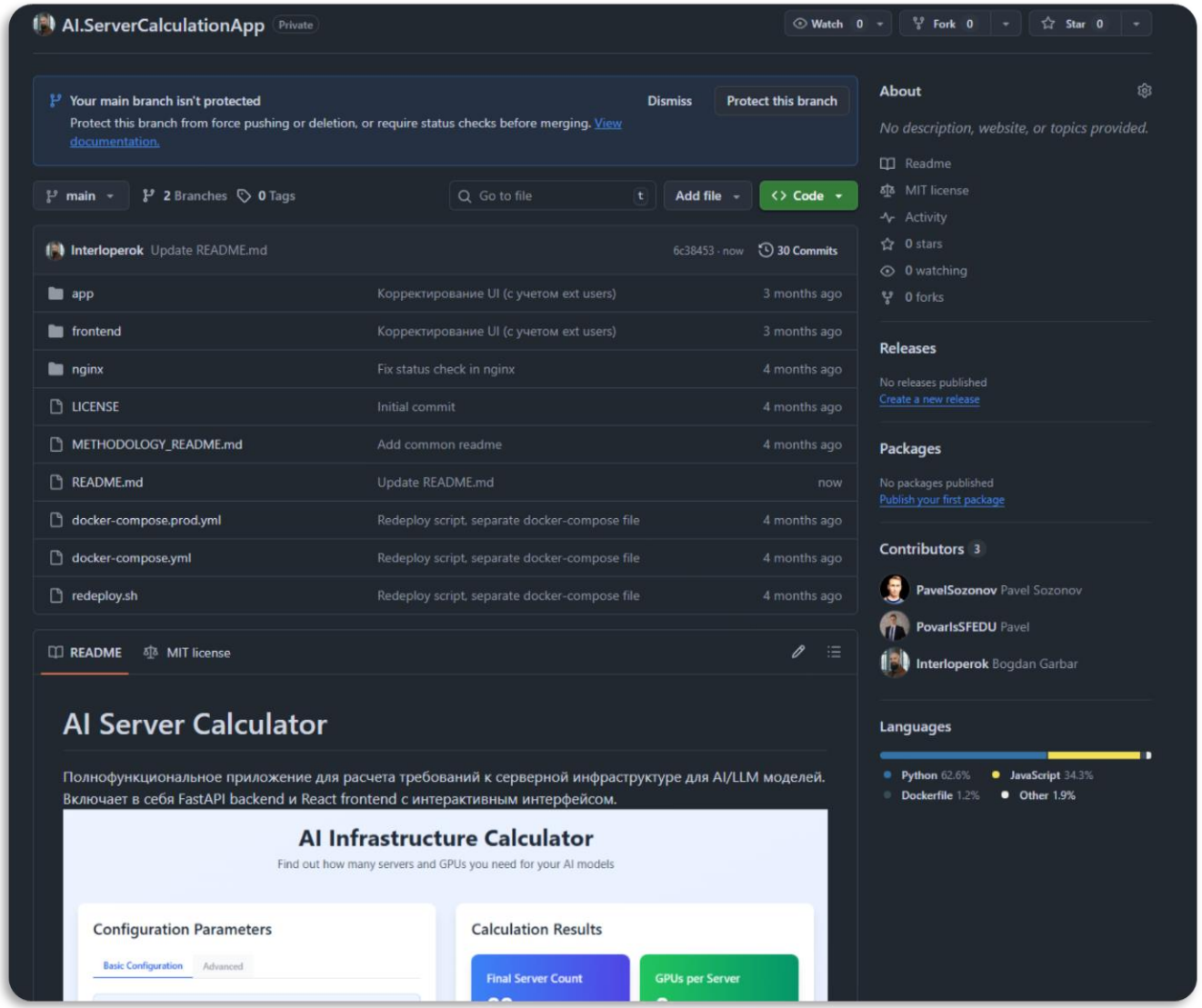
Variable	Definition	Unit	Note
N_{sim}	Number of employees simultaneously using the service per unit time	User/unit time	Calculations are performed for each user segment separately if needed (e.g., internal employees

Полный текст методики опубликован на Archive.org и в GitHub



БЕРИ И ПОЛЬЗУЙСЯ: OPEN SOURCE REPOSITORY

Здесь вы найдете все что нужно!



https://github.com/Interloperok/AI.ServerCalculationApp

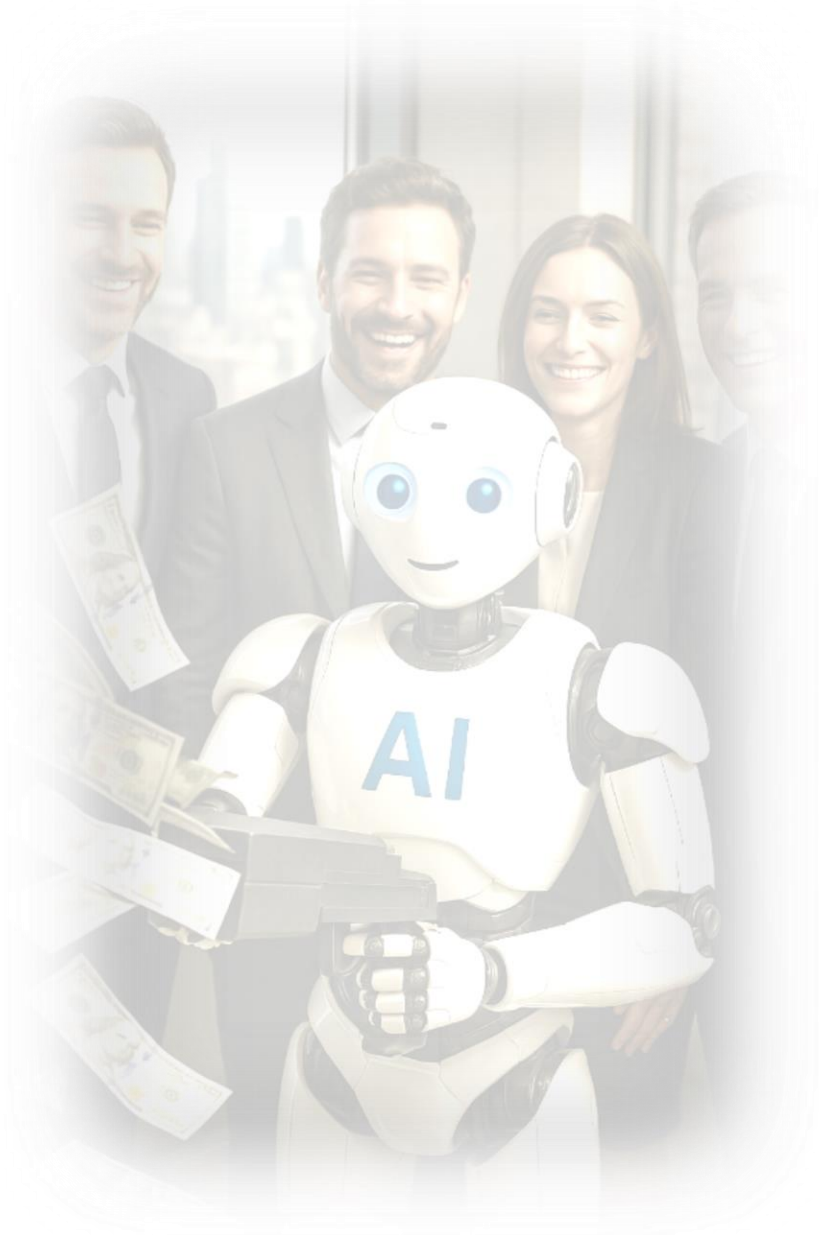
3 ШАГА НА ЗАВТРА

1.  **Соберите вход и прогоните калькулятор**

2.  **Защитите цифру перед CFO на 2-3 года**

3.  **Закупите минимум, подключите метрики и собирайте факт с последующей калибровкой**

05 ЗАКЛЮЧЕНИЕ



ТАК СКОЛЬКО «ВЕШАТЬ В ГРАММАХ»?

«ВЗЯТЬ С ЗАПАСОМ» — МОЖЕМ СЕБЕ ПОЗВОЛИТЬ, НО НУЖНО ЛИ?

МЫ ДЛЯ СЕБЯ НА ЭТИ ВОПРОСЫ ОТВЕТИЛИ ЧЕРЕЗ ФОРМАЛИЗОВАННУЮ И ПРОВЕРЯЕМУЮ ЛОГИКУ.

И ПОКАЗАЛИ, КАК МОЖНО ПРИЙТИ К ТАКИМ ВЫВОДАМ.

отпбанк

СПАСИБО ЗА ВНИМАНИЕ

AI Platform Center Lead
Богдан Гарбар