

# Симуляция распределённого обучения больших нейронных сетей с использованием Commodity серверов



Артеми́й Вишняков

Руслан Андреев

[cloud.ru](https://cloud.ru)

01

Обзор и постановка  
проблемы

02

Подходы к решению -  
концепция симулятора

03

Применение симулятора для  
анализа больших сетей

04

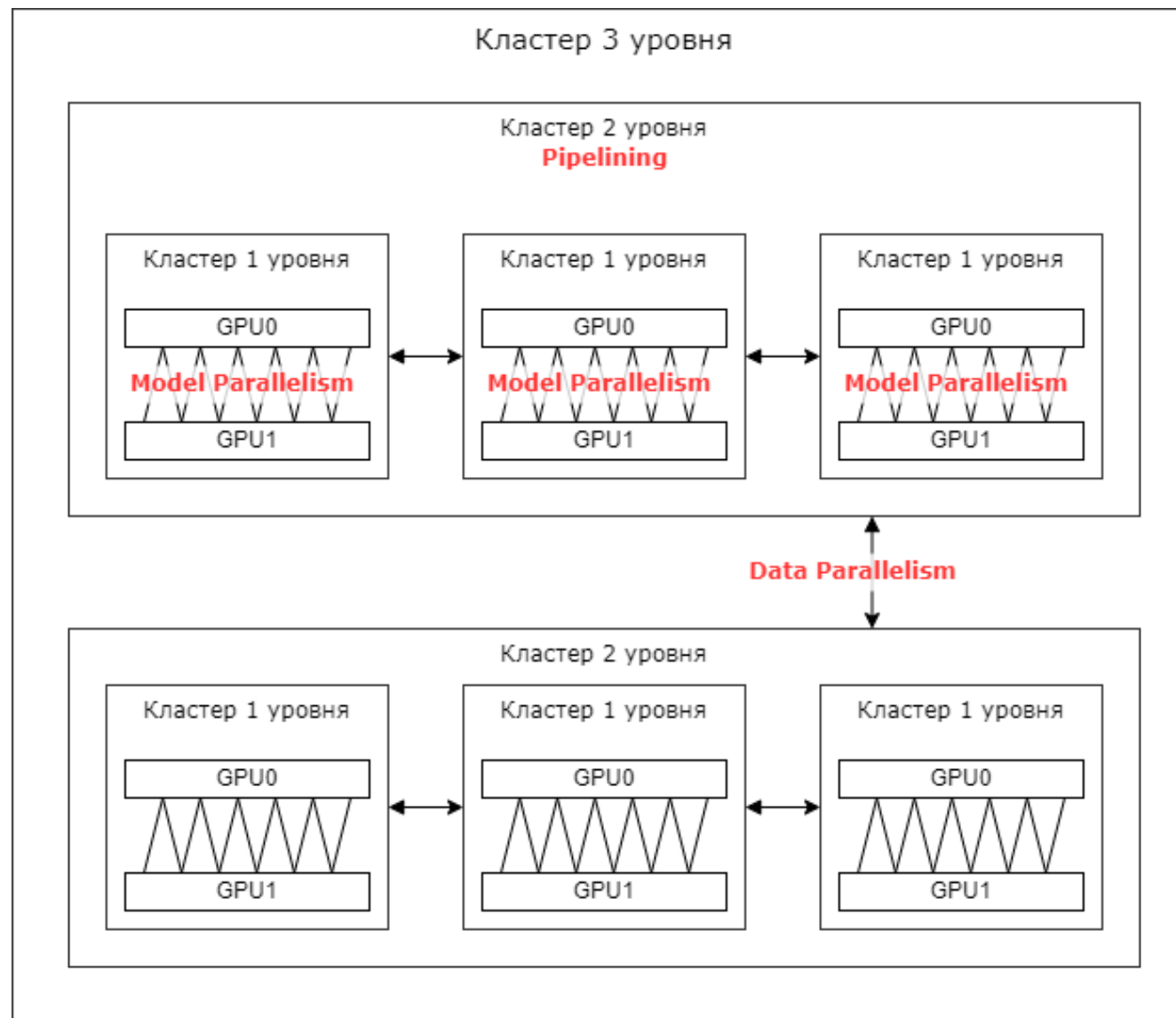
Выводы  
и заключение

- Большое количество вычислительных узлов
- Каналы передачи данных существенно различной пропускной способностью
- Многоуровневое распределённое обучение

С

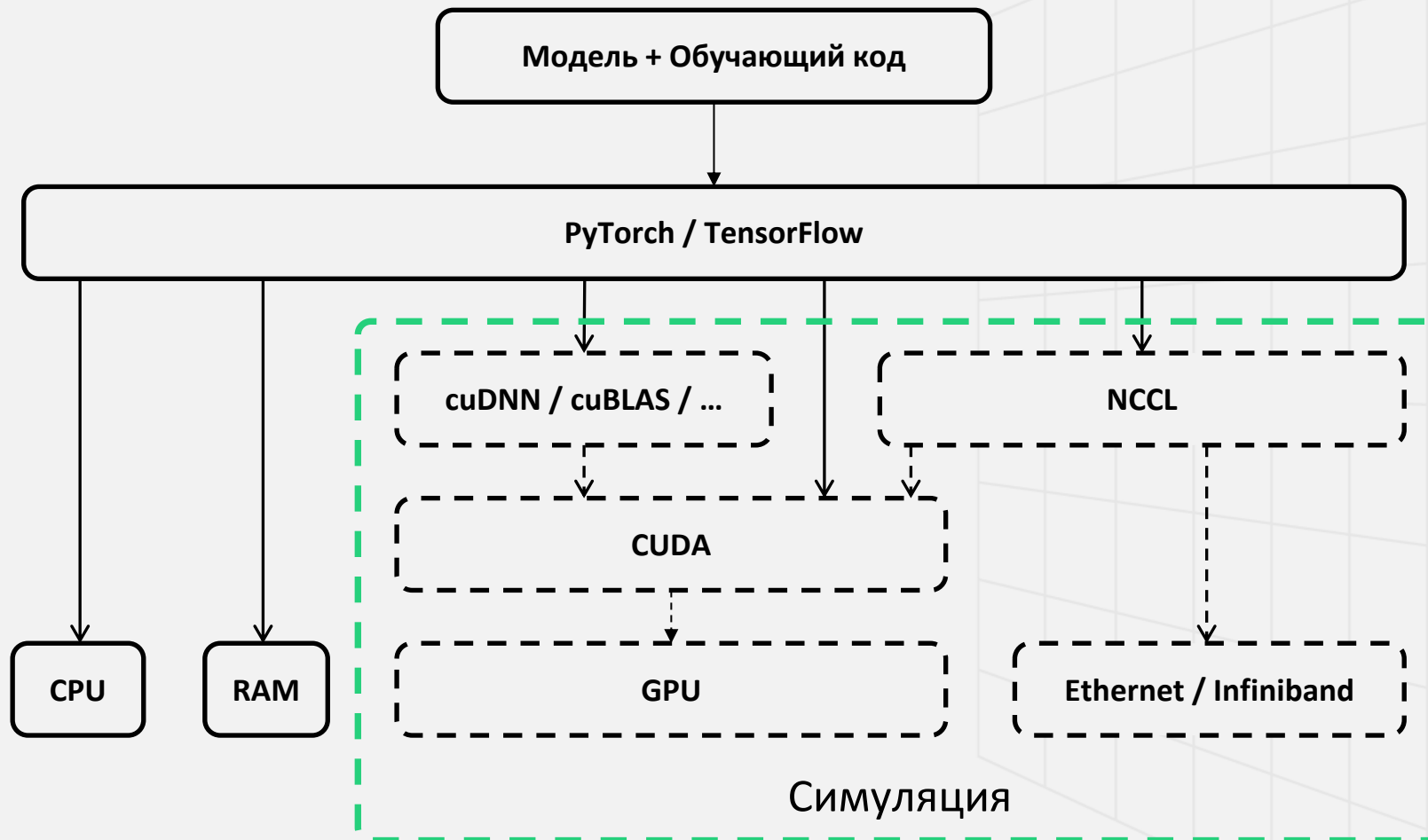


- 01 Математическая модель
- 02 Минимальный кластер
- 03 Симуляция



# КОНЦЕПЦИЯ СИМУЛЯТОРА

- Результаты вычислений не влияют на последовательность вычислений
- Mock API видеокарт
- Оценка времени вычислений
- Оценка времени передачи данных

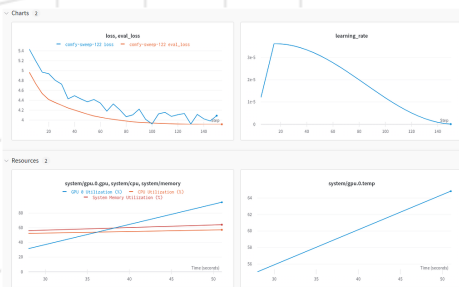


## Знакомый аналитикам интерфейс

1 ядро на симулируемую видеокарту: 6x ускорение

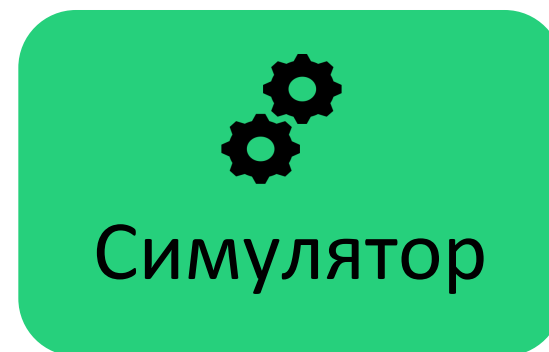
0.20 ядро на симулируемую видеокарту: 1x ускорение

Эксперимент: симуляция 3072 видеокарт на 4 только сри хостах



```
[2]: from cluster_simulator import GPUmodel, ServerConfig, Simulator, NetworkConfig

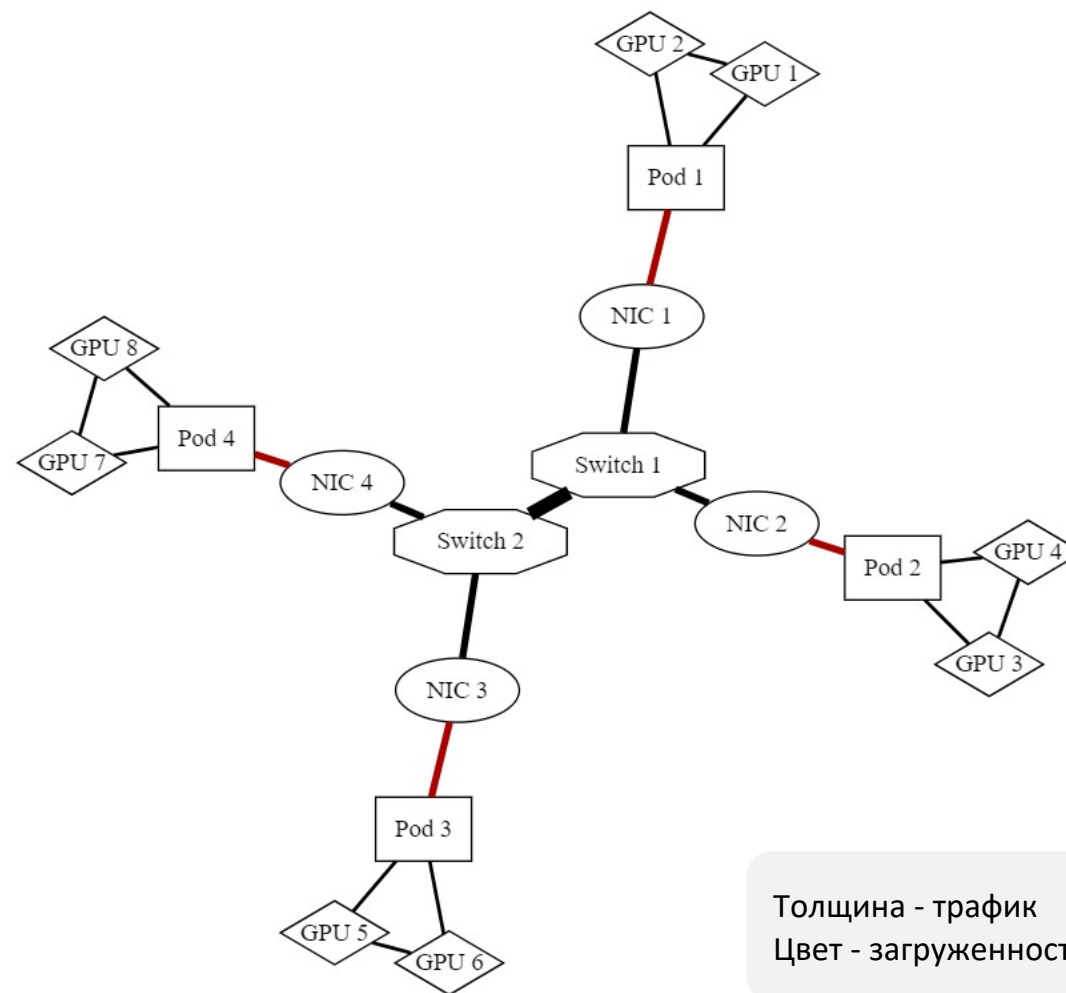
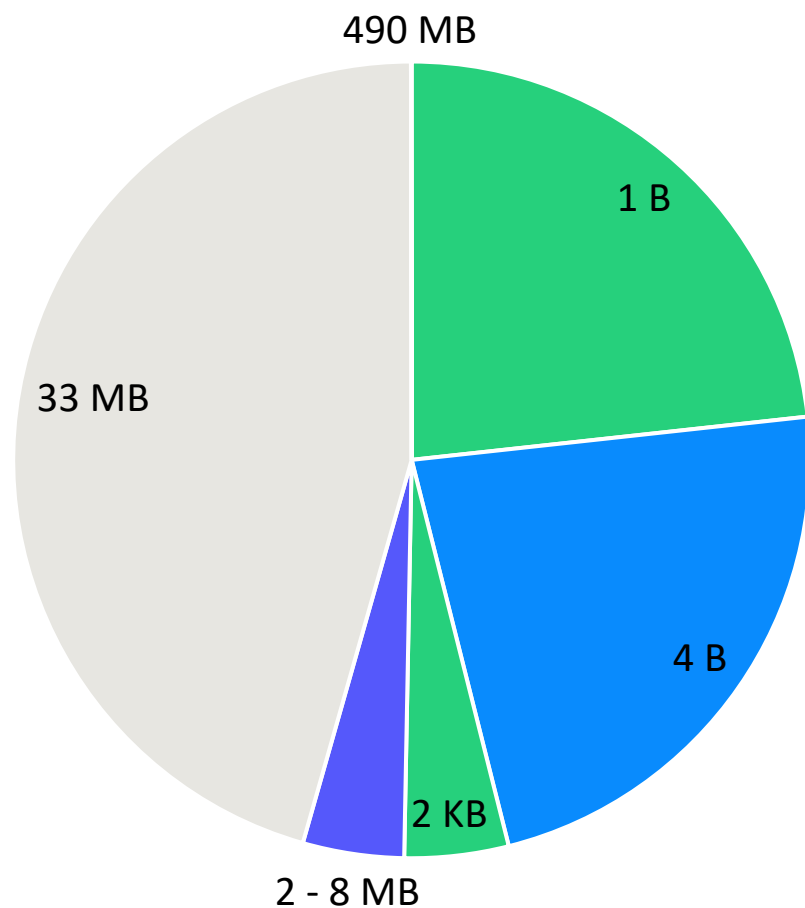
sim = Simulator(
    servers=[ServerConfig([GPUmodel.A100] * 8)] * 384,
    network_topo=NetworkConfig('fattree', **NetworkConfig.IB_EDR)
)
sim.run('launch.sh')
```



- calls.log
- network\_events.log
- stats.json
- stderr.txt
- stdout.txt



Распределение размеров транзакций  
(ZeRO 2, bloom-560m)



Толщина - трафик  
Цвет - загруженность

## Что мы научились делать:

- 1) Поиск ошибок в коде
- 2) Подбор параметров запуска
- 3) Определение загрузки каналов внутренней сети симулируемого кластера

## Ещё хотелось бы сделать:

- Оценка времени исполнения шагов конвейера модели
- Оценка эффективности использования ресурсов
- Поддержка будущих фреймворков