РОССИЙСКИЙ ФОРУМ
МИКРОЭЛЕКТРОНИКА

ШКОЛА
МОЛОДЫХ УЧЕНЫХ

ФЕДЕРАЛЬНАЯ
ТЕРРИТОРИЯ
«СИРИУС»

15–23
сентября 2025

Исследование реализаций лог-структурированного блочного устройства в ядре Linux

Гавриленко Михаил, студент
Васенина Анна Игоревна, инженер-исследователь

СПбГУ (МатМех)

qrutyq@gmail.com

Лог-структурированность

Лог-структурированность — технология хранения данных, основанная на принципе журнала. Вместо записи в случайные места — все данные записываются последовательно.

- Применение при реализации мгновенных снимков, использующих схемы Copy-On-Write и Redirect-On-Write
- Применение в кэшировании
- Повышение скорости записи на накопитель

Существующие решения

Sprite LFS

- Является файловой системой, а не блочным устройством
- Введено понятие лог-структурированности и реализована базовая функциональность

Logical Disk

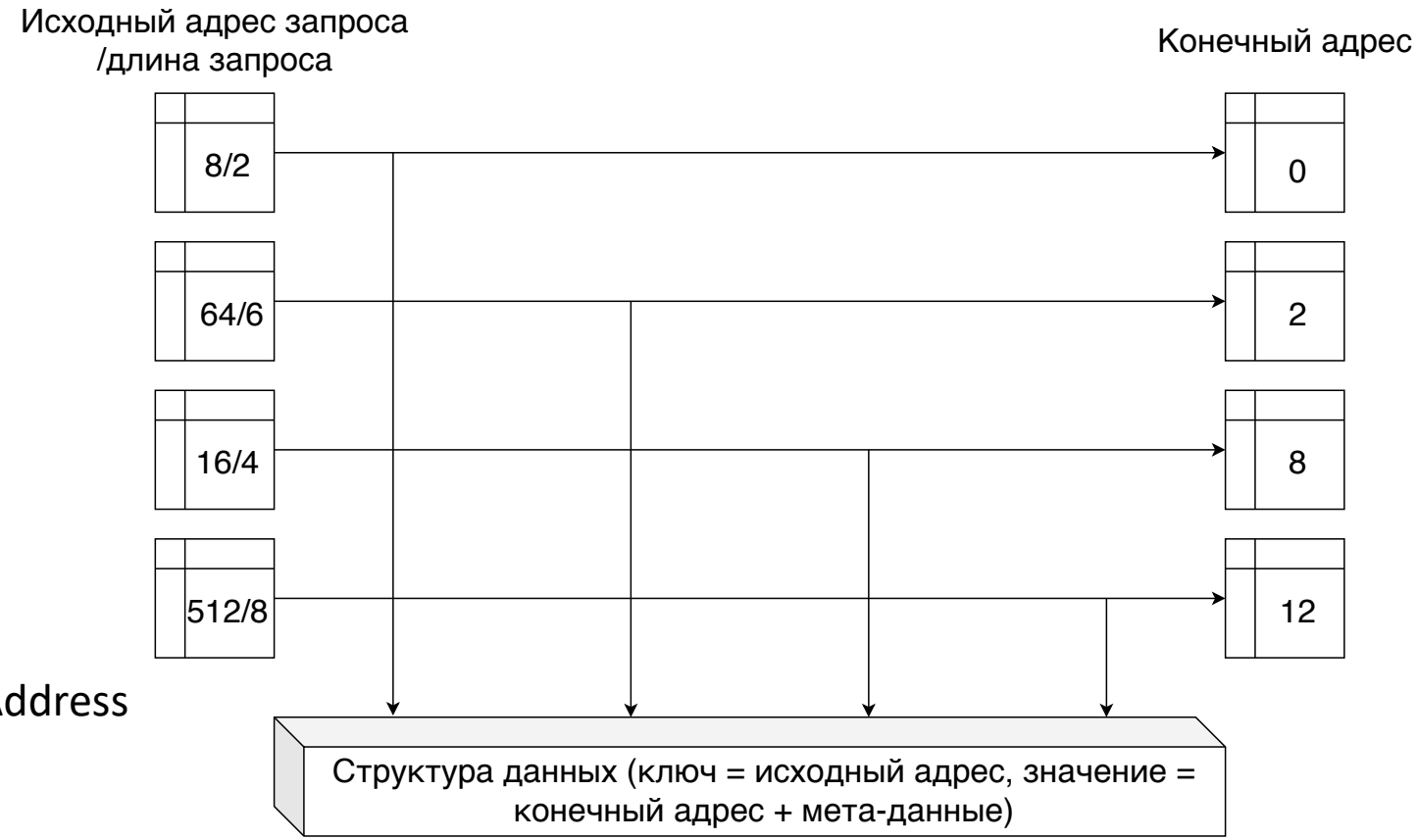
- Оптимизация файловых систем на блочном уровне
- Создан как инструмент для небольших систем

ZFS:

- Продвинутая файловая система
- Большая нагрузка на ресурсы
- Сложно интегрировать из-за величины проекта
- Нет прямого низкоуровневого доступа к данным

Производительность подхода

Производительность лог-структурированного подхода напрямую зависит от способа реализации таблицы LBA[1] отображений



[1] LBA - Logical Block Address

Постановка задачи

Целью работы является разработка модуля ядра Linux, реализующего фреймворк для тестирования производительности различных реализаций лог-структурированных блочных устройств

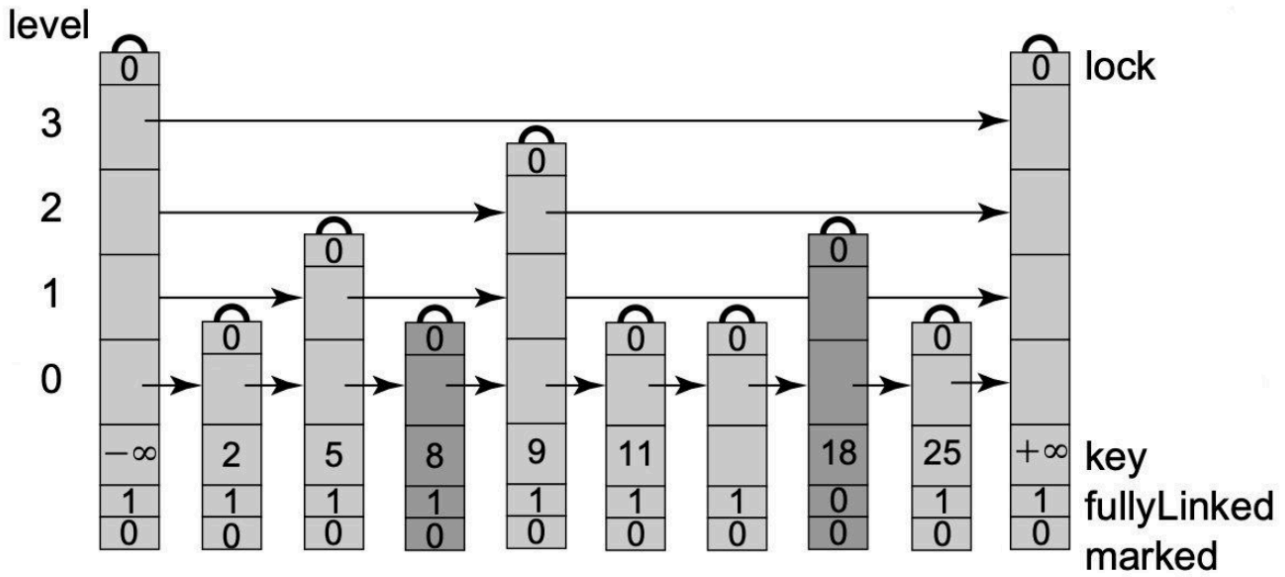
Задачи:

- Разработать драйвер блочного устройства, допускающий подключение различных структур данных для реализации лог-структурированного хранения
- Реализовать набор утилит для автоматического сбора и визуализации результатов тестирования
- Провести сравнение реализованных подходов и проанализировать полученные результаты

Реализация структуры данных — список с пропусками

За основу решения взят алгоритм Мориса Херлихи и работа Кейзера Фрейзера

- Открытая реализация с научными статьями
- Используется техника “помеченных указателей” для двух-стадийного удаления на основе single-word CAS[1]
- Реализована дополнительная MRS[2]



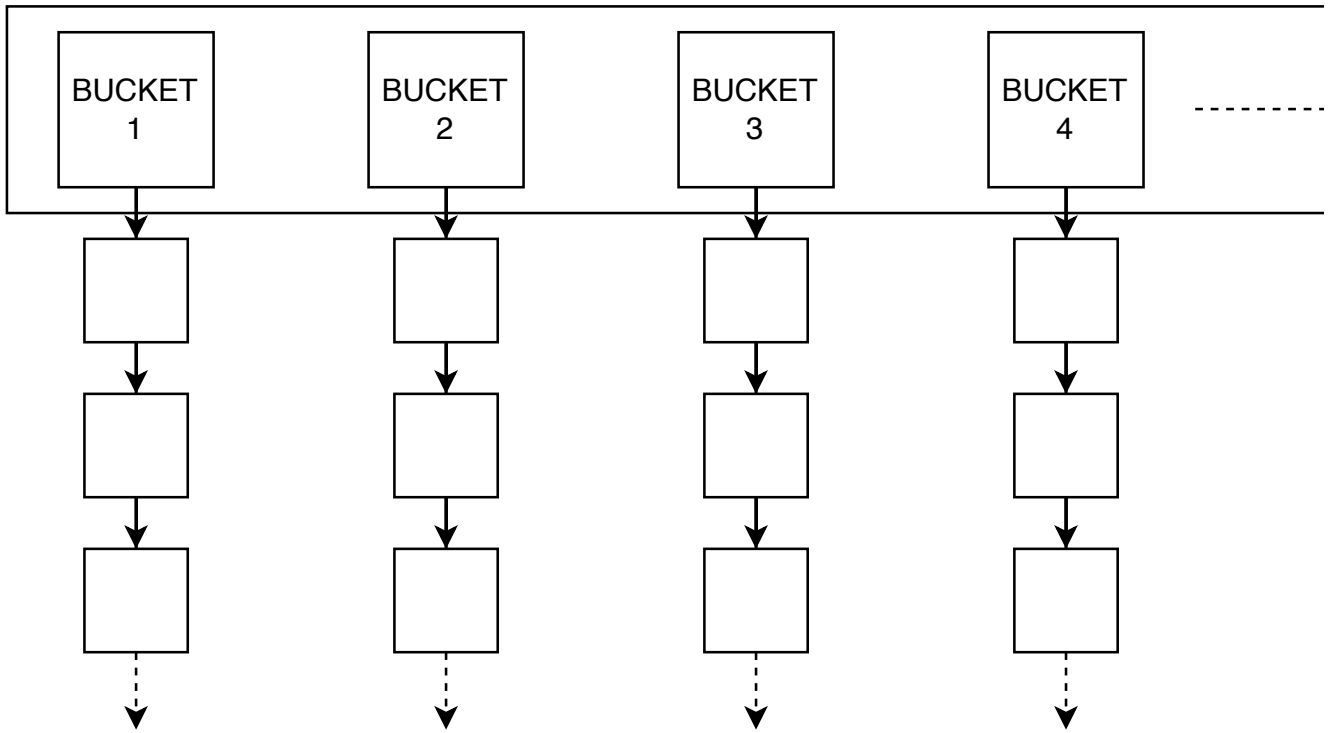
(Herlihy, Shavit. The Art of Multiprocessor Programming)

[1] CAS - Compare And Swap/Switch
[2] MRS - Memory Reclamation System

Реализация структуры данных — хеш-таблица

Было принято решение модифицировать RCU хеш-таблицу из ядра Linux

- RCU двусвязный список заменен на односвязный lock-free
- Используется техника помеченных указателей на основе single-word CAS[1]
- Реализована дополнительная MRS[2]



[1] CAS - Compare And Swap/Switch
[2] MRS - Memory Reclamation System

Экспериментальное исследование

Цель эксперимента — количественная оценка характеристик производительности и накладных расходов при использовании разных структур данных в основе лог-структурированного подхода

Были использованы такие метрики, как: **Bandwidth**, **IOPS**, **Latency**

Основные задачи эксперимента

- Производительность ввода-вывода
- Масштабируемость и параллелизм
- Потребление памяти

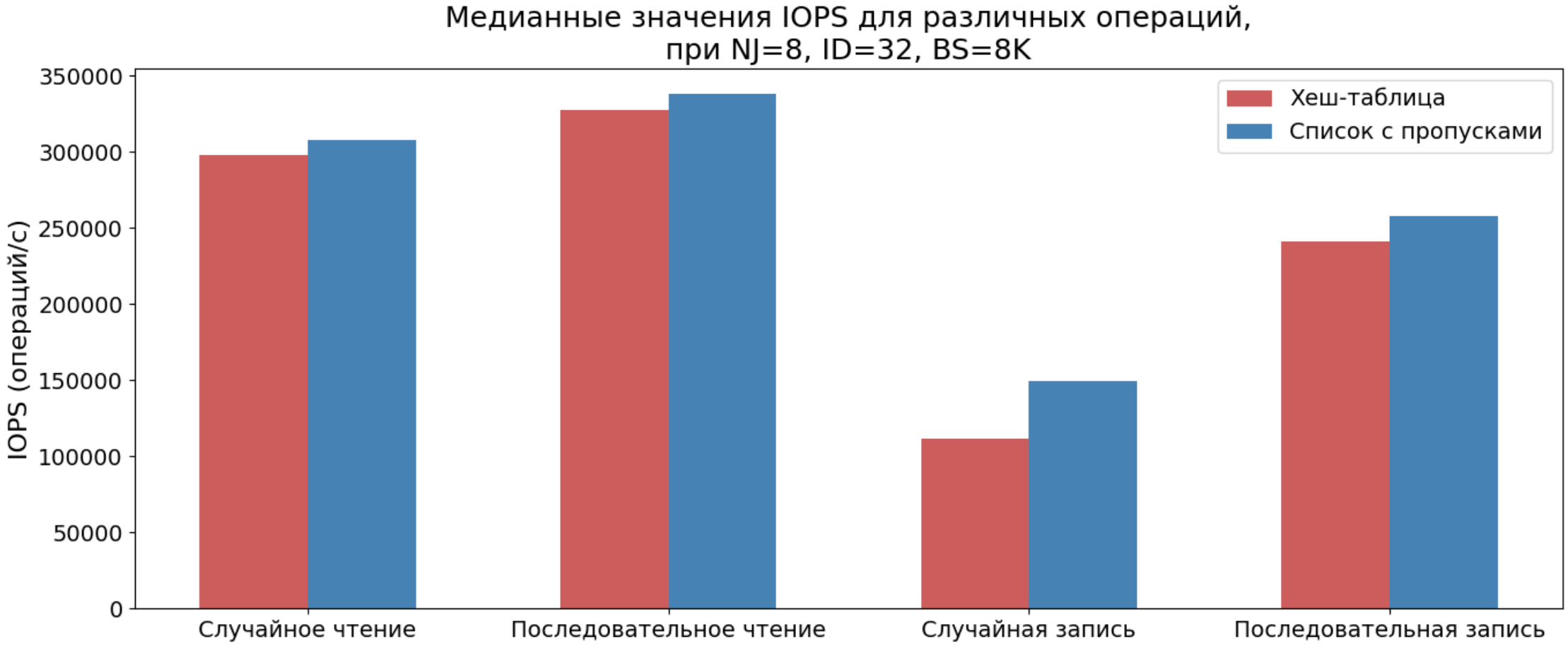
Экспериментальное исследование

Оценка производительности проводилась на сервере[1] со следующей конфигурацией:

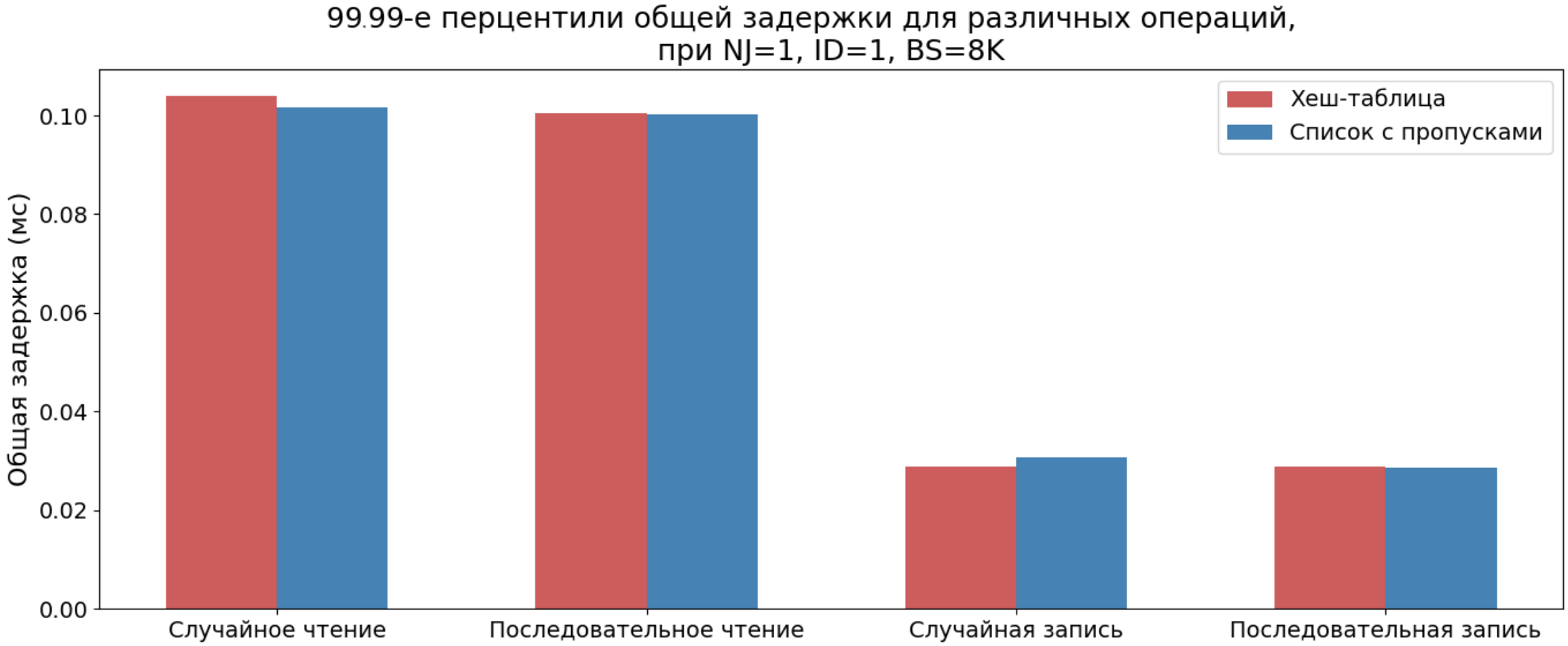
- Серверная платформа: 1U ASUS Generation E10 RS700-E10-RS12U
- Процессоры: Intel(R) Xeon(R) Gold 6338 CPU @ 2.00GHz (32 ядра, 64 потока) x 2
- ОЗУ: 512 ГБ DDR4, 3200 MHz
- Нижележащее устройство хранения: NVMe-диск SAMSUNG MZWLJ1T9HBJR-00007, 1.92 TB. (Скорость операции чтения до 7000 MB/s, операции записи до 2400 MB/s)
- ОС: Fedora Linux (Server Edition) 42, v6.14.0 kernel

[1] Оборудование было предоставлено ООО «ШВАЧЕР»

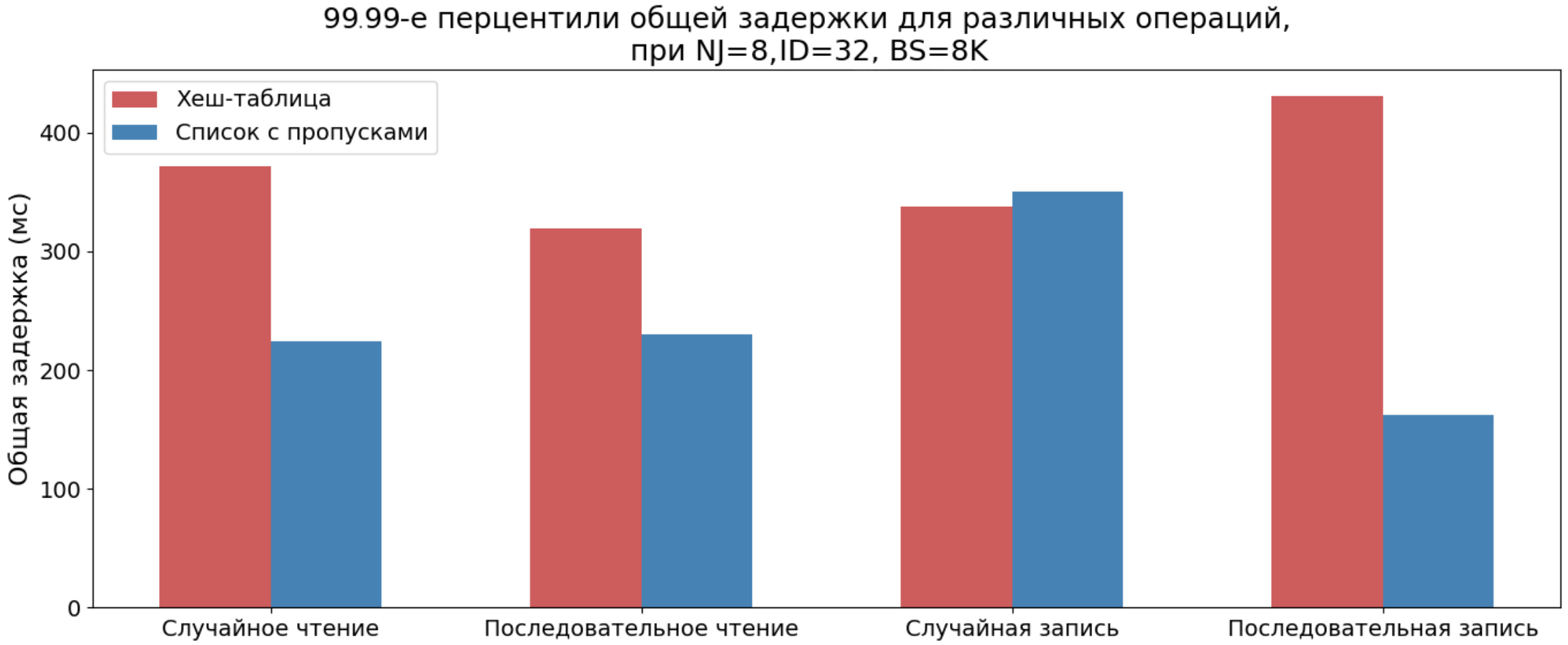
Производительность при различных нагрузках



Задержка при различных нагрузках



Задержка при различных нагрузках



Потребление памяти

Структура данных	Список с пропусками	Хеш-таблица
Узел	133 (224) Б	32 Б
Голова	32 Б	16Б + 262144 * 40 Б
Всего	16,5 (28) ГБ	4 ГБ

Результаты

- Разработан драйвер блочного устройства с возможностью подключения различных структур данных для реализации лог-структурированного хранения
- Реализован набор утилит для автоматического сбора и визуализации результатов хранения
- Проанализированы результаты измерений производительности реализованных подходов, сделан вывод об эффективности выбранных структур данных в терминах лог-структурированного подхода

С реализованной функциональностью можно ознакомиться в репозитории на Github (https://github.com/qrutyy/ls-bdd)

Возможные пути продолжения работы

- Реализация других lock-free структуры данных и анализ производительности
- Реализация блокирующих версий структур данных и сравнение производительности с lock-free версиями
- Реализация сохранения таблицы отображений LBA в энергонезависимую память
- Реализация полноценного подхода Redirect-On-Write

ШКОЛА МОЛОДЫХ УЧЕНЫХ

Спасибо за внимание!

- Благодарю компанию ООО «ШВАЧЕР» за предоставленное оборудование для тестирования и проведения экспериментов

ОФИЦИАЛЬНЫЙ САЙТ ШКОЛЫ МОЛОДЫХ УЧЕНЫХ

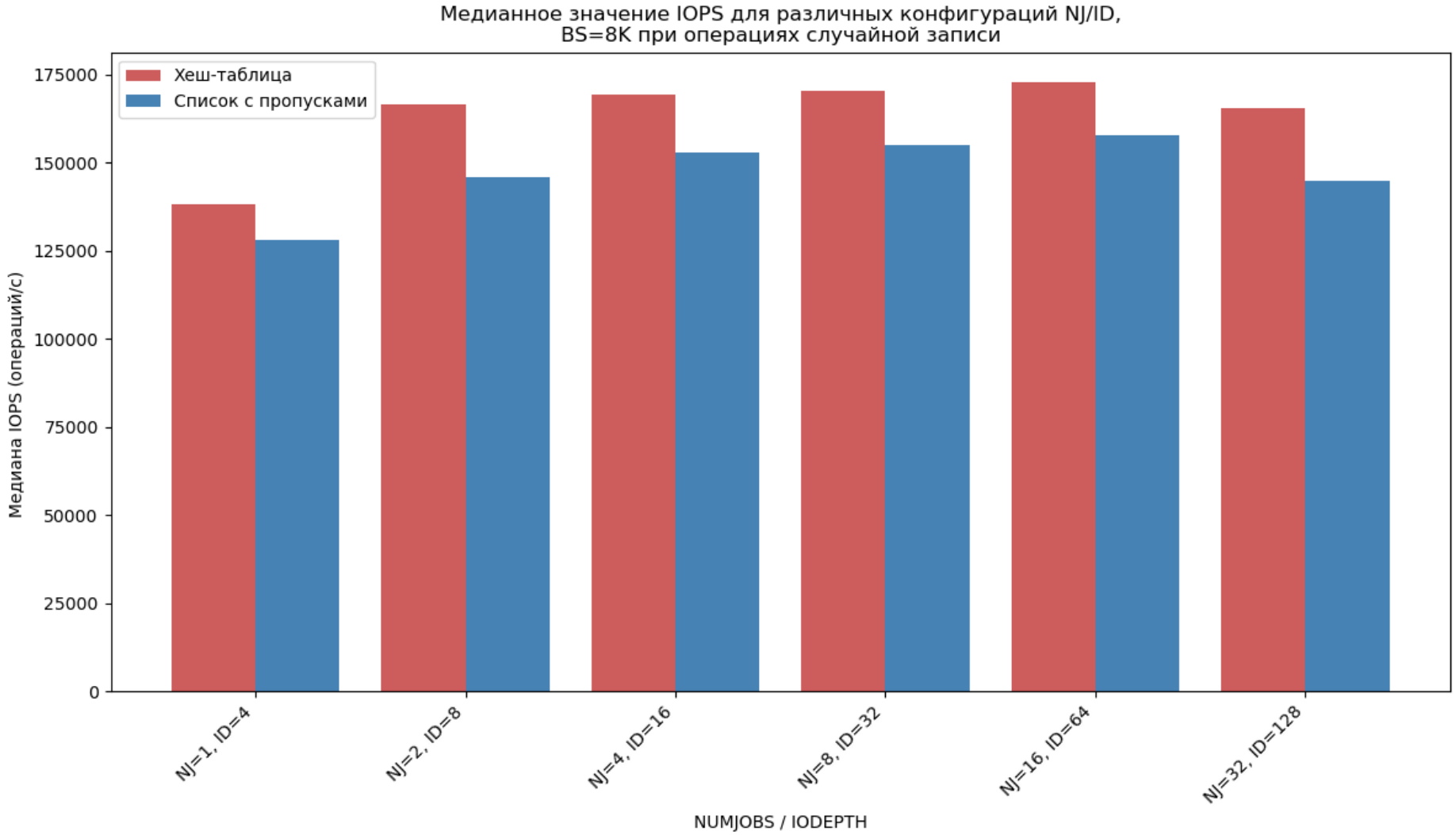
MICROELECTRONICA.PRO

ПОДПИСЫВАЙТЕСЬ И БУДЬТЕ В КУРСЕ ВСЕХ ПОСЛЕДНИХ НОВОСТЕЙ!

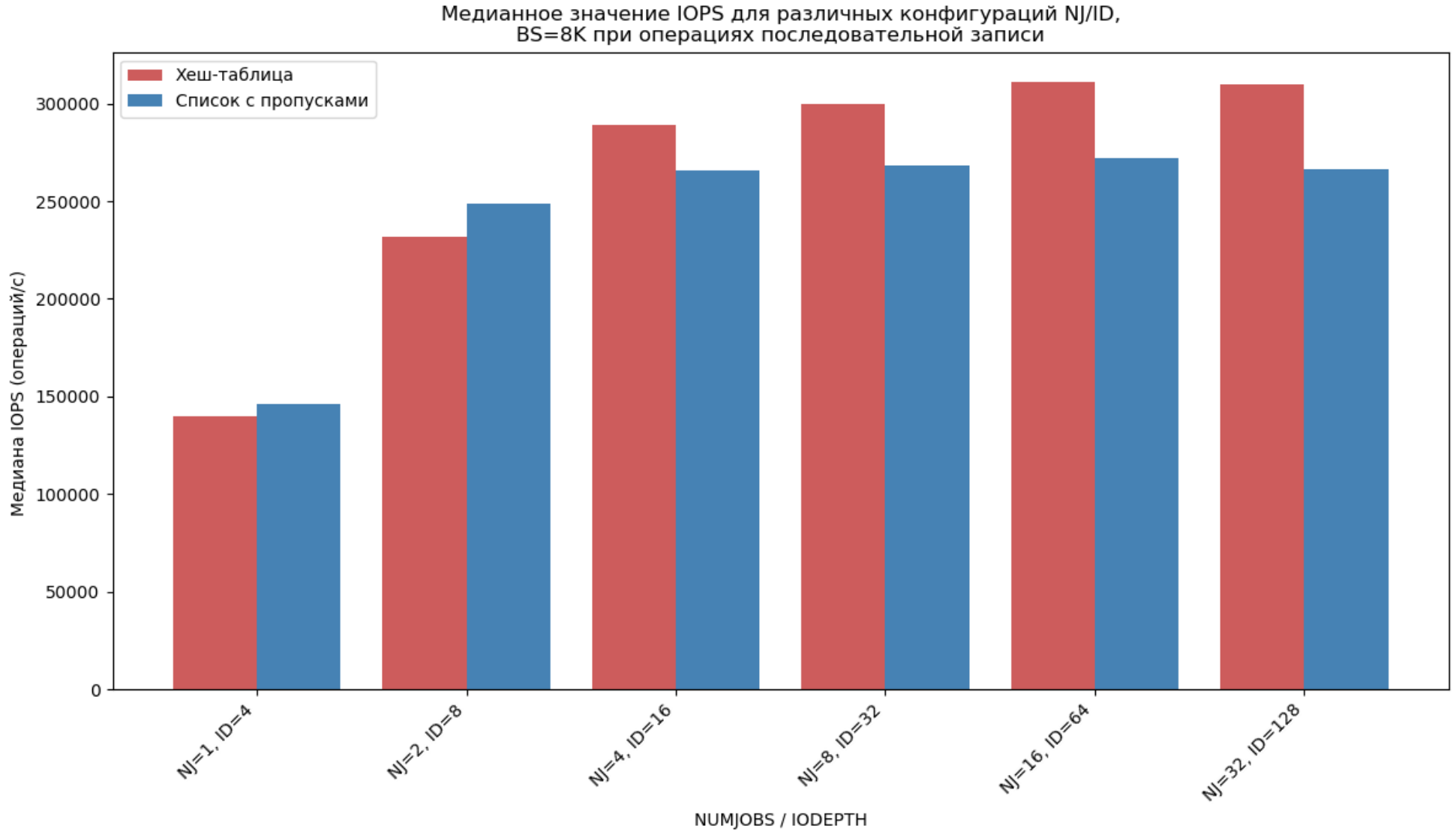
ВАШ ЛИЧНЫЙ КАБИНЕТ

Rutube

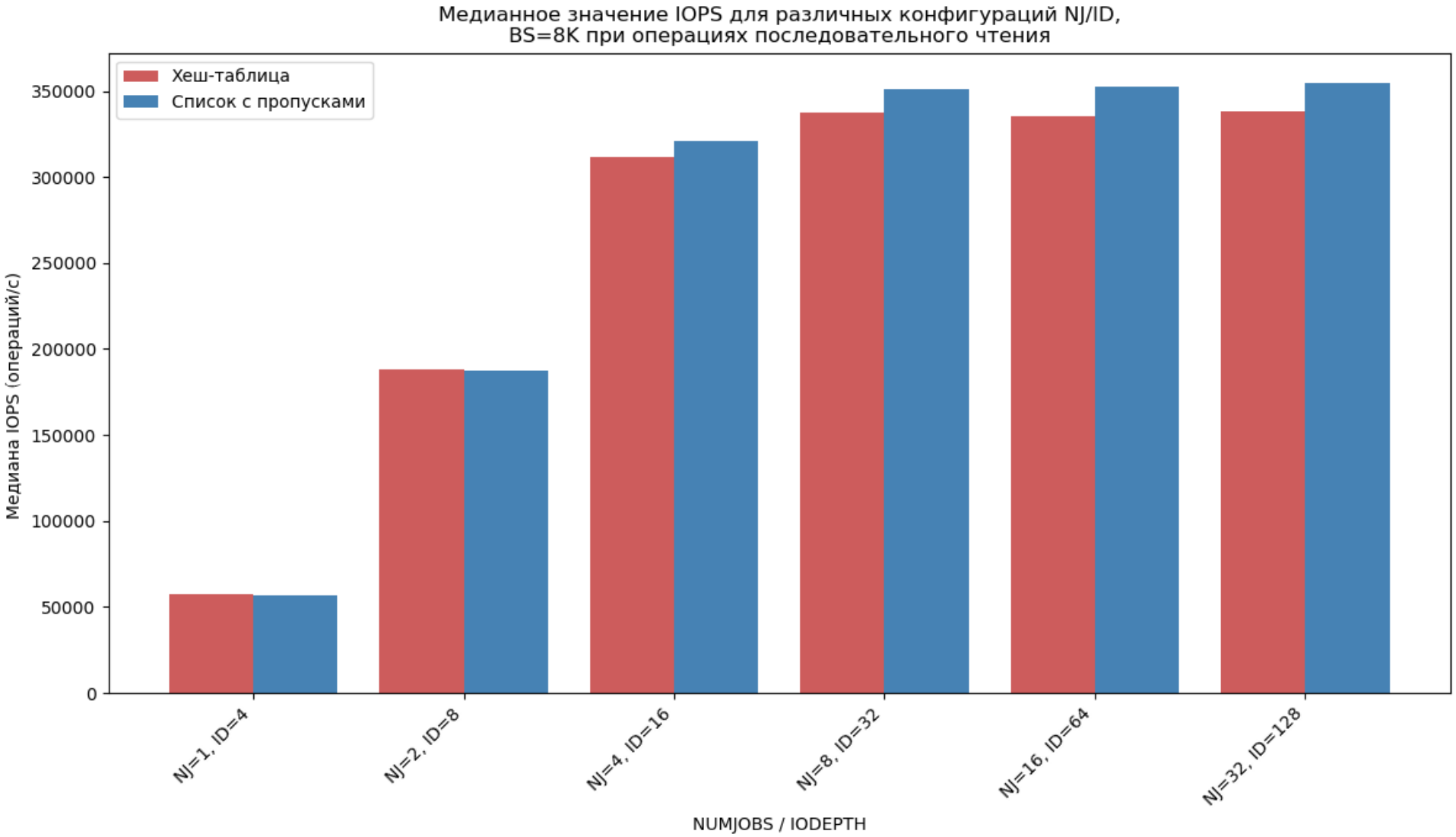
Доп. слайд (параллелизм при случайной записи)



Доп. слайд (параллелизм при посл. записи)



Доп. слайд (параллелизм при посл. чтении)



Доп. слайд (параллелизм при случайном чтении)