

# 2. HDFS. Архитектура.

BigData. Hadoop.

Преподаватель

**Леонов Евгений**



## Цели и принципы построения

- Файловая система HDFS создана для хранения большого объема неструктурированных данных
- Это означает: HDFS ваш выбор если
  - Ваши данные относительно большие (50Тб +)
  - Ваши файлы относительно большие (от 128 Мб минимальный размер)
  - Количество ваших файлов относительно невысоко (десятки миллионов)
- HDFS обеспечивает отказоустойчивость и параллелизацию обращения к файлам на основе разделения файлов на блоки данных
- HDFS обеспечивает Posix-совместимость - нет



## Отличия HDFS от других файловых систем

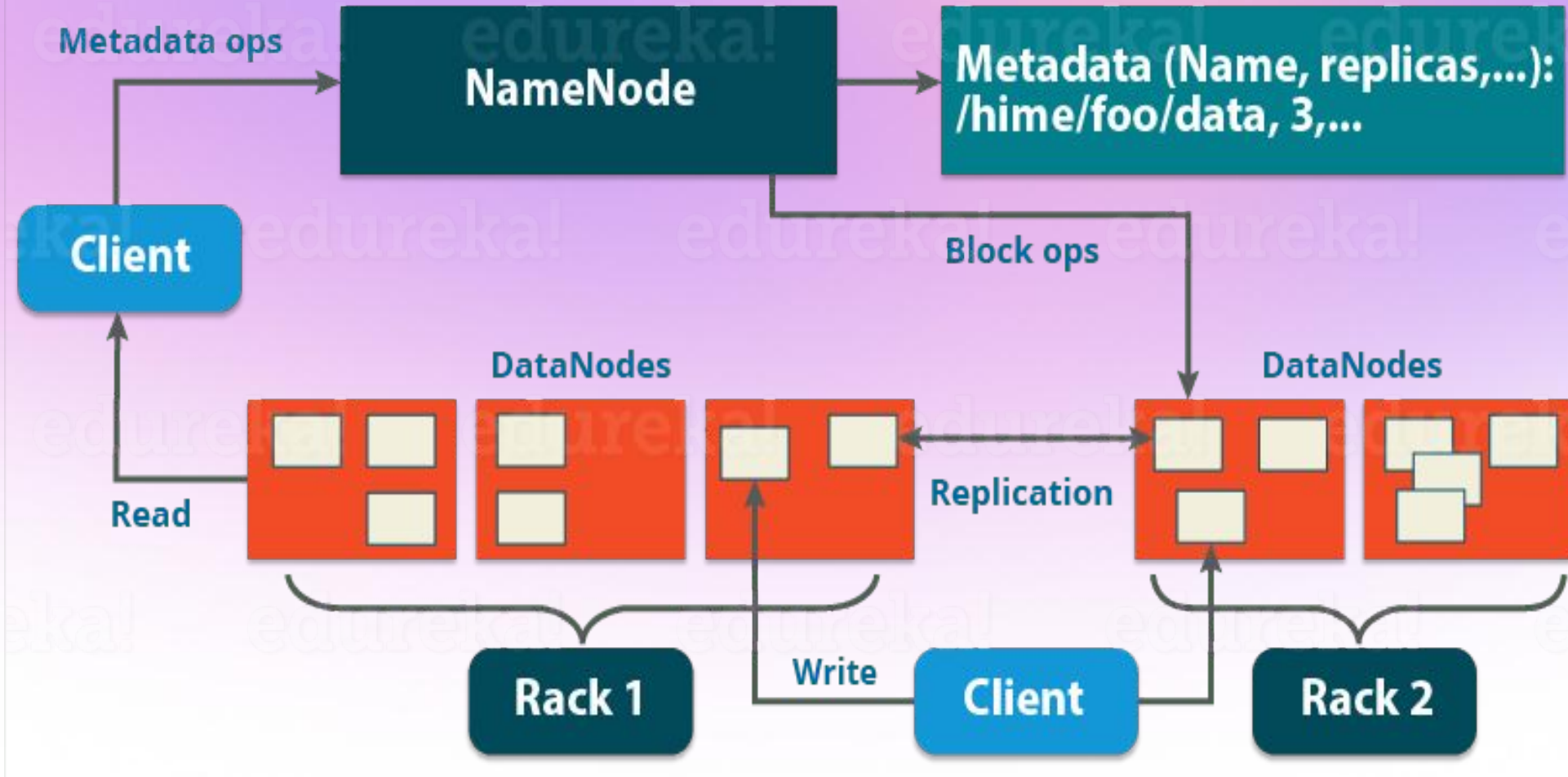
- Условные аналоги и конкуренты
  - ext2/ext3/xfs (локальная файловая система)
  - LizardFS («нормальная» кластерная ФС)
  - Oracle ASM (проприетарная файловая система созданная быть системой хранения для СУБД)
  - NFS :)))
  - Amazon S3 (Объектное хранилище от Amazon, единственный реальный конкурент)
- Самое главное отличие HDFS от всего вышеперечисленного:
  - **НЕИЗМЕНЯЕМЫЕ ФАЙЛЫ**
- То есть файл можно создать, дописать и удалить. Изменить файл уже созданный невозможно. Ну и прочитать тоже можно.



## Основные сущности HDFS

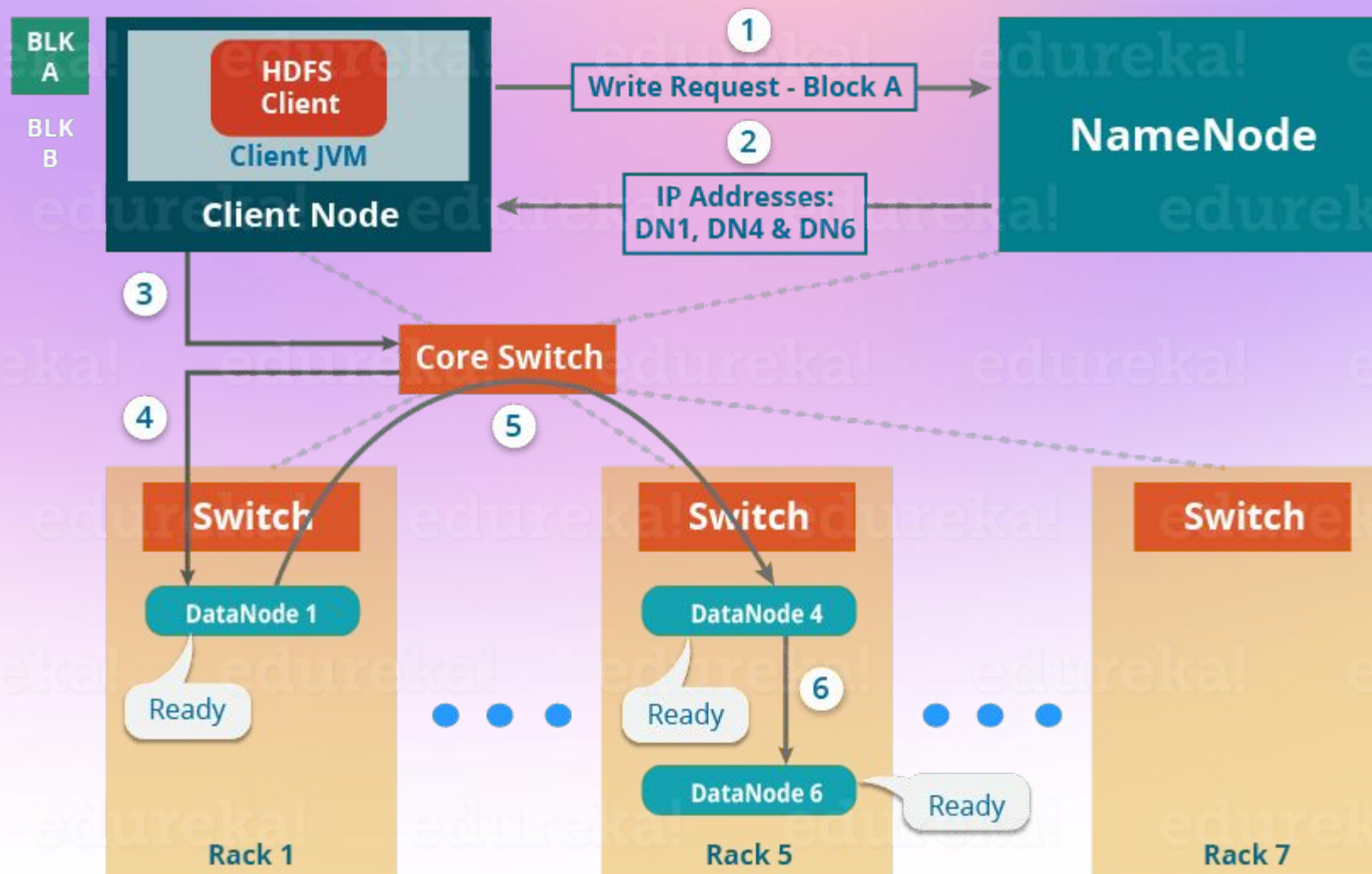
- **Хранение**
  - **Файл**
    - обычный файл, как на любой другой FS. Имеет те-же свойства (например время создания, владелец, группа и права доступа) плюс некоторые специфические - фактор репликации. Файл делится на блоки.
  - **Блок**
    - единица репликации. Кусок файла кратный размеру блока, обычно 128-256 Мб
- **Кластер**
  - **NameNode**
    - Узел, содержащий в памяти образ файловой системы
  - **DataNode**
    - Узел, содержащий собственно блоки файлов

## HDFS Architecture



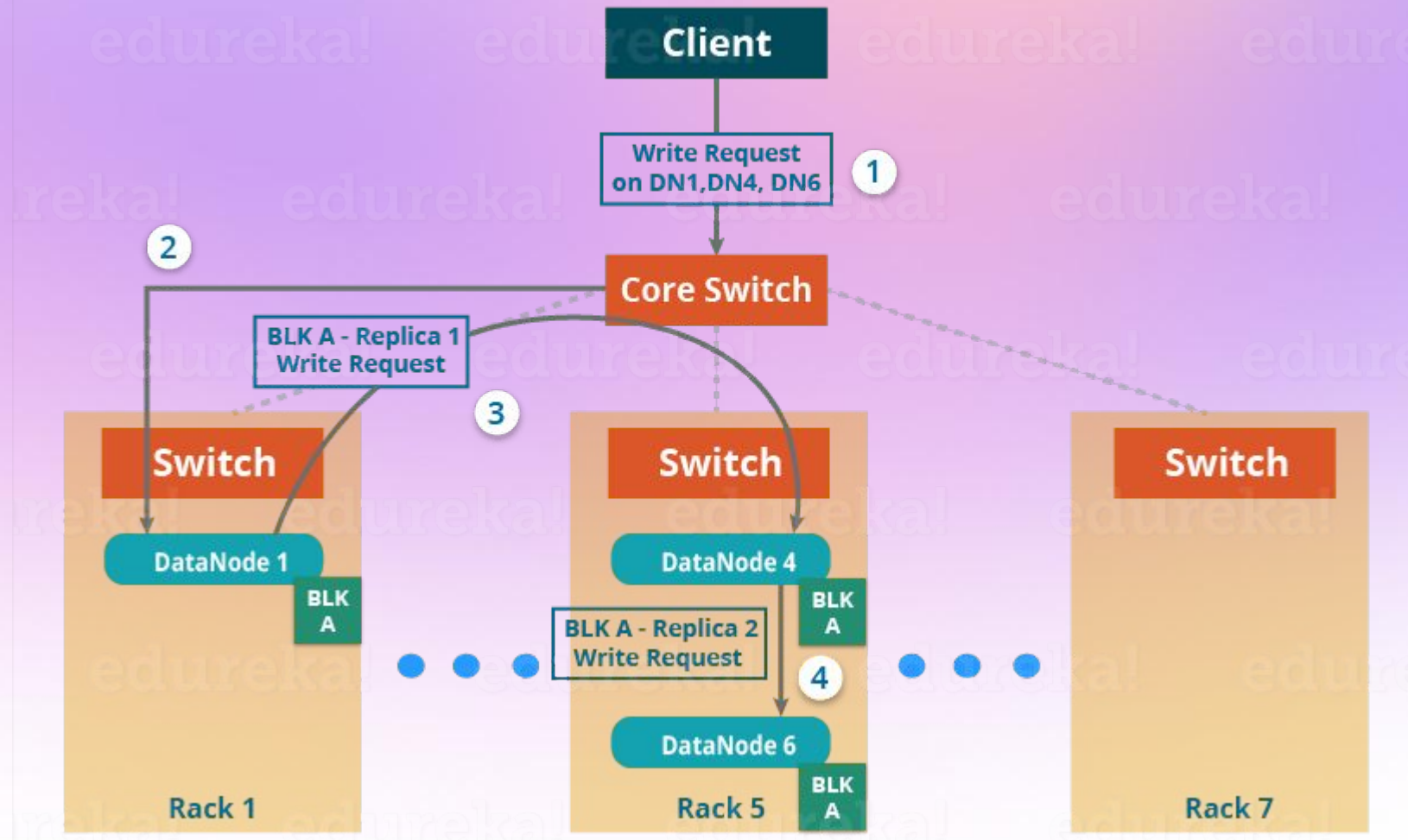
## Setting up HDFS - Write Pipeline

Запись в  
HDFS

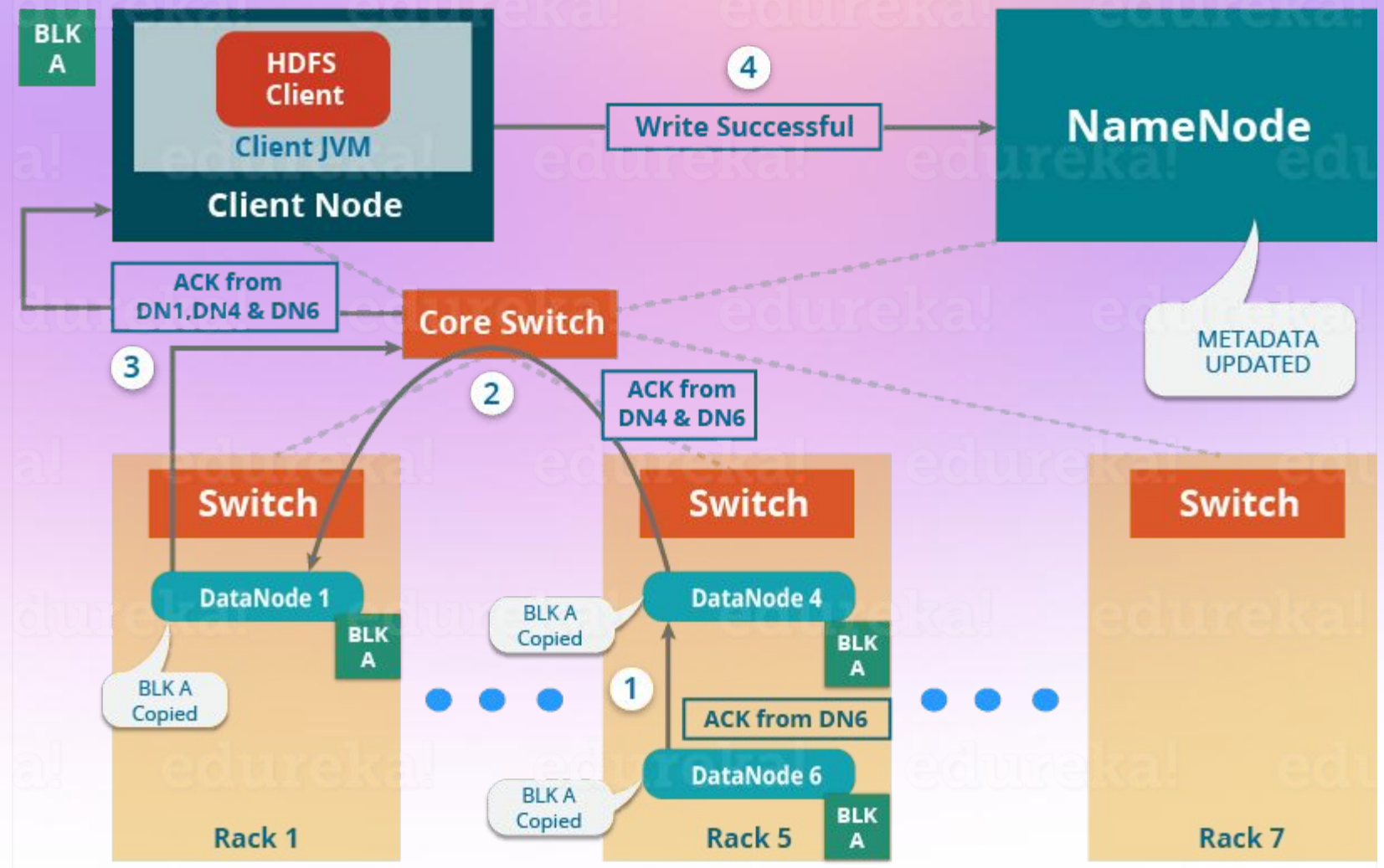




## HDFS - Write Pipeline

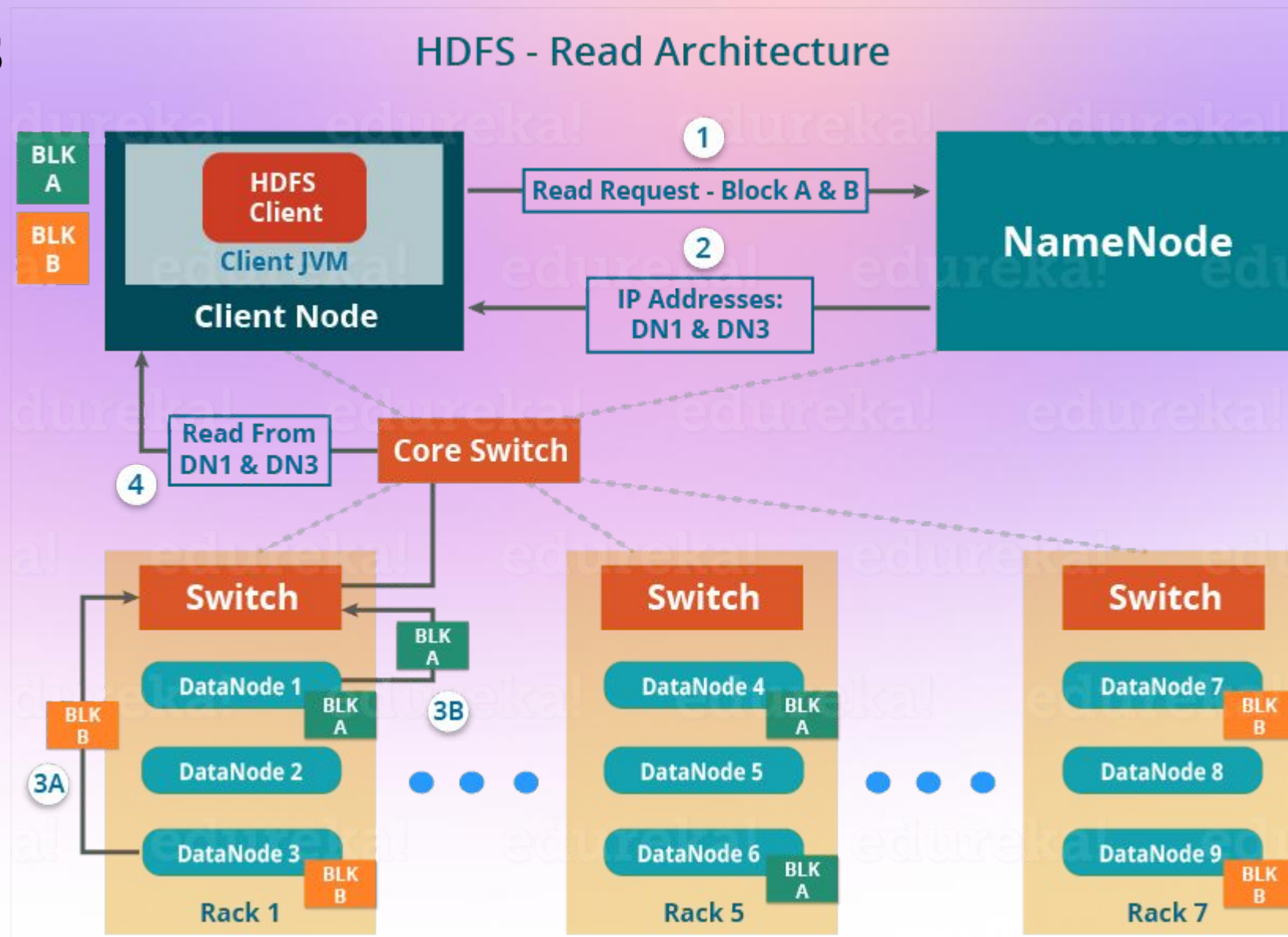


## Acknowledgement in HDFS - Write





## Чтение из HDFS



## Способы доступа к файлам на HDFS

- Доступ при помощи консольных утилит `hdfs/hadoop`
- Доступ при помощи REST через сервис WebHDFS
- Доступ при помощи примонтированного NFS — диска через HDFS NFS Gateway
- И наконец - различные API, например доступ через нативный RPC из Python через `libhdfs3`

## Разница между ver. 2 и ver. 3

- Может использоваться т.н. Erasure Coding (EC) вместо поблочной репликации
  - Меньшая избыточность (50% против 200%)
  - Некоторые операции недоступны
- Теоретически v3 должна быть медленнее, но до тестов я не добрался.
- Не видел в production еще ни разу 😊



## Что могут спросить на собеседовании

- то такое HDFS и для чего она нужна ?
- Каковы преимущества HDFS перед другими файловыми системами ?
- HDFS vs. S3

## Материалы для самостоятельного изучения

- Общая архитектура  
<https://www.edureka.co/blog/apache-hadoop-hdfs-architecture/>  
<https://www.edureka.co/blog/hadoop-yarn-tutorial/>
- О Posix-совместимости HDFS  
<https://www.quora.com/Is-HDFS-compliant-with-POSIX-Why>