Лекция 2. Распределенная файловая система HDFS

Big Data Analytics:
Approaches and Tools

ИНСТИТУТ ЦИФРОВОГО ОБРАЗОВАНИЯ МГПУ

Основные темы

- Виртуальная Java Машина (JVM)
- Назначение и отличительные особенности HDFS
- Основные компоненты HDFS
- Операции чтение/запись в HDFS
- HDFS HA
- Hadoop 3.x – HDFS

ИНСТИТУТ
ЦИФРОВОГО
ОБРАЗОВАНИЯ
МГПУ

Виртуальная Java Машина (JVM)

JVM

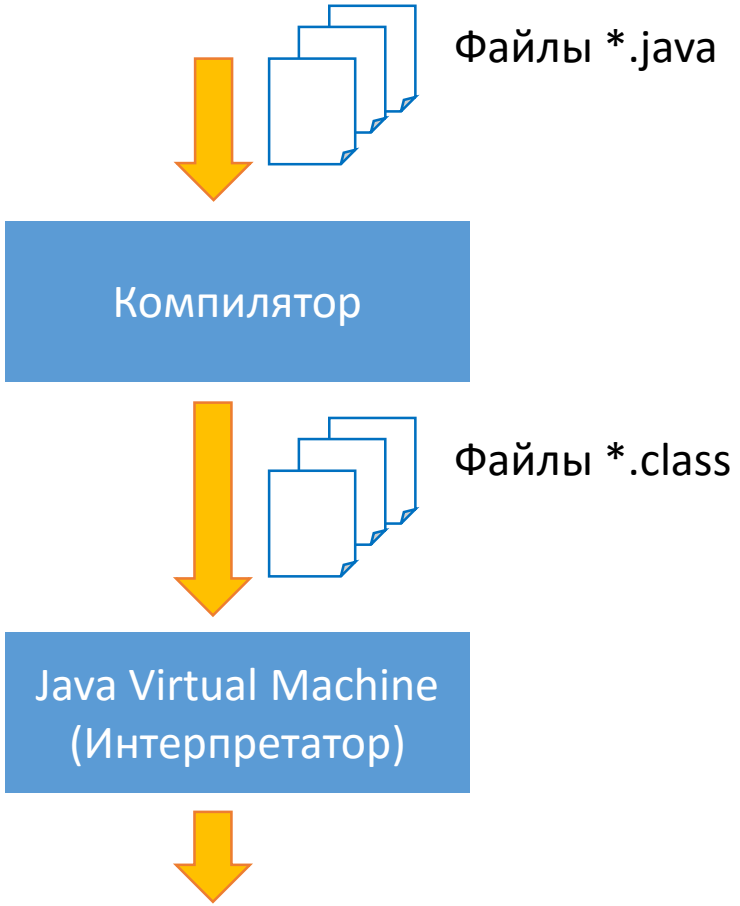
Java Virtual Machine (JVM) – спецификация для реализации выполняемой в реальном времени среды, в которой Java байт-код транслируется в машинный код

Реализация JVM – Java Runtime Environment (JRE)

Java Development Kit = JRE + дополнительные пакеты/инструменты для разработки

ИНСТИТУТ ЦИФРОВОГО ОБРАЗОВАНИЯ МГПУ

Этапы трансляции программного кода в машинный



ИНСТИТУТ ЦИФРОВОГО ОБРАЗОВАНИЯ МГПУ

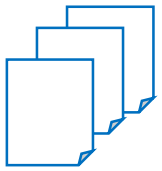
Компиляция исходного кода программы

```
void spin() {
    int i;
    for (i = 0; i < 100; i++) {
        ;    // Loop body is empty
    }
}
```

Компилятор

```
0   iconst_0      // Push int constant 0
1   istore_1      // Store into local variable 1 (i=0)
2   goto 8        // First time through don't increment
5   iinc 1 1      // Increment local variable 1 by 1 (i++)
8   iload_1       // Push local variable 1 (i)
9   bipush 100    // Push int constant 100
11  if_icmplt 5   // Compare and loop if less than (i < 100)
14  return        // Return void when done
```

Структура кучи (heap)



Файлы *.class

Загрузчик (Loader)

Структура памяти (Run-Time Data Areas)

Выполнение (Execution Engine)

Java Virtual Machine

Java Virtual Machine

Процесс ОС

Структуры памяти в JVM

Файлы *.class

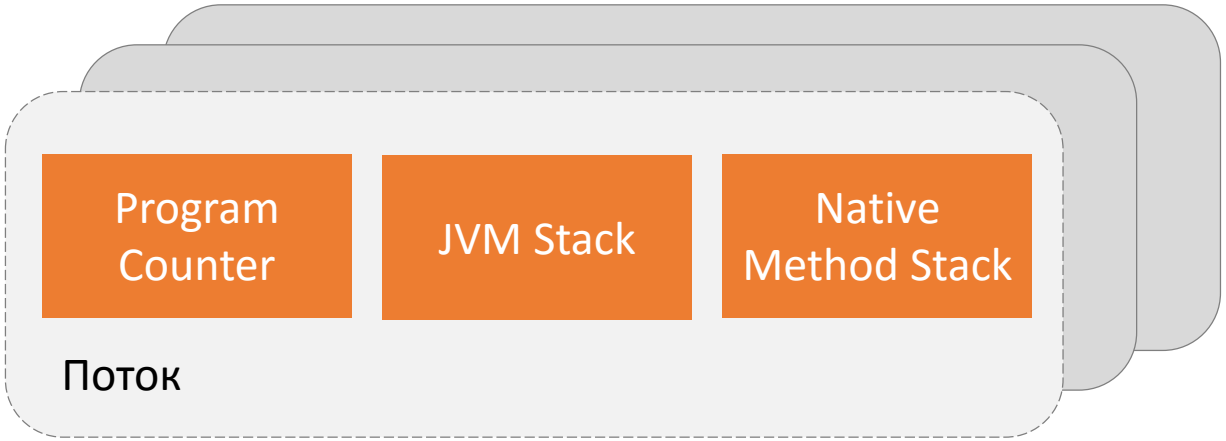
Загрузчик (Loader)

Heap | JVM Stack | Native Method Stack | Program Counter | Method Area

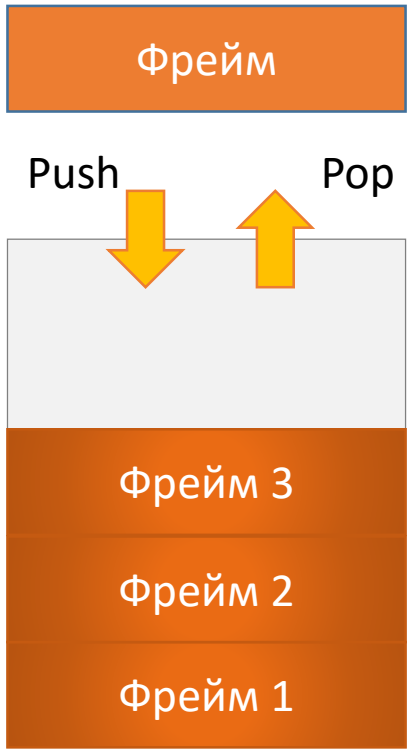
Структуры памяти (Run-Time Data Areas)

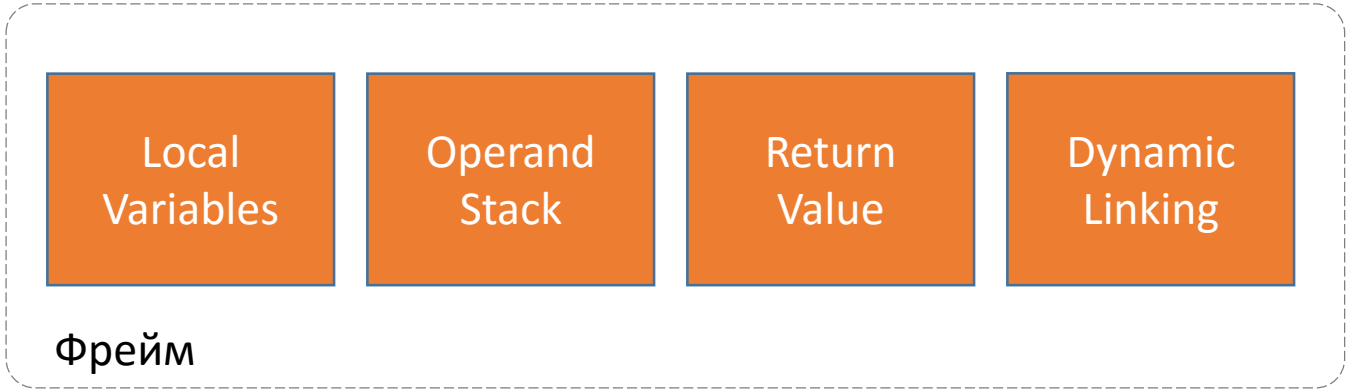
Выполнение (Execution Engine)

Потоки в JVM



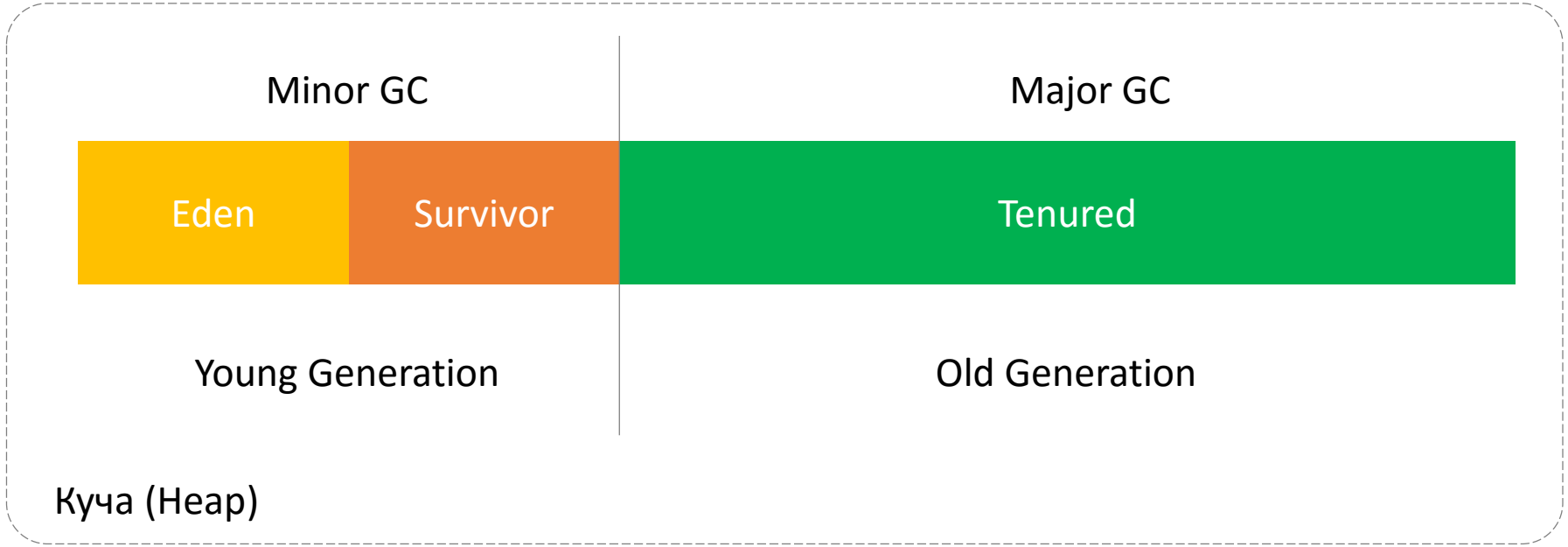
Структура кучи (heap)





ИНСТИТУТ ЦИФРОВОГО ОБРАЗОВАНИЯ МГПУ

Поколения и куча



ИНСТИТУТ ЦИФРОВОГО ОБРАЗОВАНИЯ МГПУ

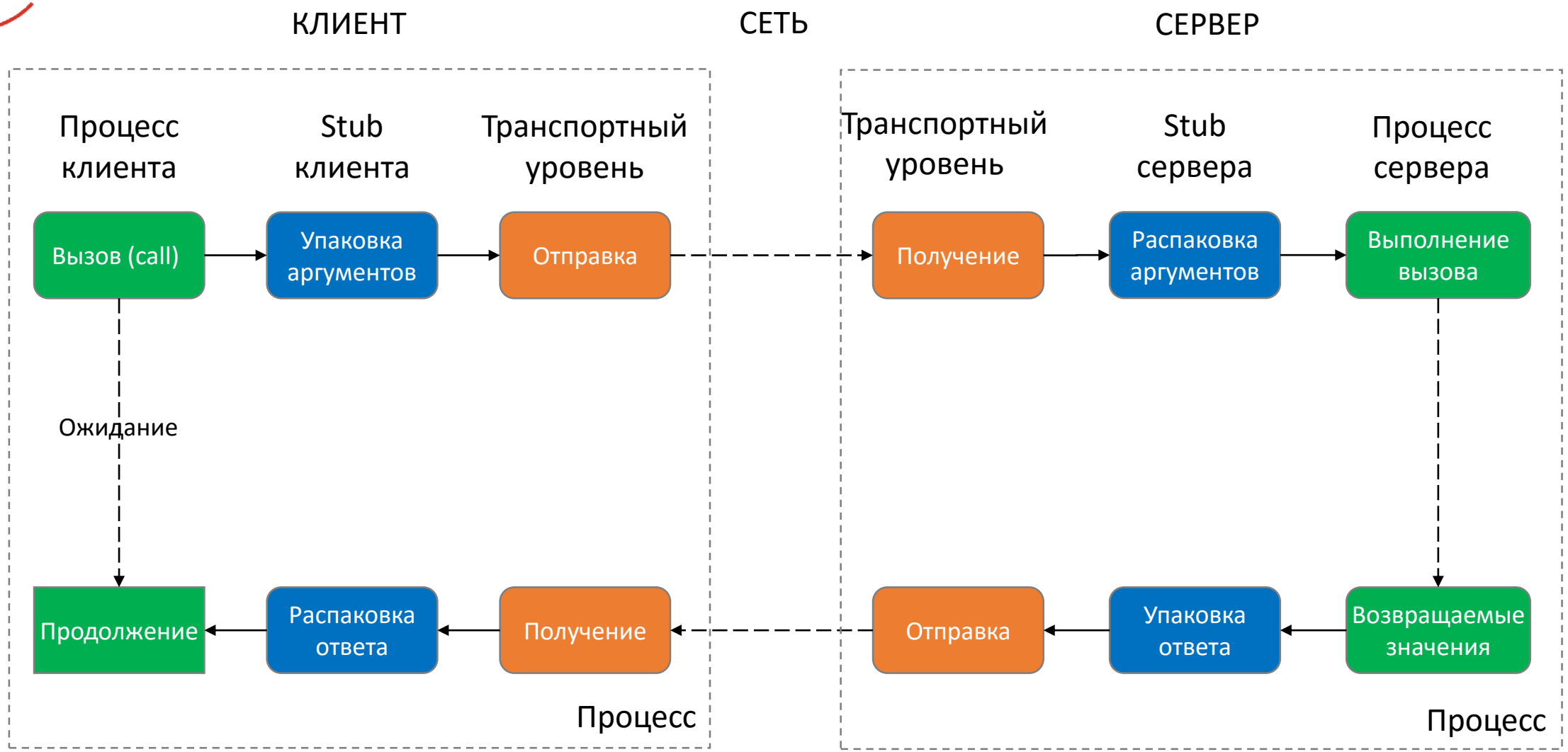
Важные параметры JVM

- Объем heap и non-heap памяти
- Количество потоков
- Количество классов, загружаемых в память
- Параметры GC

ИНСТИТУТ
ЦИФРОВОГО
ОБРАЗОВАНИЯ
МГПУ

Удаленный вызов процедур
RPC

Remote Procedure Call (RPC)



http://www.pling.org.uk/cs/nds.html

ИНСТИТУТ
ЦИФРОВОГО
ОБРАЗОВАНИЯ
МГПУ

Назначение и отличительные особенности Hadoop

ИНСТИТУТ ЦИФРОВОГО ОБРАЗОВАНИЯ МГПУ

Особенности Hadoop

- Масштабируемость
- Отказоустойчивость
- Доступность
- Вычисления приближены к данными
- Высокая пропускная способность I/O

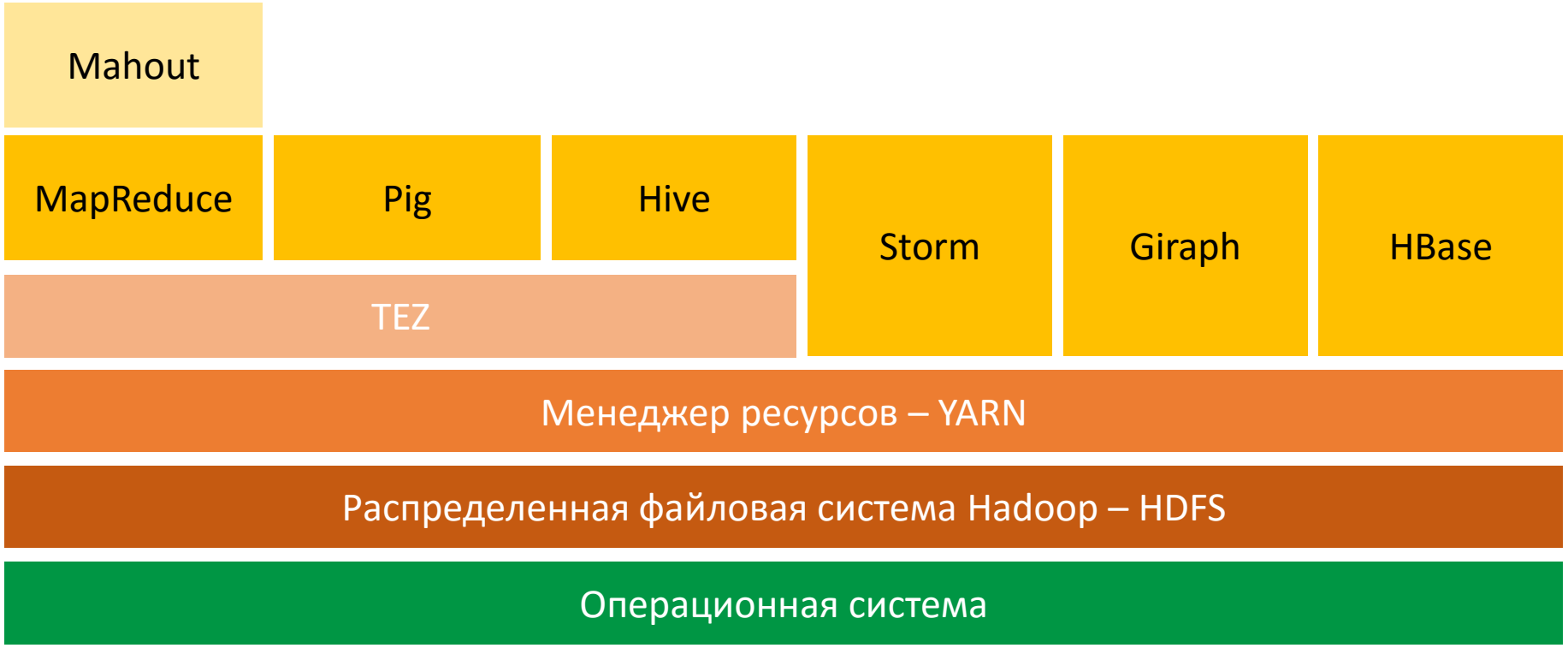
ИНСТИТУТ ЦИФРОВОГО ОБРАЗОВАНИЯ МГПУ

Сравнение с Hadoop

	Реляционная СУБД	**Hadoop**
Размещение данных	Централизованное	Распределены по вычислительным узлам
Размер данных	Гигабайты	Петабайты
Доступ	Интерактивный и batch	Batch
Изменения	Чтение и множественная запись	Запись один раз, чтение много раз
Транзакции	ACID	нет
Структура	Схема на запись	Схема на чтение
Целостность	Высокая	Низкая
Масштабируемость	Нелинейная	Линейная

ИНСТИТУТ ЦИФРОВОГО ОБРАЗОВАНИЯ МГПУ

Стек Hadoop



ИНСТИТУТ
ЦИФРОВОГО
ОБРАЗОВАНИЯ
МГПУ

Распределенная файловая система Hadoop (HDFS)

HDFS

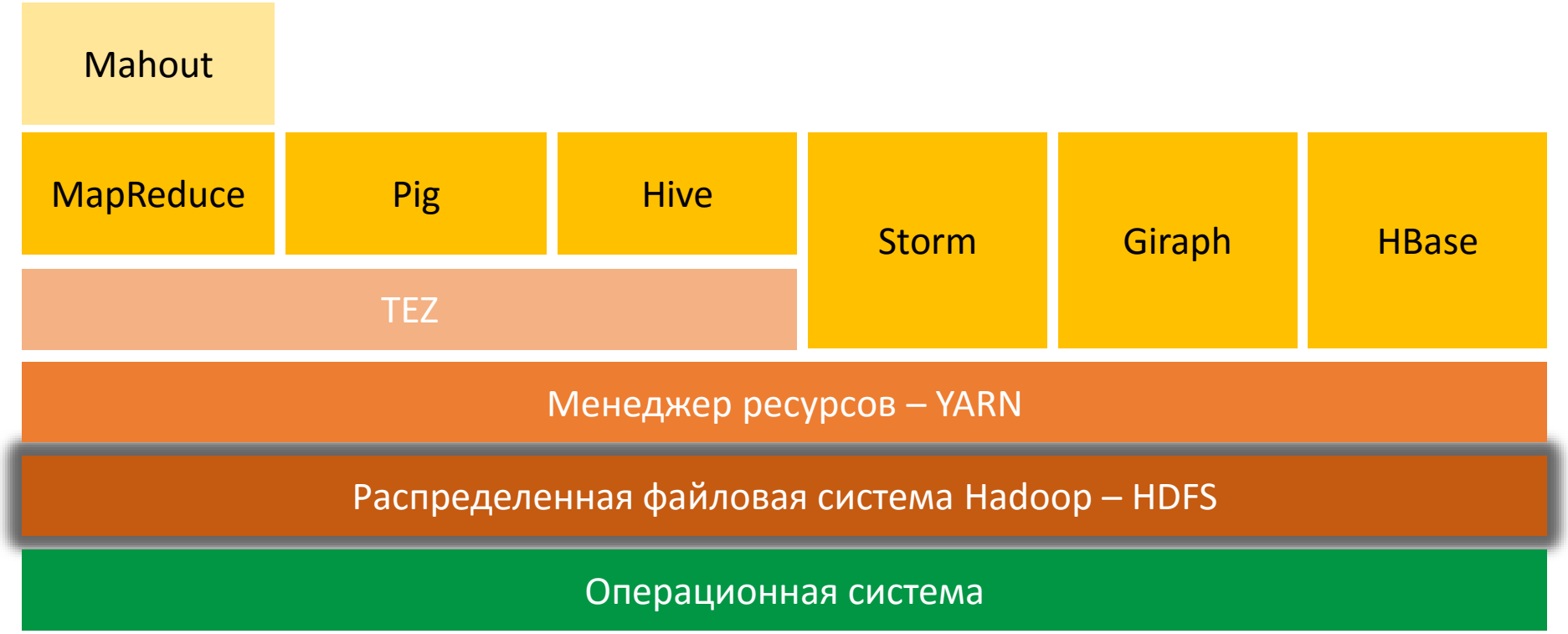
Распределенная файловая система Hadoop (Hadoop Distributed File System – **HDFS**) – файловая система, разработанная для хранения больших массивов данных и запускаемая на кластере с серверами общего назначения (commodity hardware)

- \+ Большие файлы
- \+ Потоковый доступ к данным
- \+ Серверы общего назначения

- – Низкая задержка доступа к данным
- – Множество маленьких файлов
- – Произвольный доступ и изменение данных

ИНСТИТУТ ЦИФРОВОГО ОБРАЗОВАНИЯ МГПУ

Стек Hadoop



ИНСТИТУТ
ЦИФРОВОГО
ОБРАЗОВАНИЯ
МГПУ

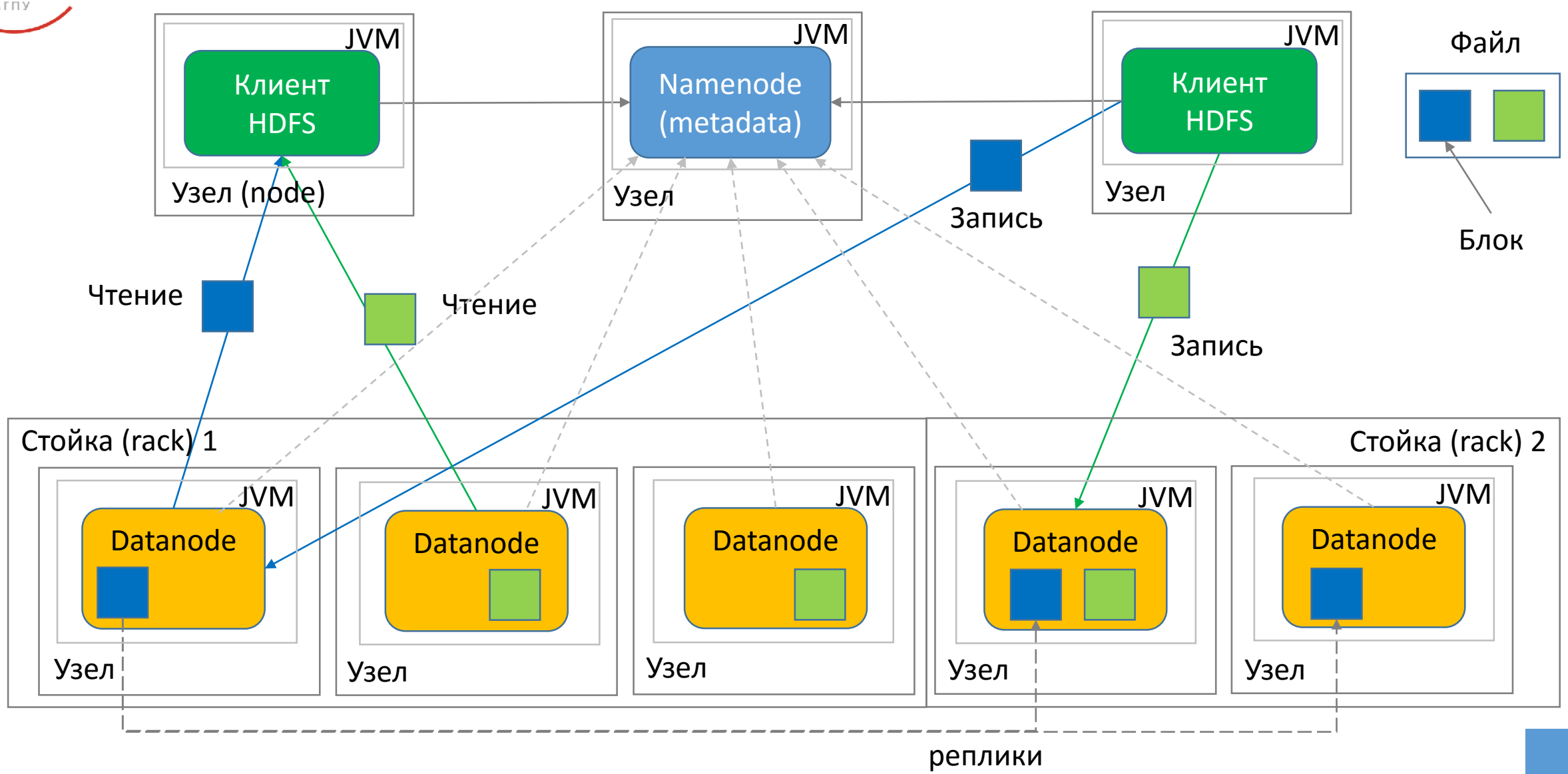
Основные компоненты HDFS

ИНСТИТУТ ЦИФРОВОГО ОБРАЗОВАНИЯ МГПУ

Словарь HDFS

- Данные
- Блок данных
- Клиент
- NameNode
- Secondary NameNode
- DataNode
- Реплика
- JournalNode
- Federation

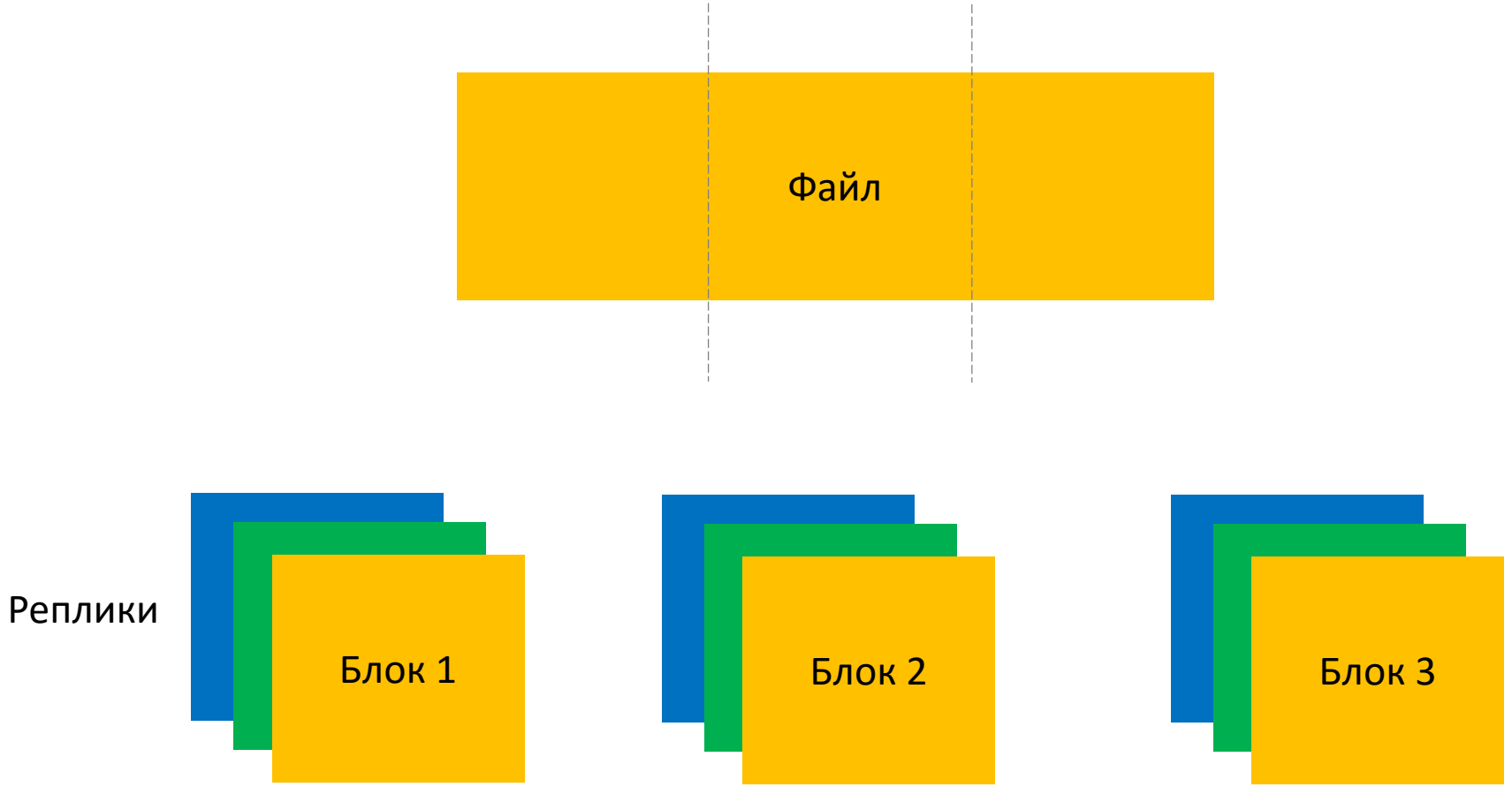
Архитектура HDFS



Блок данных

64/128/256МБ

Блоки данных – объем данных для чтения/записи



Блок данных

Зачем нужны блоки?

Если слишком маленький/большой размер блока?

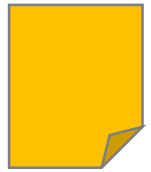
Namenode

Namenode управляет пространством файловой системы и поддерживает

- дерево файловой системы
- метаданные о всех файлах и папках
- списки: файл -> блоки -> datanodes

Namespace Image

Fsimage



Edit Log

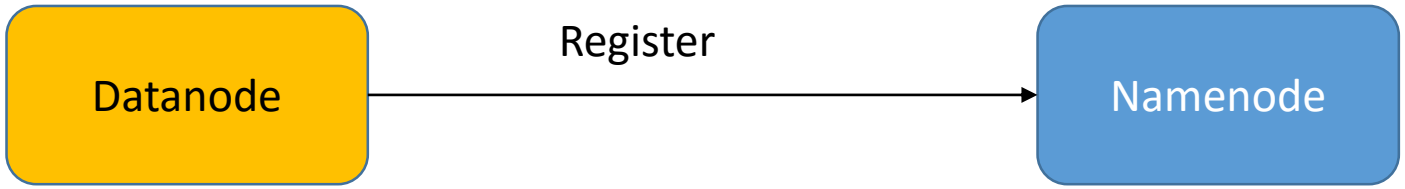
Datanode

Datanode – компонент HDFS для работы с блоками данных

- хранит и извлекает блоки по запросу
- передает namenode информацию о своих блоках
- периодически посылает namenode heartbeats

Кластер может иметь тысячи Datanodes и десятки тысяч клиентов.

Каждый Datanode может выполнять множество тасков приложений параллельно



Передаваемые сообщения. Block report и heartbeat

- **Heartbeats** от DN к NN (~ секунды – 3) передают следующую информацию:
  - Доступный объем дискового пространства
  - Сколько используется для хранения
  - Количество передаваемых данных в текущий момент

- **Block report** от DN к NN (~ часы – 1) содержит информацию о хранящихся репликах блоков данных:
  - Id блока,
  - Метка поколения
  - Размер каждого блока

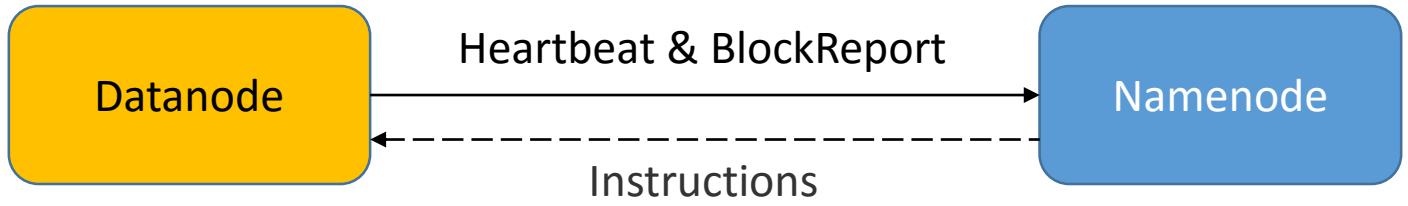
NN может обрабатывать тысячи hearbeat´ов в секунду без воздействия на другие операции NN

Управление DataNode через NameNode

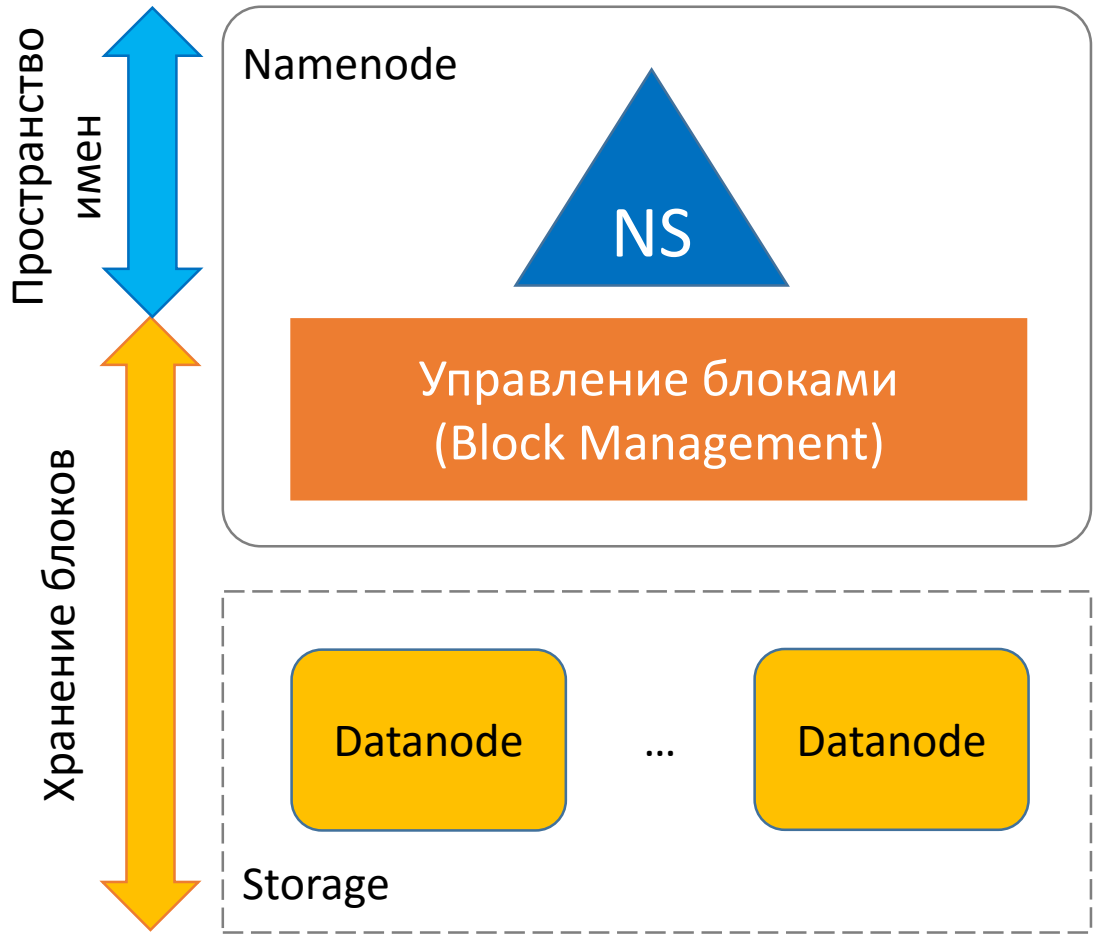
NameNode использует ответ на heartbeat’ы, чтобы отправлять **инструкции** на Datanode’ы.

Инструкции включают **команды** на:

- Репликацию блоков на другие узлы
- Удаление реплик;
- Повторную регистрацию и завершение работы Datanode;
- Отправку block report



Block report и heartbeat



Пространство имен (Namespace)

- директории, файлы и блоки
- создание, удаление, изменение, вывод список файлов и директорий.

Службы хранения блоков

Управление блоками

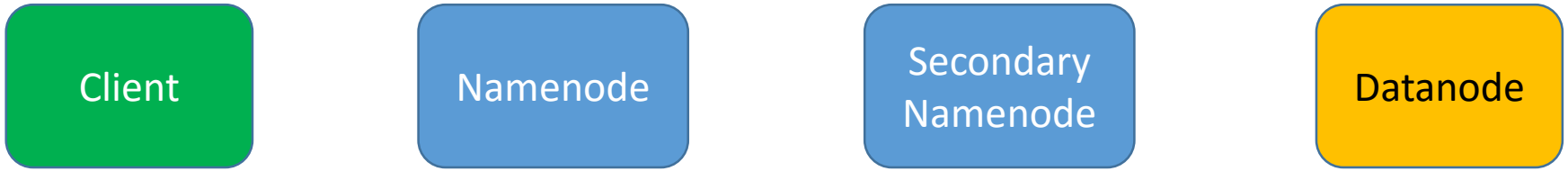
- Регистрация DN и обработка heartbeat’ов
- block reports и обработка информации о расположении блоков
- Операции над блоками: создание, удаление, изменение и получение расположения блока
- Управление репликами

Хранение (Storage)

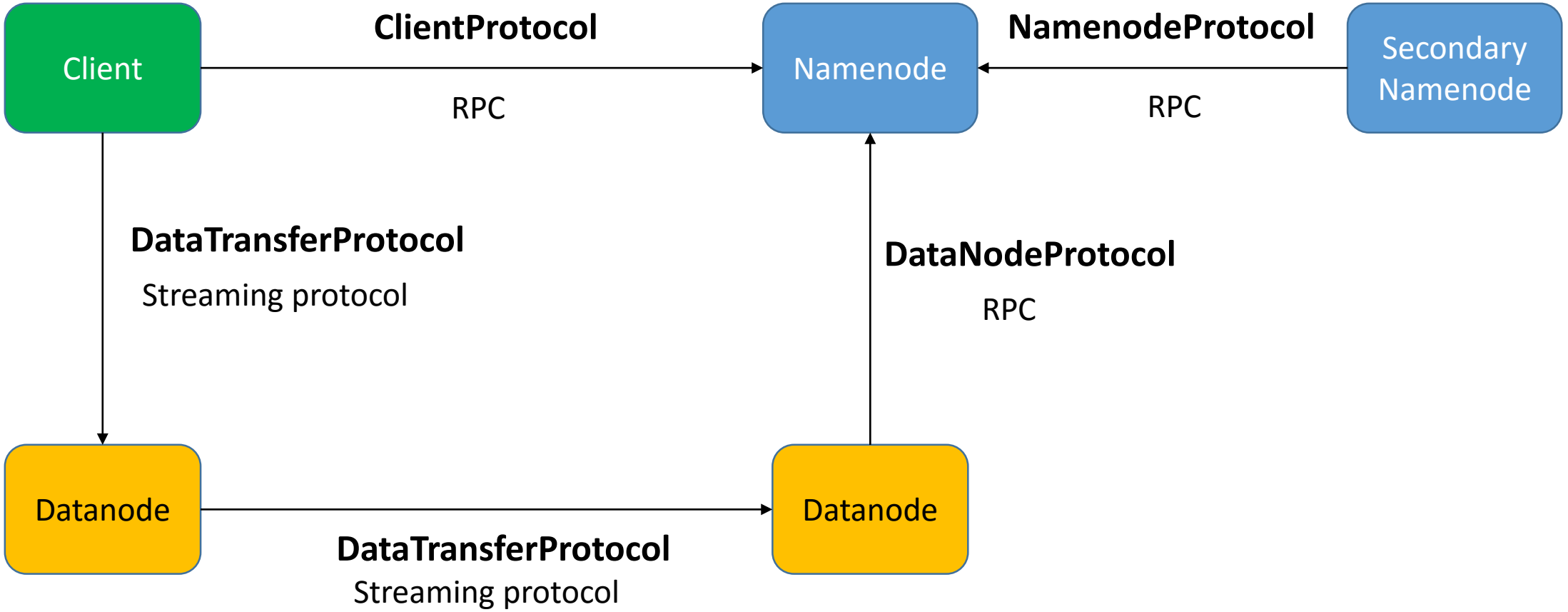
- Хранение блоков в локальной файловой системе и обеспечение доступа на чтение/запись

Сервисы/демоны в HDFS

Daemon	Environment Variable
NameNode	HADOOP_NAMENODE_OPTS
DataNode	HADOOP_DATANODE_OPTS
Secondary NameNode	HADOOP_SECONDARYNAMENODE_OPTS



Протоколы в HDFS



ИНСТИТУТ
ЦИФРОВОГО
ОБРАЗОВАНИЯ
МГПУ

Основные команды в HDFS

ИНСТИТУТ ЦИФРОВОГО ОБРАЗОВАНИЯ МГПУ

“hadoop fs” vs “hdfs dfs”

hadoop fs

- Hadoop Distributed File System (HDFS)
- Local FS,
- HFTP FS,
- S3 FS, and others

ИНСТИТУТ ЦИФРОВОГО ОБРАЗОВАНИЯ МГПУ

Манипуляции с данными

hadoop fs -mkdir [-p] <paths>

hadoop fs -copyFromLocal <localsrc> URI

hadoop fs -ls [-d] [-h] [-R] <args>

hadoop fs -copyToLocal [-ignorecrc] [-crc] URI <localdst>

hadoop fs -cp [-f] [-p | -p[topax]] URI [URI ...] <dest>

hadoop fs -df [-h] URI [URI ...]

hadoop fs -du [-s] [-h] URI [URI ...]

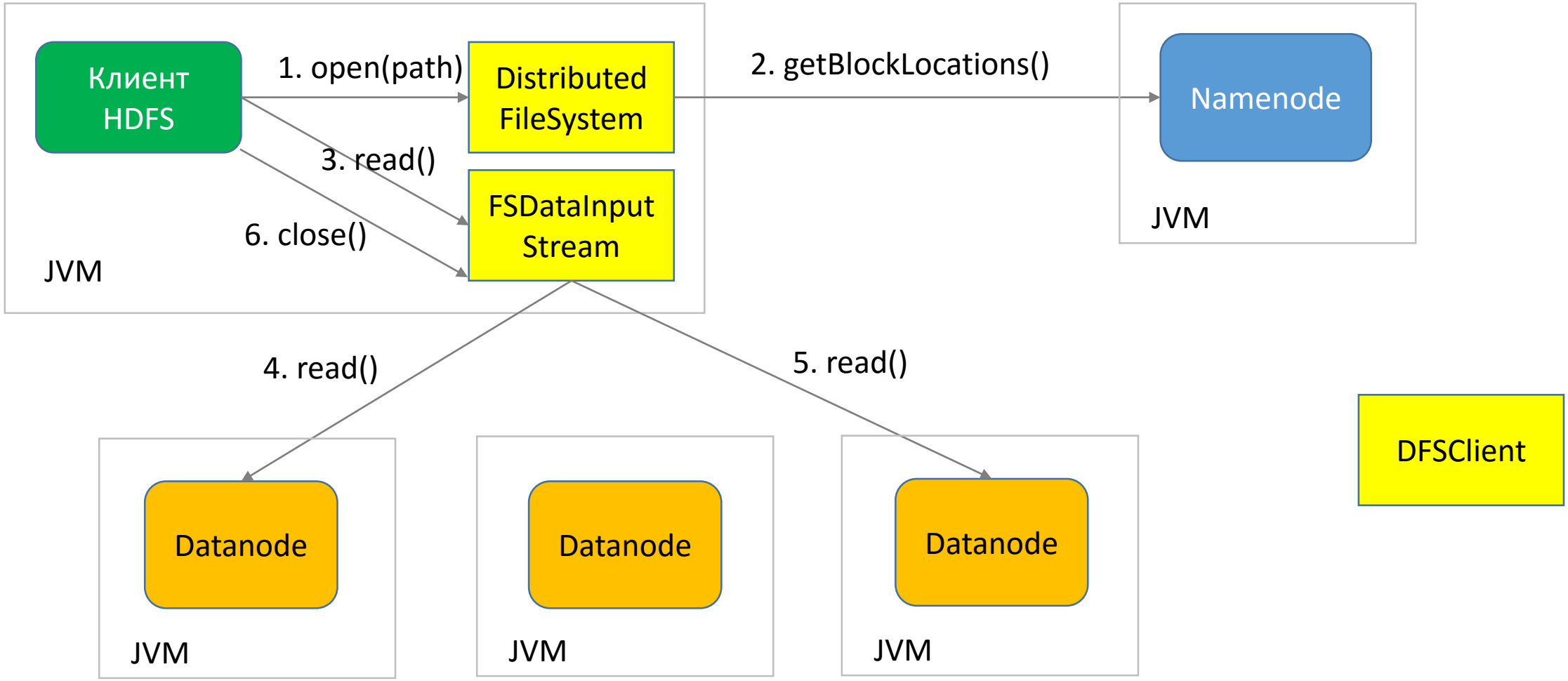
https://hadoop.apache.org/docs/r2.7.2/hadoop-project-dist/hadoop-common/FileSystemShell.html

https://hadoop.apache.org/docs/r2.7.0/hadoop-project-dist/hadoop-hdfs/HDFSCommands.html

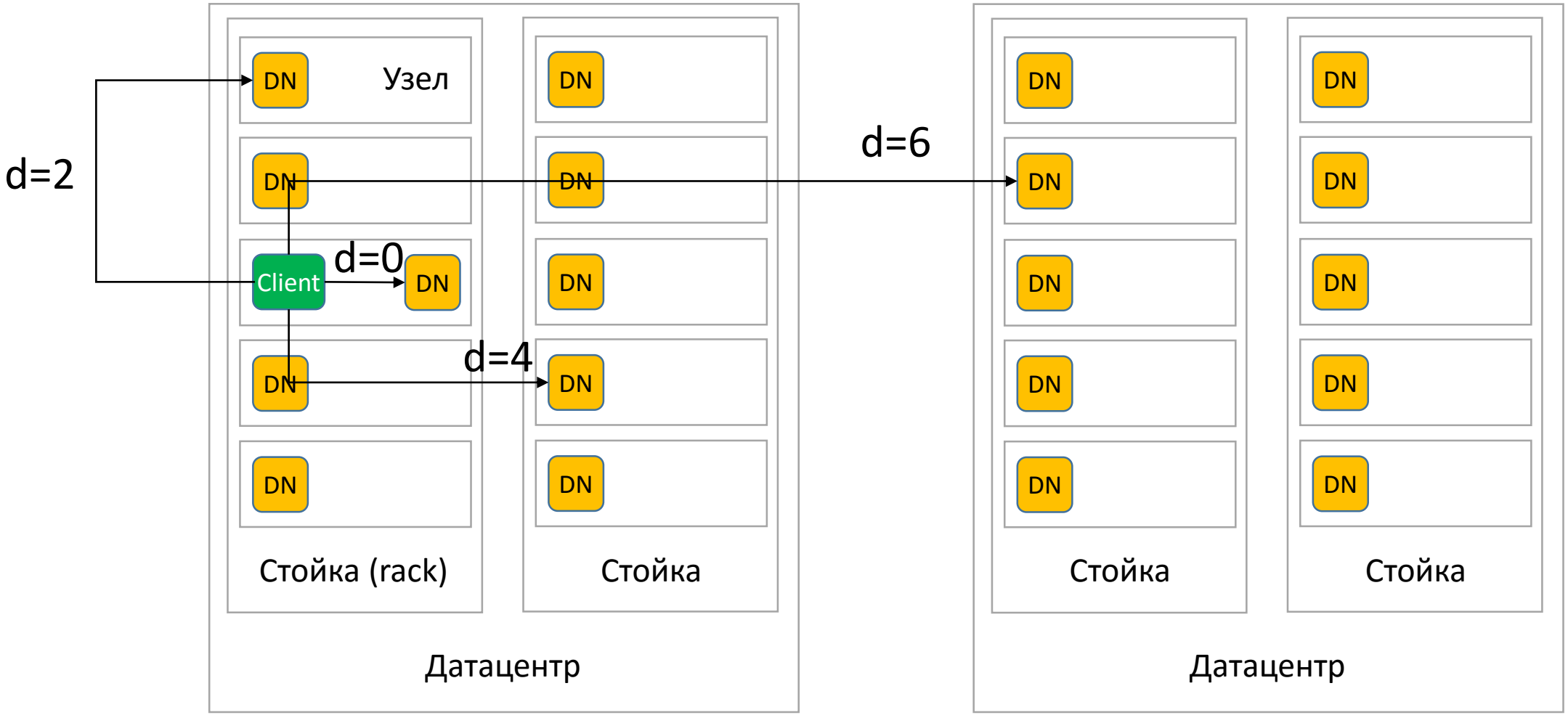
ИНСТИТУТ
ЦИФРОВОГО
ОБРАЗОВАНИЯ
МГПУ

Операции чтение/запись в HDFS

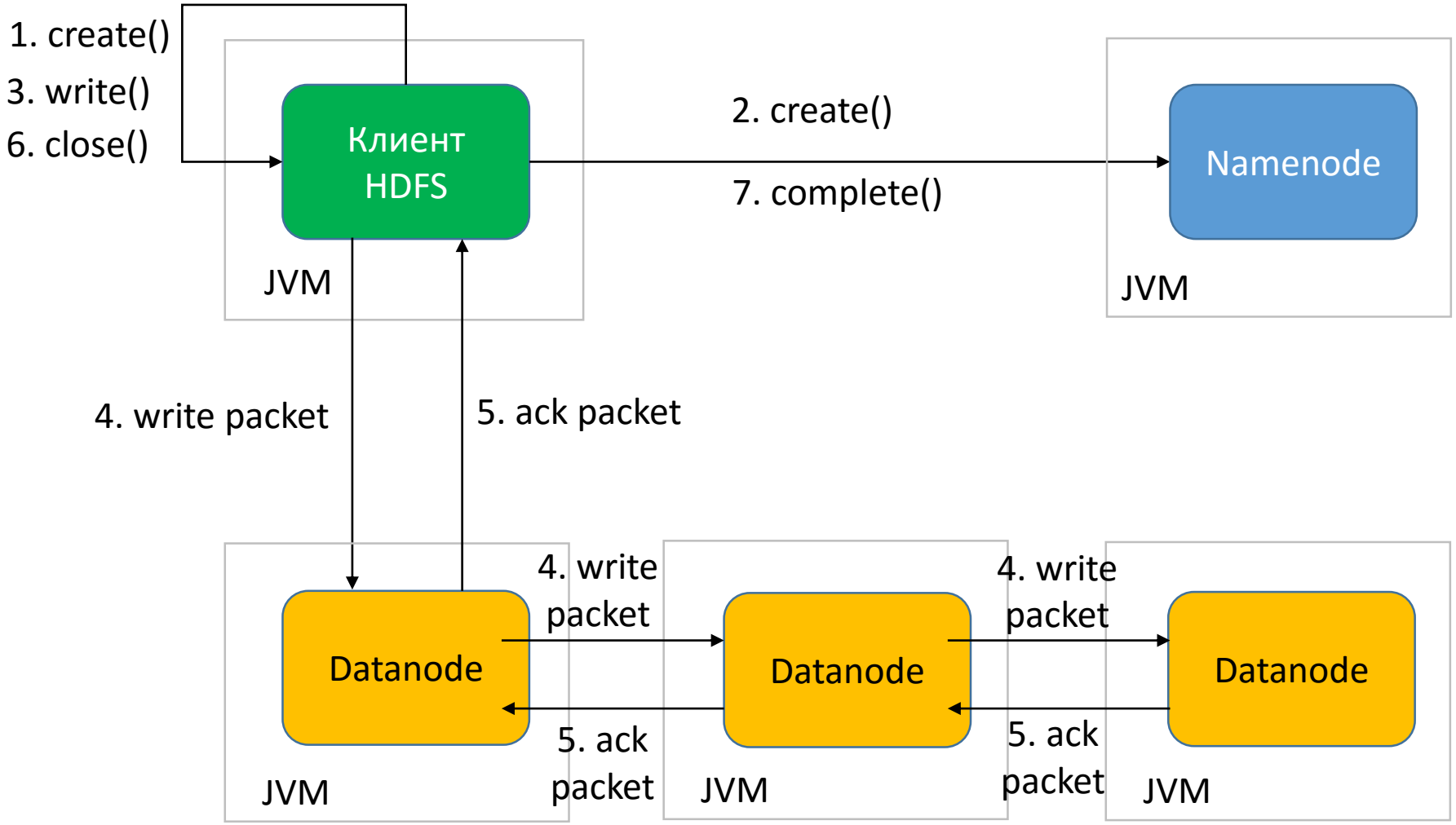
Чтение данных



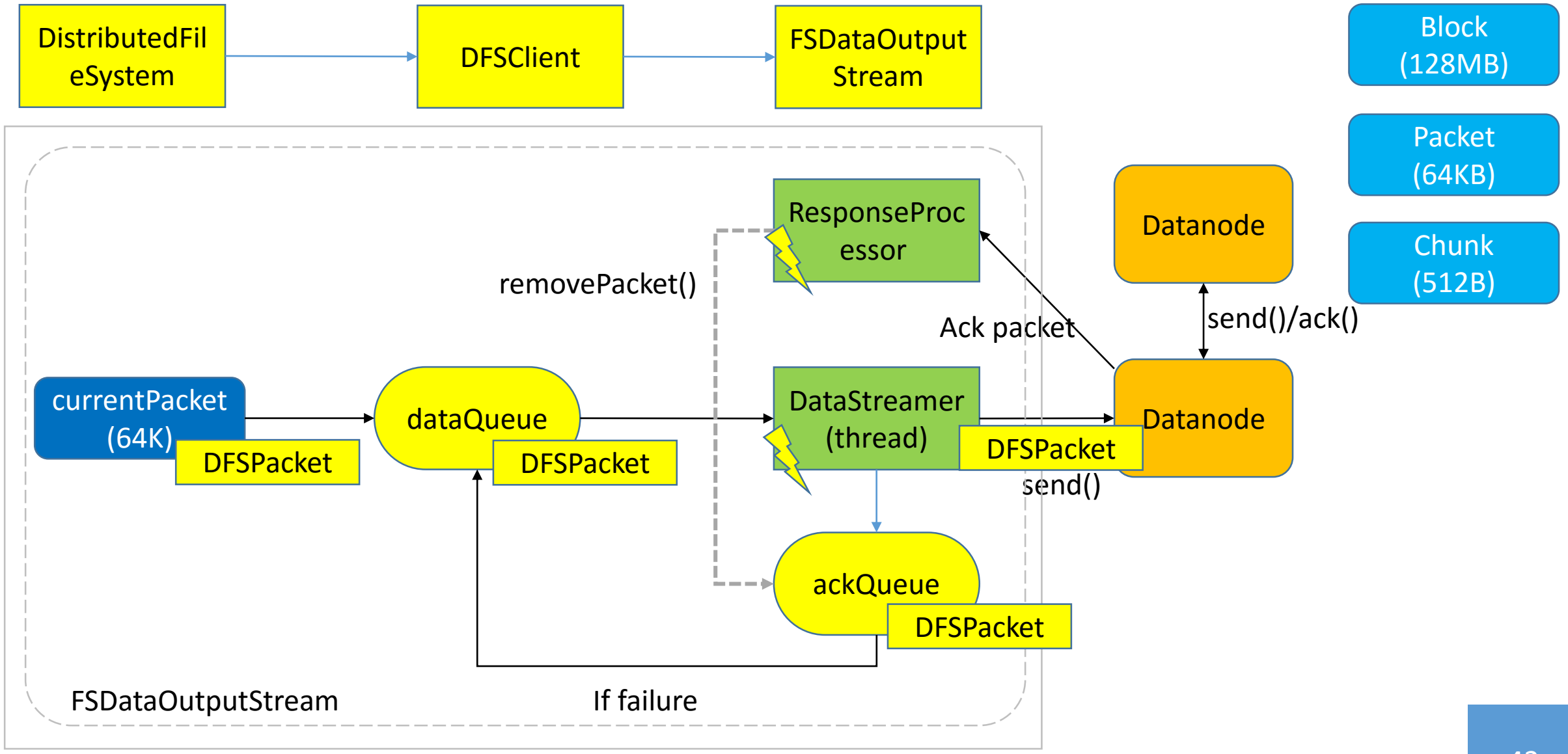
Определение расстояния



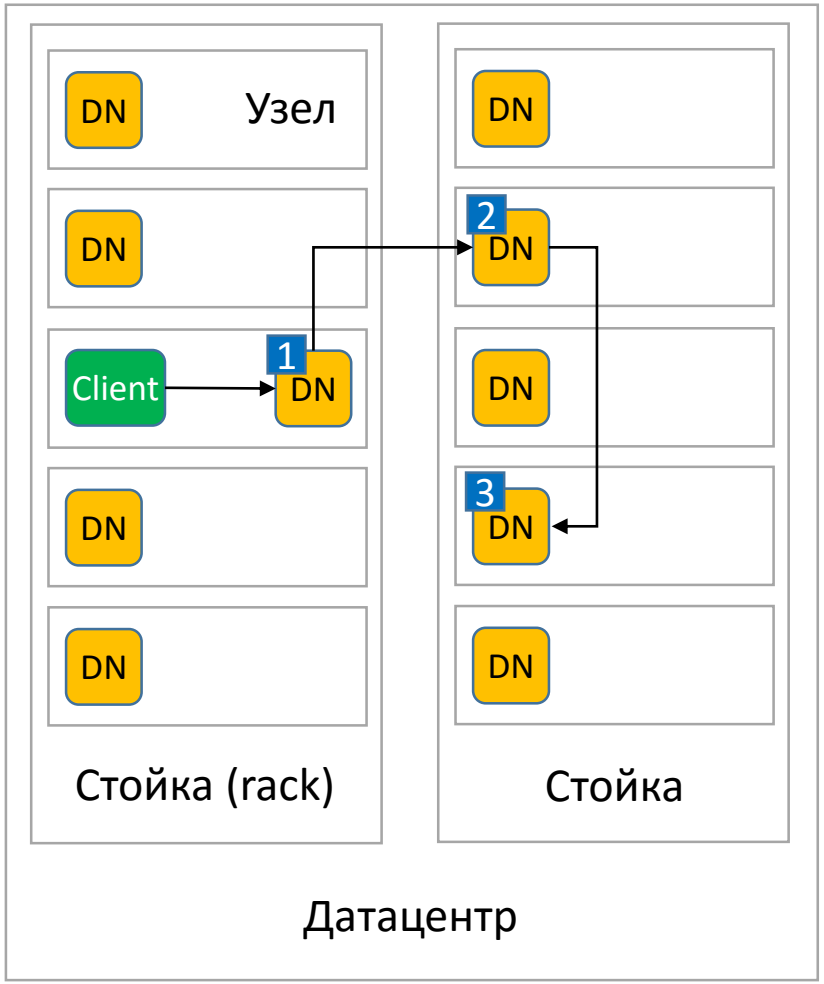
Запись данных



Запись данных. Write(), writePacket(), ackPacket()



Расположение реплик



Namenode vs Datanode

Что теряем, если выйдет из строя Datanode/Namenode?

ИНСТИТУТ ЦИФРОВОГО ОБРАЗОВАНИЯ МГПУ

Дополнительные возможности

- Secondary NameNode
- High Availability
- Federation

Secondary Namenode

Namenode хранит образ всего пространства имен файловой системы и распределение блоков по файлам в памяти

- Fsimage: checkpoint пространства имен файловой системы
- Edit logs: содержит изменения пространства имен

Checkpoint – процесс слияния старого представления fsimage в памяти с edits и запись на диск нового fsimage

Secondary Namenode загружает fsimage и edits из **Namenode**, создает новый fsimage и загружает его обратно в **Namenode**

По умолчанию интервал 3600 сек., размер edits для старта 64MB

HDFS High Availability

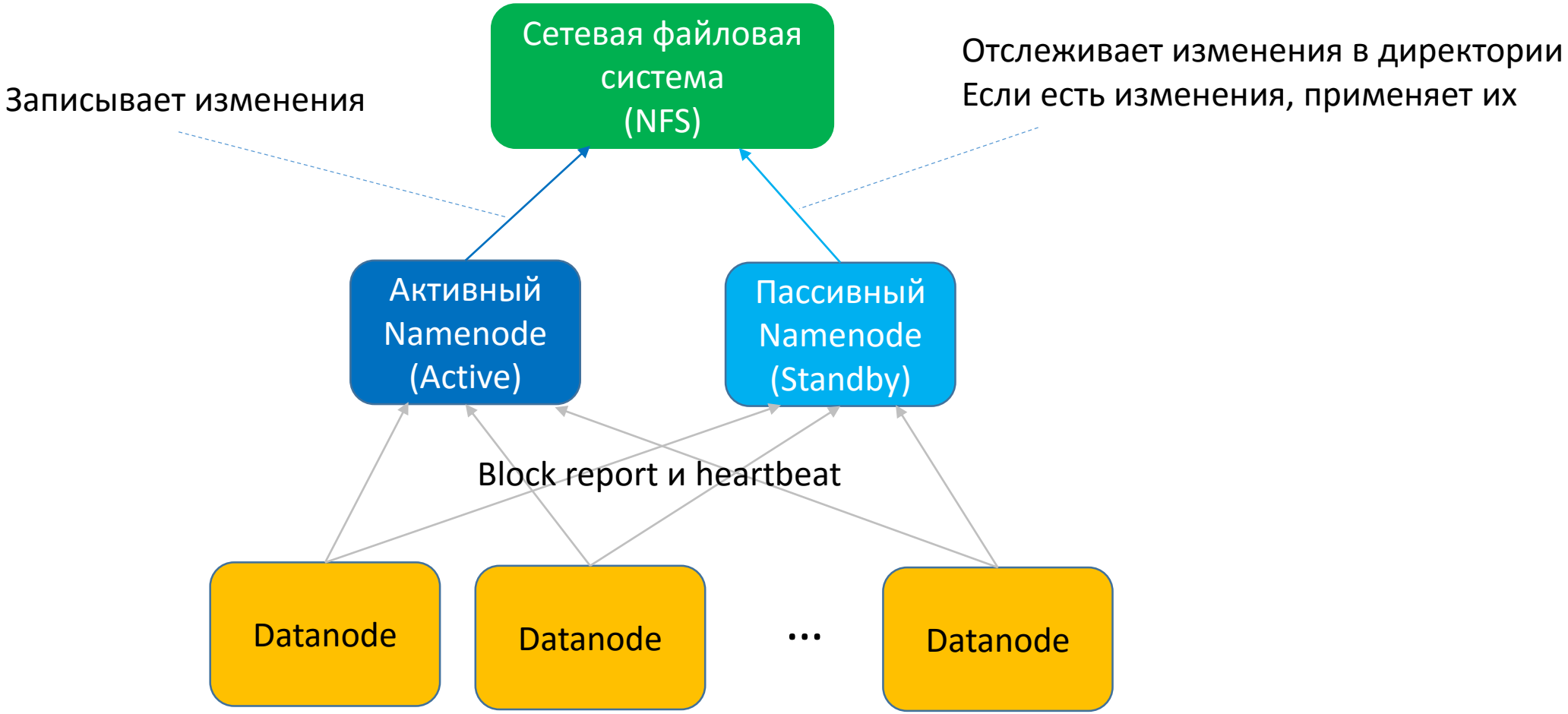
HDFS High Availability – механизм HDFS поддерживающий работу нескольких Namenode'ов (2 и более) в конфигурации **активный/пассивные**.

- Позволяет быстро восстановить данные на **пассивном Namenode**.
- **Используется** при 1) выходе из строя активного Namenode, 2) при плановой поддержке (например, при обновлении ПО на узле, где работает активный Namenode)
- **Активный Namenode** отвечает за все операции клиента в кластере
- Datanode'ы отправляют block report и heartbeat'ы всем Namenode'ам
- Пассивные Namenode также выполняют checkpoint, поэтому нет необходимости в Secondary Namenode, CheckpointNode или BackupNode

HDFS High Availability. Виды

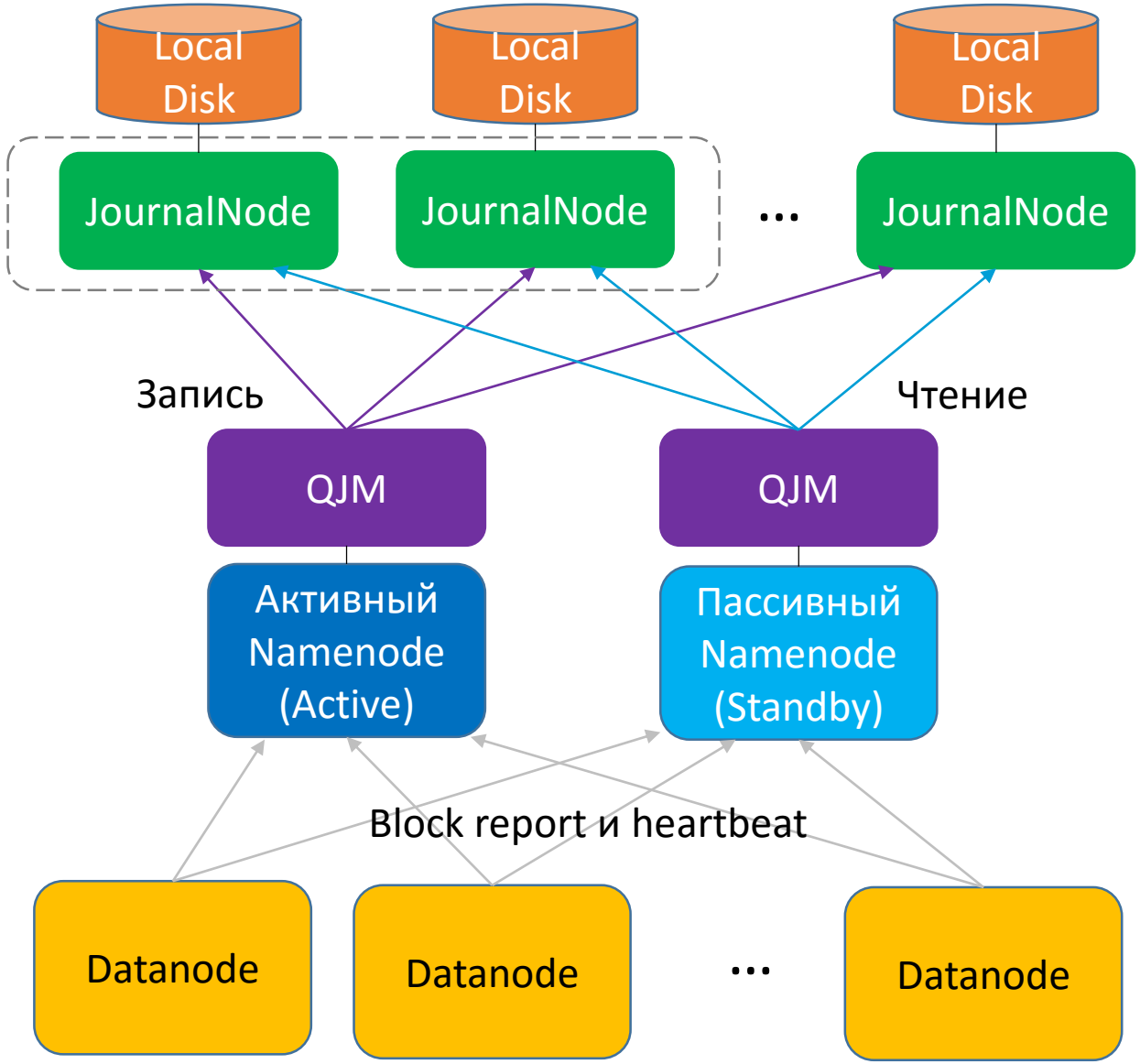
- Общая сетевая файловая система (NFS)
- Quorum Journal Manager (QJM)

HDFS HA. NFS. Синхронизация Active NN и Standby NN



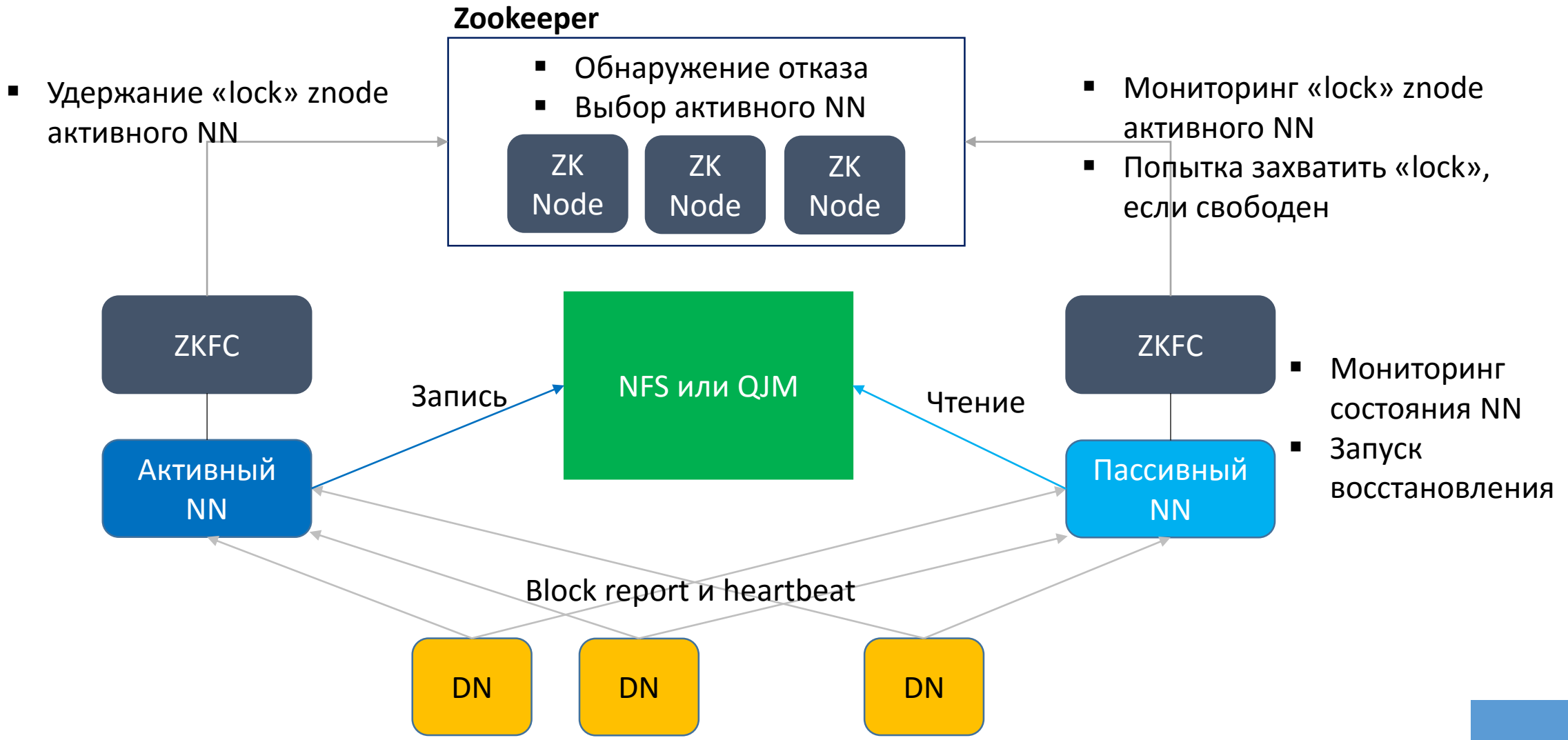
HDFS HA. QJM. Синхронизация Active NN и Standby NN

Кворум – большинство (majority) JN



HDFS HA. Автоматический запуск восстановления

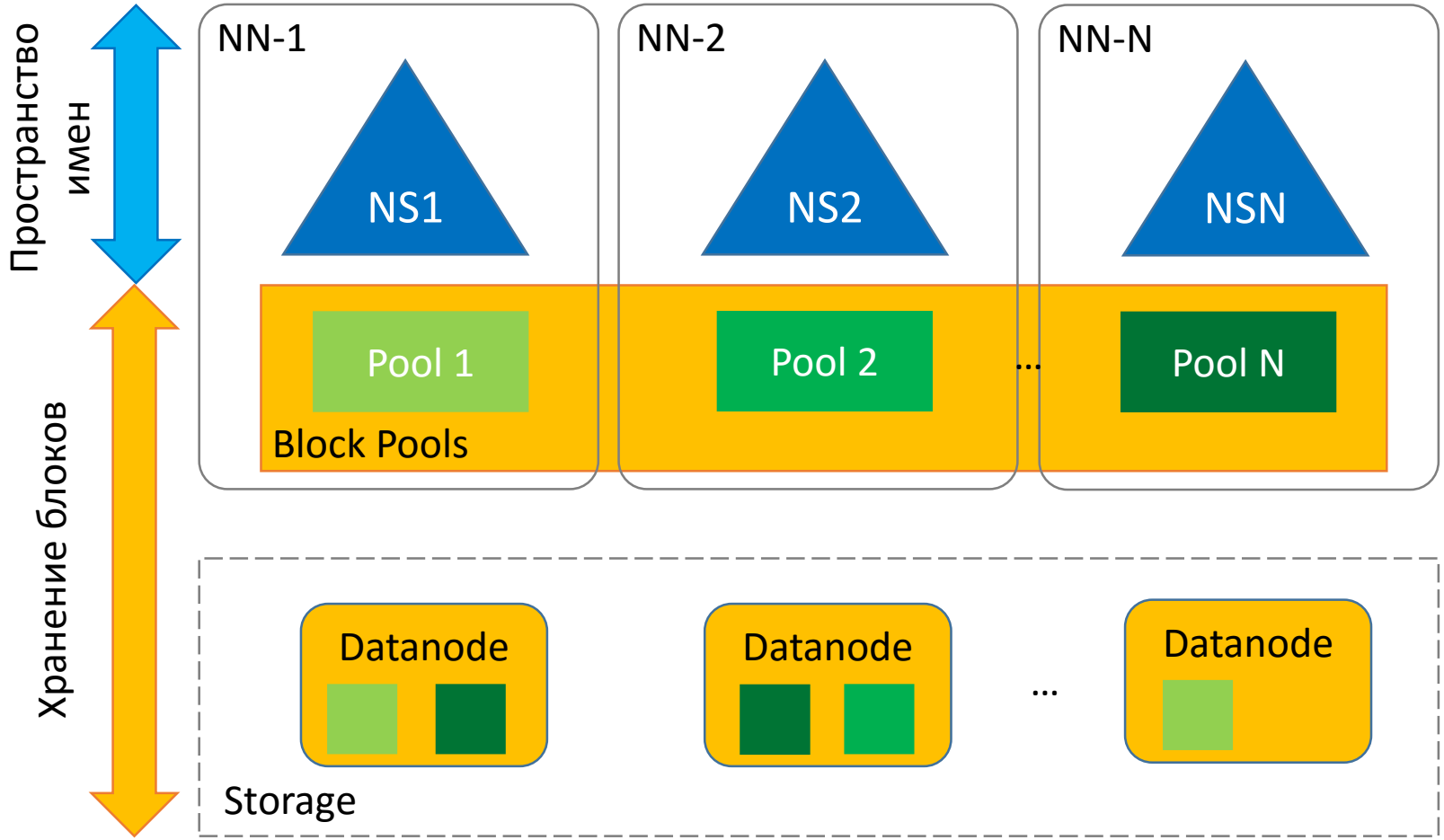
По умолчанию механизм восстановления запускается вручную



Federation

Block Pool – множество блоков, которые принадлежат одному пространству имен

Пространство имен и его **block pool** вместе называются **Namespace Volume**



ИНСТИТУТ ЦИФРОВОГО ОБРАЗОВАНИЯ МГПУ

Преимущества Federation

- Масштабируемость пространства имен
- Производительность (увеличивается пропускная способность)
- Изолированность

ИНСТИТУТ
ЦИФРОВОГО
ОБРАЗОВАНИЯ
МГПУ

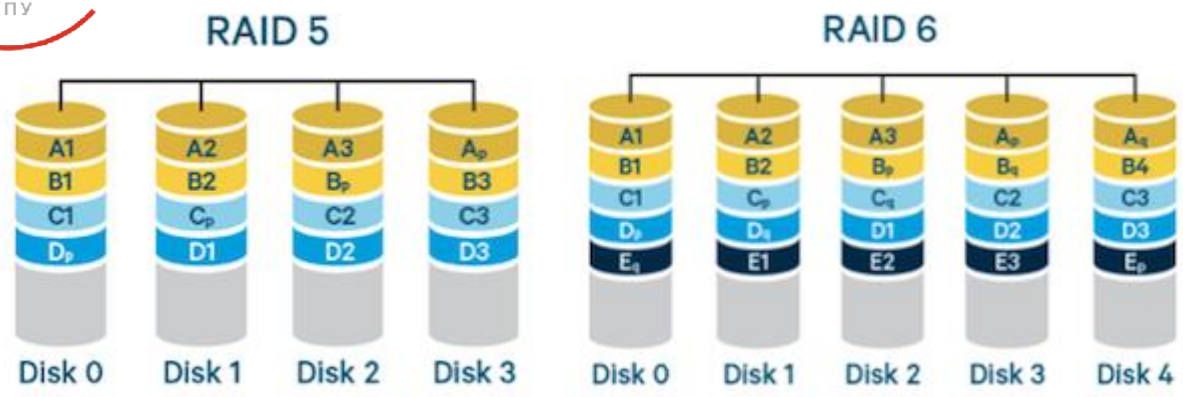
Hadoop 3.x - HDFS

ИНСТИТУТ ЦИФРОВОГО ОБРАЗОВАНИЯ МГПУ

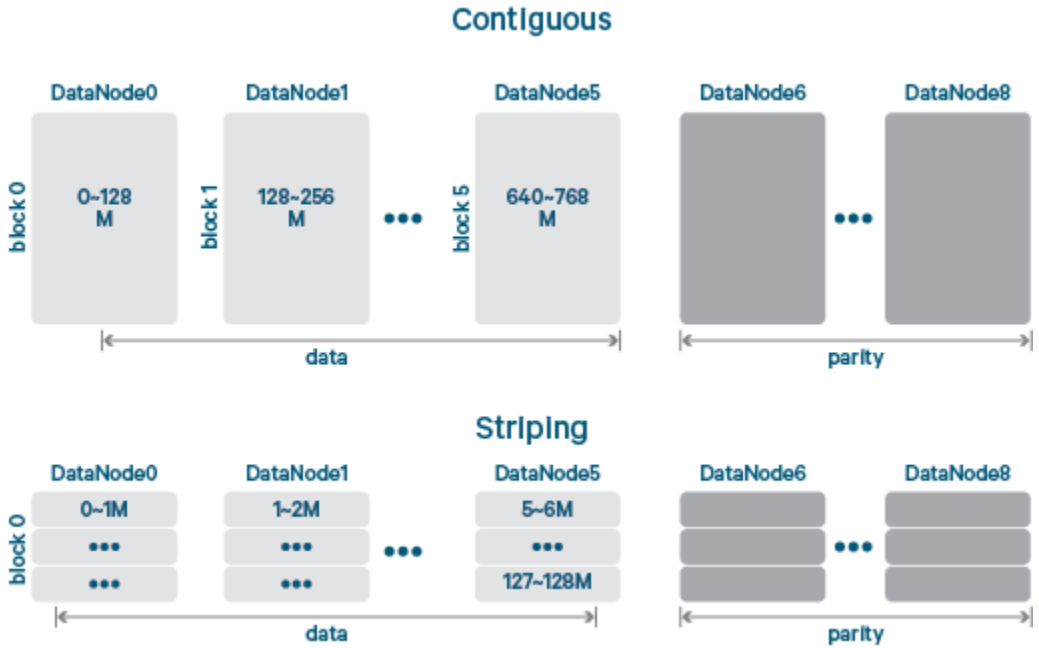
Изменения

- Intra-DataNode Balancer
- Erasure Encoding

Erasure Encoding



	Data Durability	Storage Efficiency
Single replica	0	100%
Three-way replication	2	33%
XOR with six data cells	1	86%
RS(6,3)	3	67%
RS(10,4)	4	71%



typically 64KB or 1MB

ИНСТИТУТ ЦИФРОВОГО ОБРАЗОВАНИЯ МГПУ

Источники

Hadoop (github source code)

Hadoop: The Definitive Guide, 4th Edition (book)

The Hadoop Distributed File System (book)

Check point (e-book)

Configuring Environment of Hadoop Daemons (doc)

HDFS High Availability Using the Quorum Journal Manager (doc)

HDFS Federation (doc)

Hadoop HDFS High Availability (blog)

A Guide to Checkpointing in Hadoop (blog)

Quorum-based Journaling in CDH4.1 (blog)

Introduction to HDFS Erasure Coding in Apache Hadoop (blog)