

# Операционная система Linux: Дисковые системы



Артём  
Поневин



## **Артем Поневин**

Старший инженер автоматизации,  
X5 Retail Group

---

# Предисловие

**На этом занятии мы поговорим о:**

- хранилищах и протоколах;
- видах и различиях RAID;
- LVM и его настройка в Linux;
- возможностях диагностики ввода вывода в Linux.


**По итогу занятия** вы получите представление о дисковой подсистеме Linux, научитесь настраивать программный RAID и LVM, получите представление о диагностике ввода вывода в Linux.

---

# План занятия

1. [Предисловие](#)
2. [Интерфейсы, протоколы и устройства хранения](#)
3. [Дисковая подсистема в Linux](#)
4. [Разделы на блочных устройствах](#)
5. [RAID](#)
6. [LVM](#)
7. [Диагностика ввода-вывода в Linux](#)
8. [Итоги](#)
9. [Домашнее задание](#)

В хорошем качестве картинку можно скачать [здесь](#).



# Интерфейсы, протоколы и устройства хранения

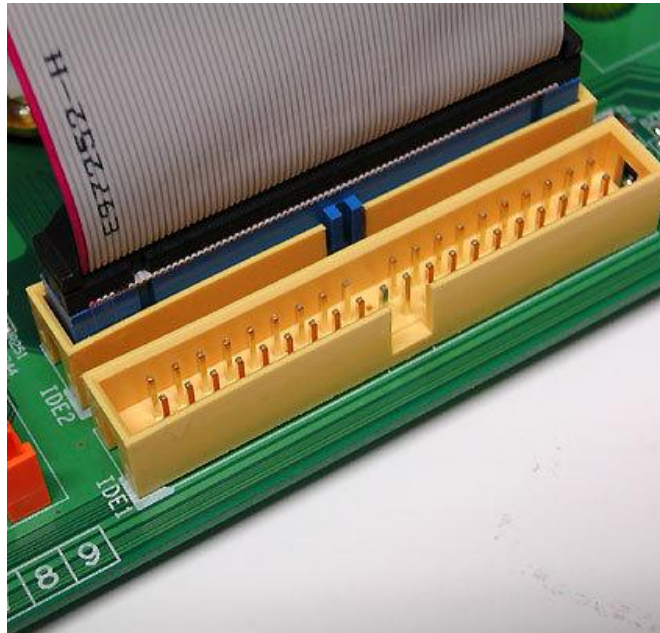
# SCSI

**SCSI** (Small Computer System Interface) – это интерфейс, предназначенный для объединения на одной шине устройств различных классов: жестких дисков, CD-ROM, приводов CD, DVD, стримеров, сканеров, принтеров и т. д.



# IDE (PATA)

**PATA** (Parallel Advanced Technology Attachment) или **IDE** (Integrated Drive Electronics) – параллельный интерфейс подключения накопителей (гибких дисков, жёстких дисков и оптических дисководов) к компьютеру.

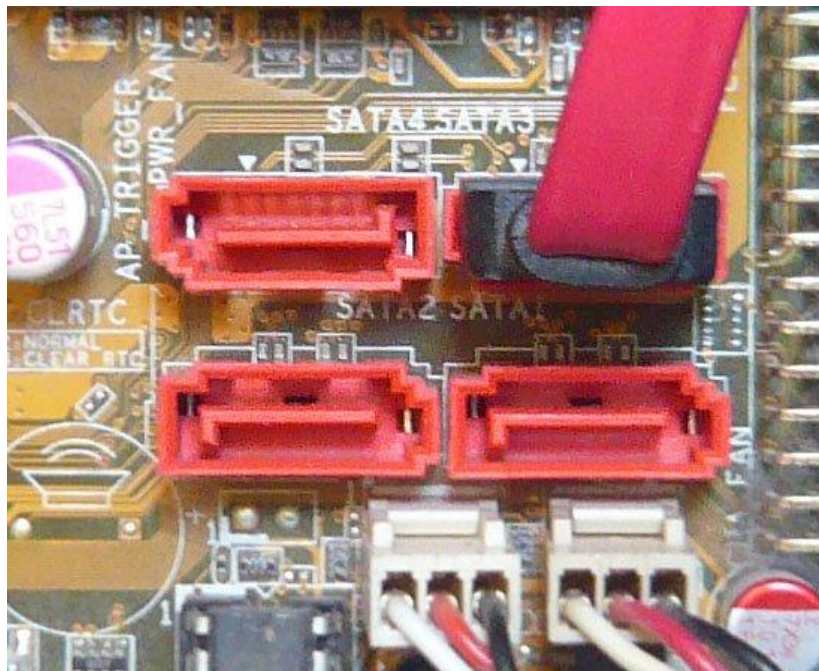




# SATA

**SATA** (Serial ATA) – последовательный интерфейс обмена данными с накопителями информации.

SATA является развитием параллельного интерфейса ATA (IDE).



# Ревизии SATA

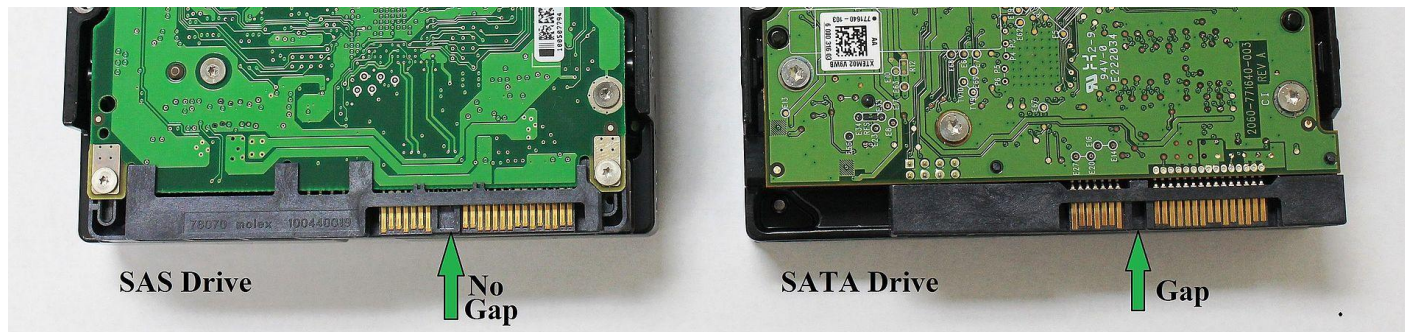
Интерфейс	Пропускная способность		Частота
	биты	байты	
SATA 1.x 1.5Gb/s	1,2 Гбит/с	150 МБ/с	1.5 Гц
SATA 2.x 3Gb/s	2,4 Гбит/с	300 МБ/с	3.0 Гц
SATA 3.x 6Gb/s	4,8 Гбит/с	600 МБ/с	6.0 Гц

# SAS

**Serial Attached SCSI (SAS)** – последовательный компьютерный интерфейс, разработанный для подключения различных устройств хранения данных, например, жёстких дисков и ленточных накопителей.

SAS разработан для **замены параллельного интерфейса SCSI** и основывается во многом на терминологии и наборах команд SCSI.

SAS обратно **совместим с интерфейсом SATA**: устройства 3 Гбит/с и 6 Гбит/с SATA могут быть подключены к контроллеру SAS, но не наоборот.



# Сравнение интерфейсов SAS и SATA

SAS	SATA
Для серверных систем	Преимущественно для настольных и мобильных систем
Использует набор команд SCSI	Использует набор команд ATA
Минимальная скорость вращения шпинделя HDD 7200 RPM, максимальная – 15000 RPM	Минимум 5400 RPM, максимум 7200 RPM
Два дуплексных порта	Один полудуплексный порт
Поддерживается Multipath I/O	Подключение по типу «точка – точка»
Очередь команд до 256	Очередь команд до 32
Можно использовать кабели до 10 м	Длина кабелей не более 1 м
Пропускная способность шины до 12 Гбит/с (в перспективе – 24 Гбит/с)	Пропускная способность 6 Гбит/с (SATA III)
Стоимость накопителей выше, иногда значительно	Дешевле в пересчете на цену за 1 Гб

# PCIe

**Peripheral Component Interconnect Express (PCI Express, PCIe)** – последовательный высокопроизводительный интерфейс для передачи данных. Имеет несколько видов разъемов для подключения различного оборудования, в том числе разъем **M2** для подключения запоминающих устройств.



# Fibre Channel

**Fibre channel** (волоконный канал) — семейство протоколов для высокоскоростной передачи данных.

**Fibre Channel Protocol (FCP)** — транспортный протокол (как TCP в IP-сетях), инкапсулирующий протокол SCSI по сетям Fibre Channel. Является основой построения сетей хранения данных.



# SAN, NAS, DAS

**DAS** – блочное устройство с диска, который физически, напрямую, подключен к хост-машине.

Необходимо поместить на нее файловую систему, прежде чем ее можно будет использовать.

Технологии для этого включают IDE, SCSI, SATA и т.д.

**SAN** – блочное устройство, которое доставляется по сети.

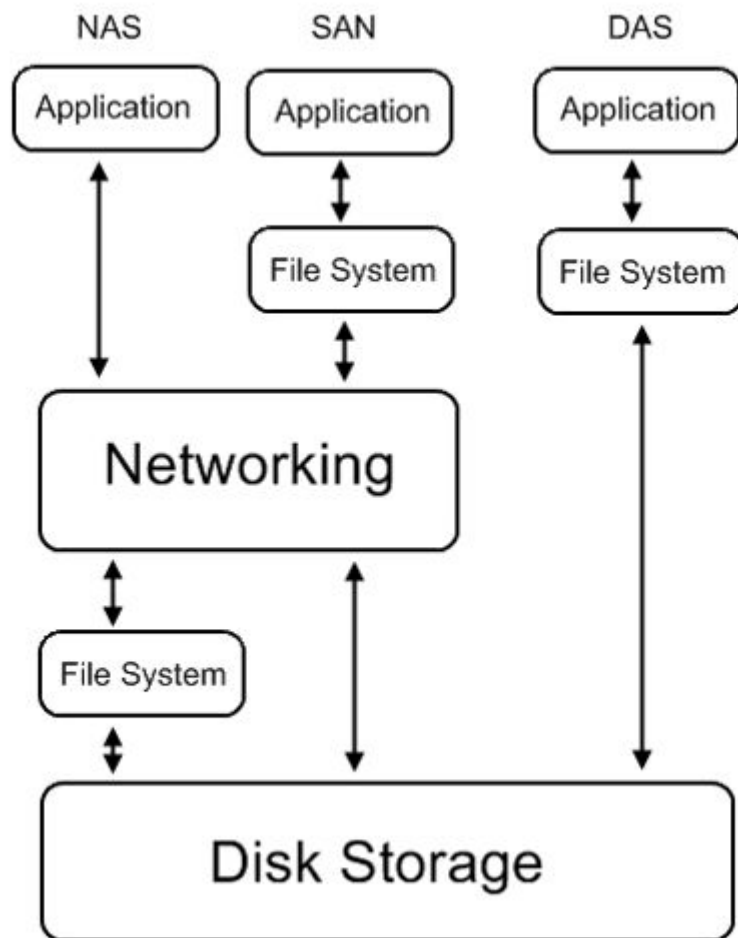
Как и DAS, вы все равно должны разместить на нем файловую систему, прежде чем она сможет использоваться.

Технологии для этого включают FibreChannel, iSCSI, FoE и т. Д.

**NAS** – файловая система, доставляемая по сети.

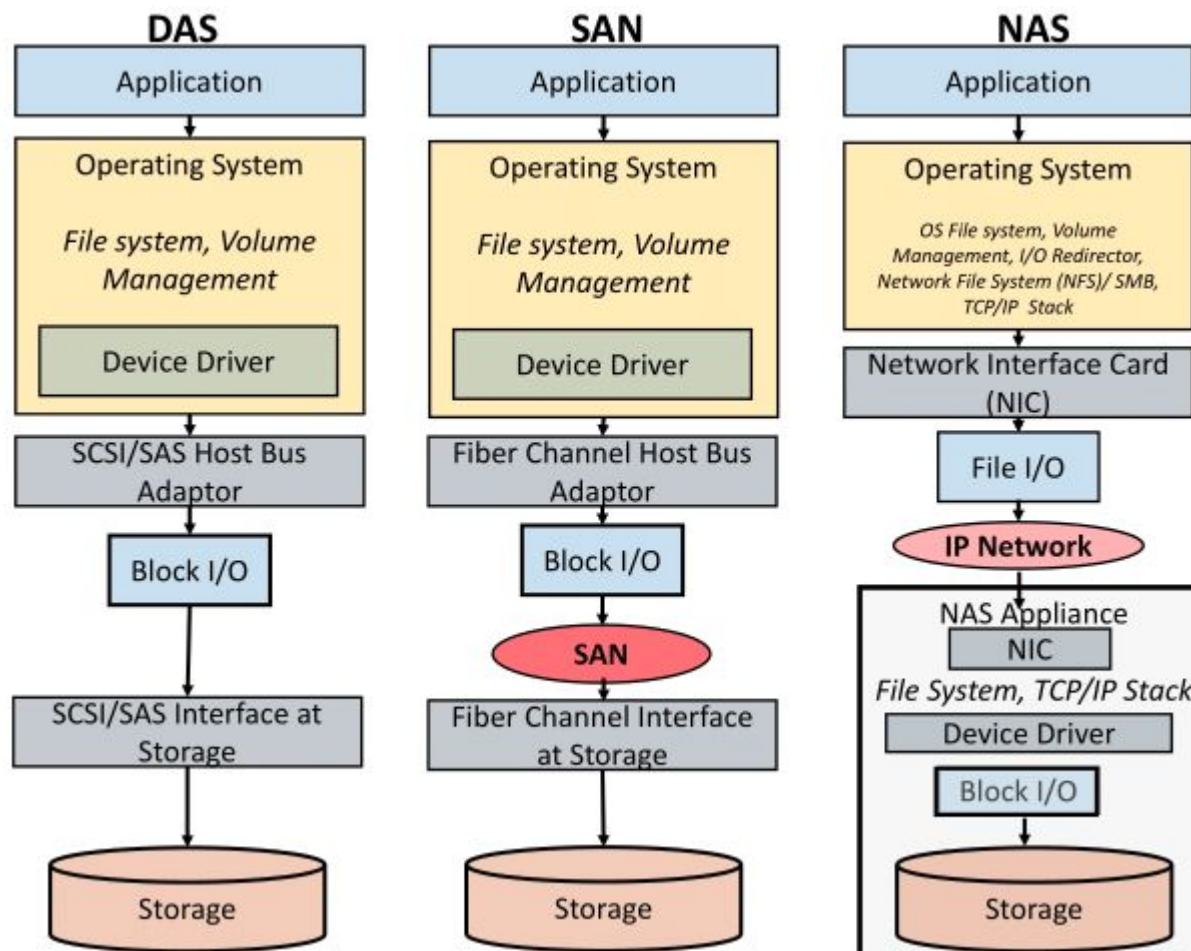
Технологии для этого включают NFS, CIFS, AFS и т.д.

# Сравнение SAN, NAS, DAS



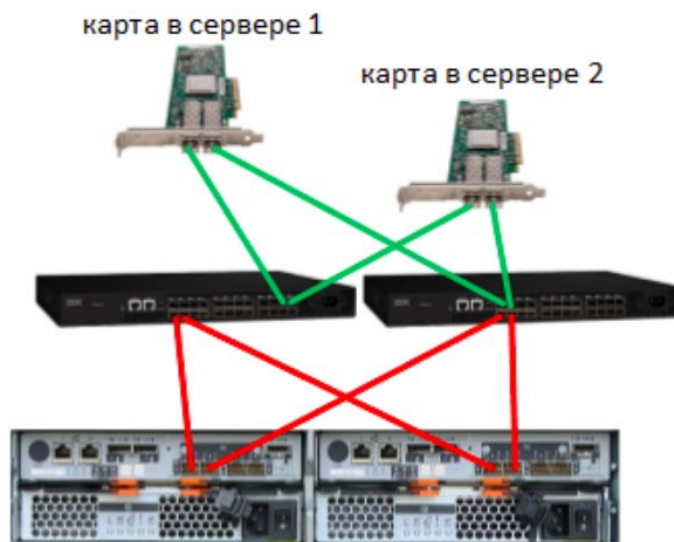


# Сравнение SAN, NAS, DAS

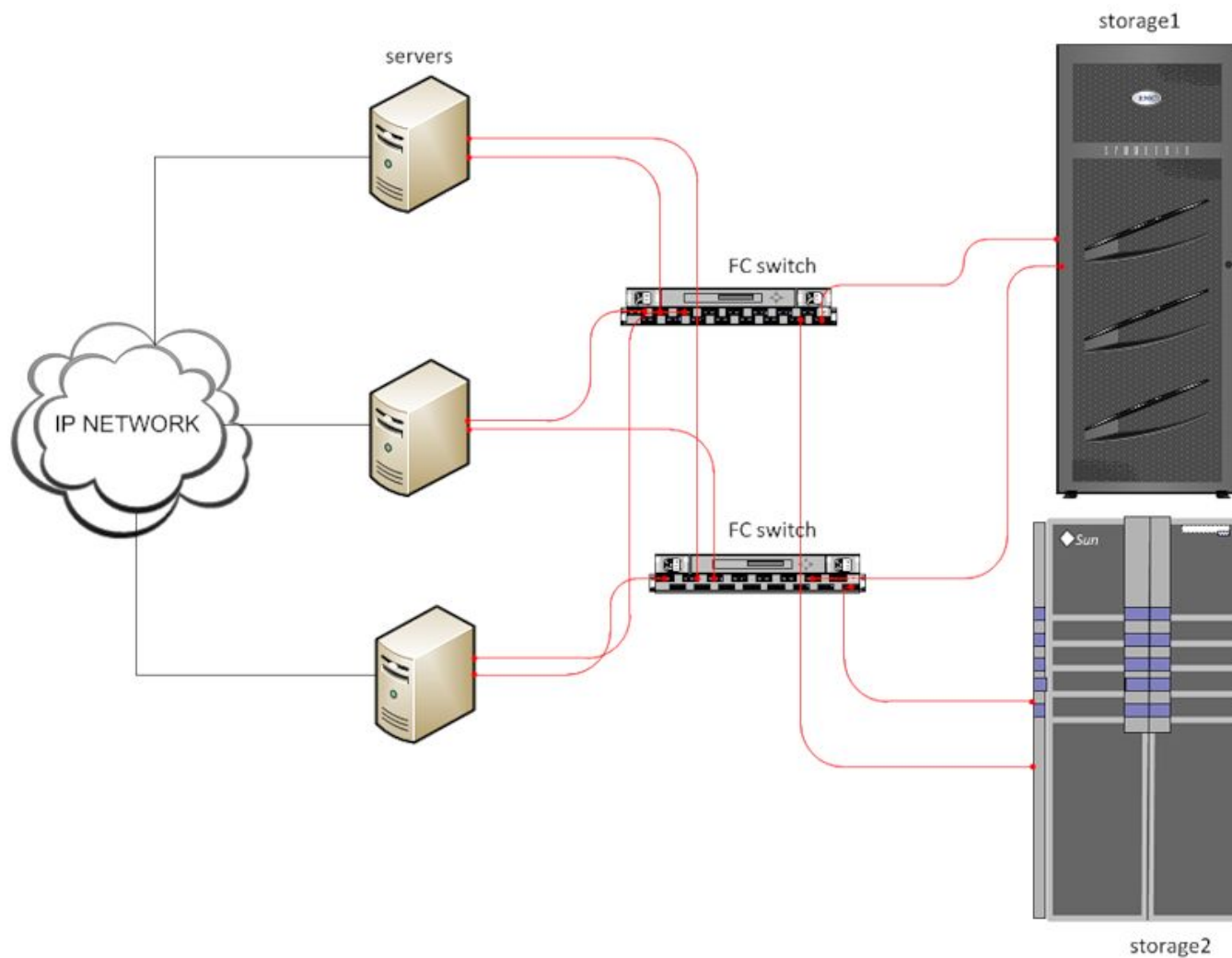


# Пример СХД

**СХД** (сеть хранения данных, SAN, storage area network) – архитектурное решение для подключения внешних устройств хранения данных (дисковые массивы, ленточные библиотеки) к серверам таким образом, чтобы операционная система распознала подключённые ресурсы как локальные.



# Архитектура СХД



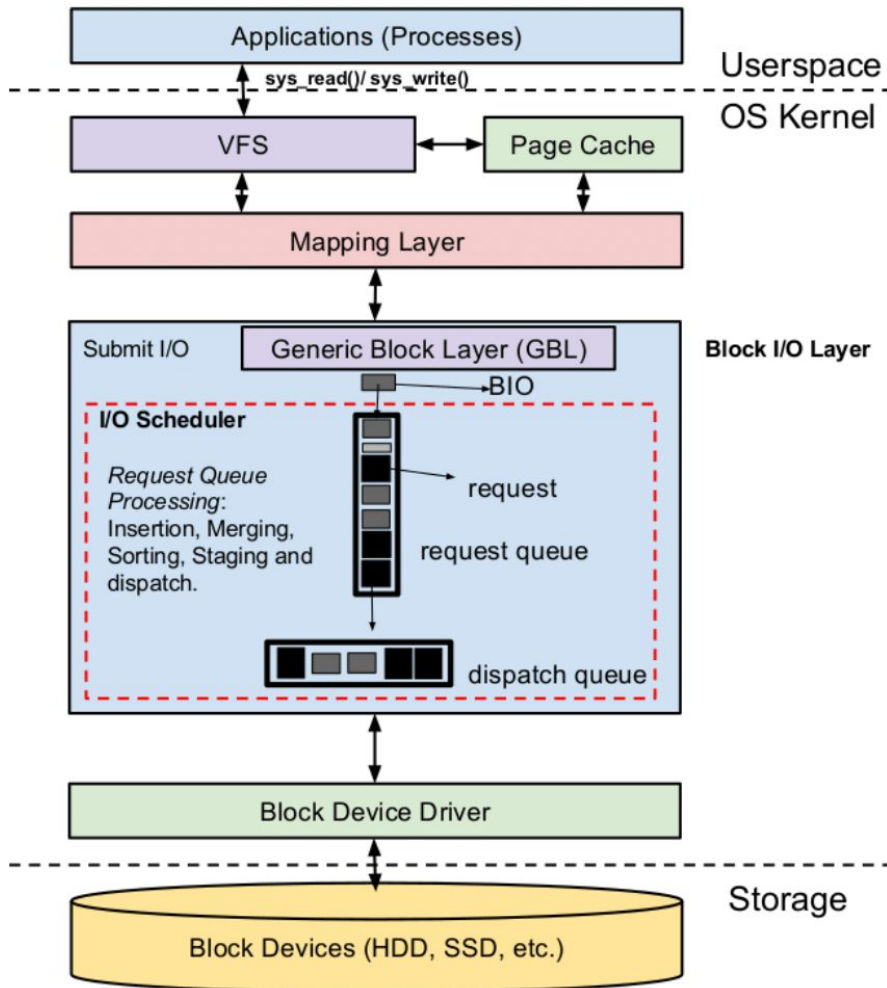
# Команды Linux для работы с аппаратным обеспечением

- **cat /proc/devices** - отображает список устройств, опознанных ядром
- **ls -la /dev/disks/by-\*** - выводит информацию о дисках, подключенных к системе
- **ls -la /dev** – каталог специального назначения, который содержит файлы устройств.
- **lsblk** - list block devices
- **lshw -short -C disk, hdparm -l /dev/sda, smartctl --all /dev/nvme0**  
– утилиты, которые отображают информацию об имеющихся дисках с точки зрения железа

---

# Дисковая подсистема в Linux

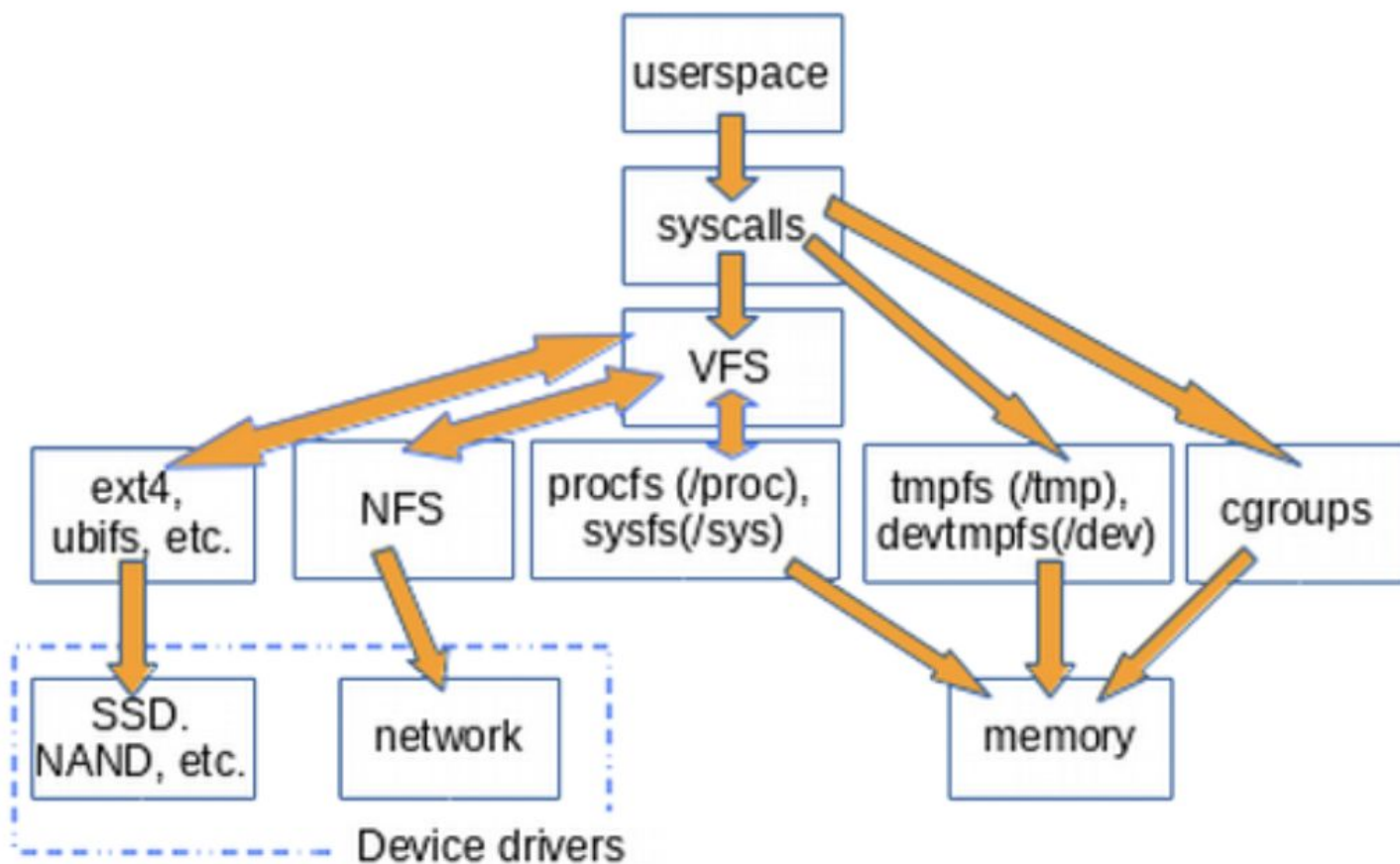
# Упрощенная архитектура работы с устройствами хранения в Linux



# Virtual File System

**Виртуальная файловая система** (англ. virtual file system — VFS) - это абстрактный уровень поверх конкретной реализации файловой системы. Она используется для доступа к локальным файловым системам(ФС) (fat32, ext4, ntfs), сетевым ФС(nfs), а также к устройствам, не предназначенным для хранения данных (procfs). VFS предоставляет единообразный доступ клиентских приложений к различным типам файловых систем. VFS декларирует программный интерфейс между ядром и конкретной файловой системой, таким образом, для добавления поддержки новой файловой системы не требуется вносить изменений в ядро операционной системы.

# Virtual File System





---

# Device mapper

Подсистема ядра Linux, которая позволяет мэппить одно блочное устройство на другое или несколько других.

➡ Это позволяет реализовать объединение нескольких реальных устройств в одно виртуальное, создавать snapshot'ы, организовывать балансировку между устройствами, делать шифрованные тома и т.д.

➡ Через device-mapper, в качестве низкоуровневого API, работают LVM, cryptsetup, dm-multipath и другие утилиты.

# Команды Linux для device mapper

- **/dev/mapper/vgdata-lvdata** – ссылки на устройства, созданные device mapper;
- **/dev/dm-0, /dev/dm-1** – устройства которые создаются device mapper;
- **/dev/vgdata/lvdata** – ссылки на устройства, созданные device mapper;
- **dmsetup ls** - отображает список устройств, созданных с помощью device mapper.

---

# **Разделы на блочных устройствах. Таблица разделов**

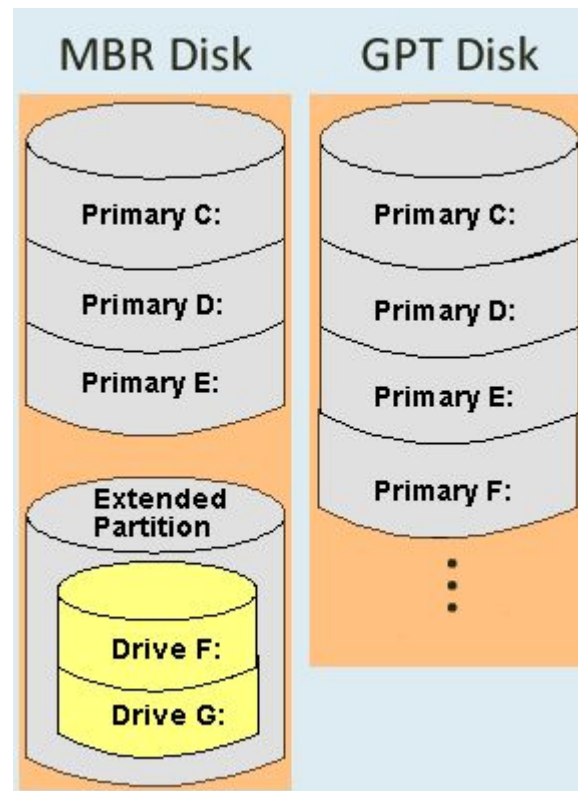
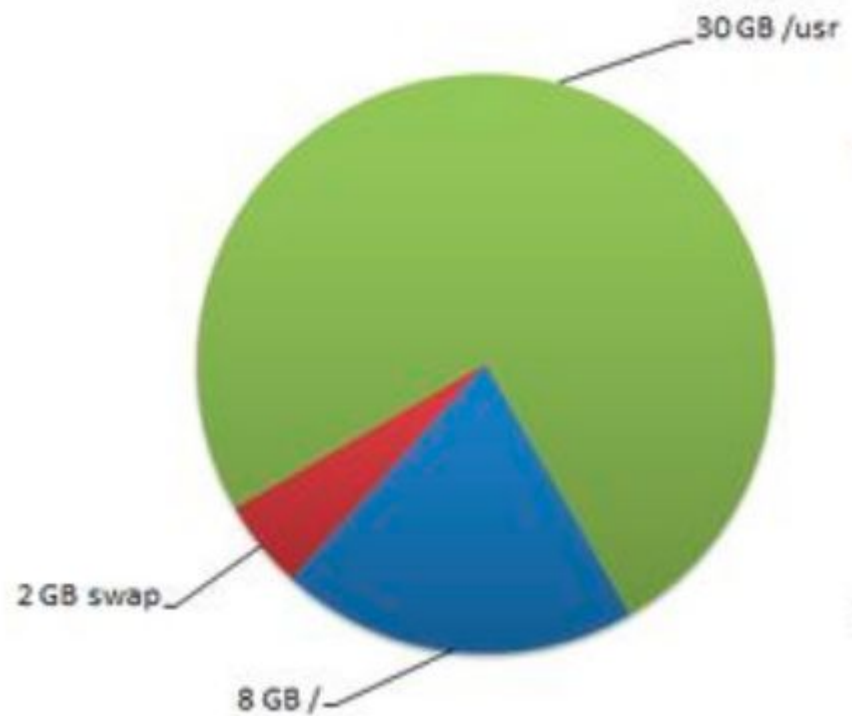
# Разделы жесткого диска

**Раздел** — часть долговременной памяти жёсткого диска или флеш-накопителя, выделенная для удобства работы, и состоящая из смежных блоков. На одном устройстве хранения может быть несколько разделов.

*Использование разделов дает возможность:*

- хранить информацию в разных файловых системах, или в одинаковых файловых системах, но с разным размером кластера;
- на одном жёстком диске можно установить несколько операционных систем

# Диаграмма разбивки жестких дисков



# Master Boot Record

MBR (Master Boot Record, главная загрузочная запись) содержит сведения о структуре жесткого диска и код для запуска операционной системы.



---

# Master Boot Record

Особенности MBR:

- можно разместить 4 раздела,
- типы разделов : Primary, Extended(Logical),
- максимальный объем диска - 2 ТБ,
- 512 байт для загрузчика,
- 64 байта для хранения таблицы разделов.

# GUID partition table

**GUID** (Globally Unique Identifier) **Partition Table** (GPT) — стандарт формата размещения таблиц разделов на физическом жестком диске. Является частью расширяемого микропрограммного интерфейса, Extensible Firmware Interface, EFI.





# GUID partition table

Особенности GPT:

- можно разместить 128 разделов(зависит от ОС);
- типы разделов больше не существуют (можно считать, что все разделы Primary);
- хранит копию таблицы разделов в конце диска.

---

# Команды Linux для работы с дисками и разделами

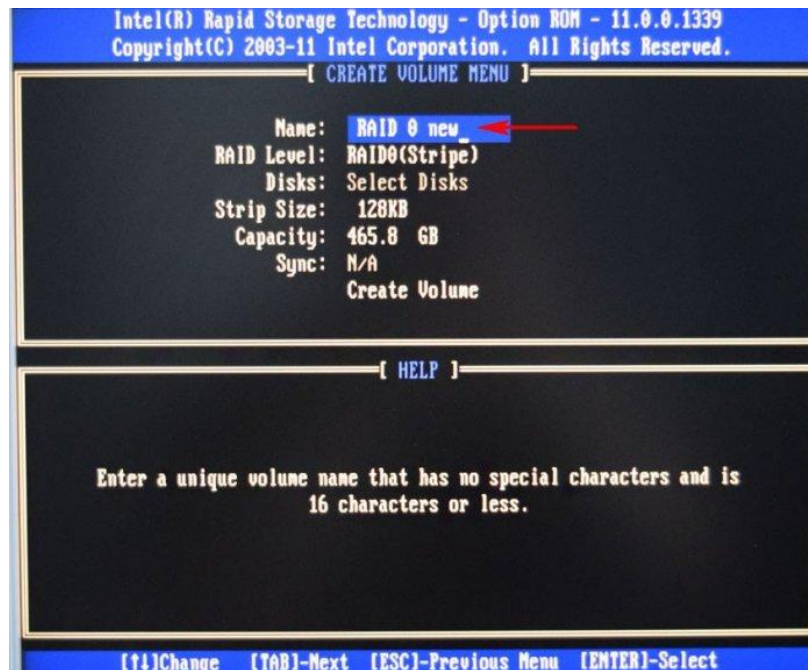
- **lsblk** – утилита выводит список блочных устройств с информацией о них;
- **blkid** – утилита отображает информацию об уникальных идентификаторах блочных устройств;
- **fdisk** – утилита позволяет управлять разделами с разметкой MBR;
- **gdisk** – утилита позволяет управлять разделами с разметкой GPT;
- **parted** – утилита позволяет управлять разделами с разметкой MBR и GPT. В отличие от **gdisk** и **fdisk** изменения применяются сразу, не требуя подтверждения;
- **cat /proc/partitions** – вывести список разделов.



# RAID

# RAID

**RAID** (Redundant Array of Independent Disks, избыточный массив независимых дисков) — технология виртуализации данных, которая объединяет несколько дисков в логический элемент для повышения производительности и отказоустойчивости.



---

# Виды RAID контроллеров

- **Аппаратный** – отдельное устройство.

Настройка производится через специальное ПО.

- **Полуаппаратный** – встроен в материнскую плату.

Настройка производится через BIOS.

- **Программный** – работает из ОС.

Настраивается утилитами из ОС: mdadm для Linux, составные тома для Windows.



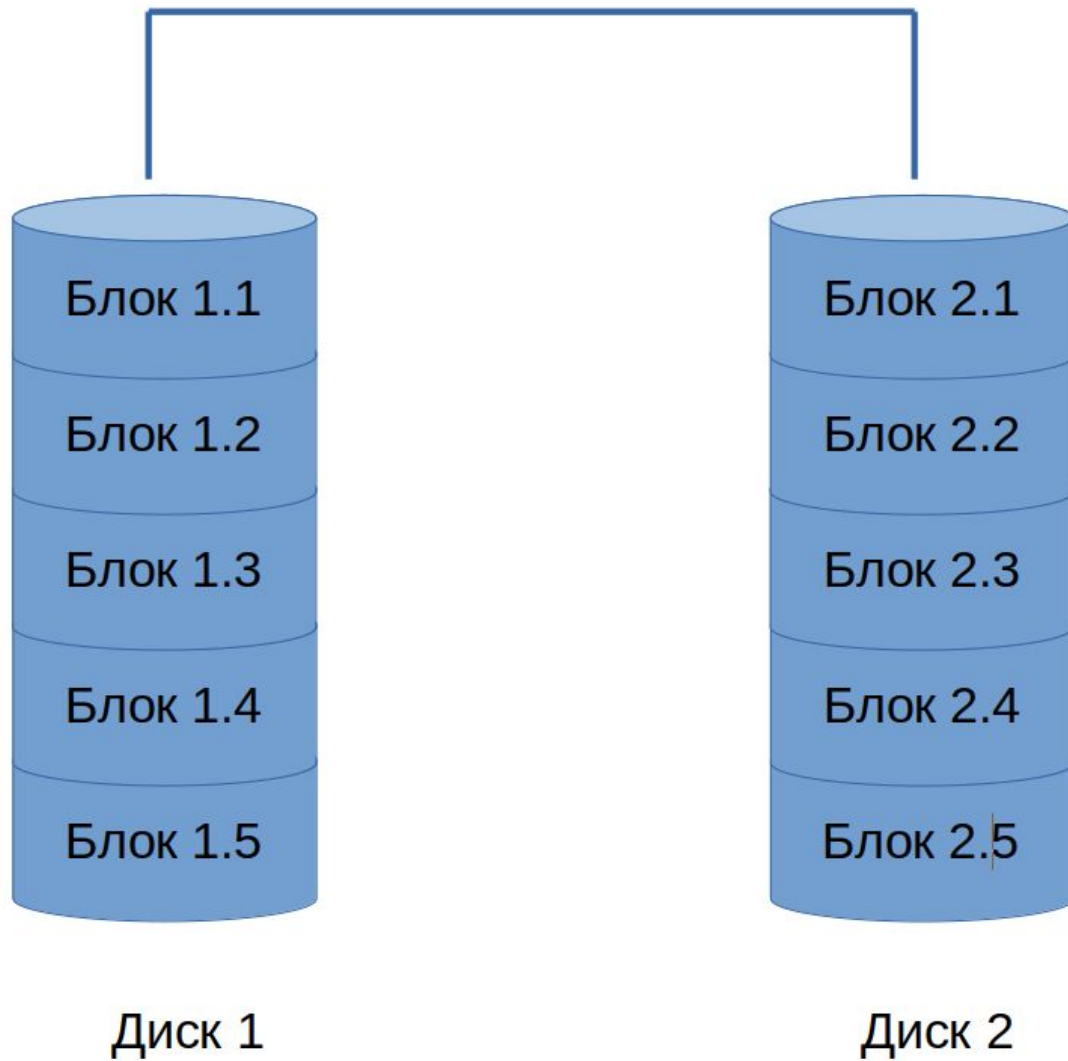
## RAID JBOD\*

В конфигурации **JBOD** данные на дисках хранятся последовательно. Например, сначала данные записываются на Диск 1. После заполнения Диска 1 данные записываются на Диск 2, затем на Диск 3 и т. д.

Два преимущества данного уровня RAID — это доступность всей общей емкости хранения дисков и простота расширения. Однако если один диск выйдет из строя, все данные будут утеряны.

\*RAID JBOD – Just a Bunch Of Disks.

# JBOD



# RAID 0

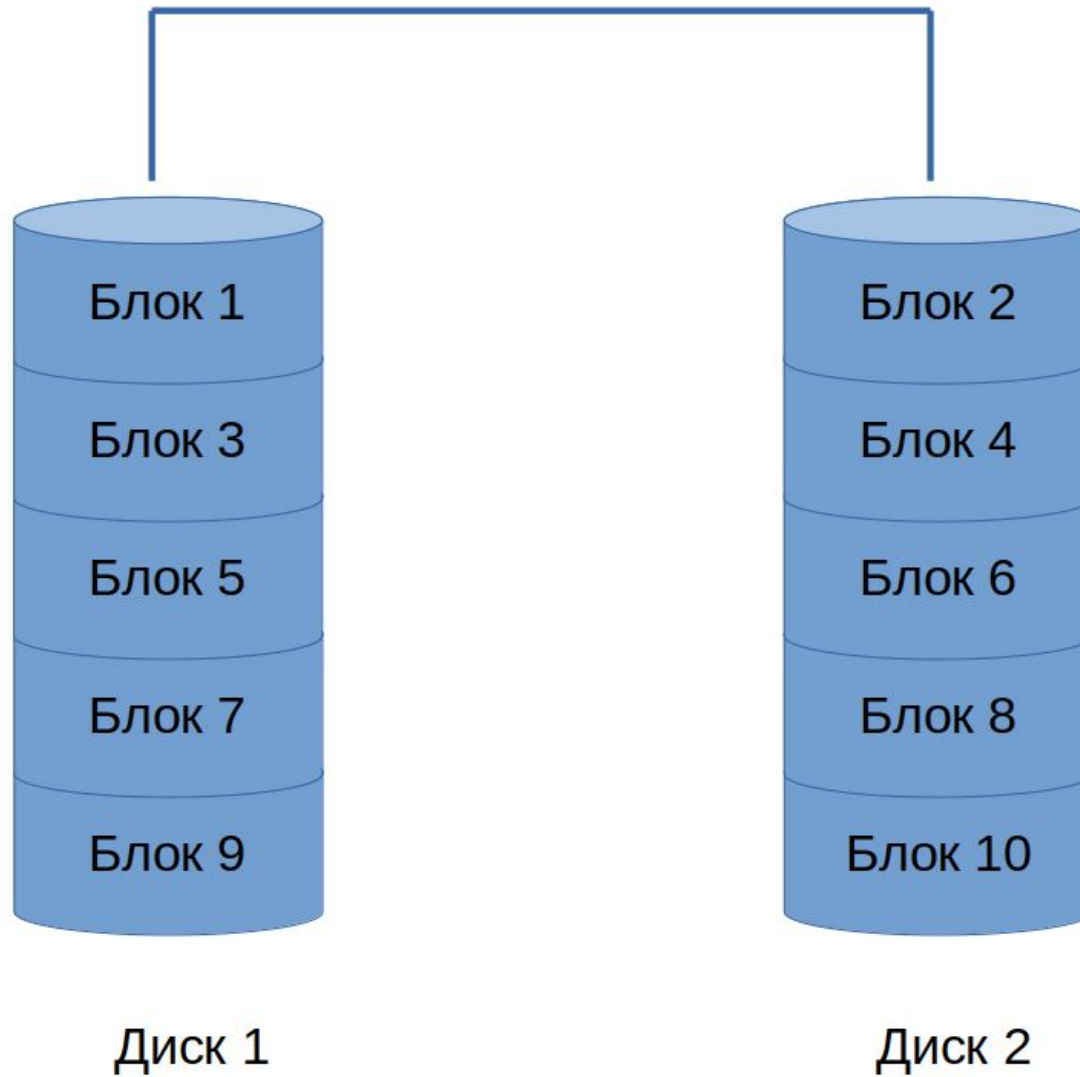
**RAID 0** (striping, чередование) — дисковый массив из двух или более жёстких дисков без резервирования.

Информация разбивается на одинаковые по длине блоки, а затем записывается поочерёдно на каждый диск в структуре.

Основное предназначение такой системы — увеличение производительности, при этом вам будет доступен полный объем всех дисков. Если один диск выходит из строя, все данные становятся недоступными.



## RAID 0





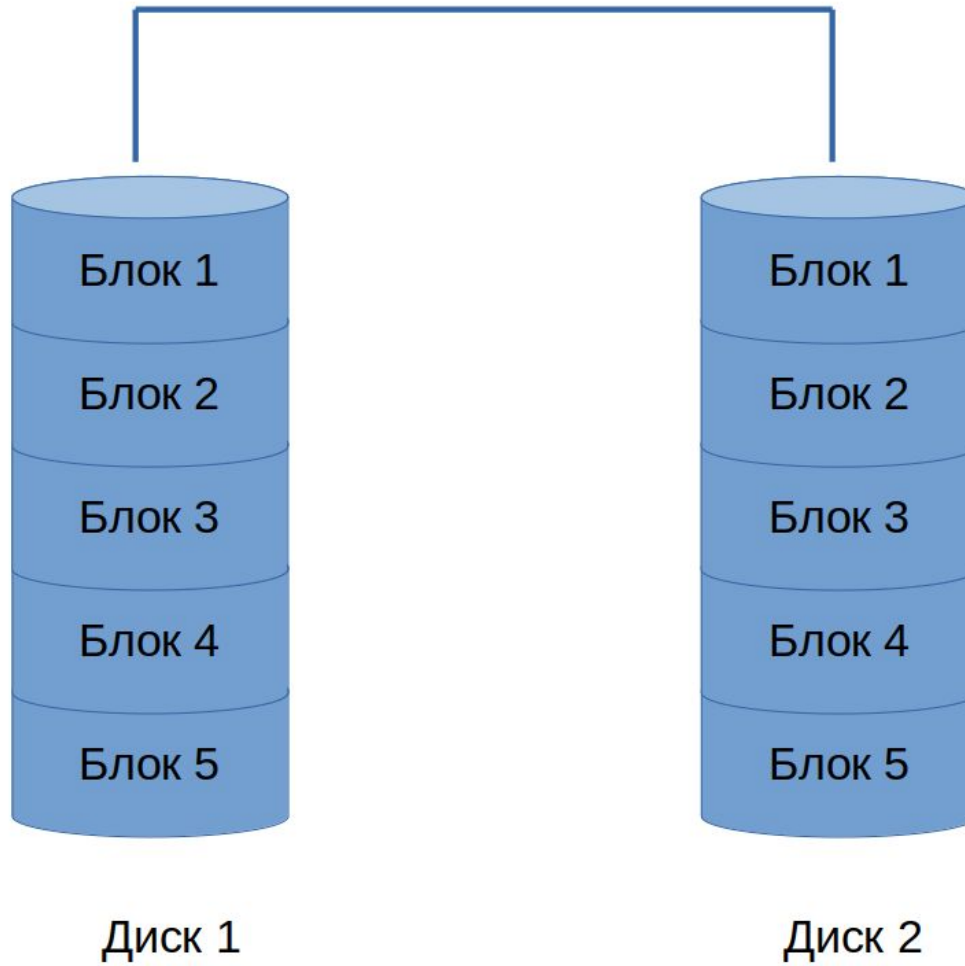
# RAID 1

**RAID 1** (mirroring, зеркалирование) — массив из двух или более дисков, являющихся полными копиями друг друга.

Обеспечивается бесперебойность работы, даже если один из дисков выйдет из строя.

Производительность остается на прежнем уровне, объем равен меньшему из дисков в массиве.

# RAID 1

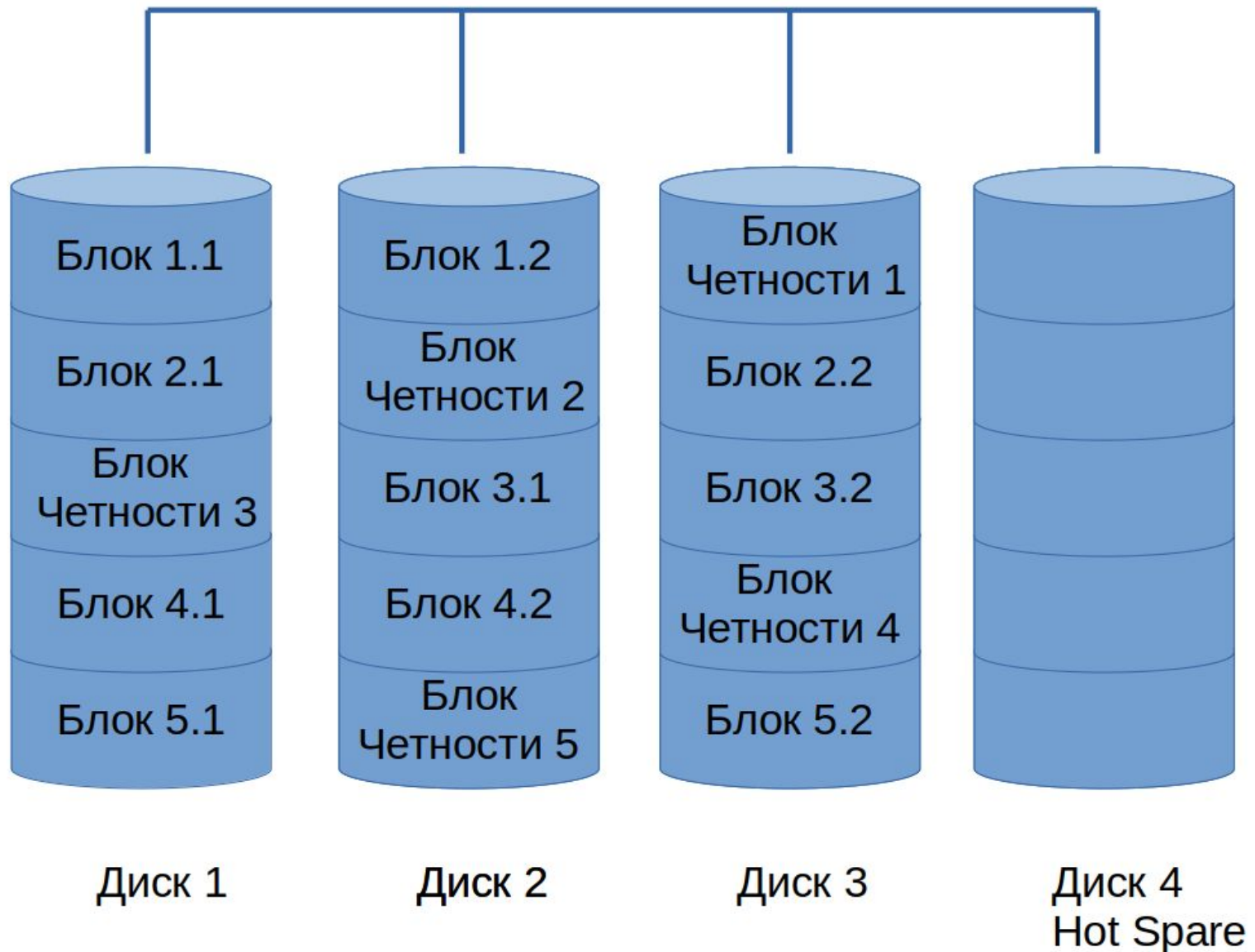


---

# RAID 5

- Данные записываются на все диски в том и в блок четности для каждого блока данных.
- Если один физический диск выходит из строя, данные из неисправного диска можно восстановить на запасной диск.
- Данные сохраняются при выходе из строя одного диска, но в случае выхода из строя второго диска до того, как данные смогли быть восстановлены на запасной диск, все данные будут потеряны.
- Для создания тома RAID 5 требуется минимум три диска.

# RAID 5



# RAID 10

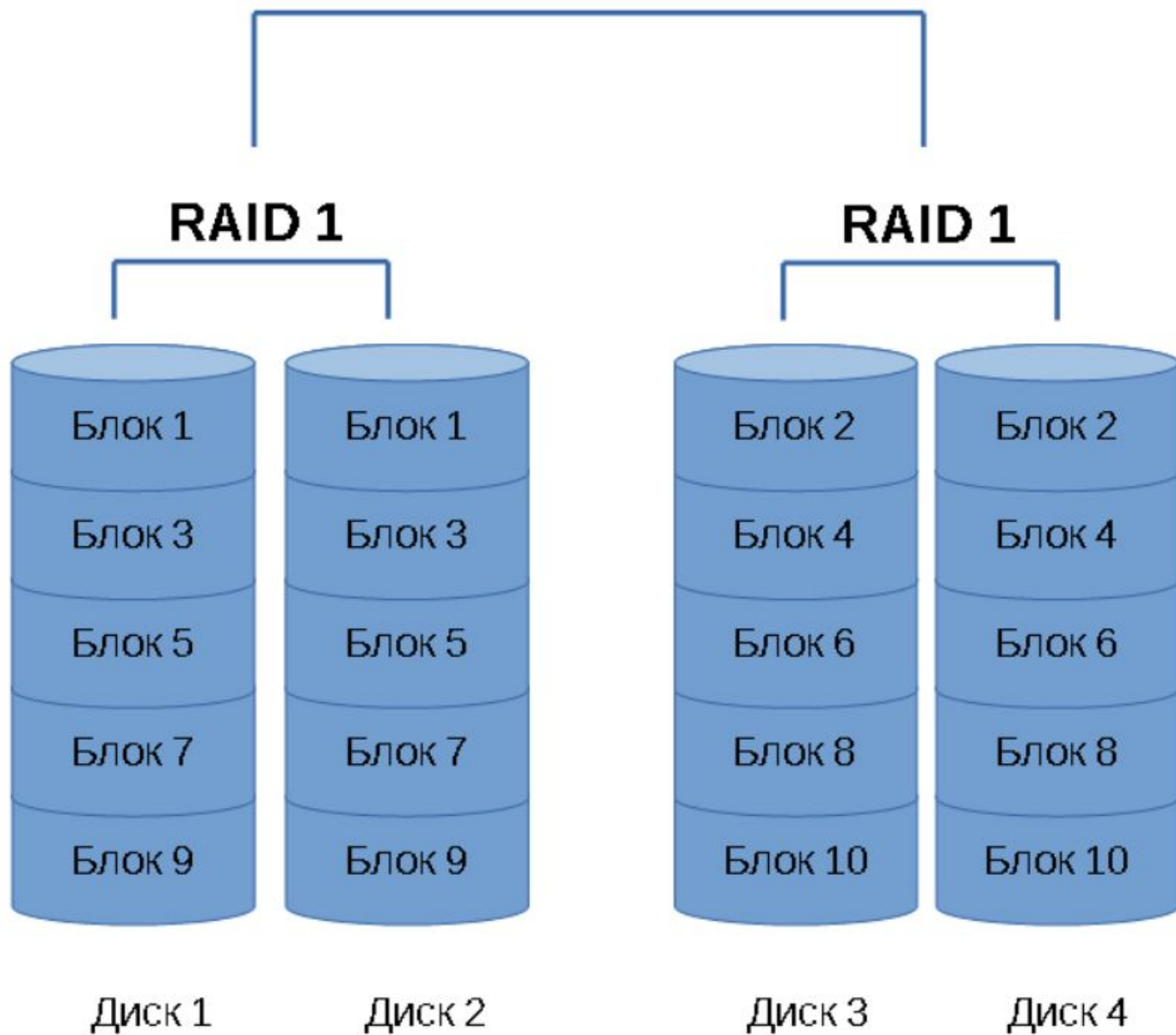
В режиме RAID 10 сочетаются защита режима RAID 1 и производительность режима RAID 0.

➡ При использовании четырех дисков в режиме RAID 10 создается два сегмента RAID 1, которые объединяются в страйп RAID 0.

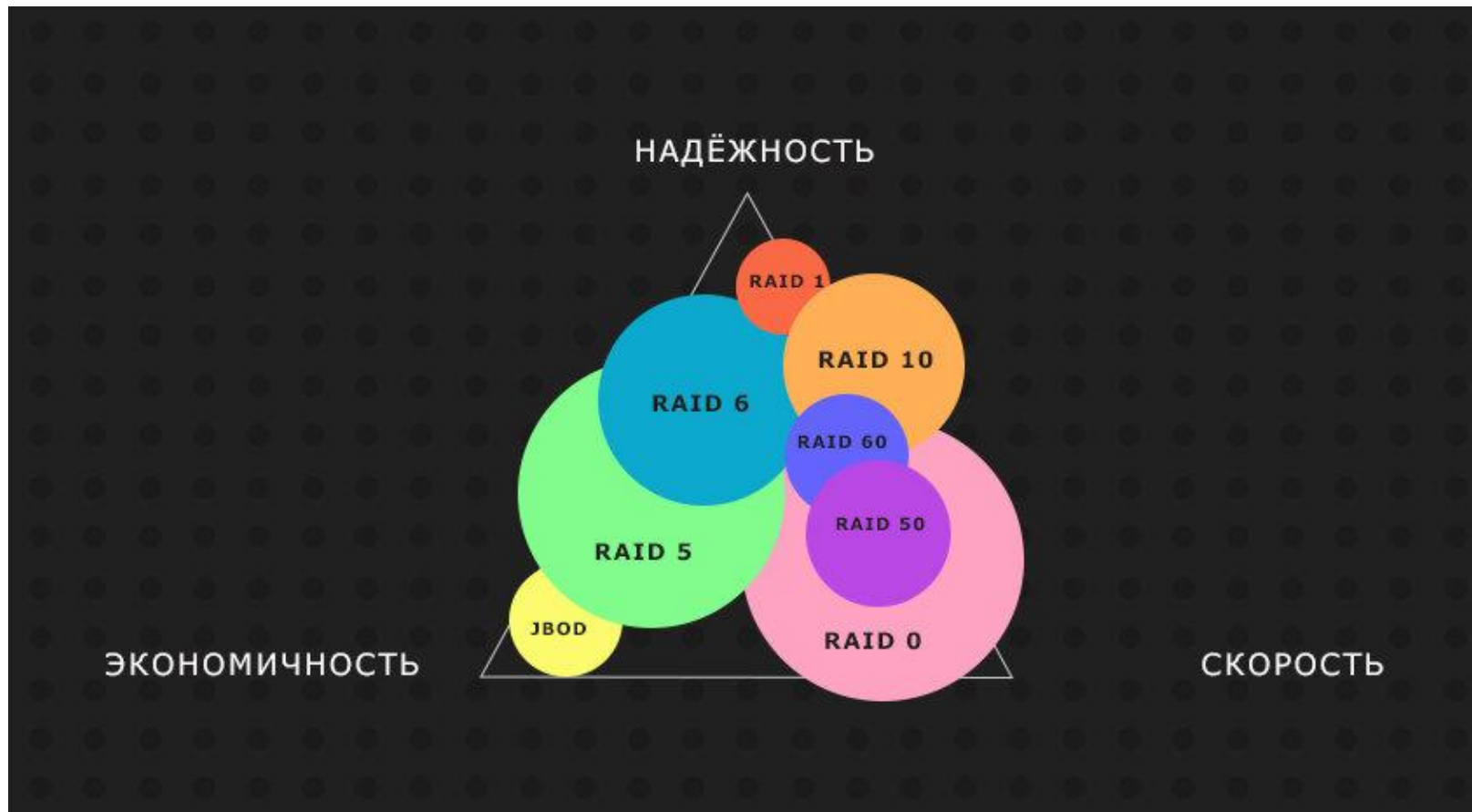
➡ При использовании восьми дисков в страйпе RAID 0 будет уже четыре сегмента RAID 1.

В режиме RAID 10 могут выйти из строя до 2х дисков в двух сегментах RAID 1.

# RAID 10 RAID 0



# Выбор RAID





# Команды Linux для работы с RAID

- `sudo yum (apt-get) install mdadm` — установка утилиты;
- `sudo mdadm --create /dev/md0 -l 1 -n 2 /dev/sd{b,c}` — создание нового массива;
- `cat /proc/mdstat` — текущее состояние;
- `/etc/mdadm.conf` — файл конфигурации.



**LVM**



# LVM

**LVM** (Logical Volume Manager, Менеджер логических томов) — это дополнительный слой абстракции между «железом» и файловой системой, позволяющий использовать разные области одного жёсткого диска и/или области с разных жёстких дисков как один логический том.

---

# Преимущества LVM

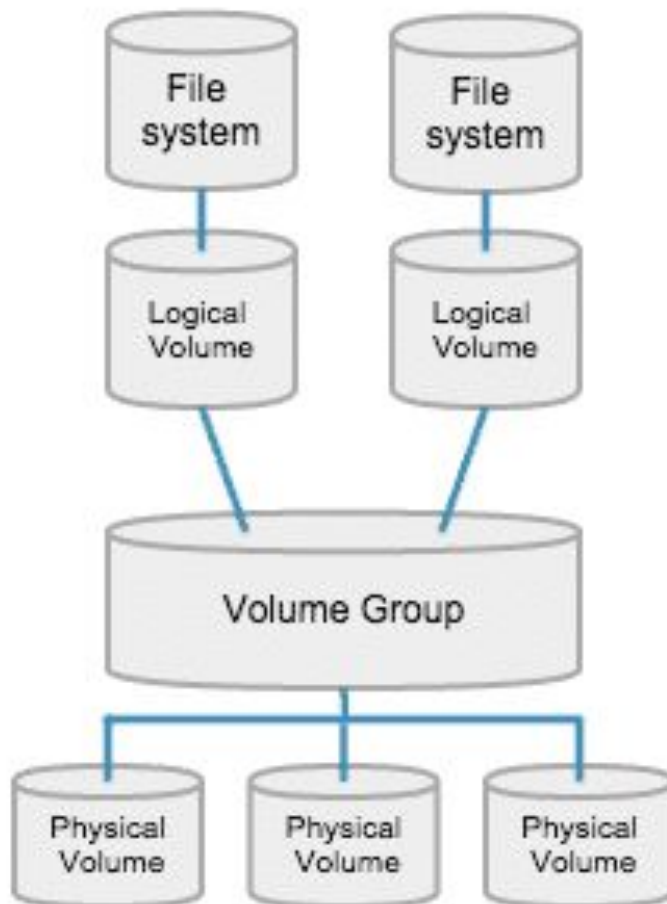
- легко изменять размер логических томов,
- можно объединять несколько физических устройств в один логический пул,
- возможное количество логических томов намного превышает возможности традиционного деления диска на разделы,
- с помощью снапшотов можно делать бэкапы логических томов.

---

## Уровни абстракции LVM

- **PV** (Physical Volumes, физические тома) — разделы или целые «неразбитые» диски;
- **VG** (Volumes Group, группа томов) — объединенные в группу физические тома (PV), из которых создается единый логический диск, который в дальнейшем можно разбивать по своему усмотрению;
- **LV** (Logical Volumes, логические разделы) — собственно раздел нового «единого диска», полученного из группы томов, который можно форматировать и использовать как обычный раздел, обычного жёсткого диска.

# Уровни абстракции LVM



---

# Команды Linux для изменения LVM

- **pvccreate** — позволяет создать физический том на жестком диске;
- **vgcreate** — позволяет создать группу томов из физических ТОМОВ;
- **lvcreate** — позволяет создать логический том в группе томов.

---

# Команды Linux для получения состояния LVM

- **pvdisplay\pvs** — позволяет отобразить информацию о физических томах;
- **vgdisplay\vgs** — позволяет отобразить информацию о группах томов в ОС;
- **lvdisplay\lvs** — позволяет отобразить информацию о логических томах.





# Диагностика ввода-вывода в Linux

---

# Команды Linux для диагностики нагрузки на дисковую подсистему

- **top** — выводит информацию о работающих в системе процессах и информацию о них;
- **iostat** — мониторинг использования дисковых разделов;
- **iotop** — аналогична утилите top, но вместо использования процессами CPU и памяти показывает работу процессов с дисками;
- **vmstat 5 5** — утилита отображает информацию об использовании CPU, памяти, дисков.
- **sar -p -d 5 3** - утилита для отображения различных параметров (статистики) работы системы.



# Итоги

---

# Итоги

Сегодня мы рассмотрели работу с дисками в Linux:

- виды носителей и протоколы;
- устройство LVM;
- виды RAID массивов;
- утилиты для диагностики ввода-вывода.

# Домашнее задание

Давайте посмотрим ваше [домашнее задание](#).

- Вопросы по домашней работе задавайте в чате учебной группы и/или в разделе “Вопросы по заданию”.
- Задачи можно сдавать **по частям**.
- Зачёт по домашней работе проставляется после того, как **приняты все задачи**.

⌘ нетология

**Задавайте вопросы и  
пишите отзыв о лекции!**

**Поневин Артем**