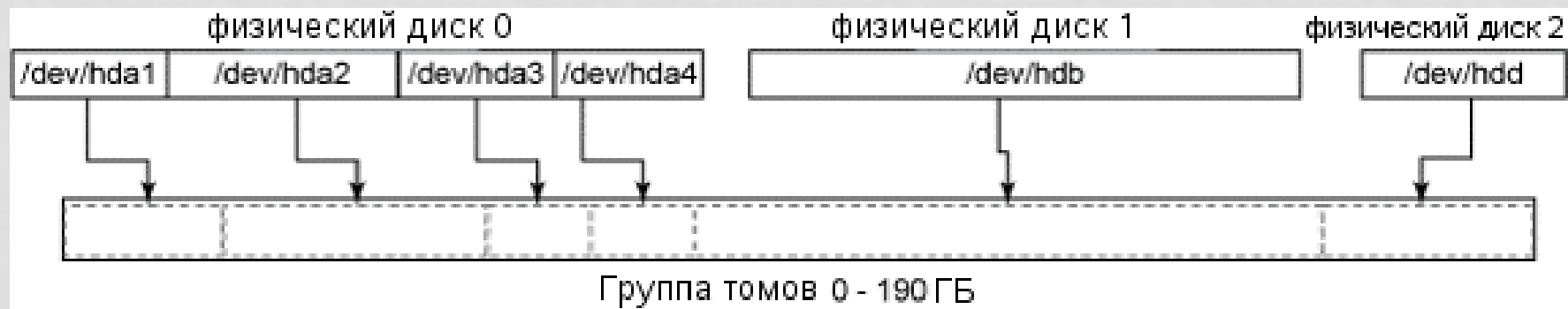


LVM

Logical Volume Management – LVM

Управление логическими томами (LVM) — это способ абстрагировать физическое управление томами в системе в высокоуровневую и, как правило, в более простую парадигму. В рамках LVM все физические диски и разделы вне зависимости от их размера и разбиения могут абстрагироваться и рассматриваться как единое хранилище данных.





LVM

- Собирая разделы и целые диски в виртуальный диск, LVM может суммировать небольшие объемы дисковой памяти в большой объединенный диск. Этот виртуальный диск в терминах LVM называется логической группой (*volume group*).

LVM также позволяет:

- Добавлять диски/разделы в дисковую группу и расширять существующие файловые системы «на лету»
- Заменить два жестких диска размером 80 ГБ одним диском на 160 ГБ без необходимости выключения компьютера для переноса системы или ручного перемещения данных между дисками
- Уменьшить размеры файловых систем и удалить диски из дисковой группы, когда их емкость больше не требуется
- Создавать внутренне согласованные резервные копии на основе мгновенных копий файловой системы (подробнее об этом будет сказано дальше)

Организация LVM

LVM состоит из трех элементов:

- Тома (Volume): физические и логические тома и группы ТОМОВ
- Диапазоны (Extent): физические и логические диапазоны
- Модуль отображения устройств (Device-mapper): модуль ядра Linux



Том

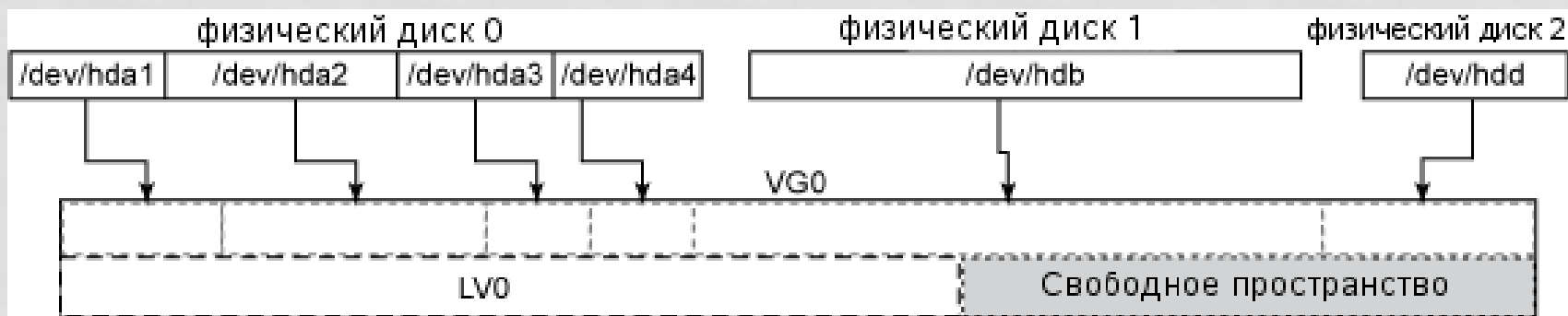
LVM

Тома

В Linux LVM используются физические тома (physical volume – PV), группы томов (volume groups – VG) и логические тома (logical volume – LV).

Физические тома являются физическими дисками или разделами физических дисков (как в `/dev/hda` или `/dev/hdb1`).

Группы томов объединяют физические тома. Группа томов может быть логически разделена на логические тома.

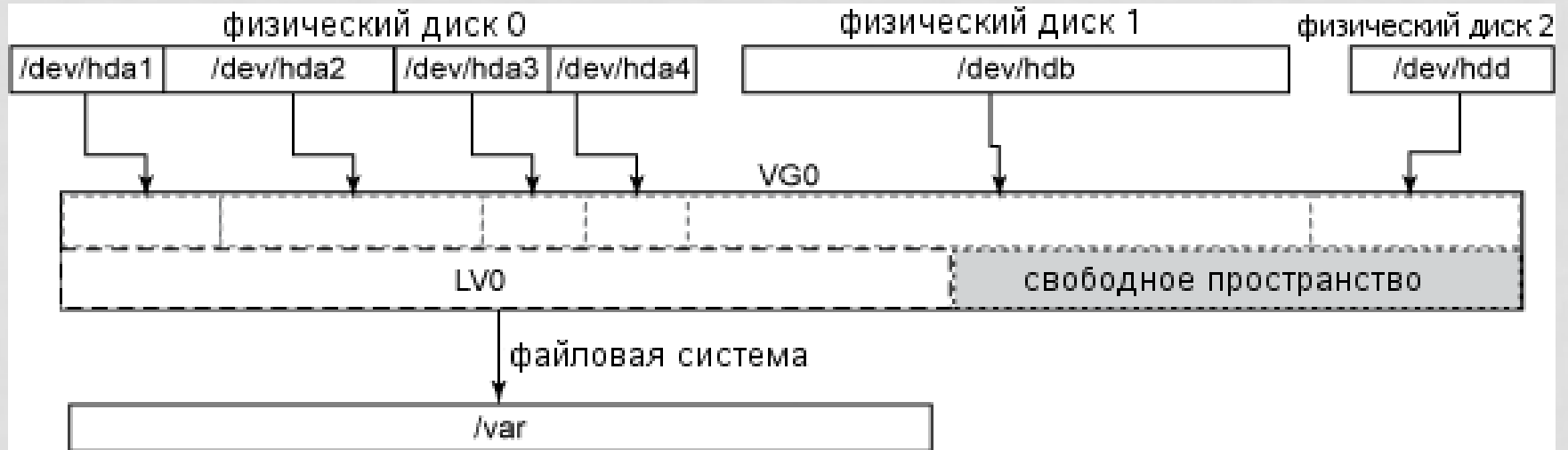


Все 4 раздела физического диска 0 (`/dev/hda[1-4]`), а также все разделы физического диска 1 (`/dev/hdb`) и физического диска 2 (`/dev/hdd`) добавлены как физические тома в группу томов VG0.

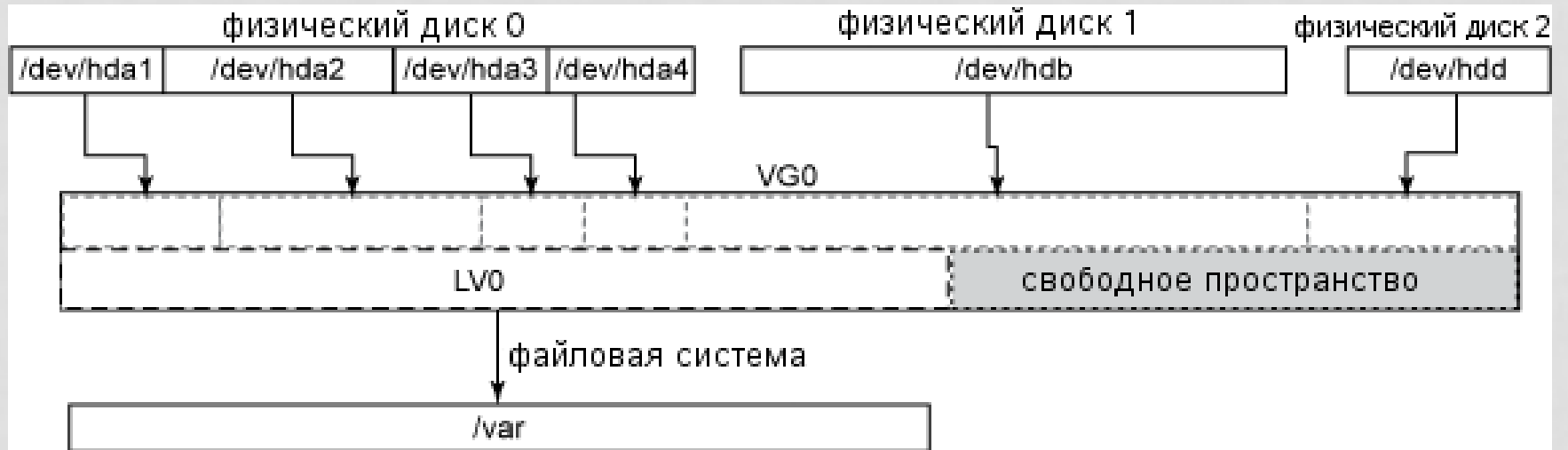
Тома

- В группе томов производится “волшебное” отображение из n в m (т.е. n физических томов представляются как m логических). Так, после назначения физических томов группе вы можете создать логический том любого размера (вплоть до суммарного объема группы томов). В примере была создана группа томов LV0 и оставлено свободное место для других логических томов (или для последующего роста LV0).
- Логические тома являются LVM-эквивалентом физических дисковых разделов—для всех практических целей они являются физическими дисковыми разделами.
- После создания логического тома вы можете использовать его с любой файловой системой, которую вы предпочитаете, и смонтировать его в некоторую точку монтирования для дальнейшего использования.

Пример



Пример



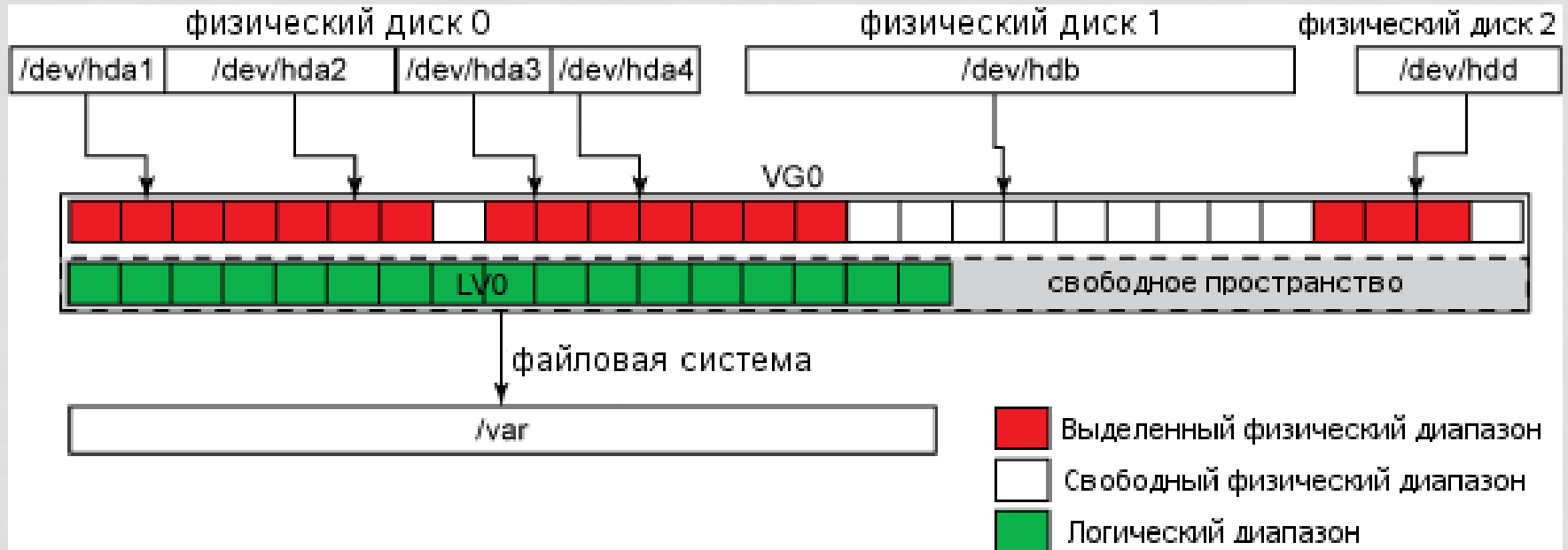
Диапазоны

Диапазоны

Чтобы можно было отобразить n физических томов в m логических, физические тома и группы томов должны состоять из базовых блоков одинакового размера; они называются *физическими диапазонами* (*physical extents — PE*) и *логическими диапазонами* (*logical extents — LE*). Даже если n физических томов отображается в m логических, PE и LE всегда находятся в соотношении 1 к 1.

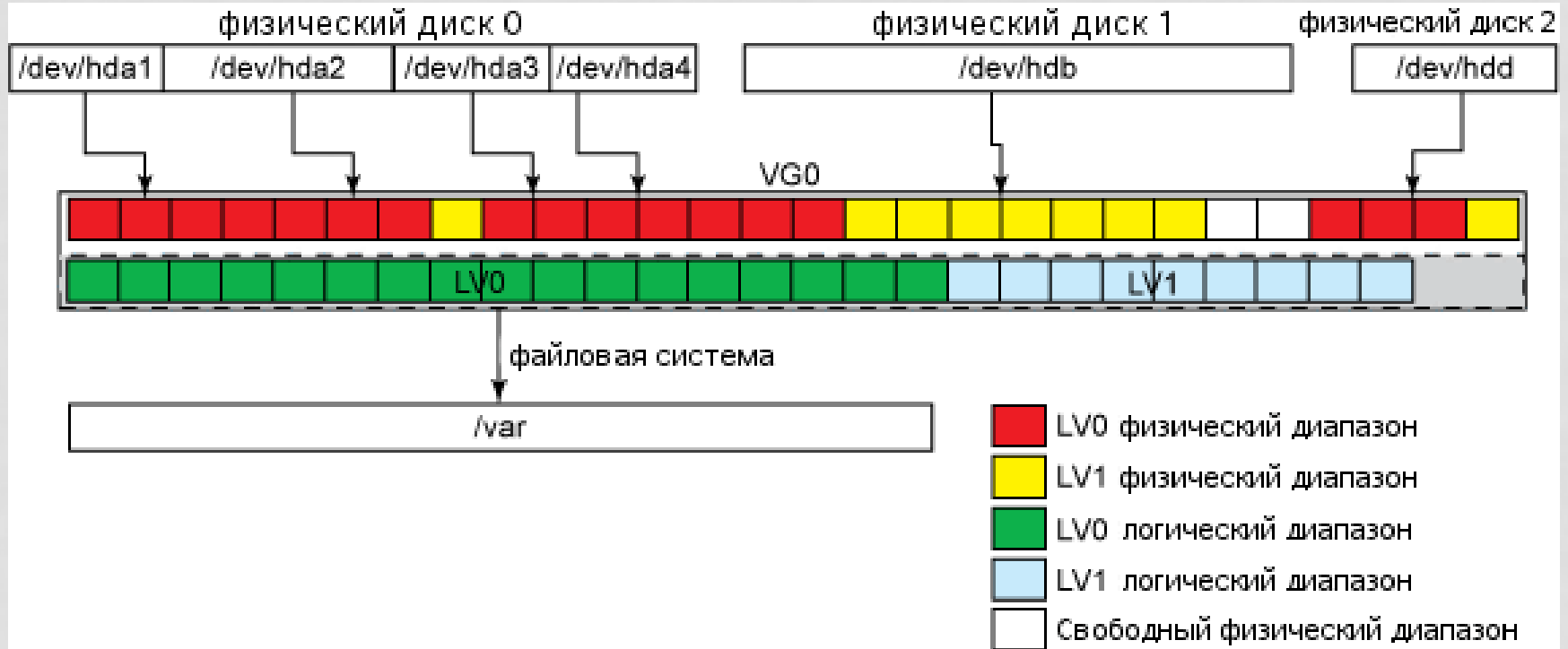
По умолчанию размер диапазона составляет 4 МБ, и для большинства конфигураций его нет необходимости изменять, поскольку производительность ввода/вывода не меняется при меньшем или большем размере диапазона.

Диапазоны



Обратите внимание на политику выделения диапазонов в примере на рисунке. LVM не всегда распределяет PE непрерывно;

Диапазоны



Модуль отображения устройств

Модуль отображения устройств

Модуль отображения устройств (также известный как `dm_mod`) является модулем ядра Linux (он может быть встроен в ядро), начиная с версии ядра 2.6.9. Его работа (как следует из названия) состоит в отображении устройств — это требование LVM.

Модуль отображения устройств

- При создании групп томов и логических томов вы можете дать им осмысленные имена (в отличие от предыдущих примеров, где использовались имена VG0, LV0 и LV1). Работа модуля отображения устройств заключается в корректном отображении этих имен на физические устройства. Если рассматривать предыдущие примеры, модуль отображения устройств создаст следующие узлы устройств в каталоге /dev файловой системы:
 - /dev/mapper/VG0-LV0
 - /dev/VG0/LV0 — это ссылка на вышеуказанное устройство
 - /dev/mapper/VG0-LV1
 - /dev/VG0/LV1 — это ссылка на вышеуказанное устройство
- В отличие от физических дисков, для группы томов нет возможности прямого доступа (это означает, что нет файла типа /dev/mapper/VG0 и нельзя использовать команду `dd if=/dev/VG0 of=dev/VG1`).

Примеры


Установка lvm

- Прежде всего для работы может понадобится установить пакет
- `sudo apt-get install lvm2`

Лирическое отступление


про apt-get

SO DO YOU, SANTIAGO, TAKE THIS
WOMAN AS YOUR WIFE ?



santi\$> apt-get
install wife_

```
santi $> apt-get install wife
#wife: The following dependencies
will be installed:
wife-house, wife-car, wife-friends,
wife-dog, wife-mother_in_law,
wife-no_more_TV_sports, wife-kid,
wife-kidlibs, wife-bricomaniplib.
#The following packages will be
upgraded:
ego-restrainer, freedom-throttler.
```



- Estimated installation
time: 47 years

Do you want to
continue? (Y/n)

!!!BUG-
ZILLA !!!



apt-get

- Команда apt-get - это консольный инструмент, который работает с пакетами АРТ, выполняющий функции:
 - установка новых программных пакетов,
 - обновление имеющихся пакетов,
 - обновления индекса списка пакетов
 - обновление все системы Ubuntu.

Команды apt-get

- Установка пакета
 - `sudo apt-get install nmap`
- Удаление пакета:
 - `sudo apt-get remove nmap`
- Обновление индекса пакетов:
 - `sudo apt-get update`
- Обновление пакетов:
 - `sudo apt-get upgrade`

Возвращаемся к lvm

- На каждом из дисков/разделов должен быть создан **физический том** (physical volume). К примеру, мы используем для LVM диск *sda* и раздел *sdb2*:
 - `pvcreate /dev/sda`
 - `pvcreate /dev/sdb2`
- На этих физических томах создаём **группу ТОМОВ**, которая будет называться, скажем, *vg1*:
 - `vgcreate vg1 /dev/sda /dev/sdb2`
- Посмотрим информацию о нашей группе ТОМОВ:
 - `vgdisplay vg1`

- Теперь в группе томов можно создать **логические тома** *lv1* и *lv2* размером 20 Гбайт и 30 Гбайт соответственно:
 - `lvcreate -n lv1 -L 20G vg1`
 - `lvcreate -n lv2 -L 30G vg1`
- Теперь у нас есть блочные устройства */dev/vg1/lv1* и */dev/vg1/lv2*.
- Осталось создать на них файловую систему. Тут различий с обычными разделами нет:
 - `mkfs.ext4 /dev/vg1/lv1`
 - `mkfs.reiserfs /dev/vg1/lv2`

Добавление физических томов

- Чтобы добавить новый винчестер sdc в группу томов, создадим физический том:
 - `pvcreate /dev/sdc`
- И добавим его в нашу группу:
 - `vgextend vg1 /dev/sdc`
- Теперь можно создать ещё один логический диск (`lvcreate`) или увеличить размер существующего (`lvresize`).

Удаление физических томов

- Чтобы убрать из работающей группы томов винчестер `sda` сначала перенесём все данные с него на другие диски:
 - `pvmove /dev/sda`
- Затем удалим его из группы томов:
 - `vgreduce vg /dev/sda`
- И, наконец, удалим физический том:
 - `pvremove /dev/sda`

Изменение размеров

- LVM позволяет легко изменять размер логических томов. Для этого нужно сначала изменить сам логический том:
 - `lvresize -L 40G vg1/lv2`
- а затем файловую систему на нём:
 - `resize2fs /dev/vg1/lv2`
 - `resize_reiserfs /dev/vg1/lv2`

Как можно пробовать

- Мы создадим виртуальные накопители и будем с ними работать. Например, можно создать 4 диска по 1 Гбайт, но можно создать другое количество большего или меньшего размера как вам хочется. Создаем сами файлы, имитирующие устройства:
 - `mkdir /mnt/sdc1/lvm`
 - `cd /mnt/sdc1/lvm`
 - `dd if=/dev/zero of=./d01 count=1 bs=1G`
 - `dd if=/dev/zero of=./d02 count=1 bs=1G`
 - `dd if=/dev/zero of=./d03 count=1 bs=1G`
 - `dd if=/dev/zero of=./d04 count=1 bs=1G`

Как можно пробовать

- Создаем loopback устройства из файлов:
 - `losetup -f --show ./d01`
 - `losetup -f --show ./d02`
 - `losetup -f --show ./d03`
 - `losetup -f --show ./d04`

- `pvcreate /dev/loop0`
- `pvcreate /dev/loop1`
- `pvcreate /dev/loop2`
- `pvcreate /dev/loop3`
- `vgcreate -s 32M vg /dev/loop0 /dev/loop1 /dev/loop2 /dev/loop3`
- `lvcreate -n first -L 2G vg`
- `lvcreate -n second -L 400M vg`
- ...