Software-RAID HOWTO

Linas Vepstas, linas@linas.org Переводчик: Максим Дзюманенко max@april.kiev.ua v0.54, 21 Ноября 1998г Дата перевода: 24 октября 2000г

RAID обозначает ”Redundant Array of Inexpensive Disks”, ”Массив недорогих дисков с избыточностью” - это путь создания быстрых и надежных дисковых систем из отдельных дисков. RAID может противодействовать отказам дисков, а также увеличивать производительность по сравнению с одиночным диском. Этот документ - tutorial/HOWTO/FAQ для пользователей расширения MD ядра Linux, соответствующих утилит, и их применения. Расширение MD программно реализует RAID-0 (striping), RAID-1 (mirroring), RAID-4 и RAID-5. Это значит - с MD не требуется специального оборудования или дисковых контроллеров для получения многих преимуществ RAID.

Содержание

Предисловие

Не смотря на то, что RAID разработан для увеличения надежности системы путем введения избыточности, он может создавать ложное чувство безопасности и уверенности, если используется неправильно. Эта ложная уверенность может привести к большой беде. В частности заметьте, что RAID разработан для защиты от *дисковых* отказов, а не отказов *питания* или ошибок *оператора*. Отказы питания, ошибки при разработке ядра, или ошибки оператора/администратора могут привести к невосстановимому повреждению данных!

RAID *не* заменяет подобающего резервирования Вашей системы. Знайте что Вы делаете, тестируйте, будьте осведомлены и сознательны!

Домашняя страница перевода - *http://dmv.webjump.com/HOWTOs/*. Обновленные версии, в первую очередь, появляются тут.

1 Введение

1. **В**: Что такое RAID?

   О: RAID означает "Redundant Array of Inexpensive Disks", - путь создания быстрых и надежных дисковых подсистем из отдельных дисков. В мире PC, "I"понимают как "Независимые"("Independent"), где маркетинговыми усилиями продолжают различать IDE и SCSI. В оригинальном понимании, "I"означает "Недорогие (Inexpensive) по сравнению мейнфреймом 3380 DASD, размером с холодильник". Хорошие дома выглядят дешевыми по сравнению с его монстрообразными устройствами, а бриллиантовые кольца - безделушками.

2. **В**: Что это за документ?

   О: Этот документ - tutorial/HOWTO/FAQ для пользователей MD расширения ядра Linux, соответствующих утилит, и их применения. Расширение MD программно реализует RAID-0 (striping), RAID-1 (mirroring), RAID-4 и RAID-5. Это означает, что для MD не требуется специального оборудования или дисковых контроллеров для достижения многих преимуществ RAID.

   Этот документ **НЕ** введение в RAID; вы должны искать его в другом месте.

3. **В**: Какие уровни RAID реализует ядро Linux?

   О: Striping (RAID-0) и линейное соединение являются частью 2.x серии ядер. Его код - продукт производственного качества; он хорошо понятен и хорошо поддерживается. Он используется в некоторых очень больших USENET серверах новостей.

   RAID-1, RAID-4 и RAID-5 - часть ядра 2.1.63 и выше. Для ранних 2.0.x и 2.1.x ядер, существуют патчи, которые реализуют эту функцию. Не считайте обязательным обновиться до ядра 2.1.63; обновление ядра - процесс трудный; *намного* проще пропатчить ранние ядра. Большинство пользователей RAID работают с 2.0.x ядрами, и там сфокусирована большая часть исторической разработки RAID. Текущие снимки - производственного качества; т.е. нет известных ошибок, но есть некоторые грубые места и не проверенные системные установки. Большое количество людей используют программный RAID в производственном окружении.

   Горячее восстановление на RAID-1 было представлено недавно (Август 1997) и должно рассматриваться как альфа качества. Горячее восстановление RAID-5 должно быть альфа качества сейчас.

   Предостережение о 2.1.x нестабильных ядрах: они менее стабильны во многих отношениях. Некоторые из новейших дисковых контроллеров (таких как Promise Ultra) поддерживаются только в 2.1.x ядрах. Однако, 2.1.x ядра часто меняются в части драйверов блочных устройств, в коде DMA и прерываний, в PCI, IDE и SCSI коде, и в драйверах дисковых контроллеров. Комбинация этих факторов в совокупности с дешевыми жесткими дисками и/или низкого качества кабелями могут привести к значительным неприятностям. Утилита `ckraid` как и `fsck` и `mount` создают значительную нагрузку на RAID подсистему. Это может привести к блокировке дисков при загрузке, где даже магическая alt-SysReq клавишная комбинация не сохранит день. Будьте осторожны с 2.1.x ядрами, и ожидайте проблем. Или вернитесь к 2.0.34 ядру.

4. **В**: Я использую старое ядро. Где я могу получить патчи?

О: Программный RAID-0 и линейный режим - присутствуют во всех версиях текущих ядер Linux. Патчи для программных RAID-1,4,5 имеются на <http://luthien.nuclecu.unam.mx/miguel/raid>. См. также квази-зеркало <ftp://linux.kernel.org/pub/linux/daemons/raid/> для патчей, инструментов и других интересных вещей.

5. **В**: Есть ли другие Linux RAID ссылки?

   О:

   - Общий обзор RAID: <http://www.dpt.com/uraiddoc.html>.
   - Общие опции Linux RAID: <http://linas.org/linux/raid.html>.
   - Последняя версия этого документа: <http://linas.org/linux/Software-RAID/Software-RAID.html>.
   - Linux-RAID архив почтовой переписки: <http://www.linuxhq.com/lnxlists/>.
   - Домашняя страница Linux Software RAID: <http://luthien.nuclecu.unam.mx/miguel/raid>.
   - Инструменты Linux Software RAID: <ftp://linux.kernel.org/pub/linux/daemons/raid/>.
   - Как установить linear/stripped Software RAID: <http://www.ssc.com/lg/issue17/raid.html>.
   - Bootable RAID mini-HOWTO: <ftp://ftp.bizsystems.com/pub/raid/bootable-raid>.
   - Root RAID HOWTO: <ftp://ftp.bizsystems.com/pub/raid/Root-RAID-HOWTO>.
   - Linux RAID-Geschichten: <http://www.infodrom.north.de/joey/Linux/raid/>.

6. **В**: Кто ответственен за этот документ?

   О: Linas Vepstas собрал это все вместе. Однако, большая часть информации, и некоторые фразы были предоставлены

   - Bradley Ward Allen <*ulmo@Q.Net*>
   - Luca Berra <*bluca@comedia.it*>
   - Brian Candler <*B.Candler@pobox.com*>
   - Bohumil Chalupa <*bochal@apollo.karlov.mff.cuni.cz*>
   - Rob Hagopian <*hagopiar@vu.union.edu*>
   - Anton Hristozov <*anton@intransco.com*>
   - Miguel de Icaza <*miguel@luthien.nuclecu.unam.mx*>
   - Marco Meloni <*tonno@stud.unipg.it*>
   - Ingo Molnar <*mingo@pc7537.hil.siemens.at*>
   - Alvin Oga <*alvin@planet.fef.com*>
   - Gadi Oxman <*gadio@netvision.net.il*>
   - Vaughan Pratt <*pratt@cs.Stanford.EDU*>
   - Steven A. Reisman <*sar@pressenter.com*>
   - Michael Robinton <*michael@bzs.org*>
   - Martin Schulze <*joey@finlandia.infodrom.north.de*>
   - Geoff Thompson <*geofft@cs.waikato.ac.nz*>
   - Edward Welbon <*welbon@bga.com*>
   - Rod Wilkens <*rwilkens@border.net*>

- Johan Wiltink *<j.m.wiltink@pi.net>*
- Leonard N. Zubkoff *<lnz@dandelion.com>*
- Marc ZYNGIER *<zyngier@ufr-info-p7.ibp.fr>*

2 Понимание RAID

1. **В**: Что такое RAID? Почему я всегда его использую?

   О: RAID - путь комбинирования нескольких дисков в одно целое для увеличения скорости и/или надежности. Существует несколько различных типов и реализаций RAID, каждый со своими преимуществами и недостатками. Например, помещая копию одинаковых данных на два диска (называется **зеркализация дисков**, или RAID уровня 1), скорость чтения может быть повышена поочередным считыванием с каждого диска зеркала. В среднем, каждый диск менее занят, т.к. он обрабатывает только половину операций чтения (для двух дисков), или 1/3 (для трех дисков), и т.д. В дополнение, зеркало может повышать надежность: если один диск выходит из строя, другой диск содержит копию данных. Различные пути комбинирования дисков в один, обозначаются **уровнями RAID**, могут обеспечить большую эффективность хранения, чем просто зеркализация, или могут изменить производительность по задержкам (времени доступа), или производительность пропускной способности (скорости передачи), для чтения или записи, в то же время поддерживается избыточность - это полезно для противодействия отказам. **Хотя RAID может защитить от отказа, он не защищает от ошибок оператора и администратора (человека), или потерь вызванных ошибками программ (возможно и ошибками собственно программной реализации RAID). Сеть изобилует трагическими историями о системных администраторах, которые неправильно устанавливали RAID, и потеряли все свои данные. RAID - не заменяет необходимость частого, регулярного планового резервного копирования.**

   RAID может быть реализован аппаратно, в виде специальных дисковых контроллеров, или программно, как модуль ядра который связывает низкоуровневый драйвер диска, и файловую систему, которая находится на нем. Аппаратный RAID - всегда "дисковый контроллер", - это устройство к которому могут подсоединяться диски. Обычно он представляет собой плату, которая вставляется в слот ISA/EISA/PCI/S-Bus/MicroChannel. Однако, некоторые RAID контроллеры - в виде ящика, который соединяется кабелями с используемым дисковым контроллером, и дисками. Меньшие из них помещаются в дисковой стойке; большие могут быть встроены в дисковый шкаф со своими собственными стойками и источником питания. Последние аппаратные RAID используют с последними и быстрейшими процессорами, что обеспечивает обычно лучшую общую производительность, несмотря на значительную цену. Это потому, что большинство RAID контроллеров поставляются с встроенными процессорами на борту и кеш-памятью, которые могут значительно разгрузить суммарную обработку главного процессора, насколько позволяет скорость поступления данных в большой кеш контроллера. Старые аппаратные RAID могут работать как "тормоз"когда используются с новейшими процессорами: вчерашние модные встроенные процессоры и кеш могут быть бутылочным горлышком, и их производительность часто превосходится чисто-программными RAID и новыми, но простыми в дру-

гих отношениях, дисковыми контроллерами. Аппаратные RAID могут иметь преимущество над чисто-программными RAID, если используют синхронизацию шпинделей дисков и знают позицию дисковых пластин относительно головок диска и желаемого дискового блока. Однако, большинство современных (дешевых) дисков не предоставляют эту информацию, во всяком случае, средства управления этим и т.о., большинство аппаратных RAID не имеет этих преимуществ. Аппаратные RAID различных производителей, версий и моделей обычно не совместимы: если RAID контроллер отказывает, он должен быть заменен на другой контроллер того-же самого типа. На момент написания (Июнь 1998), широкое разнообразие аппаратных контроллеров используется под Linux; однако, никакой из них, на текущий момент, не поставляется с утилитами конфигурации и управления, которые запускаются под Linux.

Software-RAID - набор модулей ядра, вместе с утилитами управления, которые реализуют чисто программный RAID, и не требуют необычной аппаратуры. Подсистема Linux RAID реализована в ядре, как уровень над низкоуровневыми драйверами дисков (для IDE, SCSI и Paraport устройств), и интерфейсом блочных устройств. Файловая система, будь то ext2fs, DOS-FAT, или другая, работает поверх блочного интерфейса. Программный RAID, по своей программной природе, склонен быть более гибким, чем аппаратная реализация. Обратная сторона этого - требуется больше процессорного времени, по сравнению с аппаратной реализацией. Конечно, цена не превзойденная. Кроме того программный RAID имеет одну важную отличительную особенность: он оперирует базируясь на разделах, где несколько отдельных дисковых разделов собираются вместе для создания разделов RAID. В этом отличие от большинства аппаратных решений RAID, которые объединяют вместе целые диски в массив. В аппаратных RAID, факт, что массив RAID - прозрачен для операционной системы, упрощает управление. В программном, гораздо больше конфигурационных опций и вариантов, что запутывает дело.

На момент написания (Июнь 1998), администрирование RAID под Linux далеко от простоты, и это лучше пробовать опытным системным администраторам. Теория функционирования сложна. Системные инструменты требуют модификации загрузочных скриптов. И восстановление дискового отказа непростая задача, и способствует ошибкам человека. RAID не для новичков, и полученный прирост в надежности и производительности, может запросто перевеситься излишней сложностью. Действительно, современные диски - невероятно надежны и современные процессоры и контроллеры вполне мощные. Вы можете более просто получить желаемую надежность и производительность купив диск высшего качества и/или быструю аппаратуру.

2. **В**: Что такое уровни RAID? Почему так много? Чем различаются?

О: Различные уровни RAID имеют различную производительность, избыточность, емкость, надежность и ценовые характеристики. Большинство, но не все, из уровней RAID предоставляют повышенную защиту от отказов диска. Из тех, которые предоставляют избыточность, RAID-1 и RAID-5 более популярны. RAID-1 предлагает лучшую производительность, в то же время RAID-5 применяется для более продуктивного использования имеющихся емкостей накопителей. Однако, настройка производительности - совсем иное дело, так как производительность зависит от множества различных факторов, от типа приложения, до размеров stripe-ов, блоков, и файлов. Более трудные аспекты настройки производительности откладываются до более поздних разделов этого HOWTO. Далее описывается разница между уровнями RAID в контексте реализации программного RAID в Linux.

- **RAID-linear** простое объединение разделов для создания большого виртуального раздела. Это применяется если у Вас несколько маленьких дисков, и Вы хотите создать один большой раздел. Это объединение не предлагает избыточности,

и фактически уменьшает общую надежность: если один из дисков выходит из строя, весь раздел выходит из строя.

- **RAID-1** так же называемый "зеркализацией"("mirroring"). Два (или более) раздела, все одинакового размера, каждый содержит точную копию всех данных, блок в блок. Зеркализация дает сильную защиту от отказов диска: если один диск отказывает, есть другой с точной копией данных. Зеркализация также может помочь увеличить производительность подсистемы ввода-вывода, так как запросы на чтение могут быть разделены между несколькими дисками. К несчастью, зеркализация также менее эффективна в смысле емкости: два зеркальных раздела могут вместить не больше данных, чем один раздел.
- **Striping** - базовая концепция всех других уровней RAID. stripe - непрерывная последовательность дисковых блоков. stripe может быть размером с один дисковый блок, или может состоять из тысяч. Устройства RAID разделяют содержащие их разделы дисков на stripe-ы; различные уровни RAID различаются в том, как они организуют stripe-ы, и как данные размещаются на них. Взаимодействие между размером stripe-ов, типичными размерами файлов в системе, и их положением на диске - определяет общую производительность подсистемы RAID.
- **RAID-0** подобна RAID-linear, исключая то, что компоненты разделов делятся на strip-ы и затем чередуются. Подобно RAID-linear, результат - один большой виртуальный раздел. Так же как и в RAID-linear, это не предполагает избыточности, и тоже уменьшает общую надежность: отказ одного диска ударит по всему. RAID-0 часто претендует на увеличение производительности по сравнению RAID-linear. Однако, это может быть или не быть справедливо, в зависимости от характеристик файловой системы, типичного размера файла по сравнению с размером stripe, и типа рабочей нагрузки. Файловая система `ext2fs` уже рассеивает файлы по разделу, стараясь минимально фрагментировать. Итак, на простейшем уровне, любой доступ может быть выполнен к одному из нескольких дисков, и таким образом, чередование stripe-ов по многим дискам предоставляет реальные преимущество. Однако, существует разница в производительности, она зависит от данных, рабочей загрузки, и размера stripe.
- **RAID-4** чередует stripe-ы подобно RAID-0, но требуется дополнительный раздел для размещения информации о четности. Четность используется для получения избыточности: если один из дисков отказывает, данные на оставшихся дисках могут быть использованы для воссоздания данных на отказавшем диске. Получаем N дисков с данными, и один диск с четностью, stripe четности вычисляется так - берется один stripe из каждого диска с данными, и XOR-ятся вместе. Итак, емкость (N+1)-дисков массива RAID-4 равна N, что намного лучше чем зеркализация (N+1) дисков, и почти так же хорошо, как RAID-0 на N. Заметьте, что для N=1, где один диск с данными, и один паритетный, RAID-4 эквивалентен зеркализации, при этом каждый из двух дисков копирует друг друга. Однако, RAID-4 **НЕ** дает производительности чтения зеркализации, и имеет пониженную производительность записи. По просту, это потому, что обновление паритета требует чтения старого паритета, перед тем, ка новый паритет может быть вычислен и записан. При большом количестве операций записи, паритетный диск может стать "бутылочным горлышком", т.к. каждая операция записи должна обращаться к паритетному диску.
- **RAID-5** освободжден от "бутылочного горлышка"при записи на RAID-4 размещением паритетных stripe вперемешку на каждом диске. Однако, производительность записи все еще не столь хороша, как при зеркализации, так как паритетный stripe все же должен быть считан и XOR-ен перед записью. Производительность чтения тоже не так хороша, как при зеркализации, так как, после этого, есть только одна копия данных, не две или более. Принципиальное преимущество RAID-5 над зеркализацией то, что он предоставляет избыточность и защиту

от отказа одного диска, в то же время предоставляет намного больше емкости, когда используется с тремя или более дисками.

- **RAID-2 и RAID-3** редко используются, и в некоторой степени стали устаревшими для современных дисковых технологий. RAID-2 подобен RAID-4, но размещает ECC информацию вместо паритетной. С тех пор как все современные диски реализуют ECC в себе, это предоставляет маленькую дополнительную защиту. RAID-2 может дать большую целостность данных, если пропало питание в процессе записи; однако, резервные аккумуляторы и чистое завершение работы могут дать ту же выгоду. RAID-3 подобен RAID-4, исключая то, что он использует наименьший возможный размер stripe. Как результат, любая операция чтения будет включать в себя все диски, делая перекрытие запросов ввода-вывода трудным/невозможным. Для избежания задержек при ожидания вращения, RAID-3 требует синхронизации всех шпинделей дисков. Большинство современных дисков не имеют способности синхронизировать шпиндели, или, если и имеют, не имеют нужных соединителей, кабелей, и документации производителей. Ни RAID-2 ни RAID-3 не поддерживаются драйверами программного RAID в Linux.
- **Прочие уровни RAID** определены различными исследователями и поставщиками. Многие из них представляют наложение одного типа raid поверх другого. Некоторые требуют специального оборудования, а другие защищены патентами. Нет единой схемы именования этих уровней. Иногда преимущества этих систем небольшие, или по крайне мере не проявляются пока система не слишком нагружена. Исключая размещение RAID-1 поверх RAID-0/linear, Программный RAID Linux не поддерживает никакие друге варианты.

3 Установка и установочные соображения

1. **В**: Как лучше сконфигурировать программный RAID?

   О: Я обнаружил, что планирование файловой системы одна из труднейших задач конфигурирования Unix. Для ответа на Ваш вопрос, я могу написать, что мы сделаем.
   Мы планируем следующую установку:

- два EIDE диска, 2.1Гб каждый.

```
диск раздел т.монтирования размер   устройство
  1      1       /         300M   /dev/hda1
  1      2       swap       64M   /dev/hda2
  1      3       /home     800M   /dev/hda3
  1      4       /var      900M   /dev/hda4

  2      1       /root     300M   /dev/hdc1
  2      2       swap       64M   /dev/hdc2
  2      3       /home     800M   /dev/hdc3
  2      4       /var      900M   /dev/hdc4
```

- Каждый диск на отдельном контроллере (и отдельном кабеле). Теоретически отказ контроллера и/или отказ кабеля не запретит доступ к обоим дискам. Также, мы возможно сможем получить повышение производительности от параллельных операций на двух контроллерах/кабелях.
- Установим ядро Linux в корневой (/) раздел `/dev/hda1`. Пометим этот раздел как загрузочный.
- `/dev/hdc1` должен содержать “холодную” копию `/dev/hda1`. Это НЕ raid копия, просто один-в-один копия. Только для использования в качестве восстановительного диска в случая отказа основного диска; пометим `/dev/hdc1` как загрузочный, и используем его для хранения без переустановки системы. Вы можете так-

же поместить копию `/dev/hdc1` ядра в LILO для упрощения загрузки в случае отказа.
Теоретически, в случае отказа, так я все еще могу загрузить систему вне зависимости от повреждения суперблока raid или других видов отказов и случаев, которые мне не понятны.

- `/dev/hda3` и `/dev/hdc3` будут зеркалами `/dev/md0`.
- `/dev/hda4` и `/dev/hdc4` будут зеркалами `/dev/md1`.
- мы выбрали `/var` и `/home` для зеркализации, и в разных разделах, основываясь на следующей логике:
  - `/` (корневой раздел) будет содержать относительно статическую, не изменяющуюся информацию: для всех практических применений, он должен быть только для чтения, без фактической отметки и монтирования только для чтения.
  - `/home` должен содержать "медленно изменяющиеся" данные.
  - `/var` должен содержать быстро изменяющиеся данные, включая спул почты, содержимое баз данных и логи web сервера.

  Идея использования нескольких отдельных разделов такова **если**, по некоторой странной причине, при ошибках человека, пропадении питания, или ошибках операционной системы происходят повреждения - они ограничиваются одним разделом. Типичный случай - исчезновение питания при записи на диск. Это должно привести к повреждению файловой системы, что должно быть исправлено программой `fsck` при следующей загрузке. Если даже `fsck` делает восстановление без создания дополнительных повреждений этим восстановлением, можно утешиться тем, что любые повреждения были ограничены одним разделом. В другом типичном случае системный администратор делает ошибку в процессе операции восстановления, что приводит к стиранию и разрушению всех данных. Разделы могут помочь ограничить влияние ошибок оператора.
- Разумно обдумать размещение разделов `/usr` или `/opt`. В общем, `/opt` и `/home` - лучший выбор для RAID-5 разделов, если есть еще диски. Предостережение: **НЕ** помещайте `/usr` в RAID-5 раздел. В случае серьезного отказа, вы можете обнаружить, что не можете примонтировать `/usr`, и необходимый набор утилит на нем (таких как сетевые утилиты или компилятор.) С RAID-1, если произошел отказ, и Вы не можете заставить RAID работать, Вы можете, по крайней мере, смонтировать одно из двух зеркал. Вы не можете сделать это с любым другим уровнем RAID (RAID-5, striping, или линейным соединением).

Итак, чтобы завершить ответ на вопрос:

- устанавливаем ОС на диск 1, раздел 1. Не монтируем любые другие разделы.
- устанавливаем по инструкции RAID.
- конфигурирует `md0` и `md1`.
- убеждаемся, что знаем что делать в случае отказа! Делаем ошибку администратора сейчас и не ждем реального кризиса. Эксперимент! (мы выключаем питание при дисковой активности — это нехорошо, но показательно).
- делаем несколько плохих mount/copy/unmount/rename/reboot для записи `/var` на `/dev/md1`. Делайте старательно, это не опасно.
- наслаждайтесь!

2. **В**: Какое различие между `mdadd`, `mdrun`, *и т.д.* командами, и `raidadd`, `raidrun` командами?

   О: Имена утилит сменились начиная с релиза 0.5 пакета raidtools. `md` схема именования использовалась в 0.43 и более старых версиях, в то время как `raid` используется в 0.5 и более новых версиях.

3. **В**: Я хочу запустить RAID-linear/RAID-0 на 2.0.34 ядре . Я не хочу применять raid патчи, так как они не нужны для RAID-0/linear. Где я могу взять raid-утилиты для управления?

   О: Это трудный вопрос, в самом деле, новый пакет raid утилит при сборке требует установленных патчей RAID-1,4,5. Я не знаю ни одной предкомпилированной двоичной версии raid утилит, которые доступны на текущий момент. Однако, эксперименты показывают, что бинарники raid утилит, когда скомпилированы с ядром 2.1.100, кажется хорошо работающими при создании RAID-0/linear раздела под 2.0.34. Смельчаки спрашивали об этом, и я **временно** поместил бинарники mdadd, mdcreate, и т.д. на http://linas.org/linux/Software-RAID/ Вы должны взять man страницы, и т. д. с обычного пакета утилит.

4. **В**: Могу ли я strip/зеркализировать корневой раздел (/)? Почему я не могу загружать Linux прямо с `md` диска?

   О: И LILO и Loadlin требуют не stripped/mirrored раздел для считывания образа ядра. Если Вы хотите strip/зеркализировать корневой раздел (/), вы должны создать не striped/mirrored раздел для хранения ядра(ядер). Обычно, этот раздел называют `/boot`. Тогда Вы должны либо использовать начальную поддержку виртуального диска(initrd), или патчи от Harald Hoyer <*HarryH@Royal.Net*> которые позволяют использовать stripped раздел, как корневой раздел. (Эти патчи - стандартная часть последних ядер серии 2.1.x)

   Существуют несколько подходов, которые могут быть использованы. Один подход детально документирован в Bootable RAID mini-HOWTO: <`ftp://ftp.bizsystems.com/pub/raid/bootable-raid`>.

   Как альтернативу, используйте `mkinitrd` для построения образа ramdisk, как показано ниже.

   Edward Welbon <*welbon@bga.com*> написал:

   - ... все, что нужно - скрипт для управления установкой. Для монтирования `md` файловой системы как корневой, главное - построить начальный образ файловой системы, который содержит необходимые модули и md утилиты для запуска `md`. У меня есть простой скрипт, который это делает.
   - Для загрузочной среды, у меня есть маленький **дешевый** SCSI диск (170MB я получил его за 20долларов). Этот диск работает на AHA1452, им также может быть недорогой IDE диск на родном IDE интерфейсе. От этого диска не требуется скорости, так как он предназначен, в основном, для загрузки.
   - На диске создана маленькая файловая система содержащая ядро и образ `initrd`. Начальной файловой системы должно хватать для загрузки модуля драйвера raid SCSI устройства и запуска raid раздела, который будет корневым. Тогда я делаю

     ```
     echo 0x900 > /proc/sys/kernel/real-root-dev
     ```

     (`0x900` для `/dev/md0`) и выхожу из `linuxrc`. Далее загрузка продолжается обычно.
   - Я собрал большинство функций как модули кроме драйвера AHA1452, который будит файловую систему `initrd`. Таким образом у меня очень маленькое ядро. Этот метод простой и надежный, я делаю так с 2.1.26 и никогда не было проблем, которых не мог бы запросто решить. Файловая система даже выжила несколько 2.1.4[45] тяжелых разрушений без реальных проблем.
   - В одно время у меня были размечены raid диски так, что начальные цилиндры первого raid диска содержали ядро и начальные цилиндры второго raid диска содержали образ начальной файловой системы, вместо этого я использовал начальные цилиндры raid дисков для подкачкм, так как они более быстрые цилиндры (зачем терять их на загрузку?).

- Хорошо иметь недорогой диск для загрузки, так как с него просто загрузиться и, при необходимости, можно использовать как восстановительный диск. Если Вы интересуетесь, Вы можете взглянуть на скрипт, который создает мой начальный образ ramdisk и потом запускает `LILO`.

  <http://www.realtime.net/welbon/initrd.md.tar.gz>

  Его достаточно для того, чтобы обрисовать картину. Он не очень хорош, и, конечно, можно создать более маленький образ файловой системы для начального ramdisk. Было бы проще создать его более действенным. Но он использует `LILO` как есть. Если вы сделаете любые усовершенствования, пожалуйста, отправьте копию мне. 8-)

5. **В**: Я слышал, что я могу запустить зеркализацию поверх striping. Это правда? Могу ли я запускать зеркализацию поверх петлевого устройства?

   О: Да, но не наоборот. Вы можете поместить stripe поверх нескольких дисков, и затем строить зеркализацию на базе этого. Однако, striping не может быть помещен на зеркало.

   Короткое техническое объяснение этого - особенностью linear и stripe является использования `ll_rw_blk` процедуры для доступа. `ll_rw_blk` процедура отображает дисковые устройства и сектора, но не блоки. Блочные устройства могут быть размещены одно поверх другого; но устройства, которые делают прямой, низкоуровневый доступ к дискам, такие как `ll_rw_blk`, не могут.

   На текущий момент (Ноябрь 1997) RAID не может быть создан на петлевом (loopback) устройстве, однако возможно, это скоро будет исправлено.

6. **В**: У меня есть два маленьких диска и три больших диска. Могу ли я соединить два маленьких диска в RAID-0, и затем создать RAID-5 из этого и больших дисков?

   О: Сейчас (Ноябрь 1997), для массива RAID-5, нет. Сейчас, это можно сделать только для RAID-1 повер объединенных дисков.

7. **В**: Какая разница между RAID-1 и RAID-5 для двух дисковой конфигурации (имеется в виду разница между массивом RAID-1 построенном на двух дисках, и массивом RAID-5 построенном на двух дисках)?

   О: Нет разницы в емкости. Также нельзя добавить диски ни в один из массивов для увеличения емкости (для деталей, смотрите вопрос ниже).

   RAID-1 предоставляет преимущество в производительности чтения: драйвер RAID-1 использует технологию распределенного чтения для одновременного чтения двух секторов, по одному с каждого устройства, это удваивает скорость считывания.

   Драйвер RAID-5, хотя и содержит много оптимизаций, сейчас (Сентябрь 1997) не реализует то, что паритетный диск - фактически зеркальная копия диска с данными. Таким образом, выполняется последовательное чтение данных.

8. **В**: Как я могу защититься от отказа двух дисков?

   О: Некоторые из алгоритмов RAID дают отказоустойчивость при отказе нескольких дисков, но на данный момент это не реализовано в Linux. Однако, программный RAID Linux может защитить от множественных отказов дисков размещая массив поверх массива. Например, девять дисков могут быть использованы для создания трех массивов raid-5. Затем, эти три массива могут быть объединены в один массив RAID-5. Фактически, этот тип конфигурации защищает от отказа трех дисков. Заметьте, что много дискового пространства "тратится" на избыточность информации.

```
Для NxN массива raid-5,
N=3, 5 из 9 дисков используется для паритета (=55%)
N=4, 7 из 16 дисков
N=5, 9 из 25 дисков
...
N=9, 17 из 81 дисков (=~20&процентов;)
```

В общем, массив MxN будет использовать M+N-1 дисков на паритет. Наименьшее количество пространства "теряется", когда M=N.

Другая альтернатива создать массив RAID-1 с тремя дисками. Заметьте, что все три диска содержат идентичные данные, и 2/3 пространства "теряется".

9. **В**: Я хочу понять, существует-ли что-то типа `fsck`: если раздел не был правильно демонтирован, `fsck` запускается и исправляет файловую систему более 90% времени. Так как машина способна исправлять это сама с помощью `ckraid --fix`, почему не автоматизировать это?

   О: Это возможно сделать добавлением следующие строки в `/etc/rc.d/rc.sysinit`:

   ```
   mdadd /dev/md0 /dev/hda1 /dev/hdc1 || {
       ckraid --fix /etc/raid.usr.conf
       mdadd /dev/md0 /dev/hda1 /dev/hdc1
   }
   ```

   или

   ```
   mdrun -p1 /dev/md0
   if [ $? -gt 0 ] ; then
           ckraid --fix /etc/raid1.conf
           mdrun -p1 /dev/md0
   fi
   ```

   Перед предоставлением более полного и надежного скрипта, рассмотрим теорию операций.

   Gadi Oxman написал: При неправильном завершении, Linux может быть в одном их следующих состояний:

   - При возникновении аварийного завершения дисковый кеш в памяти был синхронизирован с RAID набором; потерь данных нет.
   - При возникновении аварийного завершения в памяти дискового кеша было более новое содержимое, чем в RAID наборе; в результате повреждена файловая система и возможно потеряны данные. Это состояние может быть далее разделено на два других состояния:
     - При аварийном завершении Linux записывал данные.
     - При аварийном завершении Linux не записывал данные.

   Допустим мы используем массив RAID-1. В (2a), может случиться, что перед аварией небольшое количество блоков данных было успешно записано только на несколько из зеркал, таким образом при следующей загрузке, зеркала уже не будут идентичными.

   Если мы проигнорировали разницу в зеркалах, the raidtools-0.36.3 код балансировки чтения может выбрать для чтения блоки из любого зеркала, что приведет к противоречивому поведению (например, вывод `e2fsck -n /dev/md0` будет отличаться от запуска к запуску).

   Так как RAID не защищает от неправильного завершения, обычно нет никакого "целиком корректного" пути для устранения разницы в зеркалах и повреждений файловой системы.

Например, по умолчанию `ckraid --fix` будет выбирать содержимое первого действующего зеркала и обновлять другие зеркала. Однако, в зависимости от точного времени аварии, данные на другом зеркале могут быть более свежие, и мы можем пожелать использовать их как источник для восстановления зеркал, или, возможно, использовать другой метод восстановления.

Следующий скрипт реализует одну из самых здравых последовательностей загрузки. В частности, он принимает меры предосторожности длинными, повторяющимися `ckraid`-ов при не совместных дисках, контроллерах, или драйверах контроллеров дисков. Модифицируйте его, для соответствия своей конфигурации, и скопируйте в `rc.raid.init`. Затем вызовите `rc.raid.init` после проверки fsck-ом монтирования rw корневого раздела, но перед проверкой fsck-ом оставшихся разделов. Убедитесь, что текущий каталог в путях поиска (переменная PATH).

```
mdadd /dev/md0 /dev/hda1 /dev/hdc1 || {
    rm -f /fastboot             # force an fsck to occur
    ckraid --fix /etc/raid.usr.conf
    mdadd /dev/md0 /dev/hda1 /dev/hdc1
}
# if a crash occurs later in the boot process,
# we at least want to leave this md in a clean state.
/sbin/mdstop /dev/md0

mdadd /dev/md1 /dev/hda2 /dev/hdc2 || {
    rm -f /fastboot             # force an fsck to occur
    ckraid --fix /etc/raid.home.conf
    mdadd /dev/md1 /dev/hda2 /dev/hdc2
}
# if a crash occurs later in the boot process,
# we at least want to leave this md in a clean state.
/sbin/mdstop /dev/md1

mdadd /dev/md0 /dev/hda1 /dev/hdc1
mdrun -p1 /dev/md0
if [ $? -gt 0 ] ; then
    rm -f /fastboot             # force an fsck to occur
    ckraid --fix /etc/raid.usr.conf
    mdrun -p1 /dev/md0
fi
# if a crash occurs later in the boot process,
# we at least want to leave this md in a clean state.
/sbin/mdstop /dev/md0

mdadd /dev/md1 /dev/hda2 /dev/hdc2
mdrun -p1 /dev/md1
if [ $? -gt 0 ] ; then
    rm -f /fastboot             # force an fsck to occur
    ckraid --fix /etc/raid.home.conf
    mdrun -p1 /dev/md1
fi
# if a crash occurs later in the boot process,
# we at least want to leave this md in a clean state.
/sbin/mdstop /dev/md1

# OK, just blast through the md commands now.  If there were
```

```
# errors, the above checks should have fixed things up.
/sbin/mdadd /dev/md0 /dev/hda1 /dev/hdc1
/sbin/mdrun -p1 /dev/md0

/sbin/mdadd /dev/md12 /dev/hda2 /dev/hdc2
/sbin/mdrun -p1 /dev/md1
```

В дополнение к указанному, Вы должны создать `rc.raid.halt`, который должен выглядеть как показано ниже:

```
/sbin/mdstop /dev/md0
/sbin/mdstop /dev/md1
```

Модифицируйте оба `rc.sysinit` и `init.d/halt` для включения этого в место, где файловая система уже демонтирована при halt/reboot. (Заметьте что `rc.sysinit` демонтирует и перезагружает если `fsck` завершился с ошибкой.)

10. **В**: Могу я установить одну половину RAID-1 зеркала на один диск, который есть у меня сейчас и затем позже взять другой диск и просто его добавить?

    О: С текущими утилитами - нет, во всяком случае не простым способом. В частности, вы не можете просто скопировать содержимое одного диска на другой и затем их спаровать. Это потому, что драйвера RAID используют часть пространства в конце раздела для размещения суперблока. Это слегка уменьшает количество пространства, доступного для файловой системы; если Вы просто попробуете принудительно поставить RAID-1 на раздел с существующей файловой системой, raid суперблок перезапишет часть файловой системы и обрубит данные. Так как ext2fs файловая система разбрасывает фалы по разделу случайным образом (для избежания фрагментации), есть хороший шанс, что файл будет лежать в самом конце раздела перед окончанием диска.

    Если Вы сообразительны, я предлагаю Вам вычислить сколько места нужно под суперблок RAID, и сделать вашу файловую систему немного короче, оставив место на перспективу. Но тогда, если вы такой умный, Вы должны также быть способны модифицировать утилиты для автоматизации этого процесса. (Утилиты не так уж сложны).

    Заметка: Внимательный читатель заметит, что следующий трюк может сработать; я не пытался проверить это: Сделайте `mkraidc /dev/null`, как одним из устройств. Тогда `mdadd -rc` только одним, истинным диском (не делайте mdadd `/dev/null`). `mkraid` должен быть успешно создать raid массив, когда mdadd шаг выполняется - файловая система запущена в "деградированном"режиме, как если бы один из дисков отказал.

4 Восстановление ошибок

1. **В**: У меня установлен RAID-1 (зеркализация), и у меня пропало питание во время дисковой активности. Что мне теперь делать?

   О: Избыточность уровней RAID предназначена для защиты от отказа **диска**, не от отказа **питания**.

   Есть несколько путей восстановления в этой ситуации.

- Метод (1): Использовать raid утилиты. Он может быть использован для синхронизации raid массивов. Он не устраняет повреждение файловой системы; после синхронизации raid массивов, файловая система все еще нуждается в исправлении с помощью fsck. Raid массивы могут быть проверены `ckraid /etc/raid1.conf` (для RAID-1, или, `/etc/raid5.conf`, и т.д..)
  Запуск `ckraid /etc/raid1.conf --fix` выберет оин один диск из дисков массива (обычно первый), для использования его в качестве главной копии, и копирования его блоков на другие диски зеркала. Для обозначения диска, который должен быть использован как главный, вы можете использовать `--force-source` флаг: например, `ckraid /etc/raid1.conf --fix --force-source /dev/hdc3`. Комманда ckraid может быть безопасно запущена без опции `--fix` для проверки неактивного RAID массива без внесения изменений. Если Вы удовлетворены предполагаемыми изменениями, примените опцию `--fix` .
- Метод (2): Параноидальный, длительный по времени, не намного лучше, чем первый путь. Представим двух-дисковый массив RAID-1, состоящий из разделов `/dev/hda3` и `/dev/hdc3`. Вы можете попробовать следующее:
  (a) `fsck /dev/hda3`
  (b) `fsck /dev/hdc3`
  (c) Решите, который из двух разделов содержит меньше ошибок, или где проще восстановление, или на котором находятся нужные Вам данные. Выберите один, только один, для вашей новой "главной" копии. Предположим Вы выбрали `/dev/hdc3`.
  (d) `dd if=/dev/hdc3 of=/dev/hda3`
  (e) `mkraid raid1.conf -f --only-superblock`
  Вместо последних двух шагов, Вы можете запустить `ckraid /etc/raid1.conf --fix --force-source /dev/hdc3`, что будет быстрее.
- Метод (3): Версия для ленивых людей. Если Вы не хотите ждать завершения долгой проверки fsck, просто пропустите первые три шага выше, и начинайте прямо с последних двух шагов. Только после завершения запустите `fsck /dev/md0`. Метод (3) на самом деле замаскированный метод (1).

В любом случае, вышеуказанные шаги только синхронизируют массив raid. Для файловых систем возможно все еще необходимо устранение ошибок: для этого, нужно запустить fsck на активном, но не смонтированном устройстве.

С трех-дисковым массивом RAID-1, есть много вариантов, таких как использование двух дисков для выбора ответа. Утилиты автоматизации этого пока (Сентябрь 97) не существуют.

2. **В**: Если у меня установлен RAID-4 или RAID-5 (паритетный), и пропало питание во врем активности диска. Что мне делать?

   О: Избыточность уровней RAID предназначена для защиты от отказов **дисков**, а не от отказов **питания**.

   Так как диски в массиве RAID-4 или RAID-5 не содержат файловой системы, которую fsck может читать, есть несколько опций восстановления. Вы не можете использовать fsck для предварительной проверки и/или восстановления; Вы должны использовать сначала ckraid. Комманда `ckraid` может быть безопасно запущена без опции `--fix` для проверки неактивного массива RAID без внесения любых изменений. Когда Вы удовлетворены предложенными изменениями, примените опцию `--fix`.

   Если Вы хотите, Вы можете попробовать обозначить один из дисков как "отказавший диск". Делайте это с флагом `--suggest-failed-disk-mask`.

Только один бит должен быть установлен в флаге: RAID-5 не может восстанавливать два отказавших диска. mask - битовая маска: итак:

```
0x1 == первый диск
0x2 == второй диск
0x4 == третий диск
0x8 == четвертый диск, и т.д.
```

Или Вы можете выбрать модификацию секторов с паритетом, используя флаг `--suggest-fix-parity` flag. Это заново вычислит паритет из других секторов.

Флаг `--suggest-failed-dsk-mask` и `--suggest-fix-parity` может быть безопасно использован для проверки. Никаких изменений не будет сделано, если не указан флаг `--fix`. Итак, Вы можете экспериментировать с различными возможными вариантами восстановления.

3. **В**: Мое RAID-1 устройство, `/dev/md0` состоит из двух разделов жестких дисков: `/dev/hda3` и `/dev/hdc3`. Недавно, диск с `/dev/hdc3` отказал, был заменен на новый диск. Мой лучший друг, который не разбирается в RAID, сказал, что сейчас правильно сделать "`dd if=/dev/hda3 of=/dev/hdc3`". Я попробовал это, но все по прежнему не работает.

   О: Вы должны отстранить Вашего друга от компьютера. К счастью, не произошло никаких серьезных повреждений. Вы можете все восстановить запустив:

   ```
   mkraid raid1.conf -f --only-superblock
   ```

   При запуске `dd`, были созданы две идентичные копии раздела. Это почти правильно, исключая то, что расширение ядра RAID-1 предполагает различие в суперблоке. Итак, когда Вы пробуете активировать RAID, программа обратит внимание на проблему, и деактивирует один из двух разделов. Пересоздав суперблок, вы должны получить полностью рабочую систему.

4. **В**: У моей версии `mkraid` нет флага `--only-superblock`. Что мне делать?

   О: В новых утилитах убрали поддержку этого флага, заменив его флагом `--force-resync`. Как мне сообщили с последней версией утилит и программ работает такая последовательность:

   ```
   umount /web (что было смонтировано на /dev/md0)
   raidstop /dev/md0
   mkraid /dev/md0 --force-resync --really-force
   raidstart /dev/md0
   ```

   После этого, `cat /proc/mdstat` должно доложить `resync in progress`, и далее можно `mount /dev/md0` с этого места.

5. **В**: Мое устройство RAID-1, `/dev/md0` состоит из двух разделов: `/dev/hda3` and `/dev/hdc3`. Мой лучший друг(подруга?), который не разбирается в RAID, запустил без меня `fsck` на `/dev/hda3`, и сейчас RAID не работает. Что я должен делать?

   О: Вы должны пересмотреть свое понятие - "лучший друг". В общем, `fsck` не должна запускаться на отдельных разделах массива RAID. Предположим ни один из разделов тяжело не поврежден, потерь данных нет, и RAID-1 устройство может быть восстановлено так:

   (a) делаем резервную копию файловой системы на `/dev/hda3`

   (b) `dd if=/dev/hda3 of=/dev/hdc3`

   (c) `mkraid raid1.conf -f --only-superblock`

   Это должно вернуть ваше зеркало к работе.

6. **В**: Почему сказанное выше работает как восстанавливающая процедура?

   О: Потому что каждый компонент раздела в RAID-1 зеркале является просто точной копией файловой системы. В крайнем случае, зеркализация может быть запрещена, и один из разделов может быть смонтирован и безопасно запущен как обычная, не-RAID файловая система. Если Вы готовы перезапуститься используя RAID-1, то демонтируйте раздел, и следуйте вышеприведенным инструкциям для восстановления зеркала. Заметьте, что вышеуказанное работает ТОЛЬКО для RAID-1, и ни для какого другого уровня.

   Возможно, Вам удобнее изменить направление копирования: копировать **с** диска, который был нетронут **на** тот, который был. Просто будьте уверены, что после этого запускаете fsck на md.

7. **В**: Я смущен предыдущим вопросом, но еще не убежден. Безопасно ли запускать `fsck /dev/md0` ?

   О: Да, безопасно запускать `fsck` на устройствах `md`. Фактически, это **единственное** безопасное место запуска `fsck`.

8. **В**: Если диск медленно слабеет, будет очевидно который из них? Я беспокоюсь чтобы этого не случилось, и эта неразбериха не привела к каким-либо опасным решениям системного администратора.

   О: Как только диск откажет, драйвер нижнего уровня вернет код ошибки драйверу RAID. RAID драйвер пометит этот диск в суперблоках RAID хороших дисков как "плохой"(bad) (таким образом позже мы сможем узнать, которые из дисков хорошие, а которые нет), и продолжит работу RAID на оставшихся действующих зеркалах.

   Это, конечно, предполагает, что диск и драйвер нижнего уровня в состоянии обнаружить ошибки чтения/записи и не будут, к примеру, молча искажать данные. Это справедливо для текущих дисков (схем обнаружения ошибок используемых в них), и основы работы RAID.

9. **В**: Как насчет горячей замены?

   О: Работа далека от завершения "горячей замены". С этим свойством, можно добавить несколько "резервных" дисков в RAID набор (уровня 1 или 4/5), и как только диск отказал, он будет воссоздан на ходу на одном из резервных дисков, без необходимости остановки массива.

   Однако, для использования этого свойства, резервные диски должны быть определены на момент загрузки или должны добавляться на ходу, что требует использования специального шкафа и соединителей, которые позволяют добавлять диск при включенном питании.

   На Октябрь 97, доступна бета версия MD, которая позволяет:

   - RAID 1 и 5 восстановление на резервных дисках
   - RAID-5 восстановление паритета после неправильного завершения
   - добавление резервных дисков в уже работающий массив RAID 1 или 4/5

   По умолчанию, автоматическая реконструкция сейчас (Декабрь 97) запрещена по умолчанию, в основном по причине предварительного характера этой работы. Она может быть включена изменением значения `SUPPORT_RECONSTRUCTION` в `include/linux/md.h`.

   Если при создании в массиве сконфигурированы резервные диски и реконструкция в ядре включена, резервный диск уже будет содержать суперблок (записанный утилитой `mkraid`), и ядро будет реконструировать его содержимое автоматически (без необходимости шагов: вызов `mdstop`, замены диска, `ckraid`, `mdrun`).

Если Вы не запустили автоматическую реконструкцию, и не сконфигурировали диски с горячей заменой, рекомендуется процедура описанная Gadi Oxman <*gadio@netvision.net.il*> :

- Сейчас, как только один диск удален, набор RAID будет запущен в деградированном режиме. Для восстановления полноценного функционирования, Вы должны:
  - остановить массив (`mdstop /dev/md0`)
  - заменить отказавший диск
  - запустить `ckraid raid.conf` для реконструкции содержимого
  - запустить массив снова (`mdadd`, `mdrun`).

  Теперь массив будет запущен со всеми дисками, и снова будет защищен от отказа одного диска.

На текущий момент, не возможно назначить один резервный диск нескольким массивам. Каждый массив требует своего собственного резервного диска.

10. **В**: Хочу звуковую сигнализацию "один диск в зеркале неисправен", так как администратору-новичку узнать это проблематично.

    О: Ядро ведет протокол событий с "`KERN_ALERT`" приоритетом в syslog. Существует несколько программных пакетов, которые наблюдают за файлами syslog, и автоматически подают сигнал на PC динамик, звонят на пейджер, посылают почту, и т.д.

11. **В**: Как мне запустить RAID-5 в деградированном режиме (о одним отказавшим, и еще не замененным диском)?

    О: Gadi Oxman <*gadio@netvision.net.il*> пишет: Обычно, для запуска RAID-5 набора из n дисков вы должны:

    ```
    mdadd /dev/md0 /dev/disk1 ... /dev/disk(n)
    mdrun -p5 /dev/md0
    ```

    В случае, если один из дисков отказал, Вы все также должны `mdadd` их, как и при обычном запуске. (?? попробуйте использовать /dev/null вместо отказавшего диска ??? и посмотрите, что получится). После этого массив будет активен в деградированном режиме с (n - 1) диском. Если "`mdrun`" не удается, ядро фиксирует ошибку (например, несколько отказавших дисков, или неправильное завершение). Используйте "`dmesg`" для отображения сообщений об ошибках ядра от "`mdrun`". Если набор raid-5 поврежден исчезновением питания, в отличие от отказа диска, можно попробовать создать новый RAID суперблок:

    ```
    mkraid -f --only-superblock raid5.conf
    ```

    RAID массив не предоставляет защиту от отказа питания или краха ядра, и нельзя гарантировать корректное восстановление. Воссоздание суперблока приведет к игнорированию положения пометкой всех устройств, как "OK", как будто ничего не случилось.

12. **В**: Как работает RAID-5 при отказе диска?

    О: Типичный рабочий сценарий следующий:

    - RAID-5 массив активен.
    - Одно устройство отказывает во время активности массива.
    - Микропрограмма диска и низкоуровневые драйвера Linux диска/контроллера обнаруживают отказ и сообщают код ошибки MD драйверу.

- MD драйвер продолжает поддерживать безошибочную работу `/dev/md0` устройства для верхних уровней ядра (с потерей производительности) используя оставшиеся рабочие диски.
- Системный администратор может, как обычно, `umount /dev/md0` и `mdstop /dev/md0`.
- Если отказавшее устройство не заменено, системный администратор может запустить массив в деградированном режиме, запустив `mdadd` and `mdrun`.

13. **В**:

    О:

14. **В**: Почему нет вопроса номер 13?

    О: Если Вы заботитесь о RAID, Высокой надежности, и UPS, то, возможно, также хорошая мысль - быть суеверным. Это не повредит, не так ли?

15. **В**: Я только что заменил отказавший диск в массиве RAID-5. После пересоздания массива, `fsck` выдает очень много ошибок. Это нормально?

    О: Нет. И, если Вы запускали fsck не в "режиме только для чтения; без обновлений", вполне возможно, что у Вас повреждены даные. К несчастью, часто встречающийся сценарий - случайное изменение порядка дисков в массиве RAID-5, после замены диска. Хотя суперблок RAID содержит правильный порядок, не все утилиты используют эту информацию. В частности, текущая версия `ckraid` будет использовать информацию указанную в `-f` флаге (обычно, файл `/etc/raid5.conf`) вместо данных из суперблока. Если указанный порядок неверный, то замененный диск будет реконструирован неправильно. Симптом этого - многочисленные ошибки выдаваемые `fsck`.

    И, если вы удивлены, **да**, кое-кто потерял **все** свои данные из-за этой ошибки. **Настоятельно рекомендуется** сделать копию **всех** данных перед ре-конфигурированием RAID месива.

16. **В**: В QuickStart сказано, что `mdstop` только для того, чтобы убедиться, что диски синхронизированы. Это ДЕЙСТВИТЕЛЬНО необходимо? Не достачно де-монтирования файловой системы?

    О: Команда `mdstop /dev/md0` будет:

- помечать диски как "чистые". Это позволит нам обнаружить неправильное завершение, например из-за отказа питания или краха ядра.
- синхронизирует массив. Это менее важно после де-монтирования файловой системы, но важно если к `/dev/md0` был доступ не через файловую систему (например посредством `e2fsck`).

5 Поиск неисправностей при установке

1. **В**: Какой текущий стабильный патч для RAID в серии ядер 2.0.x?

   О: На 18 сентября 1997, это - "2.0.30 + pre-9 2.0.31 + Werner Fink's swapping patch + the alpha RAID patch". На Ноябрь 1997, это - 2.0.31 + ... !?

2. **В**: У меня не устанавливаются патчи RAID. Что не так?

О: Убедитесь, что `/usr/include/linux` символическая ссылка на `/usr/src/linux/include/linux`.

Убедитесь, что новые файлы `raid5.c`, и т.д. скопированы в свое правильное расположение. Иногда комманда patch не создает новые файлы. Попробуйте указать флаг `-f` для `patch`.

3. **В**: При компиляции raidtools 0.42, компиляция останавливается на include <pthread.h>, но его нет в моей системе. Как мне это исправить?

   О: raidtools-0.42 требует linuxthreads-0.6 с: <ftp://ftp.inria.fr/INRIA/Projects/cristal/Xavier.Leroy> Как альтернатива - glibc v2.0.

4. **В**: Я получаю сообщение: `mdrun -a /dev/md0: Invalid argument`

   О: Используйте `mkraid` для инициализации RAID перед первым использованием. `mkraid` стиранием разделов RAID гарантирует, что массив RAID изначально в согласованном состоянии. Дополнительно, `mkraid` создаст RAID суперблоки.

5. **В**: Я получил сообщение: `mdrun -a /dev/md0: Invalid argument`. Установки такие:

   - raid построен как модуль
   - была сделана обычная установочная последовательность ... mdcreate, mdadd, etc.
   - `cat /proc/mdstat` показывает

     ```
     Personalities :
     read_ahead not set
     md0 : inactive sda1 sdb1 6313482 blocks
     md1 : inactive
     md2 : inactive
     md3 : inactive
     ```

   - `mdrun -a` выдает сообщение об ошибке `/dev/md0: Invalid argument`

     О: Попробуйте `lsmod` (или, `cat /proc/modules`) чтобы увидеть, загружены ли raid модули. Если нет, Вы можете загрузить их явно командой `modprobe raid1` или `modprobe raid5`. Либо, если Вы используете автозагрузчик, и ожидаете, что их загрузит `kerneld`, а он этого не делает - это, возможно, из-за того что загрузчику не хватает информации для загрузки модулей. Отредактируйте `/etc/conf.modules` и добавьте следующие строки:

     ```
     alias md-personality-3 raid1
     alias md-personality-4 raid5
     ```

6. **В**: При `mdadd -a` я получаю ошибку: `/dev/md0: No such file or directory`. И действительно, вроде-бы нигде нет никакого `/dev/md0`. И что мне теперь делать?

   О: Пакет raid-tools должен создать устройства когда вы запускаете `make install` как root. Или Вы можете сделать следующее:

   ```
   cd /dev
   ./MAKEDEV md
   ```

7. **В**: После создания raid массива на `/dev/md0`, Я пытаюсь монтировать и получаю следующую ошибку: `mount: wrong fs type, bad option, bad superblock on /dev/md0, or too many mounted file systems`. Что не так?

О: Вы должны создать файловую систему на `/dev/md0` перед монтированием. Используйте `mke2fs`.

8. **В**: Truxton Fulton написал:

   На моем Linux 2.0.30, при выполнении `mkraid` на RAID-1 устройстве, при очистке двух отдельных разделов, я получил на консоль сообщения об ошибках `"Cannot allocate free page"`, и `"Unable to handle kernel paging request at virtual address ..."` ошибки в системном log-е. На этот момент, система стала совершенно не готова к использованию, но кажется восстановленной после этого. Работоспособность кажется восстановленной без прочих ошибок, и я успешно использую мой RAID-1. Однако ошибки смущают. Какие идеи?

   О: Это хорошо известная ошибка в ядре 2.0.30. Она устранена в 2.0.31 ядре; или откатитесь к 2.0.29.

9. **В**: Я не могу `mdrun` устройство RAID-1, RAID-4 или RAID-5. Если я пробую `mdrun` на добавленном посредством `mdadd` устройстве, я получаю сообщение `"invalid raid superblock magic"`.

   О: Проверьте, запускали ли Вы `mkraid` часть установочной процедуры.

10. **В**: Когда я обращаюсь к `/dev/md0`, ядро выплевывает кучу ошибок, таких как `md0: device not running, giving up !` и `I/O error...`. Я успешно добавил мои устройства в виртуальное устройство.

    О: для использования, устройство должно быть запущено. Используйте `mdrun -px /dev/md0` где x = l для линейного объединения, 0 для RAID-0 или 1 для RAID-1 и т.д.

11. **В**: Я создал линейное соединение из 2-х дисков. `cat /proc/mdstat` показывает общий размер устройства, но `df` показывает только размер первого физического устройства.

    О: Вы должны сделать `mkfs` на новом md-устройстве перед первым использованием, чтобы файловая система заняла все устройство.

12. **В**: Я установил `/etc/mdtab` используя mdcreate, я сделал `mdadd`, `mdrun` и `fsck` на двух моих разделах `/dev/mdX`. Все выглядит нормально перед перезагрузкой. Как только я перегрузился, я получаю ошибки `fsck` на обоих разделах: `fsck.ext2: Attempt to read block from filesystem resulted in short read while trying too open /dev/md0`. Почему?! Как это исправить?!

    О: В процессе загрузки, разделы RAID должны быть запущены перед проверкой `fsck`. Это должно быть сделано в одном из следующих скриптов. В одних дистрибутивах, `fsck` вызывается из `/etc/rc.d/rc.S`, в других - из `/etc/rc.d/rc.sysinit`. Добавьте в этот файл `mdadd -ar` *перед* запуском `fsck -A`. Еще лучше, чтобы запускался `ckraid`, если `mdadd` завершается с ошибкой. Как это сделать подробно обсуждается в вопросе 14 секции "Восстановление Ошибок".

13. **В**: Я получаю сообщение `invalid raid superblock magic` когда пытаюсь запустить массив, который состоит из разделов более 4Гб.

    О: Эта ошибка сейчас исправлена. (Сентябрь 97) Убедитесь, что у Вас свежий драйвер raid.

14. **В**: Я получаю сообщение `Warning: could not write 8 blocks in inode table starting at 2097175` когда пытаюсь запустить mke2fs на разделе более 2Гб.

O: Кажется это проблема `mke2fs` (Ноябрь 97). Временный выход - взять код mke2fs и добавить `#undef HAVE_LLSEEK` в `e2fsprogs-1.10/lib/ext2fs/llseek.c` прямо перед первым `#ifdef HAVE_LLSEEK`, затем пересобрать ядро.

15. **B**: `ckraid` сейчас не способен читать `/etc/mdtab`

    O: Формат конфигурационного файла RAID0/linear используемый в `/etc/mdtab` устарел, однако он будет поддерживаться некоторое время. Сейчас обновленные конфигурационные фалы называются `/etc/raid1.conf` и т.д.

16. **B**: Модули (`raid1.o`) не загружаются автоматически; но они загружаться через modprobe перед mdrun. Как я могу это устранить?

    O: Для автозагрузки модулей, мы можем добавить следующее в `/etc/conf.modules`:

    ```
    alias md-personality-3 raid1
    alias md-personality-4 raid5
    ```

17. **B**: Я добавил посредством `mdadd` 13 устройств, и теперь я пробую `mdrun -p5 /dev/md0` и получаю сообщение: `/dev/md0: Invalid argument`

    O: Конфигурация по умолчанию для программного RAID - 8 физических устройств. Отредактируйте `linux/md.h` изменив `#define MAX_REAL 8` на большее число, и пересоберите ядро.

18. **B**: Я не могу заставить работать `md` с разделами на последнем SPARCstation 5. Я думаю, что это связано с disk-labels.

    O: Sun disk-labels сидят в первом килобайте раздела. Для RAID-1, Sun disk-label не играют роли, так как `ext2fs` будет проскакивать метку на каждом зеркале. Для других raid уровней (0, линейного соединения и 4/5) это - проблема; (на Декабрь 97) они все еще не адресуются.

6 Поддерживаемая аппаратура и программы

1. **B**: У меня есть SCSI адаптер производителя XYZ (с несколькими каналами или без них), и диск(и) производителя(ей) PQR и LMN, будут ли они работать с md для создания linear/stripped/mirrored?

   O: Да! Программный RAID будет работать с любым дисковым контроллером (IDE или SCSI) и любыми дисками. Не должны быть идентичны ни диски, ни контроллеры. Например, зеркало RAID может быть создано с одной половиной зеркала на SCSI диске, и другой половиной на IDE диске. Диски могут не быть одного размера. Нет ограничений на смешивание и соответствие дисков и контроллеров.

   Это потому, что программный RAID работает с дисковыми разделами, не непосредственно с дисками. Одна рекомендация - для RAID уровней 1 и 5, разделы, используемые для одного набора, должны быть одного размера. Если для создания RAID 1 или 5 массивов используются неодинаковые разделы, то излишек пространства на более длинных разделах теряется (не используется).

2. **B**: У меня есть двухканальный BT-952, и на коробке написано, что он поддерживает аппаратно RAID 0, 1 и 0+1. У меня сделан RAID том из двух дисков, карта очевидно распознает их при запуске своей загрузочной процедуры BIOS. Я просмотрел исходный код драйвера, и не обнаружил ссылок на поддержку аппаратного RAID. Есть кто-либо, у кого это работает?

O: Mylex/BusLogic FlashPoint платы с RAIDPlus являются фактически программными RAID, а не аппаратными RAID. RAIDPlus поддерживается только Windows 95 и Windows NT, но не Netware или любой Unix платформой. Помимо загрузки и конфигурирования, в драйвере ОС, на самом деле, реализована поддержка RAID.

Даже если теоретически возможна поддержка Linux-ом RAIDPlus, реализация RAID-0/1/4/5 в ядре Linux намного более гибкая и должна быть большую производительность, так что мало причин поддерживать RAIDPlus непосредственно.

3. **B**: Я хочу запускать RAID с несколькими процессорами (SMP). RAID безопасен при SMP?

   O: "Я думаю, да лучший имеющийся ответ на время написания этого (Апрель 98). Много пользователей сообщают, что они без проблем используют RAID с SMP приблизительно год. Однако, на Апрель 98 (circa kernel 2.1.9x), были замечены следующие проблемы:

   - Драйвер Adaptec AIC7xxx SCSI не безопасен при SMP (Общая заметка: Adaptec адаптеры имеют длинную, продолжительную и вобще ветвистую историю проблем. Даже если они выглядят более доступными, широко распространенными и дешевыми SCSI адаптерами, необходимо их избегать. Учитывая факторы потери времени, разочарования, и поврежденных данных, Adaptec окажется самой дорогостоящей ошибкой, которую Вы когда-либо делали. Вобщем, если у Вас проблемы с SMP на 2.1.88, попробуйте патч ftp://ftp.bero-online.ml.org/pub/linux/aic7xxx-5.0.7-linux21.tar.gz Я не уверен будет ли этот патч вставлен в более поздние ядра 2.1.x. Для дальнейшей информации, взгляните на почтовые архивы для Марта 1998 на http://www.linuxhq.com/lnxlists/linux-raid/lr_9803_01/ Как обычно, из-за быстро-меняющейся природы последних ядер 2.1.x серии, проблемы описанные в этом почтовом списке могут быть или могут не быть устранены в то время, как Вы читаете этот документ.)
   - насколько известно IO-APIC с RAID-0 на SMP падает в 2.1.90

7 Модификация существующей инсталяции

1. **B**: Линейный MD расширяем? Могу я добавить новый раздел/диск, и увеличить размер существующей файловой системы?

   O: Miguel de Icaza <*miguel@luthien.nuclecu.unam.mx*> написал:

   Я изменил код ext2fs для способности поддерживать много устройств вместо обычного предположения - одно устройство на файловую систему. Итак, когда вы хотите расширить файловую систему, вы запускаете служебную программу, которая делает необходимые изменения на новом устройстве (вашем дополнительном разделе) и затем, Вы только говорите системе расширить файловую систему используя указанное устройство.

   Вы можете расширить файловую систему новым устройством во время работы системы, без остановки (и когда есть свободное время, вы можете удалить удалить устройства из ext2 набора, снова без необходимости переходить в однопользовательский режим или любого подобного хака).

   Вы можете получить патч для ядра 2.1.x с моей web страницы:

   <http://www.nuclecu.unam.mx/miguel/ext2-volume>

2. **B**: Могу я добавить диски в массив RAID-5?

   O: Сейчас (сентябрь 1997) - нет, нельзя без стирания всех данных. Утилита преобразования, позволяющая это делать еще не существует. Проблема в том, что структура и размещение массива RAID-5 зависит от количества дисков в массиве.

Конечно, вы можете добавить диски заархивировав массив на ленте, удалив все данные, создав новый массив, и восстановив данные с ленты.

3. **В**: Что случится с моим RAID1/RAID0 набором, если я сдвину одно устройство с `/dev/hdb` на `/dev/hdc`?

   Из-за глупых кабельно/корпусно/размерных дурацких решений, я создал мой RAID набор на одном IDE контроллере (`/dev/hda` и `/dev/hdb`). Сейчас я усвоил несколько вещей, я хочу перейти от `/dev/hdb` к `/dev/hdc`.

   Что должно случиться, если я просто сменю `/etc/mdtab` и `/etc/raid1.conf` файлы для отображения нового положенияе?

   О: Для RAID-0/linear, нужно только осторожно указать диски в том-же порядке. Тикам образом, в выше приведенном примере, если оригинальная конфигурация

   ```
   mdadd /dev/md0 /dev/hda /dev/hdb
   ```

   то новая конфигурация *должна* быть

   ```
   mdadd /dev/md0 /dev/hda /dev/hdc
   ```

   Для RAID-1/4/5, "RAID число" диска хранится в суперблоке RAID, и следовательно порядок в котором диски указаны не важен.

   RAID-0/linear не содержит суперблока из-за своего старого дизайна, и желания сохранить обратную совместимость со старым дизайном.

4. **В**: Могу я преобразовать двух-дисковое зеркало RAID-1 в трех-дисковый массив RAID-5?

   О: Да. Michael из BizSystems нашел хитрый способ сделать это. Однако фактически, подобно всем манипуляциям с RAID массивами содержащими данные, это опасно и склонно к человеческим ошибкам. **Перед началом сделайте резервную копию данных**.

```
Я сделаю следующие допущения:
-----------------------------------------
диски
изначально: hda - hdc
raid1 разделы hda3 - hdc3
имя массива /dev/md0

новые hda - hdc - hdd
raid5 разделы hda3 - hdc3 - hdd3
имя массива: /dev/md1

Вы должны заменить соответствующие номера дисков и разделов для Вашей
конфигурации системы. Это справедливо для всех примеров конфигурационных файлов.
-----------------------------------------
СДЕЛАЙТЕ РЕЗЕРВНУЮ КОПИЮ ПЕРЕД ТЕМ, КАК ЧТО-ЛИБО ДЕЛАТЬ
1) перекомпилируйте ядро для включения и raid1 и raid5
2) инсталируте новое ядро и проверьте наличие личных свойств raid
3) отключите избыточный раздел на массиве raid 1. Если это раздел смонтирован
как root (как у меня), Вы должны быть осторожны.

 Перегрузите компьютер без запуска raid устройств или загрузитесь с
восстановительной системы ( там должны иметься raid утилиты )
```

```
 запустите не избыточный raid1
mdadd -r -p1 /dev/md0 /dev/hda3

4) сконфигурируйте raid5 но с таким конфигурационным файлом, заметьте,
что здест нет записи hda3, а hdc3 повторяется. Это необходимо, так как
raid утилиты не хотят, чтобы Вы это делали.
-------------------------------
# конфигурация raid-5
raiddev                 /dev/md1
raid-level              5
nr-raid-disks           3
chunk-size              32

# Алгоритм размещения паритета
parity-algorithm        left-symmetric

# Резервные диски для горячей реконструкции
nr-spare-disks          0

device                  /dev/hdc3
raid-disk               0

device                  /dev/hdc3
raid-disk               1

device                  /dev/hdd3
raid-disk               2
---------------------------------------
 mkraid /etc/raid5.conf
5) активируйте массив raid5 в не избыточном режиме

mdadd -r -p5 -c32k /dev/md1 /dev/hdc3 /dev/hdd3

6) создайте файловую систему на массиве

mke2fs -b {blocksize} /dev/md1

рекомендуемый размер блока 4096, в отличие от стандартных 1024.
Это увеличит использование памяти для процедур kernel raid и
сделает размер блока равным размеру страницы. Так как у меня много небольших
файлов в моей системе, я использую компромиссное значение 2048.

7) смонтируйте где-то два устройства raid

mount -t ext2 /dev/md0 mnt0
mount -t ext2 /dev/md1 mnt1

8) переместите данные

cp -a mnt0 mnt1

9) проверьте идентичность данных
10) остановите оба массива
```

```
11) откорректируйте информацию в файле raid5.conf
  замените /dev/md1 на /dev/md0
  замените первый диск для чтения /dev/hda3

12) обновите новый массив до полного избыточного состояния
 (ЭТО РАЗРУШИТ ОСТАВШУЮСЯ НА raid1 ИНФОРМАЦИЮ)

ckraid --fix /etc/raid5.conf
```

8 Производительность, утилиты и общие ключевые вопросы

1. **В**: Я создал RAID-0 устройство на `/dev/sda2` и `/dev/sda3`. Устройство намного медленнее, чем отдельный раздел. md - это куча мусора?

   О: Для запуска устройства RAID-0 на полную скорость, у Вас должны разделы на разных дисках. Кроме того, помещая две половины зеркала на один диск Вы не получаете защиты от отказа диска.

2. **В**: Зачем использовать линейный RAID, если RAID-0 делает то же самое, но с лучшей производительностью?

   О: Не очевидно, что RAID-0 даст большую производительность; фактически, в некоторых случаях, он может сделать хуже. Файловая система ext2fs распределяет фалы по всему разделу, и пытается хранить все блоки файла вместе, в основном в целях избежания фрагментации. Таким образом, ext2fs ведет себя "как если бы"stripe-ы были (переменного размера) размером с файл. Если есть несколько дисков соединяются в один линейный RAID, это приведет к статистическому распределению файлов на оба диска. Таким образом, по крайней мере для ext2fs, линейный RAID работает во многом подобно RAID-0 с большим размером stripe. Наоборот, RAID-0 с маленьким размером stripe при одновременном доступе к нескольким большим файлам может вызвать излишнюю дисковую активность, приводящую к снижению производительности. Во многих случаях, RAID-0 может явно выигрывать. Например, представьте большой файл базы данных. Так как ext2fs пытается объединить вместе все блоки файла, велики шансы, что она заполнит только одно устройство при использовании линейного RAID, но будет разделять на много кусочков, при использовании RAID-0. Теперь представим несколько нитей (ядра) пытающихся получить произвольный доступ к базе данных. При линейном RAID, весь доступ пойдет на один диск, что не так эффективно, как параллельный доступ, создаваемый RAID-0.

3. **В**: Как RAID-0 обрабатывает ситуацию, где stripe-ы на различных разделах разного размера? stripe-ы распределяются однообразно?

   О: Для понимания этого, давайте рассмотрим пример с тремя разделами; 50Мб, 90Мб и 125Мб.

   Назовем D0 50Мб диск, D1 90Мб диск и D2 125Мб диск. Когда Вы запускаете устройство, драйвер вычисляет 'strip zones'. В этом случае, он найдет 3 зоны, определенные подобно этому:

```
Z0 : (D0/D1/D2) 3 x 50 = 150MB  всего в этой зоне
Z1 : (D1/D2)  2 x 40 = 80MB всего в этой зоне
Z2 : (D2) 125-50-40 = 35MB всего в этой зоне.
```

Вы можете видеть, что общий размер зон - размер виртуального устройства, но, в зависимости от зоны, striping различается. Z2 особенно неэффективна,так как там только один диск.

Так как `ext2fs` и большинство других файловых систем Unix распределяют файлы по всему диску, У Вас 35/265 = 13% шансов, что заполнение закончится на Z2, и не получится никаких преимуществ RAID-0.

(DOS пытается заполнить диск от начала до конца, и таким образом, старые файлы должны храниться на Z0. Однако, эта стратегия приводит к резкой фрагментации файловой системы, это причина того, что никто кроме DOS так не делает.)

4. **В**: У меня есть жесткий диск производителя X и контроллер производителя Y и я предполагаю использовать `md`. Это даст значительное увеличение производительности? Производительность в самом деле заметна?

   О: Ответ зависит от используемой Вами конфигурации.

   Производительность Linux MD RAID-0 и линейного RAID:
   Если система слишком загружена вводом-выводом, статистически, часть пойдет на один диск, а часть на другой. Таким образом, производительность увеличится по сравнению с одиночным диском. Фактическое увеличение сильно зависит от текущих данных, размера stripe, и других факторов. В системе с низким вводом-выводом, производительность эквивалентна производительности одного диска.

   Производительность чтения Linux MD RAID-1 (зеркализация):
   MD реализует балансировку чтения. То есть, код RAID-1 будет поочередно выбирать каждый из дисков (двух или более) зеркала, производя поочередное чтение с каждого диска. В случае небольшого ввода-вывода, это вовсе не изменит производительность: Вы будете ждать завершения чтения одного диска. Но, с двумя дисками и при высокой загрузке вводом-выводом, возможно получить практически удвоенную производительность, так как опреации чтения будут выполняться с каждого диска одновременно. Для N дисков в зеркале, это может увеличить производительность в N раз.

   Производительность записи Linux MD RAID-1 (зеркализация):
   Нужно ждать, пока запишутся данные на все диски зеркала. Это из-за того, что копия данных должна быть записана на каждый из дисков зеркала. Таким образом, производительность будет приблизительно эквивалентна производительности записи на один диск.

   Производительность чтения Linux MD RAID-4/5:
   Статистически, данный блок может быть на любом из дисков, и, таким образом, производительность чтения RAID-4/5 во многом подобна RAID-0. Она зависит от данных, размера stripe, и приложения. Она не будет так хороша, как производительность чтения в зеркальном массиве.

   Производительность записи Linux MD RAID-4/5:
   Она, в общем, должна быть предположительно меньше, чем у одного диска. Это из-за того, что на один диск должна быть записана информация о паритете, в то время как на другой - данные. Однако, в случае вычисления нового паритета, старый паритет и старые данные должны быть сначала считаны. Старые данные, новые данные и старый паритет должны быть объединены операцией XOR для определения новой информации о паритете: это требует циклов процессора и дополнительного доступа к дискам.

5. **В**: Какую конфигурацию RAID я должен использовать для оптимальной производительности?

   О: Ваша цель максимальная пропускная способность, или минимальные время доступа? Нет простого ответа, так как на производительность влияет много факторов:

- операционная система - будет один процесс/нить, выполнять доступ к диску или несколько?
- приложение - выполняется доступ к данным последовательно или с произвольно?
- файловая система - группируются файлы или рассредотачиваются (ext2fs группирует блоки файлов, и рассредотачивает сами файлы)
- драйвер диска - количество блоков упреждающего чтения (это настраиваемый параметр)
- CEC аппаратура - один дисковый контроллер или несколько?
- контроллер жесткого диска - может ли выполнять множество запросов или нет? Имеет ли кеш?
- Жесткий дик - размер памяти кеш-буфера – Достаточен ли будет для обработки размеров записей и желаемой частоты обращений?
- физическая организация - количество блоков в цилиндре – доступ к блокам на различных цилиндрах приведет к пере-позиционированию головки.

6. **В**: Какая оптимальная конфигурация RAID-5 для производительности?

   О: Так как RAID-5 создает загрузку ввода-вывода, которая одинаково распределена на несколько устройств, лучшая производительность будет получена, когда RAID набор сбалансирован использованием идентичных дисков, идентичных контроллеров, и одинаковым (небольшим) числом дисков на каждом контроллере.

   Однако заметьте, что использование идентичных компонент увеличивает возможность множества одновременных отказов, например из-за внезапного толчка или урона, перегрева, или скачка электричества во время грозы. Смешивание марок и моделей помогает минимизировать этот риск.

7. **В**: Какой оптимальный размер блока для массива RAID-4/5?

   О: Если используется текущая (Ноябрь 1997) RAID-4/5 реализация, строго рекомендуется создавать файловую систему с `mke2fs -b 4096` вместо 1024 байтов, по умолчанию.

   Это потому, что текущая реализация RAID-5 резервирует одну 4Кб страницу памяти на дисковый блок; если размер блока диска будет 1Кб, тогда 75% памяти, которую резервирует RAID-5 для осуществления ввода-вывода, не используется. Если размер блока диска совпадает с размером страницы памяти, тогда драйвер (потенциально) может использовать всю страницу. Итак, для файловой системы с размером блока 4096 в отличие от системы с размером блока 1024, драйвер RAID будет потенциально ставить в очередь 4 раза производя ввод-вывод с драйверам нижнего уровня без расходования дополнительной памяти.

   Заметка: пометки выше НЕ применимы драйверу программного RAID-0/1/линейного.

   Заметка: высказывание о 4Кб странице памяти применимо к архитектуре Intel x86. Размер страницы на Alpha, Sparc, и других процессорах различается; я думаю на Alpha/Sparc он 8Кб (????). Скорректируйте соответственно указанное значение.

   Заметьте: если на Вашей файловой системе много небольших файлов (файлов размером менее 10Кб), значительная часть дискового пространства может быть потеряна. Это из-за того, что файловая система распределяет дисковое пространство частями размером в блок. Выделение больших блоков маленьким файлам приводит к потерям дискового пространства: таким образом, Вы можете поставить небольшой размер блока, получить большую эффективность использования емкости, и не беспокоиться о "потерянной"памяти из-за несоответствия размера блока размеру страницы памяти.

   Заметка: большинство "типичных" систем не содержат много маленьких файлов. То есть, хотя могут быть тысячи небольших файлов, это будет приводить к потере

только от 10 до 100Мб, что, возможно, приемлимо, учитывая производительность, на много-гигабайтном диске.

Однако, для серверов новостей, может быть десятки и сотни тысяч небольших файлов. В этом случае, меньший размер блока, и таким образом сохраненная емкость, может быть более важной, чем более эффективный ввод-вывод.

Заметка: существует экспериментальная файловая система для Linux, которая пакует маленькие фалы и группы файлов в один блок. Она имеет большую производительность, если средний размер файла намного меньше размера блока.

Заметка: Будущие версии могут реализовать схемы, которые лишат смысла вышеприведенную дискуссию. Однако, это сложно реализовать, так как динамическое распределение на ходу может привести к мертвым-блокировкам (dead-locks); текущая реализация выполняет статическое предварительное выделение.

8. **В**: Как размер куска (размер stripe) влияют на производительность моего RAID-0, RAID-4 или RAID-5 устройства?

   О: Размер куска - количество смежных данных на виртуальном устройстве, которы смежные и на физическом устройстве. В этом HOWTO, "кусок"и "stripe"подразумевают одно и то же: что часто называется "stripe"в другой документации по RAID, в MD man страницах называется "кусок"("chunk"). Stripe-ы или куски применимы только к RAID 0, 4 и 5, так как stripe-ы не используются в зеркализации (RAID-1) и простом соединении (линейный RAID). Размеры stripe влияют на задержку, пропускную способность, и конкуренцию между отдельными операциями (возможность одновременного обслуживания перекрывающихся запросов ввода-вывода). Предполагая использование файловой системы ext2fs, и текущих правил ядра для упреждающего чтения, большие размеры stripe почти всегда лучше, чем маленькие размеры, и размеры stripe от почти четырех до полного цилиндра диска наилучшие. Чтобы понять это требование, рассмотрим эффективность больших stripe на маленьких файлах, и маленьких stripe на больших файлах. Размер stripe не влияет на производительность чтения на маленьких файлах: для массива из N дисков, файл имеет 1/N вероятность попасть целиком в один stripe на любой диск. Таким образом, и задержка и производительность чтения сравнима с чтением одного диска. Предположим, что маленькие файлы статистически хорошо распределяются по файловой системе, (и, на файловой системе ext2fs, они дожны), грубо в N раз более упорядочены, конкурентные чтения должны быть возможны без значительных коллизий между ними. Наоборот, если используются очень маленького размера stripe-ы, и последовательно читается большой файл, то чтение будет выдаваться всем дискам массива. Для чтения одного большого файла, задержка будет почти двойная, так как увеличивается вероятность нахождения блока в трех четвертях оборота диска или далее. Однако заметьте аргумент: пропускная способность может увеличиться почти в N раз для чтения одного большого файла, так как N дисков могут читать одновременно (то есть, если используется упреждающее чтение, то все диски остаются активными). Но есть другй контр-аргумент: если все диски уже заняты чтением файла, то попытки одновременного чтения второго или третьего файла приведут к значительной борьбе, разрушив производительность, так как алгоритмы управления диском будут двигать головками вдоль пластины. Таким образом, большие stripe-ы будут почти всегда приводить к большей производительности. Единственное исключение - случай, при использовании хорошего алгоритма упреждающего чтения, где один поток в одно время читает один большой файл, и он требует наивысшей возможной производительности. В этом случае желательны небольшие stripe-ы.

   Заметьте, что этот HOWTO ранее рекомендовал небольшие размеры stripe-ов для спула новостей или других систем с множеством мелких файлов. Это плохой совет, и вот почему: спулы новостей содержат не только много маленьких файлов, но также и большие суммарные файлы, также как и большие каталоги. Если суммарный

файл более одного stripe, его чтение задействует много дисков, замедляя все, так как каждый диск выполняют позиционирование. Подобным образом, текущая файловая система ext2fs просматривает каталоги в линейной, последовательной манере. Таким образом, чтобы найти данный файл или inode, в средней части будет прочитана половина каталога. Если этот каталог простирается на несколько stripe-ов (несколько дисков), чтение каталога (такое как при команде ls) будет очень медленным. Спасибо Steven A. Reisman <*sar@pressenter.com*> за эту поправку. Steve также добавил следующее:

> Я обнаружил, что использование 256k stripe дает намного лучшую производительность. Я подозреваю, что оптимальный размер должен быть размером с цилиндр диска (или, может быть размером с кеш диска). Однако, современные диски содержат зоны с различным количеством секторов (и размер кеша варьируется в зависимости от модели диска). Невозможно гарантировать, что stripe-ы не будут пересекать границу цилиндра.

Утилиты позволяют задавать размер в Кбайтах. Вы можете указать его величиной с размер страницы Вашего CPU (4Кб на x86).

9. **В**: Каков правильный stride при создании файловой системы ext2fs на разделе RAID? Под stride я подразумеваю -R флаг в команде `mke2fs`:

```
mke2fs -b 4096 -R stride=nnn  ...
```

Кокое должно быть значение nnn?

О: Флаг `-R stride` используется, чтобы указать файловой системе размер RAID stripe-ов. Так как только RAID-0,4 и 5 использует stripe-ы, а RAID-1 (зеркализация) и линейный RAID не используют, этот флаг применим только к RAID-0,4,5.

Знание размера stripe-а позволяет `mke2fs` выделять блок и битовый поля inode так, что они не все хранятся на одном физическом устройстве. Неизвестный помощник написал:

> Прошлой весной я заметил, что один диск из пары всегда больше занят вводом-выводом, и отследил - это из-за этих блоков мета-данных. Ted добавил опцию `-R stride=`, в мой вариант ответа и предложение обходного варианта.

Для файловой системы с блоком 4Кб, с размером stripe в 256Кб, нужно использовать `-R stride=64`. Если Вы не доверяете флагу `-R`, Вы можете получить подобный эффект другим путем. Steven A. Reisman <*sar@pressenter.com*> написал:

> Другое соображение - файловая система используемая на устройстве RAID-0. Файловая система выделяет ext2 8192 блоков в группу. У каждой группы есть свой набор inode-ов. Если есть 2, 4 или 8 дисков, эти inode скапливаются на первом диске. Я распределили inode-ы по всем дискам, указав mke2fs выделять только 7932 блоков на группу.

Некоторые страницы mke2fs не описывают флаг `[-g blocks-per-group]` используемый при этой операции.

10. **В**: Где в загрузочных скриптах я могу вставить команду `md`, так, чтобы все автоматически стартовало в процессе загрузки?

О: Rod Wilkens <*rwilkens@border.net*> написал:

> Вот что я сделал: вставил "`mdadd -ar`" в "`/etc/rc.d/rc.sysinit`" прямо после загрузки модулей, и перед проверкой дисков "`fsck`". Таким образом, Вы можете вставить устройство "`/dev/md?`" в "`/etc/fstab`". Затем вставить "`mdstop -a`" прямо после де-монтирования всех дисков "`umount -a`", в файле "`/etc/rc.d/init.d/halt`".

Для raid-5, Вы должны посмотреть на код возврата `mdadd`, и если он ошибочен, сделать

```
ckraid --fix /etc/raid5.conf
```

для восстановления любых повреждений.

11. **В**: Меня интересует возможно ли установить striping для более, чем 2 устройств в `md0`? Это для сервера новостей, и у меня есть 9 дисков... Нужно ли говорить, что мне нужно больше, чем два. Это возможно?

    О: Да. (описать как это сделать)

12. **В**: Когда программный RAID превосходит аппаратный RAID?

    О: Обычно, аппаратный RAID считается производительнее программного RAID, так как аппаратные контроллеры, часто содержат большой кеш, и могут лучше выполнять планирование параллельных операций записи. Однако, интегрированный программный RAID может (и дает) определенное преимущество при реализации в операционной системе.

    Например, ... мммм. Мы обходим молчанием темное описание кеширования реконструированных блоков в буферный кеш ...

    На дуальных PPro SMP системах, мне рассказывали, что производительность программного RAID превышала производительность плат аппаратного RAID известных производителей с кратностью от 2 до 5 раз.

    Программный RAID также очень интересная опция для избыточных серверных систем высокой готовности. В такой конфигурации, два CPU подсоединены к одному набору SCSI дисков. Если один север рушится или отказывается отвечать, то другой сервер может `mdadd`, `mdrun` и `mount` массив программного RAID, и продолжить работу. Этот режим работы не всегда возможен с многими аппаратными RAID контроллерами, из-за состояний конфигурации которое аппаратные контроллеры могут поддерживать.

13. **В**: Если я обновляю версию моих raidtools, приведет ли это к проблемам манипулирования старых raid массивами? Коротко, должен ли я пересоздать мои массивы RAID при обновлении raid утилит?

    О: Нет, по крайней мере до смены старшего номера версии. MD версия x.y.z состоит из трех подверсий:

    ```
    x:      старший номер версии.
    y:      младший номер версии.
    z:      номер уровня патча.
    ```

    Версия x1.y1.z1 драйвера RAID поддерживает RAID массив с версией x2.y2.z2 в случае (x1 == x2) и (y1 >= y2).

    Различные номера патчей (z) для тех же (x.y) версий разработаны по большей мере совместимыми.

    Младший номер версии увеличивается всякий раз, когда код RAID массива изменяется таким образом, что он несовместим с старыми версиями драйвера. Новые версии драйвера должны поддерживать совместимость с старыми RAID массивами.

    Старший номер версии увеличивается, если более не имеет смысла поддерживать старые RAID массивы в новом коде ядра.

    Для RAID-1, не правдоподобно, чтобы ни дисковый уровень ни структура суперблока изменились в ближайшее время. Скорее всего любые оптимизации и новые свойства (реконструкция, многопоточные утилиты, горячая замена, и т.п.) не отразятся на физическом размещении.

14. **В**: Комманда `mdstop /dev/md0` говорит, что устройство занято.

    О: Есть процесс, который держит открытым файл на `/dev/md0`, или `/dev/md0` все еще смонтирован. Завершите процесс или `umount /dev/md0`.

15. **В**: Существуют утилиты измерения производительности?

    О: Существует новая утилита, называемая `iotrace` в каталоге `linux/iotrace`. Она читает `/proc/io-trace` и анализирует/строит графики из его вывода. Если Вы чувствуете, что блочная производительность Вашей системы слишком низкая, просто посмотрите на вывод iotrace.

16. **В**: Я читал исходники RAID, и видел, что там определено значение `SPEED_LIMIT` равное 1024Кб/сек. Что это значит? Это ограничивает производительность?

    О: `SPEED_LIMIT` используется для ограничения скорости реконструкции RAID при автоматической реконструкции. По существу, автоматическая реконструкция позволяет Вам `e2fsck` и `mount` сразу после неправильного завершения, без предварительного запуска `ckraid`. Автоматическая реконструкция также используется после замены отказавшего диска.

    Для избежания подавления системы при реконструкции, нить реконструкции контролирует скорость реконструкции и уменьшает ее, если она слишком высока. Предел 1Мб/сек был выбран как разумная норма, которая позволяет реконструкции завершаться умеренно быстро, при создании только небольшой нагрузки на систему, не мешая другим процессам.

17. **В**: Как насчет "синхронизации шпинделей"или "дисковой синхронизации"?

    О: Синхронизация шпинделей используется для поддержания вращения нескольких дисков с одинаковой скоростью, так что пластины дисков всегда точно выровнены. Это используется некоторыми аппаратными контроллерами для лучшей организации записи на диски. Однако, в программном RAID, эта информация не используется, и синхронизация шпинделей может даже снизить производительность.

18. **В**: Как я могу установить пространства для подкачки используя raid 0? Должна ли быть striped подкачка на 4+ дисках быть быстрой?

    О: Leonard N. Zubkoff отвечает: Да она действительно быстра, но Вам не нужно использовать MD для получения striped подкачки. Ядро автоматически разделяет подкачку по нескольким пространствам подкачки с одинаковым приоритетом. Например, следующие записи из `/etc/fstab` разделяют подкачку по пяти дискам в три группы:

```
/dev/sdg1       swap    swap    pri=3
/dev/sdk1       swap    swap    pri=3
/dev/sdd1       swap    swap    pri=3
/dev/sdh1       swap    swap    pri=3
/dev/sdl1       swap    swap    pri=3
/dev/sdg2       swap    swap    pri=2
/dev/sdk2       swap    swap    pri=2
/dev/sdd2       swap    swap    pri=2
/dev/sdh2       swap    swap    pri=2
/dev/sdl2       swap    swap    pri=2
/dev/sdg3       swap    swap    pri=1
/dev/sdk3       swap    swap    pri=1
/dev/sdd3       swap    swap    pri=1
/dev/sdh3       swap    swap    pri=1
/dev/sdl3       swap    swap    pri=1
```

19. **В**: Я хочу получить максимальную производительность. Я должен использовать несколько контроллеров?

    О: Во многих случаях, ответ - да. Используя несколько контроллеров для параллельного доступа к дискам увеличивает производительность. Однако, действительное приращение фактически зависит от вашей конфигурации. Например, как сообщили (Vaughan Pratt, Январь 98) что один 4.3Гб Cheetah подключенный к Adaptec 2940UW может дать до 14Мб/сек (без использования RAID). Установив два диска на один контроллер, и используя конфигурацию RAID-0 привело к увеличению производительности до 27 Мб/сек.

    Заметьте,что 2940UW контроллер - "Ultra-Wide"SCSI контроллер, теоретически способный к пакетным передачам 40Мб/сек, так что указанные измерения не неожиданность. Однако, более медленный контроллер подключенный к двум быстрым дискам будет бутылочным горлышком. Также заметьте, что большинство внешних SCSI подключений (таких как секции с лотками горячей замены) не могут работать на 40Мб/сек, из-за проблем с кабелями и электрических шумов.

    Если Вы разрабатываете систему с несколькими контроллерами, помните, что большинство дисков и контроллеров в среднем работает на 70-85% их максимальной скорости.

    Также заметьте, что использование одного контроллера на диск может, по всей вероятности, уменьшить простой системы из-за отказа контроллера или кабеля (теоретически – только в случае правильной обработки драйвером отказа контроллера. Не все драйвера SCSI представляются способными обрабатывать эту ситуацию без паники или иных блокировок).

9 Высокая готовность RAID

1. **В**: RAID может помочь мне противостоять потерям данных. Но как я могу быть уверенным, что система работает, настолько долго, насколько возможно, и не склонна к поломкам? В идеале, я хочу, чтобы система работала 24 часа в день, 7 дней в неделю, 365 дней в году.

   О: Высокая готовность дело трудное и дорогостоящее. Тяжело Вам пробовать сделать систему отказоустойчивой, тяжело и еще более дорого ее получить. Следующие подсказки, советы, идеи и не проверенные слухи могут помочь Вам в этом.

   - IDE диски могут отказать так, что отказавший диск на на IDE шлейфе будет препятствовать работе хорошего диска на том же кабеле, таким образом это выглядит как отказ двух дисков. Так как RAID не защищает от отказа двух дисков, нужно либо подключать по одному диску на IDE шлейф, или, если два диска, то они должны относиться к различным RAID томам.
   - SCSI диски могут отказать так, что отказавший диск на SCSI цепочке может мешать работе любого устройства в цепочке. Режим отказа приводит к замыканию общего (разделяемого) контакта готовности устройства; так как этот контакт общий, не может быть выполнен никакой арбитраж, пока замыкание не устранено. Таким образом, два диска в одной цепочке SCSI не должны относиться к одному RAID массиву.
   - Подобные замечания применимы к контроллерам дисков. Не подгружайте каналы к одному контроллеру; используйте несколько контроллеров.
   - Не используйте все диски одного производителя или модели. Не так уж редко сильная гроза отключала два или более диска. (Да, мы все используем подавители бросков напряжения, но они вовсе не совершенны). Другими убийцами являются жара и плохое охлаждение диска. Дешевые диски часто работают перегретыми. Использование различных марок дисков и контроллеров уменьшает вероятность

того, что что бы то ни было отключит один диск (перегрев, физический удар, вибрация, бросок электричества) и одновременно повредит другие диски.

- Для защиты от отказа контроллера или процессора, можно построить окружение SCSI диска с "двумя хвостами", т.е. подсоединенным к двум компьютерам. Один компьютер монтирует файловую систему в режиме чтение-запись, а другой монтирует ее в как только-чтение, и действует как горячий резерв. Когда горячий резерв, определяет, что главный компьютер отказал (с помощью watchdog), он выключает питание главного (для гарантии, что он в самом деле не работает), и затем выполняет fsck и перемонтируется в режиме чтение-запись. Если кто-либо запустит это, дайте мне знать.
- Всегда используйте UPS, и выполняйте правильное завершение. Хотя неправильное завершение может не повредить диски, запуск ckraid даже на маленьких массивах мучительно медленный. Вы должны избегать запуска ckraid насколько это возможно. Или Вы можете покопаться в ядре и запустить отлаживаемый код горячей реконструкции ...
- SCSI кабеля, как известно, очень темпераментные создания, и склонны создавать всяческие проблемы. Используйте кабеля наивысшего качества, из тех которые можно купить. Используйте так называемый bubble-wrap, чтобы убедиться, что ленточные кабеля не слишком близко расположены и не влияют друг на друга. Тщательно соблюдайте ограничения на длину.
- Взгляните на SSI (Serial Storage Architecture). Хотя она несколько дорога, по слухам менее склонна к ошибочным режимам, чем SCSI.

10 Вопросы ожидающие ответов

1. **B**: Если, по соображениям стоимости, я пробую зеркало состоящее из медленного диска и быстрого диска, достаточно ли S/W силен для балансировки чтения или это замедлит скорость до скорости медленного?

2. **B**: Для тестирования "чистой"производительности диска... Существует ли символьное устройство для прямого(raw) чтения/записи вместо `/dev/sdaxx`, которое мы можем использовать для оценки производительности на raid устройствах?? Существует ли GUI утилита для наблюдения дисковой производительности??

11 Список пожеланий для MD и сопутствующего ПО

Bradley Ward Allen <*ulmo@Q.Net*> написал:

Идеи включают:

- Параметры загрузки для указания ядру какие устройства - MD устройства (без “`mdadd`")
- Сделать MD прозрачным для “`mount`"/“`umount`" без использования “`mdrun`" и “`mdstop`"
- Интегрировать “`ckraid`" в ядро, и запускать его при необходимости

(Итак в общем, все, что я могу предложить - избавиться от утилит и поместить их в ядро; так я это вижу, это - файловая система, а не игрушка.)

- Работа с массивами, которые могут свободно пережить одновременный, или в разное время, отказ N дисков, где N - целое > 0 устанавливаемое администратором
- Лучшая обработка застывания ядра, отключения питания, и других внезапных завершений

- Не отключать целый диск, если только часть его отказала, если ошибки секторов занимают менее 50% доступа при 20 попытках различных обращений, то просто продолжаем игнорируя эти сектора этого отдельного диска.
- Плохие сектора:
  - Механизм для сохранения плохих секторов в другом месте диска.
  - Если существует обобщенный механизм для маркировки деградировавших плохих блоков, которые вышестоящие уровни файловой системы могут распознать, использовать его. Программный он или нет.
  - Возможен альтернативный механизм извещения вышестоящего уровня, что размер диска более маленький, прямо выравнивая для вышестоящего уровня сдвигать части исключенных областей диска. Это может помочь с деградированными блоками.
  - Недостаток вышеуказанных идей, сохранять маленькое (устанавливаемое администратором) количество пространства для резервирования плохих блоков (равномерно распределенное по диску?), и использовать его (как можно более близко) вместо плохих блоков, при их появлении. Конечно, это не эффективно. Более того, ядро должно вести log каждый раз при нахождении RAID массивом плохого сектора и делать это с "`crit`" уровнем предупреждения, просто дать понять администратору, что его диск содержит пылинку в себе (или соприкосновение головки с пластиной).
- Программно-переключаемые диски:

  "запретить этот диск"
  должно блокировать, пока ядро не завершит проверки наличия данных для сбрасывания на диск при завершении (таких как завершение XOR/ECC/других коррекций ошибок), затем освободить диск от использования (чтобы его можно было вынуть и т.п.);

  "разрешить этот диск"
  должно, при соответствии, `mkraid` новый диск и затем запустить его для использования для ECC или любых операций, расширяя RAID5 массив;

  "изменить размер массива"
  должно переопределять общее чило дисков и чило избыточных дисков, и результатом должно быть изменение размера массива; без потери данных, хорошо было бы сделать это должным образом, но я потратил много времени пытаясь описать, как это должно делаться; в любом случае, необходим режим, где массив будет блокироваться (возможно, на несколько часов (ядро должно заносить что-то в log каждые десять секунд));

  "разрешение диска при сохранении данных"
  это должно сохранить данные на диске как есть и перемещать их, как положено, на систему RAID5, так что ужасающее сохранение и восстановление не должно происходить каждый раз когда кто-то "поднимает"систему RAID5 (вместо этого, может быть проще сохранять только один раздел вместо двух, он может поместится на первый как сжатый gzip-ом файл);

  "пере-разрешение диска"
  должно быть операторской подсказкой операционной системе попробовать ранее отказавший диск (это должен быть, как я думаю, просто вызов запрещения, а потом разрешения).

Прочие идеи не из сети:

- finalrd аналог для initrd, для упрощения raid на корневой файловой системе.
- режим только-чтение для raid, чтобы упростить вышесказанное

- Помечать RAID том как чистый всякий раз когда не сделано "частичной записи". – То есть, всякий раз нет транзакций записи, котрые были зафиксированы на одном диске, но все еще не завершены на другом диске.

  Добавить период "неактивности записи"(для избежания частого позиционирования головок на суперблок RAID при относительной занятости RAID тома).