Основы построения файловых систем

Сегодня мы поговорим про ZFS

ZFS (Zettabyte File System) – файловая система, изначально написанная для Sun Solaris.

Интересным образом объединяет RAID, менеджер томов и собственно ФС.

Некоторые из проблем связки LVM + FS

Что хочется	Что есть
Удобный для администратора интерфейс управления томами и ФС	
Быстрые снимки состояния ФС (для окружений вроде Solaris Zones или Linux Containers)	
FSCK, который не занимал бы бесконечно много времени.	
Защита от случайных повреждений содержимого дисков.	

Некоторые из проблем связки LVM + FS

Что хочется	Что есть
Удобный для администратора интерфейс управления томами и ФС	RAID reshape + resizefs или lvextend + resizefs
Быстрые снимки состояния ФС (для окружений вроде Solaris Zones или Linux Containers)	
FSCK, который не занимал бы бесконечно много времени.	
Защита от случайных повреждений содержимого дисков.	

Некоторые из проблем связки LVM + FS

Что хочется	Что есть
Удобный для администратора интерфейс управления томами и ФС	RAID reshape + resizefs или lvextend + resizefs
Быстрые снимки состояния ФС (для окружений вроде Solaris Zones или Linux Containers)	LVM COW snapshots при изменении одного из снимков раздела уменьшают производительность всех остальных. LVM ничего не знает о том, что находится на разделе, поэтому снапшот части ФС сделать невозможно.
FSCK, который не занимал бы бесконечно много времени.	
Защита от случайных повреждений содержимого дисков.	

Некоторые из проблем связки LVM + FS

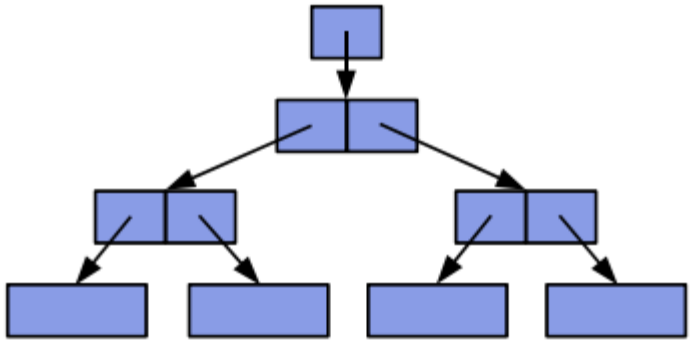
Что хочется	Что есть
Удобный для администратора интерфейс управления томами и ФС	RAID reshape + resizefs или lvextend + resizefs
Быстрые снимки состояния ФС (для окружений вроде Solaris Zones или Linux Containers)	LVM COW snapshots при изменении одного из снимков раздела уменьшают производительность всех остальных. LVM ничего не знает о том, что находится на разделе, поэтому снапшот части ФС сделать невозможно.
FSCK, который не занимал бы бесконечно много времени.	Чтобы прочесть диск размером 10Tb, надо около суток.
Защита от случайных повреждений содержимого дисков.	

Некоторые из проблем связки LVM + FS

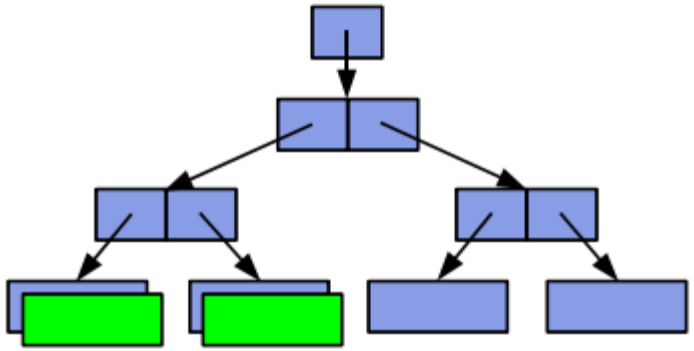
Что хочется	Что есть
Удобный для администратора интерфейс управления томами и ФС	RAID reshape + resizefs или lvextend + resizefs
Быстрые снимки состояния ФС (для окружений вроде Solaris Zones или Linux Containers)	LVM COW snapshots при изменении одного из снимков раздела уменьшают производительность всех остальных. LVM ничего не знает о том, что находится на разделе, поэтому снапшот части ФС сделать невозможно.
FSCK, который не занимал бы бесконечно много времени.	Чтобы прочесть диск размером 10Tb, надо около суток.
Защита от случайных повреждений содержимого дисков.	В ext4, XFS, UFS, NTFS, … такой защиты не предусмотрено, поскольку такие повреждения были слишком маловероятны в то время, когда они разрабатывались.

Copy-On-Write Transactions

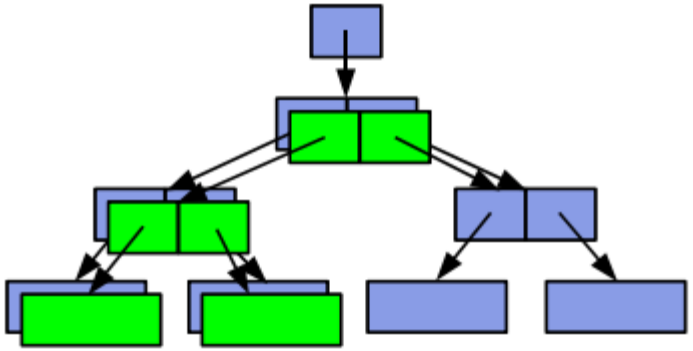
1. Initial block tree



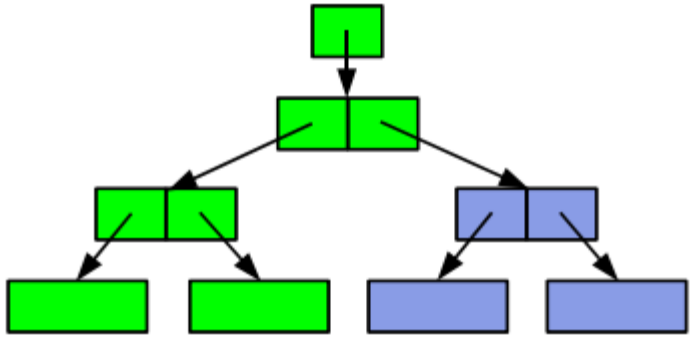
2. COW some blocks



3. COW indirect blocks



4. Rewrite uberblock (atomic)



Что мы получаем от COW

- Транзакционность изменений без использования журнала.
- Бесплатные снимки состояния: достаточно не удалять старую версию данных.
- Мы никогда не перезаписываем данные, поэтому у RAIDZn не бывает RAID write hole.

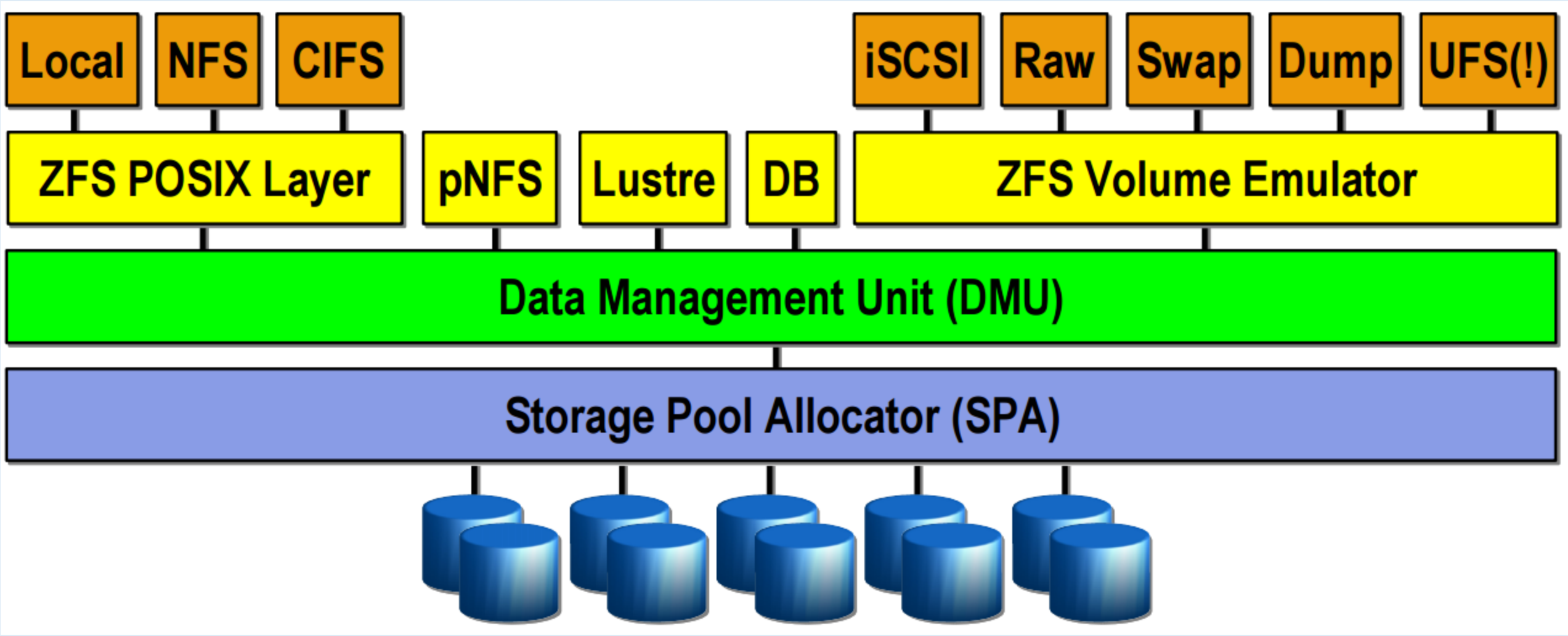
- При изменении элемента в ФС надо обновить всё, что находится между ним и корнем.

Что мы получаем от COW

- Транзакционность изменений без использования журнала.
- Бесплатные снимки состояния: достаточно не удалять старую версию данных.
- Мы никогда не перезаписываем данные, поэтому у RAIDZn не бывает RAID write hole.

- При изменении элемента в ФС надо обновить всё, что находится между ним и корнем.
- **Для удаления элемента нужно свободное место на разделе.**

ZFS IO stack


Local
NFS
CIFS
iSCSI
Raw
Swap
Dump
UFS(!)
ZFS POSIX Layer
pNFS
Lustre
DB
ZFS Volume Emulator
Data Management Unit (DMU)
Storage Pool Allocator (SPA)

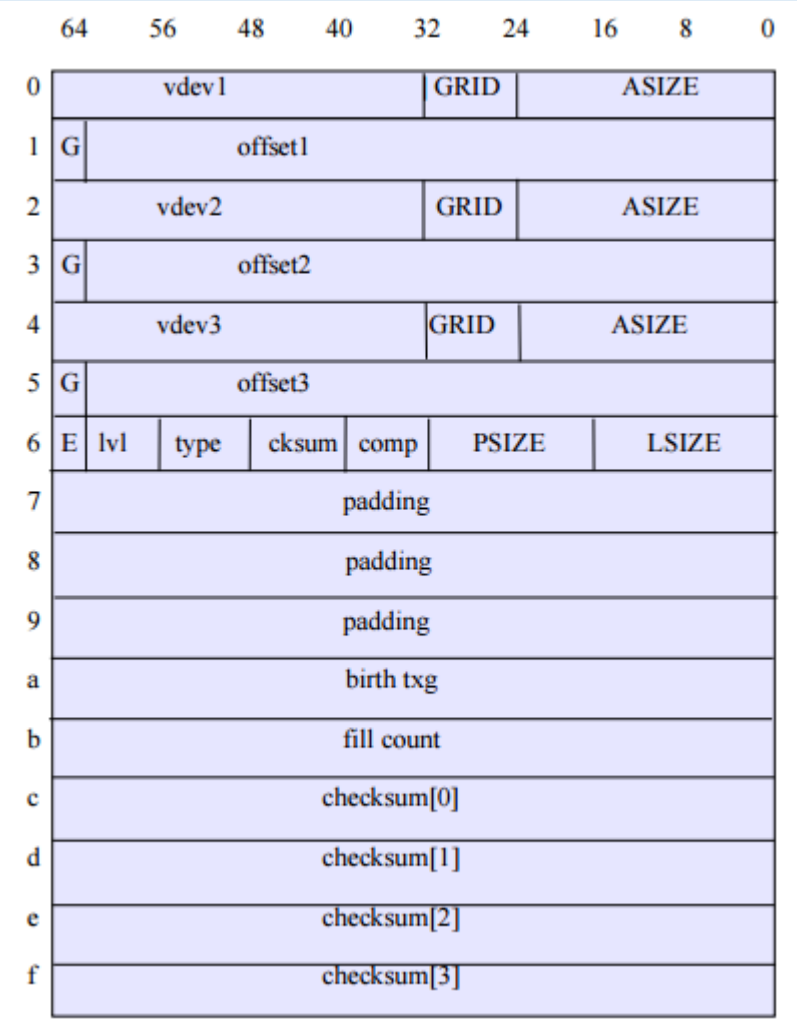
Storage Pool Allocator

- Блоки данных хранятся на virtual devices (vdevs)
- Vdev могут быть отдельными дисками, зеркалами из дисков, или группами RAIDZn

- Место выделяется блоками размера от 512B до 128Kb, экстентов нет,
- Блоки выделяются из vdev по кругу (round-robin),
- Чтобы упростить выделение места на vdev, они разделяются на metaslabs (аналогично block groups в ext4 или allocation groups в XFS),
- В каждом metaslab место учитывается с помощью дерева интервалов и журнала выделения/освобождения блоков.

Storage Pool Allocator

ZFS block pointer



Блок может иметь до трёх копий:

- одна для пользовательский данных,
- две – для метаданных в пределах одной ФС,
- три – для метаданных пула.

Для каждого блока мы храним время его создания – пригодится для снимков состояния.

Для каждого блока мы храним контрольную сумму, притом сумма хранится отдельно от данных.

Sun microsystems

Trends in Storage Integrity

- Uncorrectable bit error rates have stayed roughly constant
  - 1 in 10^{14} bits (~12TB) for desktop-class drives
  - 1 in 10^{15} bits (~120TB) for enterprise-class drives (allegedly)
  - Bad sector every 8-20TB in practice (desktop and enterprise)
- Drive capacities doubling every 12-18 months
- Number of drives per deployment increasing
- → Rapid increase in error rates
- Both silent and “noisy” data corruption becoming more common
- Cheap flash storage will only accelerate this trend

Sun microsystems

Measurements at CERN

- Wrote a simple application to write/verify 1GB file
  - Write 1MB, sleep 1 second, etc. until 1GB has been written
  - Read 1MB, verify, sleep 1 second, etc.
- Ran on 3000 rack servers with HW RAID card
- After 3 weeks, found 152 instances of silent data corruption
  - Previously thought “everything was fine”
- HW RAID only detected “noisy” data errors
- Need end-to-end verification to catch silent data corruption

LVM + FS vs. ZFS

Что хочется	Что есть в ZFS
Удобный для администратора интерфейс управления томами и ФС	Увеличить доступное место – одна команда, которая добавляет vdev. Дальше ZFS просто использует этот vdev.
Быстрые снимки состояния ФС (для окружений вроде Solaris Zones или Linux Containers)	Copy-on-write
Защита от случайных повреждений содержимого дисков.	Checksums, online scrub.
FSCK, который не занимал бы бесконечно много времени.	Online scrub and resilver.

Снимки состояния и время жизни блока

При перезаписи данных мы создаём новый блок, а старый должны удалить. Как быть, если есть снимки состояния ФС?

Снимки состояния и время жизни блока

При перезаписи данных мы создаём новый блок, а старый должны удалить. Как быть, если есть снимки состояния ФС?

Для блоков мы храним время создания. Если оно больше, чем время создания последнего снимка, то на этот блок никто не может ссылаться и его можно удалить.

Снимки состояния и время жизни блока

При перезаписи данных мы создаём новый блок, а старый должны удалить. Как быть, если есть снимки состояния ФС?

Для блоков мы храним время создания. Если оно больше, чем время создания последнего снимка, то на этот блок никто не может ссылаться и его можно удалить.

Как быть с блоками, на которые больше не ссылается снимок A, но продолжает ссылаться более поздний снимок B?

Снимки состояния и время жизни блока

При перезаписи данных мы создаём новый блок, а старый должны удалить. Как быть, если есть снимки состояния ФС?

Для блоков мы храним время создания. Если оно больше, чем время создания последнего снимка, то на этот блок никто не может ссылаться и его можно удалить.

Как быть с блоками, на которые больше не ссылается снимок A, но продолжает ссылаться более поздний снимок B?
Поместить в dead list снимка A.

Ещё один dead list.

Как быть с файлами, которые открыты, но все имена которых удалены? Их содержимое надо освободить, когда они будут закрыты.

Ещё один dead list.

Как быть с файлами, которые открыты, но все имена которых удалены? Их содержимое надо освободить, когда они будут закрыты.

ZFS ведёт delete queue – список файлов без имени, которые надо удалить в будущем.

Ещё один dead list.

Как быть с файлами, которые открыты, но все имена которых удалены? Их содержимое надо освободить, когда они будут закрыты.

ZFS ведёт delete queue – список файлов без имени, которые надо удалить в будущем.

Неочевидная особенность: delete queue тоже имеет COW-семантику, поэтому файл без имени, даже закрытый, не может быть удалён, пока не будут удалени снимки, сделанные пока он ещё был открыт.

Бесплатный (?) block-level deduplication

ZFS для каждого блока считает криптографический хеш. Совпадение таких хешей – надёжный признак совпадения блоков.

Идея: давайте вести DeDuplication Table – список встреченных в ФС хешей блоков. Если новый блок имеет хеш, который встречается в DDT, то его не надо писать на диск – можно сделать ссылку на уже существующий блок.