



NETAPP TECHNICAL REPORT

# Использование Thin Provisioning в системах хранения NetApp

Rick Jooss

Январь 2008 | TR-3483

## Коротко о главном

Традиционные дисковые системы хранения заставляют администраторов хранения выделять место под нужды приложений в момент создания соответствующего LUN или тома. Проблема заключается в том, что на момент создания LUN или тома, потребности приложения или задачи, как правило, неясны, что вынуждает выделять это пространство со значительным запасом. Системы хранения NetApp™ предлагают мощные и гибкие возможности распределения пространства, дающие администратору системы хранения методы снизить затраты, сохраняя возможности для системы хранения соответствовать требованиям задач по производительности и емкости. Этот документ описывает конфигурации и рекомендации по эффективному распределению пространства хранения для SAN с использованием NetApp FAS.

## Оглавление

1 Введение .....	3
2 Необходимые требования к читателю .....	3
3 THIN PROVISIONING.....	3
3.1 Определение .....	3
3.2 WAFL — то, что позволяет этому работать.....	4
4 Использование пространства LUN .....	4
4.1 С точки зрения хоста или системы хранения.....	4
4.2 Емкость LUN, используемая с течением времени.....	6
4.3 Возврат неиспользуемого пространства (Space Reclamation) .....	7
4.4 Space Reclamation с точки зрения хоста или системы хранения.....	7
5 Параметры конфигурирования в DATA ONTAP .....	8
5.1 LUN Reservation.....	8
5.2 Параметры Guarantees .....	9
5.3 Fractional_reserve .....	12
5.4 Snap Reserve.....	13
5.5 Заполнение тома на примере .....	13
5.6 Autosize.....	16
5.7 Autodelete .....	17
6 Эффективное распределение пространства при использовании снимков .....	18
6.1 Thin provisioning при использовании снимков .....	18
6.2 Наилучшие конфигурации: Provisioning с использованием снимков .....	18
7 Эффективное распределение пространства при использовании LUN .....	20
7.1 Thin provisioning пространства LUN .....	20
7.2 Наблюдение за доступным пространством.....	20
7.3 SnapDrive for Windows и thin provisioned LUN .....	20
7.4 Рекомендуемые конфигурации .....	21
7.5 Thin Provisioning при расширении LUN (LUN Growing).....	22
8 Выводы .....	22

## 1 Введение

Этот документ описывает, как эффективно распределять пространство хранения на системах NetApp для использования в сетях Fibre Channel или iSCSI. Эффективность хранения исторически всегда была невысока, по причине сложностей процесса распределения или изменения распределенного единожды пространства на традиционных архитектурах дисковых массивов хранения, чтобы соответствовать изменяющимся требованиям задач. Системы хранения NetApp, с момента внедрения технологии FlexVol™ в версии с Data ONTAP® 7.0, обеспечивают предельно гибкую парадигму распределения пространства, позволяющую пользователю минимизировать неиспользуемое пространство. Этот документ не охватывает вопросы использования thin provisioning при использовании LUN с FlexClone™, так как эти методы рассмотрены в других работах.

## 2 Необходимые требования к читателю

Этот документ адресован архитекторам информационных систем и систем хранения, разрабатывающим решения, с использованием систем хранения NetApp FAS. Для того чтобы воспользоваться методами и процедурами, описанными в этом документе, читателю необходимо:

- Иметь базовые представления об устройстве систем хранения NetApp FAS, и применяемых в них программных решениях, в области Fibre Channel и/или iSCSI
- Принять во внимание, что этот документ основывается на возможностях версии Data ONTAP release 7.2.4 и позднее.

## 3 THIN PROVISIONING

### 3.1 Определение

Суть thin provisioning в предоставлении хостам большего пространства, чем присутствует физически на системе хранения. Системы хранения NetApp всегда имели такую, обычно непростую в реализации возможность для распределения пространства, с доступом по протоколам FC и iSCSI, а реализация технологии FlexVol в семействе Data ONTAP 7.0 значительно увеличила ее гибкость. Примером использования thin provisioning будет вариант, когда на системе хранения, физическим размером 5000GB, администратором выделены 15 LUN-ов различным хостам, размером по 500GB каждый. В данном примере администратор хранения распределил 7500GB пространства, хотя система хранения имеет физически только 5000GB используемого пространства. Разумеется, если все 15 хостов одновременно задействуют по 500GB выделенных им, то это будет проблемой. Администратор системы хранения должен будет наблюдать за уровнем использования пространства на системе хранения, и добавлять доступную емкость, когда это потребуется.

В различных отраслях промышленности и бизнеса такой подход активно используется. Например, компания, занимающаяся водоснабжением, не рассчитывает свои ресурсы по напору воды на вариант, когда все ее потребители откроют краны водопровода «на полную мощность». Компания, обычно, рассчитывает на то, что только некоторый разумный процент пользователей воспользуется водопроводом в данный конкретный момент. Аналогично и в банковской сфере, большинство банков не держит в хранилищах огромное количество наличных, в расчете на то, что все клиенты банка разом захотят получить наличные по своим счетам разом. Как правило, они

рассчитывают, что только некоторый процент клиентов может обратиться в какой-то момент времени за наличными, и это дает им значительную гибкость возможностей. Оба эти примера показывают, что базовая концепция thin provisioning широко применяется в мире, в том числе она может быть использована и в области хранения данных. Очевидно, что thin provisioning имеет тем большую применимость и преимущества перед «традиционным» методом, чем больше емкость системы хранения, и чем больше пользователей с различным характером потребления дискового пространства ее используют.

### 3.2 WAFL — то, что позволяет этому работать

WAFL, являющийся «слоем виртуализации» в Data ONTAP, это то, что позволяет задействовать thin provisioning. Когда создается LUN, то он не занимает конкретных блоков на томе NetApp. Вместо этого, блоки занимают на томе NetApp только тогда, когда данные на самом деле на него записываются. Это позволяет администратору выделять, с точки зрения подключенных серверов, больше пространства, чем физически имеется на системе хранения.

## 4 Использование пространства LUN

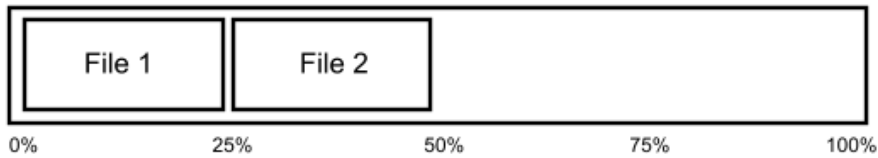
### 4.1 С точки зрения хоста или системы хранения

Пользователи часто неправильно понимают ситуацию с использованным местом внутри LUN. Причина ошибки в том, что подавляюще большинство LUN-ов несут на себе файловую систему, и то, насколько полна файловая система, не соответствует тому, насколько полон сам LUN. Предположим, что файловая система сконфигурирована так, что занимает все пространство LUN, при этом LUN будет всегда заполнен файловой системой, и заполнен настолько, насколько заполнена находящаяся на нем файловая система. С точки зрения системы хранения, процент использованного LUN будет всегда увеличиваться, и никогда не уменьшается, если только не используется процесс space reclamation, о котором мы поговорим далее.

График ниже поможет понять, почему степень использования LUN никогда не уменьшается, если не использовать space reclamation. График показывает теоретический случай, когда файловая система и LUN используются для файлового хранения, и проводятся записи и удаления из файловой системы. Шаг 1 и 2 показывают, как и ожидаемо, что файловая система и LUN оба сообщают равную величину использованного пространства, когда файл записывается на файловую систему. На шаге 3 возникает разница между тем, что сообщает хост или файловая система с тем, что видит система хранения. На этом шаге, файл 1 и 2 удалены, что значит, что блоки, использованные этими файлами, считаются освобожденными и доступными файловой системе. Единственный файл, все еще присутствующий в файловой системе, это файл 3, так что система сообщает о всего 25% заполненности. Однако система хранения по-прежнему говорит о 75% использованного LUN. Причина того, что система хранения по-прежнему показывает 75% заполнения, в том, что у системы хранения нет способов узнать, что данные, записанные в файл 1 и 2 больше не нужны. Их пространство на рисунке закрашено, потому что, «с точки зрения системы хранения», эти блоки все еще содержат какие-то данные. Когда файл удаляется с файловой системы, то файловая система не посылает какого-либо сигнала или команды на систему хранения. Она просто изменяет указатель в своих таблицах размещения на эти файлы, и записывает, что данные блоки могут впрямь использоваться для записи. Факт того, что блоки «удаленных файлов» по-прежнему содержат данные этих файлов, подтверждается многочисленными утилитами «восстановления данных после удаления». На шаге 4, файл 4, как показано, записывается в последние 25% пространства LUN. Этот шаг показывает, что файловая

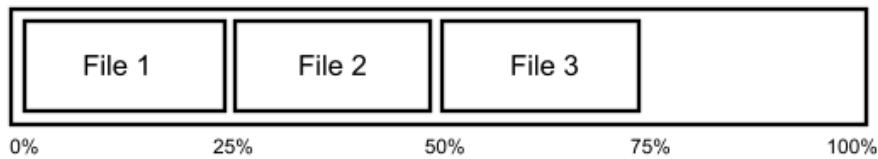
система не обязательно записывает данные в начало LUN. После шага 4, каждый блок на LUN был хотя бы раз записан, и LUN является «заполненным» с точки зрения системы хранения. Как указывалось ранее, процент заполнения LUN никогда не снижается, если не использовать специальные средства space reclamation, и шаги от 5 до 7 показывают это: вне зависимости от того, что происходит с LUN со стороны хоста, он остается «заполненным» на 100% объема.

**Шаг 1** Записываем 2 файла, каждый по 25% общего объема LUN



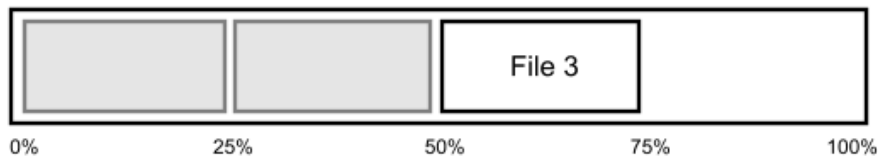
Сервер - 50% занято  
Сторадж - 50% занято

**Шаг 2** Записываем третий файл, объемом 25% LUN



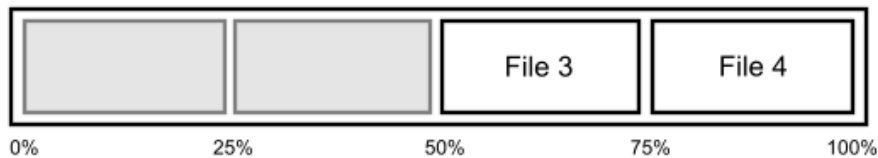
Сервер - 75% занято  
Сторадж - 75% занято

**Шаг 3** Удаляем файлы 1 и 2



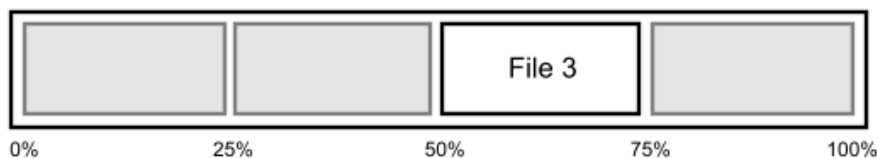
Сервер - 25% занято  
Сторадж - 75% занято

**Шаг 4** Записываем файл 4



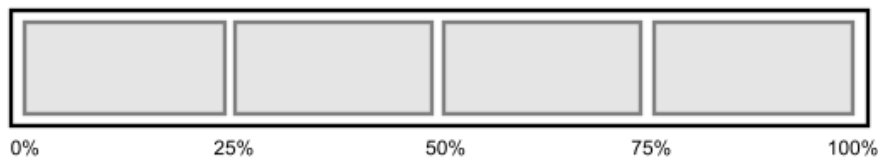
Сервер - 50% занято  
Сторадж - 100% занято

**Шаг 5** Удаляем файл 4



Сервер - 25% занято  
Сторадж - 100% занято

**Шаг 6** Удаляем все файлы



Сервер - 0% занято  
Сторадж - 100% занято

**Шаг 7** Записываем файл 5



Сервер - 25% занято  
Сторадж - 100% занято

Рис. 1) Емкость LUN — с точки зрения хоста и системы хранения.

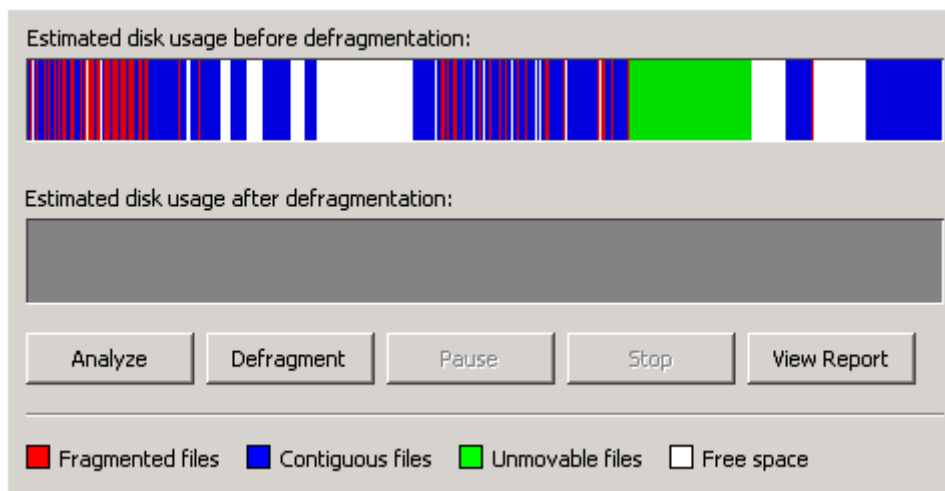


Рис. 2) Использование LUN

#### 4.2 Емкость LUN, используемая с течением времени

Рисунок ниже показывает, сколько пространства LUN используется с течением времени, с точки зрения системы хранения, без процесса space reclamation. Крутизна кривой и скорость достижения 100% зависит от множества параметров, таких как тип файловой системы или приложения, размера записываемых на файловую систему файлов, и объемов файловой системы, занятых под собственные нужды. Время, которое проходит до момента заполнения LUN может варьироваться от часов до лет. Важный момент на этом графике заключается в том, что использованное пространство, с точки зрения системы хранения, никогда не освобождается, если не использовать space reclamation.

Эта проблема присуща всем системам хранения на рынке, а не только системам NetApp.

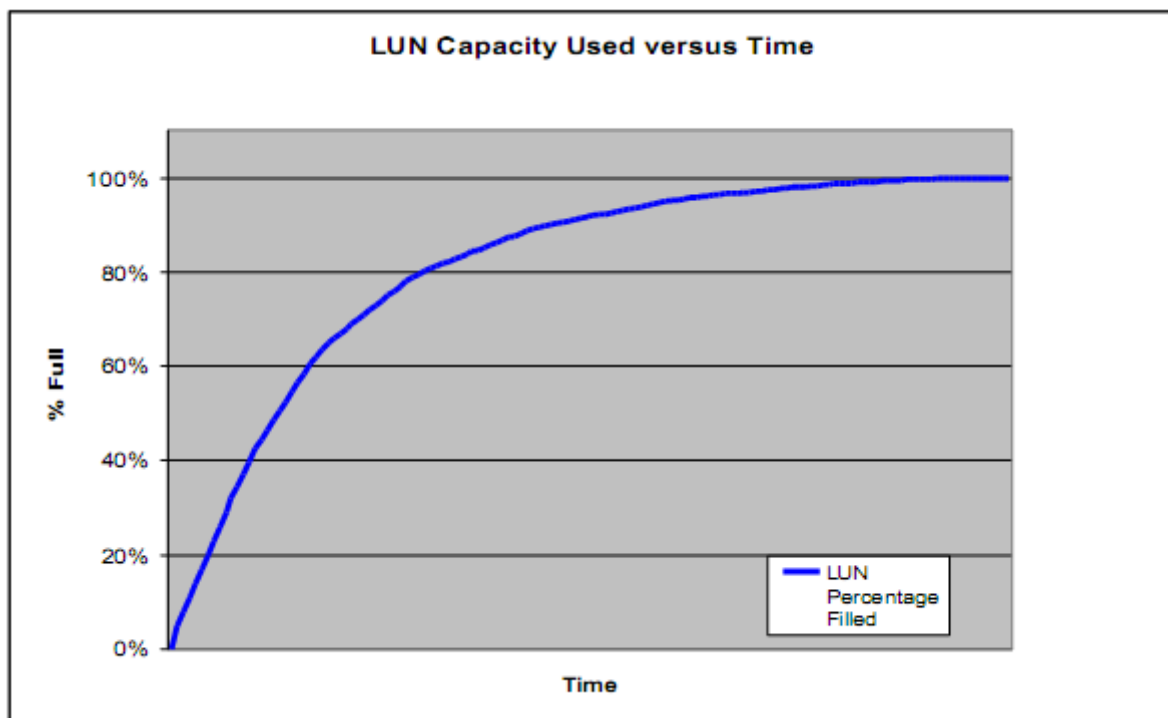


Рис. 3) Емкость LUN с течением времени.

### 4.3 Возврат неиспользуемого пространства (Space Reclamation)

Space Reclamation это процесс, позволяющий системе хранения освободить блоки, более не используемые операционной системой хост-сервера. После этого, такие блоки могут вновь использоваться системой хранения для каких-то иных целей, например для хранения данных снимков. Впервые *space reclamation* появилась в SnapDrive for Windows version 5.0, и работает с файловой системой NTFS.

### 4.4 Space Reclamation с точки зрения хоста или системы хранения

Пример, показанный в разделе 4.1, демонстрирует, как соотносится использование места на файловой системе хоста и на LUN с ходом времени. Space Reclamation может, в теории, устранить разницу между использованным пространством на LUN его количеством, использованным файловой системой хоста. На практике *space reclamation* не устраняет эту проблему полностью, но сильно снижает ее действие.

Первые три шага в примере выше одинаковы как со *space reclamation*, так и без нее. Шаг #4 показывает результат, после запуска *space reclamation*. Space reclamation производится с помощью **SnapDrive for Windows**, и проходит на файловой системе хоста, определяя на ней группы свободных блоков. Затем она посылает специальную команду на систему хранения, которая говорит ей, что эти блоки больше не содержат данных. Блоки временно локируются, чтобы не дать приложению использовать эти блоки во время прохождения процесса *reclamation*. После этого, система хранения освобождает эти блоки.

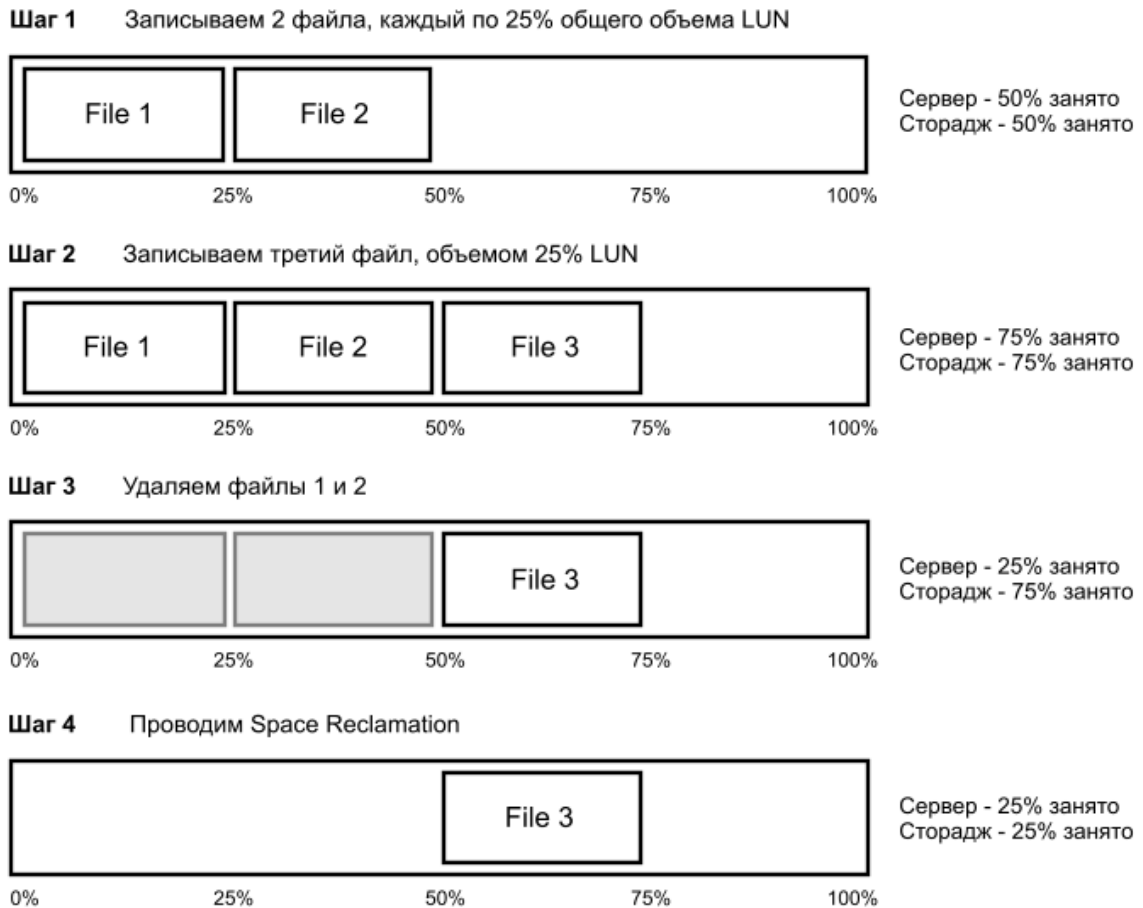


Рис. 3) Емкость LUN — с точки зрения хоста и системы хранения, с использованием space reclamation

## 5 Параметры конфигурирования в DATA ONTAP

Существует некоторое количество переменных и опций в Data ONTAP, смысл которых важно понимать, перед тем, как приступать к конфигурированию thin provisioning.

### 5.1 LUN Reservation

LUN reservation (не следует путать с logical unit locking reservations в протоколах SCSI-2 или SCSI-3) определяет то, сколько пространства для LUN будет зарезервировано, или выделено предварительно, на соответствующем ему томе. С включенным резервированием (значение по умолчанию) пространство на томе уменьшается в момент создания LUN. Например, если LUN, размером 20GB создан на томе, с объемом свободного места, равном 80GB, объем свободного места на томе станет равным 60GB в момент создания LUN, даже если в LUN на этот момент ничего не будет записано. Если резервирование выключено, то пространство начнет уменьшаться только по мере проводимых в LUN записей. Если LUN размером 20GB был создан без включенного LUN space reservation, то количество свободного места на томе останется 80GB и будет постепенно уменьшаться, по мере того, как данные будут записываться на LUN.



## 5.2 Параметры Guarantees

С появившимися в Data ONTAP 7.0 томами типа flexible появилась и концепция space guarantees, которая позволяет пользователю определить то, как будет резервироваться или выделяться пространство на aggregate.

- Volume — уровень гарантии «volume» означает, что потребованное томом типа FlexVol пространство будет всегда доступно на его aggregate. Это значение по умолчанию для тома FlexVol. С параметром space guarantee установленном в «volume» пространство отнимается, или «гарантируется», у aggregate в момент создания тома. Пространство резервируется на aggregate вне зависимости от того, используется ли оно для хранения данных или нет.

Пример, показывающий создание тома, размером 20GB. Команда df показывает величину использованного места на aggregate до, и после создания тома. Команда показывает, как 20GB забирается с aggregate сразу, как только том создан, даже если никакие данные еще не записаны на него.

```
netapp1> df -A -g aggr0
```

Aggregate	total	used	avail	capacity
aggr0	85GB	0GB	85GB	0%
aggr0/.snapshot	4GB	0GB	4GB	0%

```
netapp1> vol create flex0 aggr0 20g
```

Creation of volume 'flex0' with size 20g on hosting aggregate 'aggr0' has completed.

```
netapp1> df -g /vol/flex0
```

Filesystem	total	used	avail	capacity	Mounted on
/vol/flex0/	16GB	0GB	16GB	0%	/vol/flex0/
/vol/flex0/.snapshot	4GB	0GB	4GB	0%	/vol/flex0/.snapshot

```
netapp1> df -A -g aggr0
```

Aggregate	total	used	avail	capacity
aggr0	85GB	20GB	65GB	23%
aggr0/.snapshot	4GB	0GB	4GB	0%

Так как пространство резервируется на aggregate сразу, то операции записи на том не занимают на нем больше места, чем уже занято.

- None — Том типа FlexVol с уровнем гарантии «none» вообще не резервирует пространство на aggregate в момент создания. Пространство на aggregate занимает только по мере того, как на том начнут записываться данные. В примере видно, как, по сравнению с рассмотренным выше параметром гарантии «volume», создание тома не уменьшает количество места на aggregate. Даже в случае создания LUN, которое производится по умолчанию с включенным space reservation, оно не занимает место на aggregate.

Операция записи в space-reserved LUN, на том, созданный с параметром *guarantee = none*, завершится с ошибкой, если aggregate не имеет для этого достаточно свободного места. LUN reservation гарантирует, что LUN будет иметь достаточно места на томе, однако в случае *guarantee = none* это не гарантирует, что и том имеет достаточно места на aggregate.

```
netapp1> df -A -g aggr0
```

Aggregate	total	used	avail	capacity
aggr0	85GB	0GB	85GB	0%
aggr0/.snapshot	4GB	0GB	4GB	0%

```
netapp1>
```

```
netapp1> vol create noneflex -s none aggr0 20g
```

```
Creation of volume 'noneflex' with size 20g on hosting aggregate  
'aggr0' has completed.
```

```
netapp1>
```

```
netapp1> df -g /vol/noneflex
```

Filesystem	total	used	avail	capacity	Mounted on
/vol/noneflex/	16GB	0GB	16GB	0%	/vol/noneflex/
/vol/noneflex/.snapshot	4GB	0GB	4GB	0%	/vol/noneflex/.snapshot

```
netapp1>
```

```
netapp1> df -A -g aggr0
```

Aggregate	total	used	avail	capacity
aggr0	85GB	0GB	85GB	0%
aggr0/.snapshot	4GB	0GB	4GB	0%

```
netapp1> lun create -s 10g -t windows /vol/noneflex/foo
```

```
Mon Nov 24 15:17:28 EST [array1: lun.vdisk.spaceReservationNotHonored:notice]:  
Space reservations in noneflex are not being honored, either because the volume  
space guarantee is set to 'none' or the guarantee is currently disabled due to  
lack of space in the aggregate.
```

```
lun create: created a LUN of size: 10.0g (10742215680)
```

```
netapp1>
```

```
netapp1> df -g /vol/noneflex
```

Filesystem	total	used	avail	capacity	Mounted on
/vol/noneflex/	16GB	10GB	6GB	0%	/vol/noneflex/
/vol/noneflex/.snapshot	4GB	0GB	4GB	0%	/vol/noneflex/.snapshot

```
netapp1>
```

```
netapp1> df -A -g aggr0
```

Aggregate	total	used	avail	capacity
aggr0	85GB	0GB	85GB	0%
aggr0/.snapshot	4GB	0GB	4GB	0%

- File — с уровнем *guarantee = file* на aggregate гарантируется наличие места для перезаписи LUN, созданного с параметром *space-reserved*. *Fractional reserve*, опция уровня тома, она рассматривается в этом документе ниже, установлена на 100% и не настраивается для этого типа *guarantee*. Тип *guarantee = file* в основе соответствует типу *guarantee = none*, за исключением того, что учтены и действуют резервирования пространства для LUN и файлов. Пример ниже схож с таким же для *guarantee=none*, за исключением того, что при создании LUN он сразу займет место на aggregate, так как для LUN использует резервирование пространства под свое содержимое. Так как *space reservation* действует для этого типа *guarantee*, то команда `lun create` не выдаст предупреждения, подобного тому, что мы видели в предыдущем примере.

```
netapp1> df -A -g aggr0
```

Aggregate	total	used	avail	capacity
aggr0	85GB	0GB	85GB	0%
aggr0/.snapshot	4GB	0GB	4GB	0%

```
netapp1>
```

```
netapp1> vol create noneflex -s file aggr0 20g
```

```
Creation of volume 'noneflex' with size 20g on hosting aggregate
'aggr0' has completed.
```

```
cnr11>
```

```
netapp1> df -g /vol/noneflex
```

Filesystem	total	used	avail	capacity	Mounted on
/vol/noneflex/	16GB	0GB	16GB	0 %	/vol/noneflex/
/vol/noneflex/.snapshot	4GB	0GB	4GB	0%	/vol/noneflex/.snapshot

```
netapp1>
```

```
netapp1> df -A -g aggr0
```

Aggregate	total	used	avail	capacity
aggr0	85GB	0GB	85GB	0%
aggr0/.snapshot	4GB	0GB	4GB	0%

```
netapp1>
```

```
netapp1> lun create -s 10g -t windows /vol/noneflex/foo
```

```
lun create: created a LUN of size: 10.0g (10742215680)
```

```
netapp1>
```

```
netapp1> df -g /vol/noneflex
```

Filesystem	total	used	avail	capacity	Mounted on
/vol/noneflex/	16GB	10GB	6GB	0%	/vol/noneflex/
/vol/noneflex/.snapshot	4GB	0GB	4GB	0%	/vol/noneflex/.snapshot

```
netapp1>
```

```
netapp1> df -A -g aggr0
```

Aggregate	total	used	avail	capacity
-----------	-------	------	-------	----------

aggr0	85GB	10GB	75GB	12%
aggr0/.snapshot	4GB	0GB	4GB	0%

### 5.3 Fractional\_reserve

*Fractional reserve* это опция тома, которая определяет то, сколько пространства Data ONTAP резервирует для перезаписанных данных при наличии снэпшотов с LUN и файлов с параметром *space-reserved*, которое будет использоваться, когда займется все прочее место на томе. Значение по умолчанию равно 100%, однако, с появлением функциональности **autodelete**, NetApp, в общем случае, рекомендует устанавливать его значение в *fractional\_reserve = 0* и пользоваться возможностью *autodelete*. Data ONTAP забирает (или резервирует) это пространство, как только создается первый же снэпшот, и начинает использовать это пространство только тогда, когда все остальное пространство на томе полностью заполнится. Например, как показано в левой части рисунка, на томе, размером 100GB, находятся два LUN размером 20GB каждый. Предположим, что LUN-ы заполнены. Значение по умолчанию в Data ONTAP равное *fractional\_reserve = 100* резервирует 40GB (2 x 20GB) пространства на томе, чтобы быть уверенным в том, что на томе всегда останется достаточно места для обоих LUN-ов и их снэпшотов, даже если LUN-ы будут полностью перезаписаны. Этот вариант показан в середине. На правой части рисунка показан вариант, когда *fractional\_reserve* установлен в размере 60% при этом, когда снэпшот создается, вместо резервирования 40GB на томе, Data ONTAP резервирует 24GB (60% \* [2 x 20GB]).

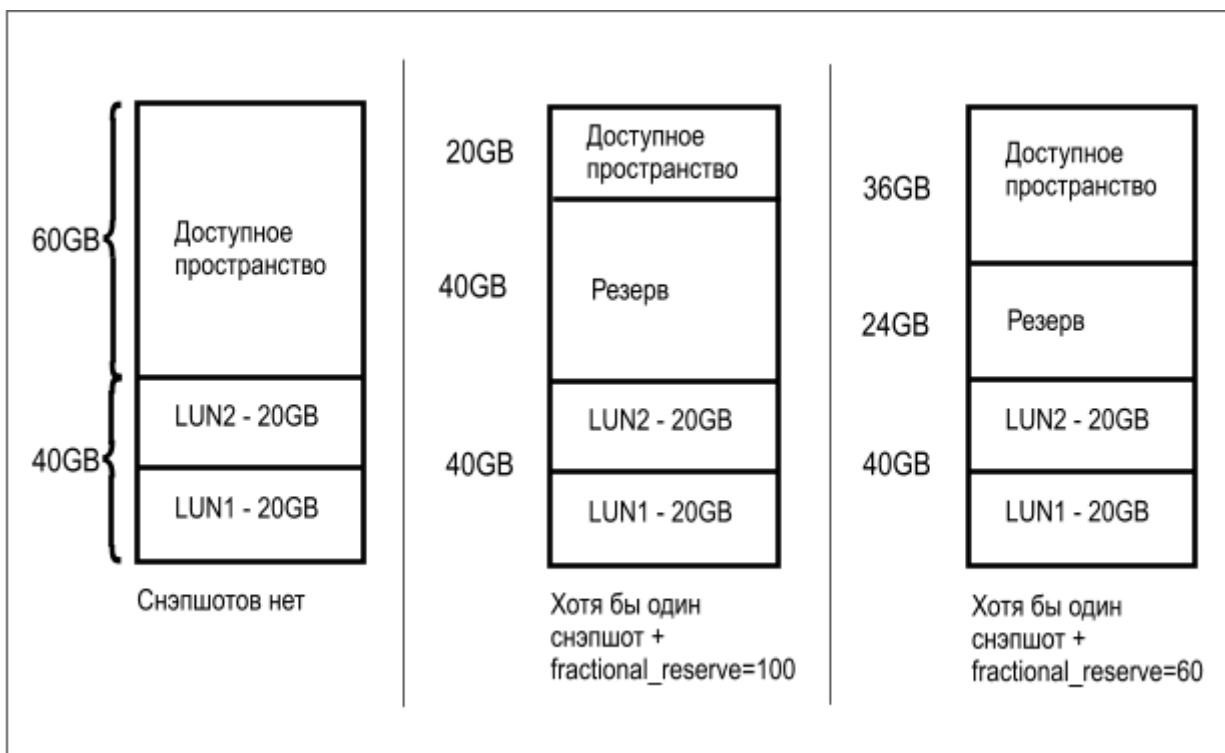


Рис. 4) Fractional\_reserve.

Зарезервированный объем можно посмотреть, запустив команду `df` с ключом `-r`. Это пространство используется, только когда Data ONTAP сообщает об исчерпании места на томе. Пока том не заполнен целиком, место под перезапись снэпшотов (*snapshot overwrites*) берется из пространства самого тома, и лишь после заполнения его на 100% начинает использоваться из пространства резерва.

## 5.4 Snap Reserve

Эта переменная определяется на уровне тома, и устанавливается в процентах от размера тома. Data ONTAP отнимает определенное количество (20% по умолчанию) пространства тома, доступного для создания LUN-ов, или для использования с CIFS или NFS. Когда снимоту нужно место для хранения своих данных, он использует эту зарезервированную область.

В качестве поведения по умолчанию, когда пространство *snap reserve* заполнено, снимоты начинают использовать оставшееся свободное пространство тома. На самом деле, конечно же, согласно технологии WAFL, *snap reserve* это не какие-то физически выделенные блоки для использования их в снимотах, а просто некое логическое определение.

## 5.5 Заполнение тома на примере

Следующие рисунки показывают вам пошаговый пример того, как используется пространство создания тома, вплоть до заполнения тома данными снимотов. В этом примере значения *lun reserve*, *fractional\_reserve*, и *snap reserve* оставлены в значениях по умолчанию.

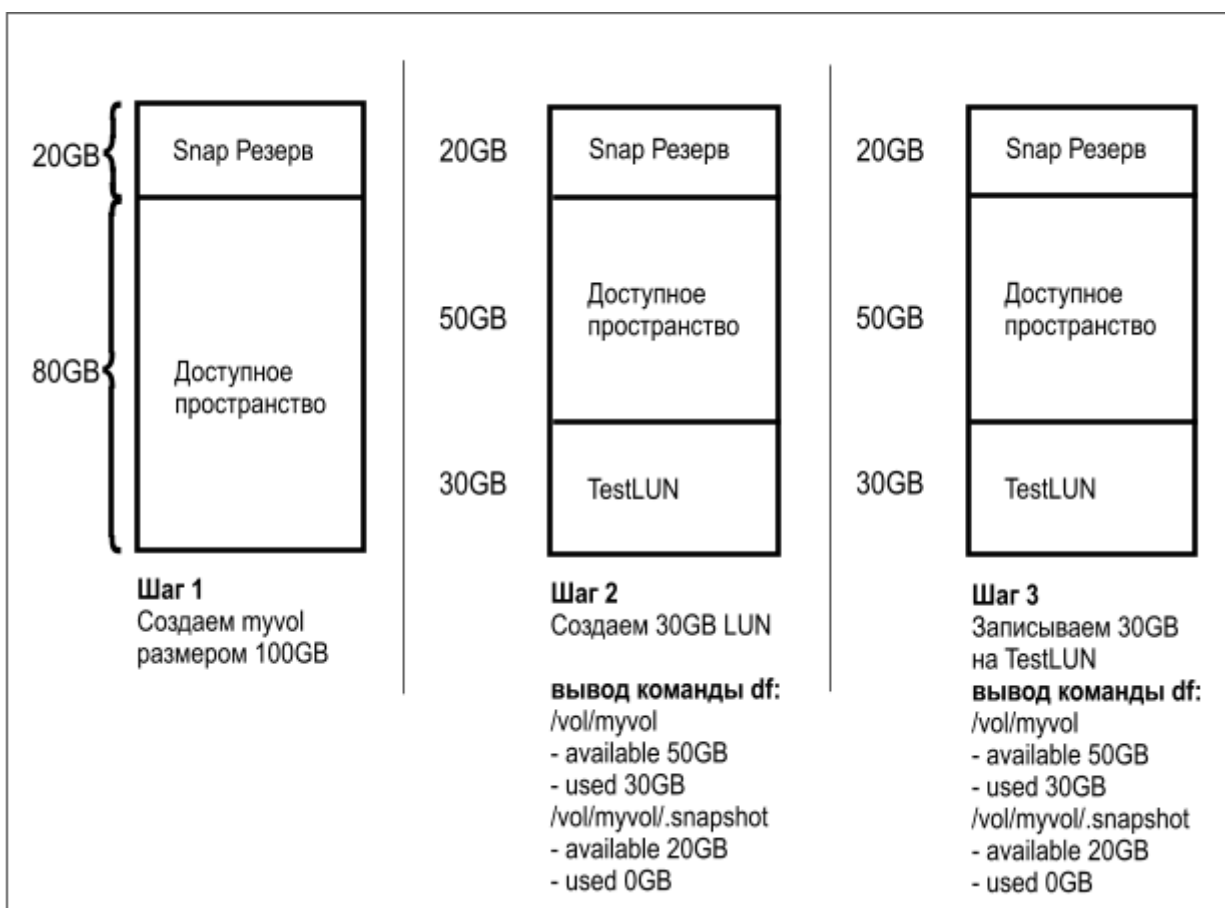


Figure 5) Заполнение тома — Шаги #1–#3.

- Шаг #1 — Создан том, размером 100GB. Доступное пространство равно 80GB так как, по умолчанию, *snap reserve* равен 20%.
- Шаг #2 — Создан тестовый LUN размером 30GB с именем «TestLUN». Так как по умолчанию для LUN используется резервирование, то 30GB немедленно вычтутся из доступного пространства тома.

- Шаг #3 — LUN заполняется данными, и, так как пространство под LUN было зарезервировано сразу при его создании, то никаких изменений в размерах свободного пространства тома не происходит.

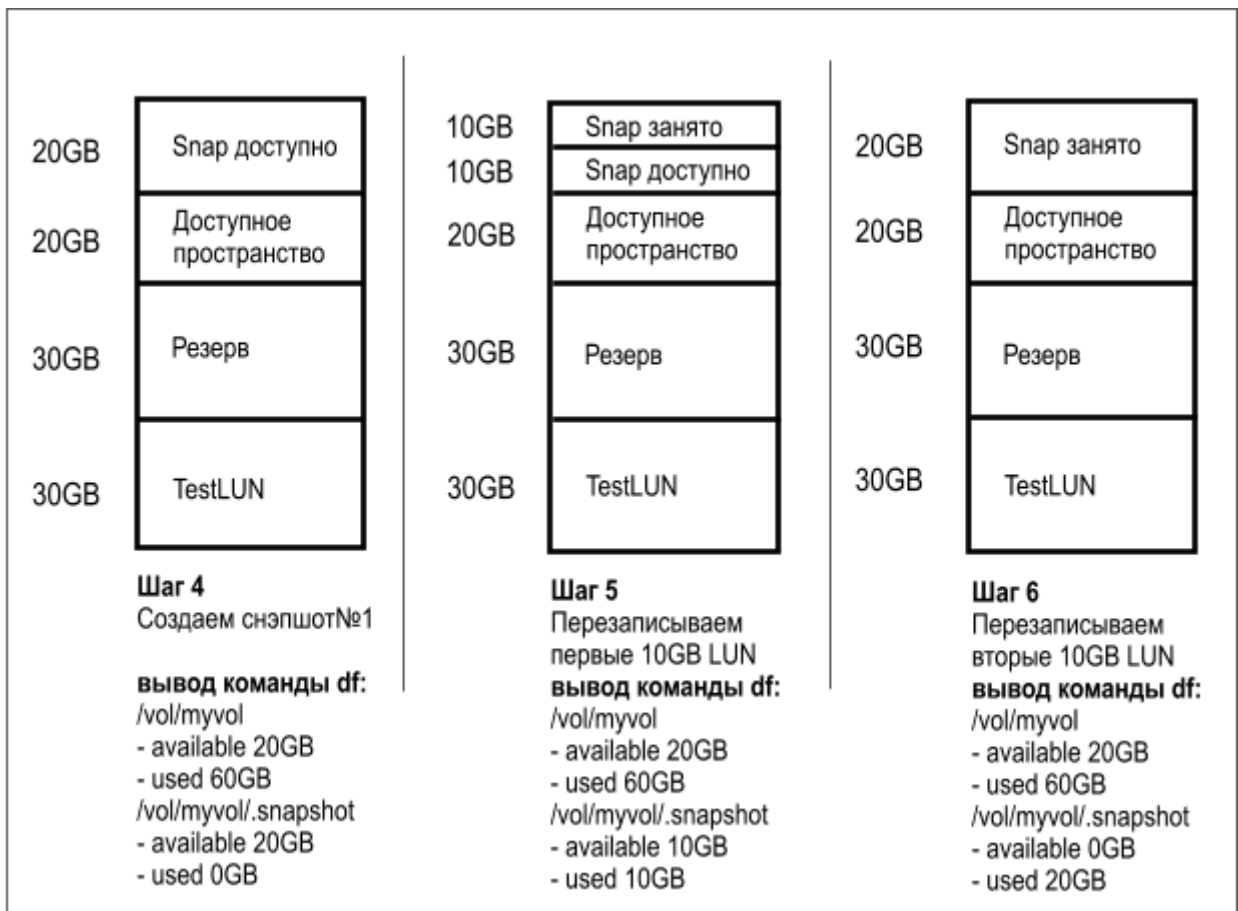


Figure 6) Заполнение тома — Шаги #4–#6.

- Шаг #4 — Создан снимок, что вызывает резервирование 30GB места на томе, что необходимо для обеспечения достаточного пространства для снимка в том случае, если даже весь LUN целиком будет перезаписан. Как показано, если *fractional\_reserve* будет установлен в значение менее 100, размер резервированного пространства будет менее 30GB.
- Шаг #5—10GB данных перезаписано на TestLUN. Так как снимок его уже существует, то перезапись данных LUN означает, что для снимка начнет использоваться свободное пространство самого тома. Первый участок будет использован из *snapshot reserve area*. Это видно по тому, что покажет команда *df* для тома */vol/myvol/*.
- Шаг #6 — Другие 10GB данных перезаписаны на TestLUN. Это использует остальное пространство *snapshot reserve area*, которое теперь полностью занято.

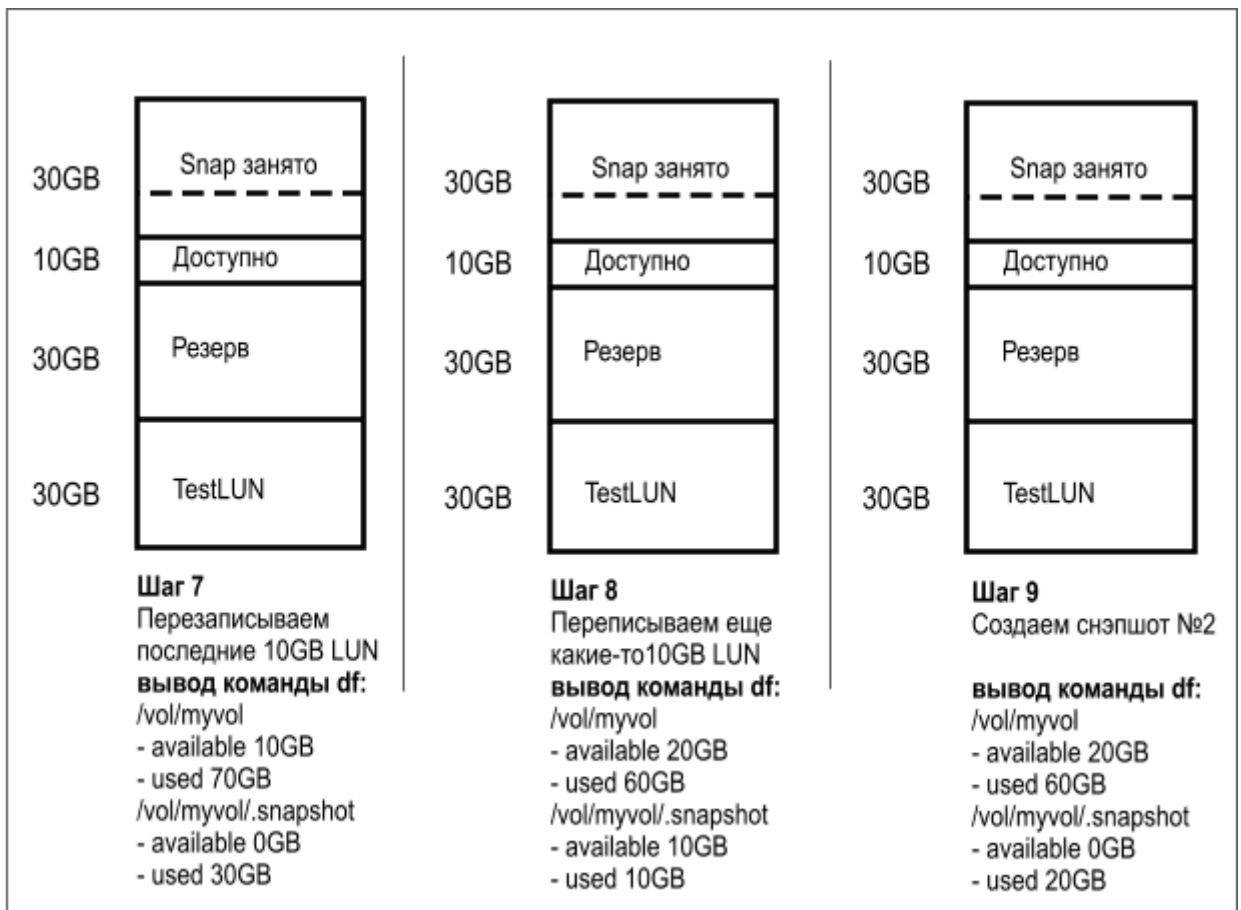


Figure 7) Заполнение тома — Шаги #7–#9.

- Шаг #7 — Следующие 10GB данных перезаписываются на TestLUN. Даже если *snapshot reserve area* целиком заполнена, запись можно продолжать, и снэпшоты не окажутся потеряны или повреждены, так как по-прежнему остается место на пространстве самого тома. Эти записи просто берут место у пространства самого тома, которое уменьшается с 20GB, как было показано на предыдущем шаге, до 10GB после шага #7. Пространство в снэпшотах, которое выходит за пределы *snap reserve* может быть посчитано дважды в выводе команды *df*, так как оно показано как «занятое» (колонка *used*) сперва в объеме основного тома, а потом в пространстве *snap reserve*. Интересно отметить, что величина использованного на томе пространства в выводе команды *df* при этом будет превышать 100%.
- Шаг #8 — Перезаписываются следующие 10GB данных на TestLUN. В этом случае, так как на предыдущем шаге, весь LUN уже был перезаписан, дополнительным данным снэпшота не нужно места. Изменений в величинах использованного пространства не происходит.
- Шаг #9 — Создан второй снэпшот. В отличие от ситуации с созданием первого снэпшота, создание дополнительных снэпшотов не вызывает немедленного изменения в объемах использованного пространства. Создание дополнительного снэпшота не влияет напрямую на величину использованного пространства. Однако когда количество снэпшотов увеличивается, возможен рост занятого места в *snapshot reserve space*, так как множественные перезаписи блоков скорее всего будут «заперты» в отдельных снэпшотах.

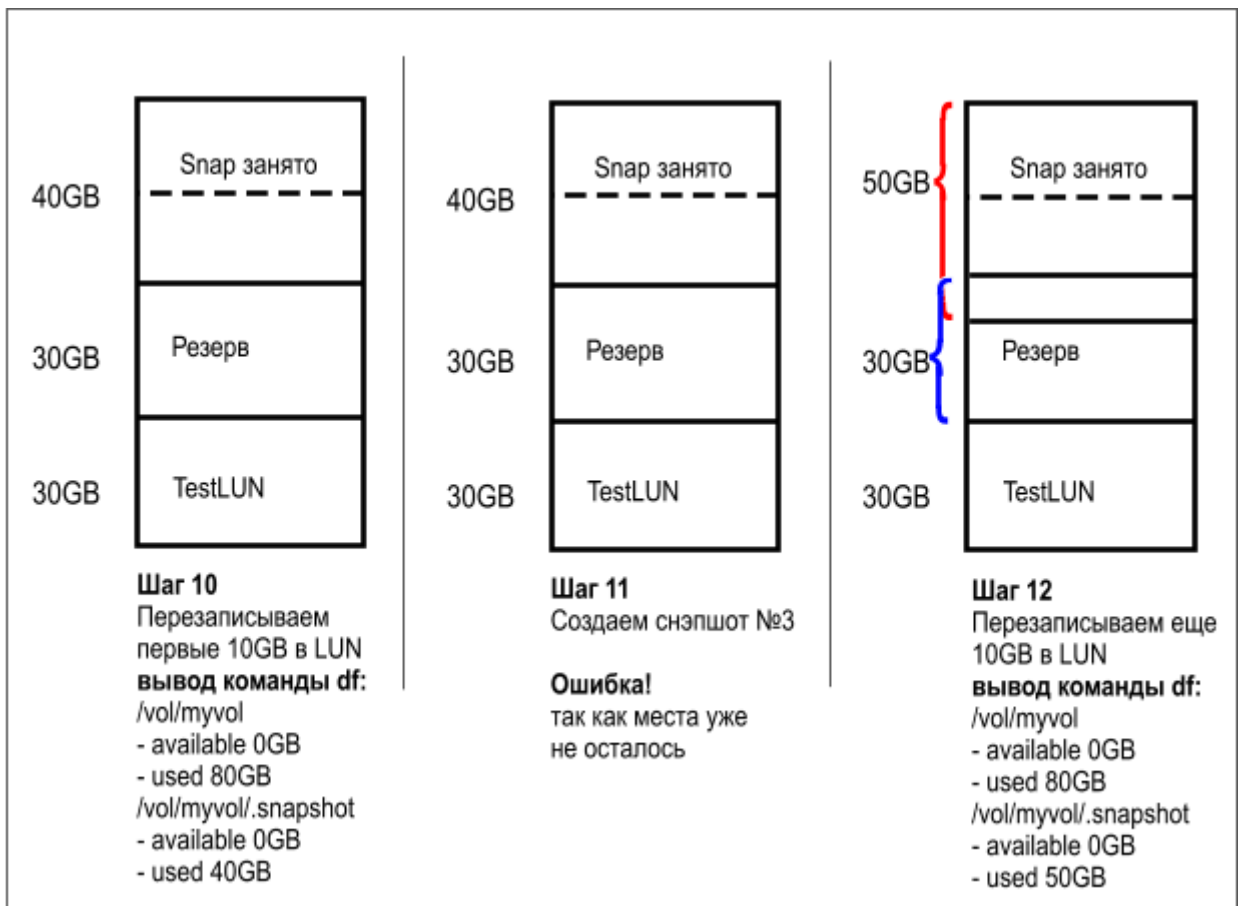


Figure 8) Заполнение тома — Шаги #10–#12.

- Шаг #10 — 10GB содержимого LUN перезаписано. Так как снимок только что был создан на предыдущем шаге, то не важно, какая часть 10GB в LUN перезаписана, так как пространство все равно нужно для сохранения данных снимота. В этом случае, 10GB доступного пространства будет взяты вне пространства тома. На этом месте том сообщает о том, что он заполнен, на нем невозможно создать другой LUN, а также записать на него файлы по NFS/CIFS, создать еще снимоты, или записать не в зарезервированное место.
- Шаг #11 — Делается попытка создать третий снимот. Операция завершается неудачей, так как том заполнен.
- Шаг #12 — Следующие 10GB данных перезаписываются на TestLUN. Даже несмотря на то, что том полон, записи в TestLUN проводятся успешно, и снимот остается на месте, так как сохраняется свободное место в зарезервированной области. Единственное изменение в выводе команды df будет в колонке used для области *snap reserve*. Так как *fractional\_reserve* установлен в 100%, записи на LUN не будут прерваны по причине нехватки места. Если *fractional\_reserve* установлен в значение менее 100%, и перезаписи продолжаются, в какой-то момент запись на LUN может остановиться по причине исчерпания места.

Этот пример показывает, что область резервирования используется только в случае, когда не остается другого свободного места на томе.

## 5.6 Autosize

Этот параметр тома (доступен, начиная с Data ONTAP 7.1 и новее) определяет, что том должен автоматически увеличиться, чтобы избежать исчерпания его емкости при наполнении. Эта опция



доступна только для томов типа *flexible*. Возможно определить то, как быстро будет прирастать том, с помощью ключа *-i*. Шаг прироста по умолчанию 5% от размера тома при его создании. Также можно определить максимально возможный размер тома с помощью ключа *-m*. Если параметр тома *autosize* включен, то по умолчанию максимальный размер автоматически увеличивающегося тома задан в 120% от исходного размера тома.

## 5.7 Autodelete

Этот параметр тома (доступен с версии Data ONTAP 7.1) позволяет Data ONTAP удалять снимки, если они соответствуют заданным условиям. Эти условия называются «триггер» (*trigger*) и могут быть настроены таким образом, что снимок будет автоматически удален при выполнении одного из следующих условий:

- *Volume* — Пространство тома близко к исчерпанию. Это пространство сообщается в первой строке вывода для каждого тома, командой *df*. Следует отметить, что том может быть полностью заполнен, даже если еще осталось место в области *snap\_reserve*.
- *Snap\_reserve* — Пространство *snap reserve* близко к исчерпанию.
- *Space\_reserve* — Пространство для перезаписи близко к исчерпанию. Это пространство, определенное LUN-ами с включенным резервированием пространства (*space reservations enabled*) и опцией *fractional\_reserve*. Это зарезервированное пространство не будет заполняться, пока и том, и пространство *snap\_reserve* на нем не будут целиком заполнены.

Настоятельно рекомендуется устанавливать триггер на значение *volume*.

Порядок, в котором будут удаляться снимки, задается следующими тремя опциями:

- *Delete\_order* — Эта опция определяет, будут ли удаляться в первую очередь наиболее старые, или наоборот, наиболее новые снимки.
- *Defer\_deleted* — Эта опция позволяет пользователю определить группу снимков, которая должна быть удалена, только если других доступных снимков нет. Это делает возможным отложить удаление созданных пользователем снимков, снимков, созданных по расписанию, или снимков с заданным префиксом.
- *Commitment* — Эта опция определяет то, как будут обрабатываться снимки, используемые SnapMirror®. Если она установлена в *try*, то снимки будут удалены только в том случае, если они не заблокированы. Если она установлена в *disrupt*, то эти снимки будут удалены, даже если они в этот момент были заблокированы. В этом случае всегда будет удаляться последний снимок. Во многих конфигурациях удаление последнего снимка в SnapMirror нежелательно, так как потребуются репликация полной новой базовой копии, чтобы продолжить операцию репликации. Если, например, система-источник и система-получатель расположены на различных, удаленных площадках, то пересоздание реплики может занять значительное время и быть довольно дорогостоящей.

Алгоритм, который определяет порядок удаления снимков в первую очередь ищет снимки, не соответствующие критерию *defer\_delete*, затем использует порядок, определенный в *delete\_order* чтобы определить, следует ли удалить наиболее старый, или напротив, наиболее свежий снимок. Если такие снимки не найдены, то критерий *defer\_delete* при выборе игнорируется. Если по-прежнему не обнаруживаются подходящих снимков, тогда, в зависимости от опции *commit* под удаление попадают снимки, созданные SnapMirror.

Удаление снэпшотов прекращается в случае, когда свободное пространство освобождается до уровня, заданного в переменной *target\_free\_space* variable, по умолчанию она определена в 80%.

Если включены одновременно и *autosize* и *autodelete*, и триггер *autodelete* установлен на *volume*, то опция тома *try\_first* будет определять то, что будет делаться в первую очередь – приращиваться том, или удаляться снэпшоты.

## 6 Эффективное распределение пространства при использовании снэпшотов

### 6.1 Thin provisioning при использовании снэпшотов

Подавляющее количество клиентов NetApp пользуются на этих системах хранения технологией создания снэпшотов, лидирующей на рынке, и встроенной в OS Data ONTAP. Всякий раз, когда используется снэпшот, необходимо зарезервировать место для хранения его данных.

Размер пространства, занятый снэпшотом, зависит от многих факторов. Некоторые, из наиболее значимых, это характер и поведение приложения и промежутки времени, прошедший с момента взятия снэпшота. Например, если снэпшот создан в ходе работы средства репликации данных SnapMirror, то срок жизни такого снэпшота относительно невелик, и хранится сравнительно небольшое их количество, поэтому для них нет смысла особо создавать резервирование.

### 6.2 Наилучшие конфигурации: Provisioning с использованием снэпшотов

Так как существует множество опций настройки, то существует множество различных конфигураций для распределения пространства хранения (provisioning) с использованием преимуществ, даваемых использованием снэпшотов. Каждая из них имеет свои преимущества и недостатки. Перечисленные ниже конфигурации, это рекомендованные NetApp варианты:

- Конфигурация с использованием **autogrow**

Guarantee	= volume
LUN reservation	= enabled
fractional_reserve	= 0%
snap_reserve	= 0%
autodelete	= on / oldest_first
autogrow	= on
try_first	= volume_grow

Большое преимущество этой конфигурации состоит в использовании свободного пространства на aggregate как совместно используемого для прироста объема свободного пространства, всеми находящимися на нем томами. При использовании значения *guarantee* равного *volume*, могут быть достигнуты различные уровни thin provisioning для отдельных приложений, и настроены индивидуально величины прироста для каждого из томов. Степень использования пространства хранения также просто отслеживается простым наблюдением за величиной объемов тома и aggregate.

Как и для всех конфигураций, совместно использующих свободное место, тома не являются на 100% независимыми один от другого, так как используют общее пространство свободного места, когда им понадобится вырасти. Это значит, что если другой том уже использовал свободное место на aggregate, то это может помешать другому тому расширяться.

Необходимо обратить внимание на то, что если клоны (FlexClone или клоны LUN) создаются как снимки, то они воспрепятствуют их удалению при выполнении *autodelete*.

Этого риска можно избежать, если том будет иметь размер, вдвое (или более) превышающий размер LUN-ов, и хранить только один снимок.

С использованием этой конфигурации, том будет иметь размер, достаточный для размещения на нем LUN-ов и данных снимков.

Например, если требования к тому следующие:

- 5 x 100GB LUNs
- 3% данных в день изменяется
- Снимки сохраняются на протяжении 14 дней

Это означает, что необходимо иметь как минимум  $500\text{GB} + ((500\text{GB} * 3\%) * 14) = 710\text{GB}$ . В идеальном случае следует взять объем слегка больший, на случай непредвиденного роста размеров снимков.

Если снимки требуют больше места, чем запланировано, можно попробовать использовать опцию *autogrow*, если невозможно воспользоваться *autodelete* для освобождения места.

- Конфигурация с использованием **autodelete**

Guarantee	= volume
LUN reservation	= enabled
fractional_reserve	= 0%
snap_reserve	= 0%
autodelete	= volume / oldest_first
autogrow	= off
try_first	= snap_delete

Эта конфигурация сходна с рассмотренной выше, но без опции *autogrow*. Эта конфигурация отдает приоритет сохранению доступа к LUN, перед сохранением доступа к снимкам. Если том становится близок к переполнению, то снимки начинают с него удаляться, чтобы сохранить LUN в онлайн.

Одним из преимуществ данной конфигурации является понятность и простота контроля за использованием пространства, путем простого контроля за объемом тома. Кроме этого тома независимы друг от друга, что означает то, что каждое приложение можно настроить независимо. Эта конфигурация также хороша тем, что при удалении снимков, будут удалены самые старые снимки, в то время, как наиболее свежие будут сохранены. Недостатком является то, что конфигурация не использует доступное пространство на aggregate как общий пул свободного пространства.

Также как и в случае предыдущей конфигурации, следует контролировать то, что если клоны (FlexClone или клоны LUN) создаются как снимки, то они воспрепятствуют их удалению при выполнении *autodelete*.

С использованием этой конфигурации, том будет иметь размер, достаточный для размещения на нем LUN-ов и данных снимков.

Например, если требования к тому следующие:

- 5 x 100GB LUNs
- 3% данных в день изменяется
- Снэпшоты сохраняются на протяжении 14 дней

Это означает, что необходимо иметь как минимум  $500\text{GB} + ((500\text{GB} * 3\%) * 14) = 710\text{GB}$ . В идеальном случае следует взять объем слегка больший, на случай непредвиденного роста размеров снэпшотов.

## 7 Эффективное распределение пространства при использовании LUN

### 7.1 Thin provisioning пространства LUN

Как мы уже определили выше, с использованием thin provisioning мы можем показать для хост-серверов, подключенных к системе хранения, больше места, чем физически имеется на ней. Для систем, потребности которых растут, использование thin provisioning позволяет администратору системы хранения выделять пространство более быстро и эффективно.

В долгосрочной перспективе, когда емкость системы хранения меньше предоставленного с помощью thin provisioning объема хранения, использование thin provisioning помогает экономить средства ИТ-бюджета.

### 7.2 Наблюдение за доступным пространством

Если используется thin provisioning, важно, чтобы свободное пространство на системе хранения своевременно контролировалось и наблюдалось. Правильно администрируемая система (например, своевременно обновляемая патчами и обновлениями безопасности и т.д.) не то же самое, что правильно наблюдаемая.

### 7.3 SnapDrive for Windows и thin provisioned LUN

Начиная с версии SnapDrive for Windows 5.0 стало возможным создавать LUN с отключенным резервированием пространства.

Так как выбор thin provision для LUN входит в зону ответственности администратора системы хранения, а не администратора хост-сервера, следует действовать следующим образом. Необходимо создать конфигурационный файл на самой системе хранения, который будет указывать SnapDrive for Windows создавать LUN на данном томе в режиме без резервирования пространства.

Для каждого тома, где SnapDrive for Windows должен создать LUN-ы без резервирования, необходимо создать файл с именем **sdparams.conf**, и со строкой в нем:

```
space_reservation_new_lun = off
```

Файл может быть создан либо из консоли системы хранения командой `wrfile` или через NFS/CIFS share на томе. Когда такая строка в этом файле будет найдена программой SnapDrive, то все создаваемые LUN-ы будут создаваться с отключенной *space reservations*. Начиная с версии SnapDrive for Windows 5.0 все проверки наличия space reservations, при запуске и подключении LUN в SnapDrive, были удалены.

## 7.4 Рекомендуемые конфигурации

Существует много способов сконфигурировать систему хранения NetApp для использования LUN с thin provisioning; каждый имеет свои преимущества и недостатки. Следует отметить, что можно иметь как том с thin provisioning, так и без, на одной системе хранения, и даже на одном aggregate . LUN-ы для критичных приложений следует конфигурировать без thin provisioning, тогда же, как LUN-ы для других приложений могут быть с thin provisioning. Приведенная конфигурация рекомендуется как одна из оптимальных:

Volume Guarantee	= none
Configuration guarantee	= none
LUN reservation	= enabled
fractional_reserve	= 0%
snap_reserve	= 0%
autodelete	= volume / oldest_first
autosize	= off
try_first	= snap_delete

Преимущества данной конфигурации в том, что свободное место на aggregate используется как общий пул свободного пространства. Недостатки конфигурации в высоком уровне зависимости между томами, и в том, что уровень thin provisioning не может быть настроен для тома индивидуально. При использовании этой конфигурации общий размер томов будет больше, чем доступно пространства на aggregate. С использованием этой конфигурации, администратор системы хранения будет планировать тома так, чтобы контролировать и управлять используемым пространством на aggregate.

- Конфигурация **autogrow/autodelete**

guarantee	= volume
LUN reservation	= disabled
fractional_reserve	= 0%
snap_reserve	= 0%
autodelete	= volume / oldest_first
autosize	= on
try_first	= autogrow

Преимуществом этой конфигурации является то, что она позволяет, если это необходимо, тонко настраивать уровень thin provisioning для каждого приложения. При использовании этой конфигурации, размер тома определяет или гарантирует количество места, доступного для LUN-ов этого тома. Уровень aggregate обеспечивает совместно используемый объема доступного пространства для всех томов, расположенных на нем. Если LUN-ы или снэпшоты требуют больше места, чем доступно на томе, том автоматически будет увеличен, заняв больше места на несущем его aggregate.

Величина thin provisioning задается для каждого тома отдельно, позволяя администратору, например, задать величину тома в 95% общего размера LUN-ов тома для более критичных приложений, и в 80% для менее критичных. Возможно настраивать то, как много совместно используемого пространства на aggregate будет использовано тем или иным томом, в случае использования опции *autogrow*, задав ее величину.

Преимуществом отключенного *LUN space reservation* является то, что снэпшоты могут использовать пространство, ненужное LUN-ам. Это также не опасно в плане исчерпания места и

для LUN-ов, так как в этом случае, функция *autodelete* удалит занявшие лишнее место снимки. Следует, однако, помнить, что снимки, использованные для создания из них клонов, не могут быть удалены с помощью *autodelete*.

## 7.5 Thin Provisioning при расширении LUN (LUN Growing)

Согласно способу использования LUN-ов, один из лучших способов выделить место для LUN на системе хранения NetApp, это распределить столько места, сколько ему нужно в данный момент, и позволить ему расти, по мере необходимости в дополнительном пространстве.

В отличие от других систем хранения, Data ONTAP позволяет такую стратегию с помощью своих средств гибкого и простого распределения пространства на стороне самой системы хранения. Кроме этого, NetApp предлагает программное средство SnapDrive, которое полностью автоматизирует изменение размера файловой системы, обеспечивает все необходимые действия как на хост-сервере, так и на системе хранения. Эти технологии позволяют администратору системы хранения наращивать размеры хранилища по мере необходимости, устраняя при этом проблему избыточно выделенного пространства хранения.

## 8 Выводы

Как мы рассмотрели выше, используя системы хранения NetApp и технологии FlexVol, возможно использовать методы thin provisioning как для LUN так и для пространства хранения снимков, что обеспечивает значительную эффективность и снижение затрат. Рекомендованные конфигурации, рассмотренные детально в этом документе, это отличная стартовая точка, которую вы можете модифицировать, как вам необходимо в случае конкретного решения, чтобы в точности соответствовать вашим нуждам. Разнообразные механизмы, как самого Data ONTAP, так и приложений верхнего уровня, таких как SnapDrive и SnapManager for Exchange, позволяют администратору настраивать уровень thin provisioning для точного соответствия потребностям системы и инфраструктуры в целом, и достигать максимума экономии ресурсов и средств.