Violin Memory: СХД для силиконовых ЦОД

Сравнение архитектурных особенностей all-flash массивов на флэш-модулях от компании Violin Memory с all-flash массивами построенными на "традиционных" SSD-дисках, а также с массивами на базе HDD.

Алексей Аверин – технический директор Violin Memory Россия/СНГ.

Введение

All-flash массивы – один из самых быстро растущих секторов рынка и границы их архитектурных особенностей размываются. Между тем, в настоящее время можно выделить два класса all-flash СХД: построенные на условно "традиционных" SSD-накопителях и на базе PCIe флэш-модулях.
В первом случае производитель СХД выступает как системный интегратор, покупая на рыке SSD и интегрируя их в свои массивы. Во втором – разрабатывается контроллер для флэш-чипов "с нуля" с учетом особенностей работы флэш. Violin Memory пошла по второму пути, разработав совместно с материнской компанией Toshiba флэш-контроллер VIMM (или Violin Intelligent Memory Module).

Рассмотрим некоторые отличительные особенности этих двух подходов.

Чем измерять производительность приложений?

Как правило, когда стоит выбор СХД, то в первую очередь обращают внимание на показатели IOPs и потоковую производительность. Между тем, вот уже 2 года все производители как флэш-технологий, так и флэш-решений ведут упорную борьбу за другой показатель – время отклика/ожидания (latency).

Как показывает статистика, при работе с внешними СХД большинство OLTP-приложений чувствительны к времени отклика, а не к IOPs, т.к. бОльшую часть времени приложение находится в состоянии ожидания данных от СХД.

Время отклика традиционной СХД – 1-20 мс (вне зависимости от класса: high-end или mid-range), при этом возможна деградация производительности.

Среднее время ожидания ввода/вывода на процессоре сервера может достигать 80% и более (рис. 1).

Медленный ввод-вывод приводит к следующему:

- к замедлению работы приложения;
- снижению эффективности вычислительных ресурсов (ожидание ответа от СХД вместо работы);
- к необходимости «перезакладывать» СХД (кэш-память, емкость, контроллеры, шкафы);
- к неэффективной совокупной стоимости владения (TCO).

Массивы на базе SSD позволяют получить время отклика чуть лучше, чем при использовании жестких дисков, а массивы с разработанной "с нуля" архитектурой обеспечивают стабильно низкое время отклика – < 1 мс. При этом задержки Violin-массивов не зависят от процентного соотношения операций чтения/записи. Для all-flash массивов на SSD это не так: и зависимость намного больше, и, как правило, задержки больше.

Время отклика флэш-СХД Violin Memory 6248 и 6264 < 350 мкс. При этом достигается следующее:

- кардинально сокращается время ожидания ввода/вывода на процессоре сервера – до 5% (рис. 2);
- КПД сервера увеличивается до 20 раз;
- ускорение OLTP-приложений от 2 до 20 раз;

Рис. 1. В существующих ЦОД традиционная СХД (на жестких дисках) является "узким местом" при работе OLTP-приложений, т.к. бОльшую часть времени приложение находится в состоянии ожидания данных.

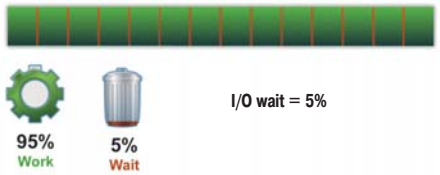

Рис. 2. При использовании флэш-СХД Violin Memory время ожидания сокращается до 5%.

Табл. 1. Примеры эффективности использования массивов Violin на различных нагрузках.

Область	Тип нагрузки	Влияние	Результат
Бизнес-аналитика (SAP)	Комплексные запросы	4 мин → 3 сек	80x улучшение
BI-нагрузка	ETL	1 час → 10 мин	6x улучшение
VDI(VMware View, Intel Sandy Bridge)	Рабочие места	10 → 25 польз. на ядро	2.5x больше польз./ядро (2,560 ядер!)
OLTP-отчеты (SAP BW)	Комплексные запросы	24 часов → 3 часа	8x улучшение
CRM-система (Remedy)	Пользовательские запросы	8-10 мин → 30 сек	20x улучшение
Биллинг (Keenan)	Месячный биллинг	72 час → 22 час	330% улучшение
Управление цепочками поставок (SAP Sales Distribution, Materials Management)	ERP	7 час → 2 час	350% улучшение
VDI (XenServer, Intel)	Рабочие места	6 → 15 польз. На ядро	Меньше портов SAN (112) и серверов (56) для 3000 пользователей

- появляются дополнительные возможности консолидации вычислительных мощностей;
- сокращение требуемого ПО СУБД, лицензируемого по ядрам сервера;
- снижаются операционные расходы в ЦОД до 80%.

Тестирование массивов Violin на различных нагрузках компанией "КРОК" (*см. SN № 1/53, 2013, www.storagenews.ru)* показало повышение производительности приложений до 80 раз (табл. 1).

Насколько важен профиль OLTP-нагрузки?

Одна из ключевых особенностей массивов Violin – автоматическая балансировка нагрузки на уровне флэш-памяти – VIMM, что позволяет увеличивать нагрузку до максимального значения при практически неизменном времени отклика (рис. 3).

Важная особенность флэш-чипов состоит в том, что другую физику по сравнению с магнитной записью, используемой в жестких дисках, – к операциям чтения и записи добавляется стирание. Другой момент – операции записи по длительности занимают гораздо больше времени, чем чтение. Это приводит к тому, что при использовании многих SSD-дисков

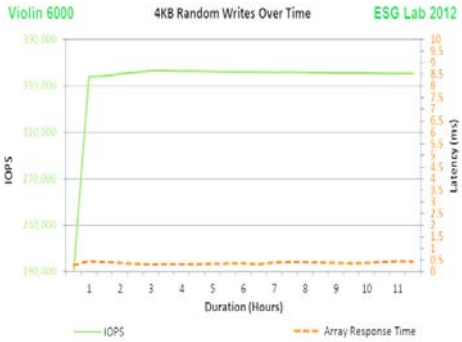

Рис. 3. Производительность флэш-СХД Violin Memory время не меняется по мере заполнения массива.

даже небольшой процент активной записи (например, 10% – запись, 90% – чтение) в ряде случаев может приводить к многократному (в отдельных случаях на порядок и более) снижению производительности.

Алгоритмы Violin Memory построены так, что система одинаково быстро справляется как с чтением, так и записью. Это является ее ключевым преимуществом перед существующими аналогами, использующими SSD-диски.

Всегда ли нужна "always on" дедупликация?

Что такое "always on" (всегда включенная) дедупликация?

Возрастающая популярность флэш-технологии в мире обеспечивается, прежде всего, как снижением ее стоимости, так и возможностью резко снизить задержки на случайных операциях ввода-вывода, что, в свою очередь, позволяет на порядок повысить скорость работы приложений. Однако флэш-технология несет и ряд проблем органически ей присущих.

Одна из них – износ флэш-ячеек. От этой особенности не может избавиться ни один поставщик all-flash массивов. Флэш NAND-ячейка способна выполнить только фиксированное число операций записи, прежде чем ее производительность существенно снизится. И в какой-то момент ячейка становится неработоспособной. Чтобы устранить эту проблему каждый флэш-контроллер имеет функцию управления жизнью флэш-ячейки.

Дедупликация – это процесс удаления избыточных блоков данных. Она достигла совершеннолетия в 1990-х годах и изначально разрабатывалась для резервного копирования. Дедупликация позволяла писать на ленту меньшее количество данных, снижая потребность в лентах и одновременно уменьшая влияние окна для резервного копирования (меньше данных – меньше окно) на работу продуктивных приложений. Позднее дедупликация стала использоваться в новом контексте – как способ уменьшения износа флэш-памяти за счет записи меньшего объема данных. В частности, ряд производителей all-flash массивов на SSD поддерживают дедупликацию. При этом опция "всегда включено" во флэш-массивах используется лишь с целью повысить ресурс флэш-памяти, а не из-за того, что от дедупликации выигрывают приложения.

Подход "Always On" дедупликации

Все производители all-flash массивов, которые используют твердотельные диски (SSD), зависят от поставщика SSD, который обеспечивает управление флэш-ячейками на контроллере SSD. Существует также программное обеспечение, устанавливаемое на контроллера в массиве, где дедупликация может быть использована для уменьшения количества операций записи, следовательно, для продления срока службы флэш-памяти в массиве.

Некоторые приложения очень хорошо используют дедупликацию, например, VDI (virtual desktop infrastructure), которые снижают объем хранимых данных до 90% (и, следовательно, уменьшают степень износа флэш-памяти). Виртуальная серверная инфраструктура (VSI – Virtual server infrastructure) также позволяет извлекать большую пользу от дедупликации, экономя до 65% объема хранения (и, соответственно, сокращая износ флэш-памяти). Тем не менее, не все приложения являются подходящими для дедупликации.

Проблемы с подходом "Always On" дедупликации

Базы данных являются примером приложения, которое не может быть использована с дедупликацией. Существует небольшое преимущество при использовании дедупликации для баз данных, хотя не такое значительное (коэффициент снижения объема данных – 1,3–1,5[1]), как с VDI или VSI. Большой проблемой является то, каким образом СУБД хранит данные. Реляционные СУБД, такие как Oracle, не имеют повторяющихся блоков данных, потому что каждый блок в таблицы (логический контейнер, в котором хранятся таблицы и индексы) содержит уникальный ключ в начале и контрольную сумму, содержащую часть этого ключа, в конце. В результате достигается незначительная экономия пространства при увеличенных задержках из-за работающей дедупликации.

Вполне возможно, что эффект от дедупликации может быть достигнут, например: 1) клиентами, хранящими копии своих баз данных на одном массиве (но это место лучше использовать для моментальных снимков); 2) в случае дедуплицирования незанятого пространства (но для этого лучше использовать "тонкое выделение ресурсов" – thin provisioning); 3) в случае дедуплицирования служебных файлов (Oracle сознательно хранит несколько копий логов, управляющих файлов и т.д.).

Другая нагрузка, которая обычно не подходит для дедупликации – шифрование данных. Шифрование, в соответствии с концепцией, – уникальный поток данных, где дедупликация только добавляет задержку. Если есть необходимость в дедуплицировании зашифрованных данных, необходимо иметь доступ к незашифрованным данным, чтобы система хранения могла идентифицировать дубликаты. Это означает, что шифрование данных не может выполняться в рамках приложения, если необходимо дедуплицировать данные. Любая обработка зашифрованных данных при хранении должна проводиться очень осторожно, чтобы сохранить безопасность данных.

Решение Violin: другой подход

Violin использует другой подход, поскольку полностью самостоятельно управляет флэш-архитектурой и не нуждается в дедупликации для повышения срока службы флэш-памяти. Поэтому дедупликация может безболезненно отключаться в тех случаях, где она неэффективна или несет дополнительные риски.

Violin Flash Fabric Architecture™ (FFA) работает напрямую с флэш-ячейками и может управлять сроком службы на уровне массива. Это позволяет избавиться не только от "горячих" флэш-ячеек, которые способствуют их преждевременному выгоранию, но и получить повышенную производительность за счет встроенного в архитектуру параллелизма. При выполнении операций записи и управлении флэш (в том числе и сбор мусора) все действия производятся на уровне массива, ресурс флэш продлевается и при этом не вносятся задержки как на ряде all-flash SSD-массивах.

Единое пространство для больших данных

С анонсированием в июле 2014 г. линейки Concerto 7000 массивы Violin стали предоставлять 264 Тбайт единого пространства для развертывания файловых систем и томов данных. В отличие от SSD-массивов, это пространство доступно без каких-либо накладных расходов и дополнительных задержек.

Упрощение управления

В заключение – о некоторых отличительнных особенностях all-flash массивов в сравнении с традиционными на жестких дисках.

При использовании массивов Violin Memory существенно упрощается вся архитектура СХД и, соответственно, управление – отпадает необходимость в построении многоуровневых RAID (0, 1, 4, 5, 6,10 и др., включая и кэширование данных на уровне СХД) с использованием различных типов дисков – SAS, SATA, Fibre-Channel или SSD, если весь объем данных для работы продуктивных приложений можно разместить на одном массиве. При этом обеспечивается надежность на уровне 99,9999%, а также прогнозируемое стабильное время доступа к данным.

В системе отсутствует единая точка отказа на всех уровнях, а замену всех компонент и микрокода можно производить без останова системы. Надежность на уровне работы ячеек обеспечивается запатентованными механизмами работы с флэш-чипами, которые продлевают жизнь ячеек до 10 раз (постоянная очистка ячеек, исправление сбоев ячеек, равномерное распределение нагрузки на ячейки).

Алексей Аверин,
Violin Memory Россия/СНГ.

1) IDC, "Why Inline Data Reduction Is Required for Enterprise Flash Arrays", *Sponsored by: Violin Memory, Eric Burgener, September 2014.*