МАСШТАБИРОВАНИЕ OLTP ПРИЛОЖЕНИЙ:

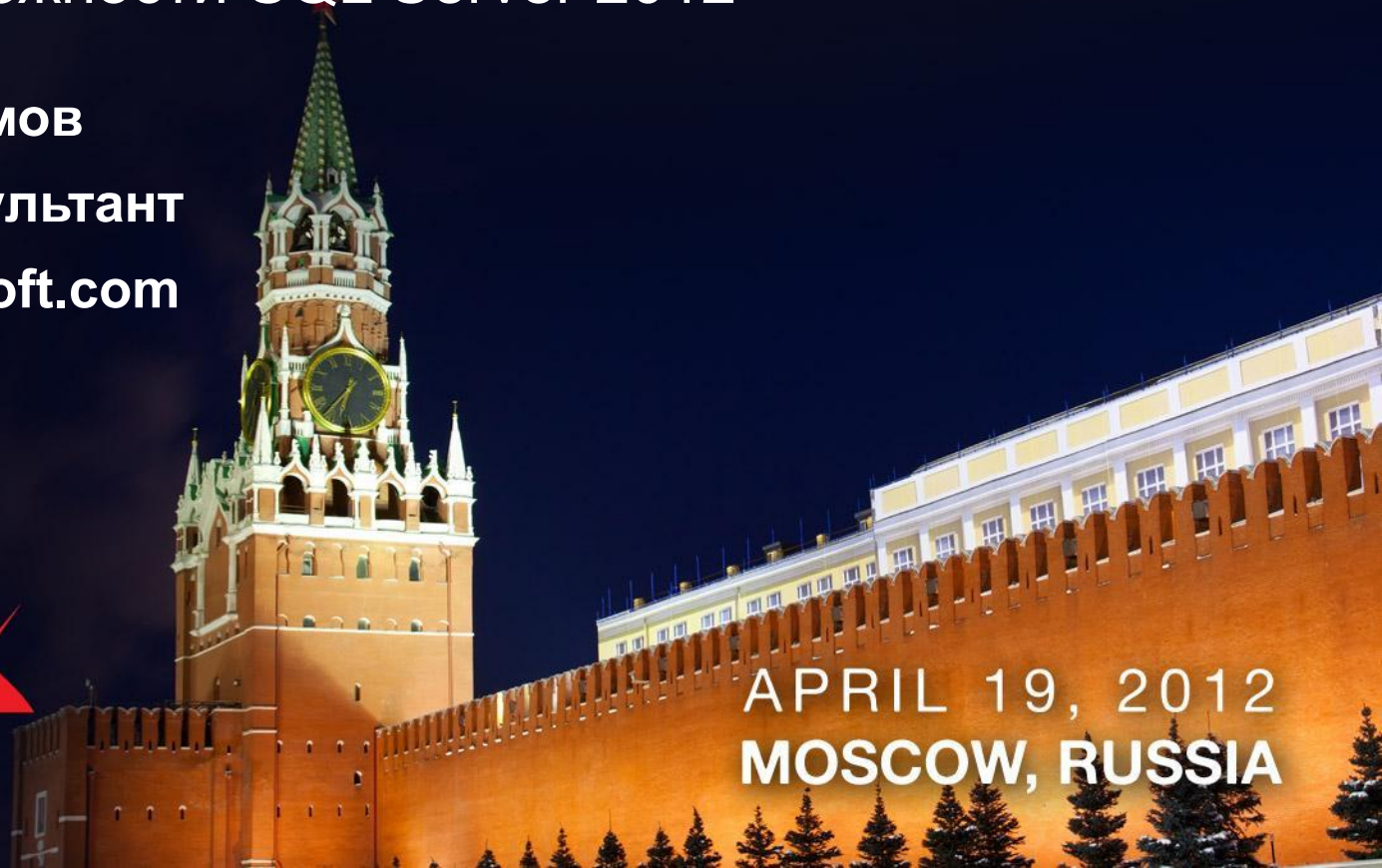
Проектирование приложений и оптимизация серверной части

Новые возможности SQL Server 2012

Дмитрий Артемов

Старший консультант

dimaa@microsoft.com



PASS SQLRally RUSSIA 2012

Аннотация

- SQL Server многократно доказал возможность поддержки самых нагруженных OLTP приложений
- SQL Server 2012 предлагает дополнительные возможности в части масштабируемости и производительности
- Возможные подходы по оптимизации ПО и серверной части для обеспечения масштабируемости

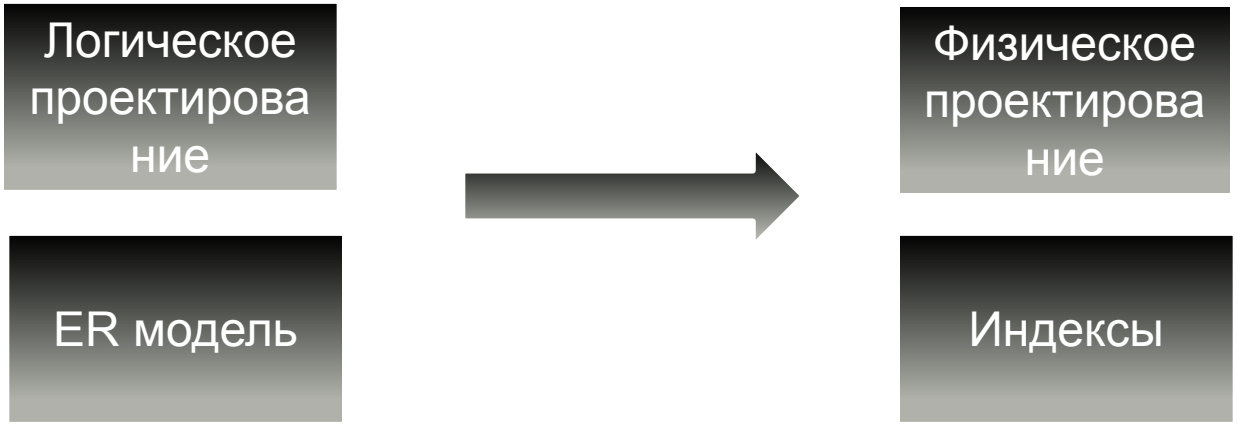
План

- Особенности OLTP нагрузки
- Принципы построения OLTP приложений
- Что определяет и что мешает масштабируемости
- SQL Server 2008 R2 - обеспечение производительности и масштабируемости
- SQL Server 2012 - обеспечение производительности и масштабируемости
- SQL Server 2012 Distributed replay для тестирования (регулярного, при обновлении, проверке совместимости,…)
- Вертикальное масштабирование - "тяжелое железо"
- Заключение

Особенности OLTP нагрузки

- Обычно организуется бизнес приложениями
- Интенсивные запись и чтение
- Относительно «мелкие» вставки и обновления
- Высокая интенсивность транзакций: десятки тысяч в секунду
- Транзакции обычно коротки: захватывают 1–3 таблицы
- Иногда транзакции реализованы многоступенчатыми: например для финансовых приложений
- Работа с относительно небольшими объемами данных

Принципы построения приложений



- При проектировании следует учитывать, что SQL Server ориентирован на работу с наборами данных (set-oriented processing)
- Для проектирования рекомендуется Visual Studio\Entity Framework и DTA (Database Tuning Advisor) для оптимизации индексов
  - DTA – только инструмент и не всесилен

Что следует учитывать при проектировании

- Любое приложение требует качественного проектирования, высоко нагруженное – особенно
  - Качественный логический (E-R Model) и физический (индексы) дизайн – верный путь к производительности
- SQL Server работает с наборами данных, перебирать индивидуальные записи – путь к проблемам
- Помимо качественного проектирования нужно помнить, что БД требует обслуживания
  - Обновление статистики
  - Обслуживание индексов
- Для облегчения работы можно использовать DTA, но думать все равно придется
- Множественные соединения -> риск некачественных планов
  - Промежуточные таблицы часто работают быстрее
- Теперь о физическом дизайне...

Что следует учитывать при физическом проектировании

- Изменения физического дизайна могут быть вызваны
  - Требованиями производительности
  - Требованиями доступности
  - Требованиями разграничения доступа
  - Требованиями аудита
- По возможности следует разделять файлы данных и журналы транзакций
- Не пожалейте время на анлиз работы/использования индексов
  - Sys.dm_db_index_...
- Дисковая активность OLTP приложений требует от стойки большого числа операций ввода\вывода в секунду
- Дисковая активность аналитических приложений требует от дисковой стойки большого объема ввода\вывода (МБ\Сек)
  - Смешанные приложения требуют и того и другого
  - 1 ядро способно перемолоть до 200 Мб\сек
  - 1 диск (классический) способен обеспечить 150-180 операций ввода\вывода в секунду
  - Канал **Диск-Контроллер-HBA-Процессор** должен иметь соответствующую мощность

Рекомендации по кластерным индексам

Очень кратко, полноценное обсуждение требует отдельного разговора

- Хороши для ранговых запросов
  - Разумно создавать на часто используемых полях, участвующих в запросе в форме (JOIN и WHERE с указанием “=“, “<“, “>“, “BETWEEN”)
  - Если возврат невелик, некластерный индекс также эффективен
  - Предпочтителен на узких, монотонных, редко изменяемых полях с невысокой повторяемостью данных
- Ничего не дается даром:
  - Обновления реорганизуют таблицу (page split)
    - Негативное влияние на производительность
    - Со временем фрагментация может нарастать (GUID – идеальный пример)

Рекомендации по некластерным индексам

- Создавайте для полей, часто участвующих в поиске
- Используйте на узких полях с невысокой повторяемостью
- Используйте на foreign key constraints (для поддержки join)
- Проверьте возможность использования «покрывающих» индексов
  - Проверьте возможность использования included полей
- Не забывайте: стоимость сопровождения
  - Интенсивные обновления требуют дополнительных усилий по актуализации индексов
- Рассмотрите возможность удаления индексов с небольших таблиц

План

- Особенности OLTP нагрузки
- Принципы построения OLTP приложений
- Что определяет и что мешает масштабируемости
- SQL Server 2008 R2 - обеспечение производительности и масштабируемости
- SQL Server 2012 - обеспечение производительности и масштабируемости
- SQL Server 2012 Distributed replay для тестирования (регулярного, при обновлении, проверке совместимости,…)
- Вертикальное масштабирование - "тяжелое железо"
- Заключение

Что определяет и что мешает масштабируемости

Измерения

- Интенсивность транзакций
- Число одновременно работающих пользователей
- Объем данных и темпы роста объемов

Ресурсы

- ЦПУ
- Память
- Ввод\вывод
- Сеть

Основной подход в проектировании и оптимизации: разделяй и властвуй

Типичные проблемы перегруженного CPU

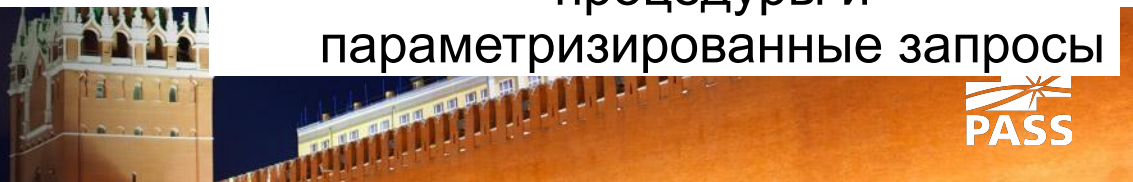
Симптомы

- Компиляция и рекомпиляция планов
  - Plan reuse < 90% - предмет для анализа
- Параллельные планы исполнения
  - Ожидания типа CXPACKET > 10% от суммарного времени ожидания
- Большое число runnable tasks или значительные ожидания SOS_SCHEDULER_YIELD

Причины

- Запросы не параметризированы
- Неэффективные планы
- Недостаточное использование хранимых процедур (большинство запросов отсылаются напрямую)
- MAXDOP > 1
- Статистика устарела
  - Trace flag 2371 для SQL Server 2008 R2 SP1
- Массивные сканирования
  - Как результат неэффективных планов
    - Как результат устаревшей статистики
- Изменения настроек SET внутри SP

Следует максимально использовать хранимые процедуры и параметризированные запросы



Типичные проблемы перегруженной стойки

Симптомы

- Высокое значение времени отклика дисковой подсистемы (> 30 msec) для «шпиндельных» устройств
- В числе первых находятся ожидания - ASYNCH_IO_COMPLETION, IO_COMPLETION, LOGMGR, WRITELOG, PAGEIOLATCH_x

Причины

- Массивные сканирования (для некачественных планов)
- Отсутствие покрывающих индексов
- Смешанный характер приложения:
  - Одна и та же БД обслуживает OLTP и аналитику
- Чрезмерная нагрузка на TempDB
- Недостаток шпинделей, узкий канал через HBA

OLTP приложения создают множество мелких операций ввода\вывода случайного характера (Random IO)

Типичные проблемы блокировок

Симптомы

- Высокие значения ожиданий на блокировку записей или latch
- Видны при мониторинге
  - sp_configure “blocked process threshold” и Profiler “Blocked process Report”
  - Наиболее значимые ожидания групп LCK_x. Видны по данным sys.dm_os_wait_stats.

Причины

- Завышенный уровень изоляции
- Высокие затраты на обслуживание индексов
- Эскалация блокировок
- Низкая производительность подсистемы ввода\вывода
- Проблема генерации последовательных номеров

Использование RCSI/Snapshot isolation может помочь делу

Типичные проблемы использования памяти

Симптомы

- Показатель Page life expectancy < 300 сек
  - Для серверов с большим объемом памяти и гораздо большее значение может быть тревожным
- SQL Cache hit ratio < 99%
- Lazy writes/sec постоянно активны
- Ошибки Out of memory

Причины

- Слишком много массивных сканирований (I/O)
- Неоптимальные планы
- Внешнее (от других процессов) давление по памяти

Постарайтесь избавиться от сканирований в планах

Используйте WSRM для иных чем SQLServer процессов на сервере

План

- Особенности OLTP нагрузки
- Принципы построения OLTP приложений
- Что определяет и что мешает масштабируемости
- SQL Server 2008 R2 - обеспечение производительности и масштабируемости
- SQL Server 2012 - обеспечение производительности и масштабируемости
- SQL Server 2012 Distributed replay для тестирования (регулярного, при обновлении, проверке совместимости,…)
- Вертикальное масштабирование - "тяжелое железо"
- Заключение

Средства обеспечения производительности и масштабируемости в среде SQL Server 2008 R2

- Оптимизация планов запросов
  - Plan guides
  - Optimize for Unknown
- Указания по эскалации блокировок
- Resource governor
- Развитые средства диагностики – Xevent, DMV’s
- Поддержка более чем 64 процессоров
- Динамическая привязка процессоров (affinity - программная или аппаратная)
- Поддержка «горячей» замены процессоров
- Сжатие данных
  - Особенно, если есть проблемы со вводом\выводом
- Секционирование
  - До 15000 секций
- Snapshot Isolation, RCSI
- Utility Control Point для группового мониторинга серверов

Plan Guide

- Направить оптимизатор в верном направлении с фиксированным планом
- Обеспечивает стабильность поведения
- План можно извлечь напрямую из кеша
- При невозможности применения запрос все равно выполняется
- **Использовать при невозможности изменить приложение**
- Простой пример
  - *SELECT TOP 1 * FROM Sales.SalesOrderHeader ORDER BY OrderDate DESC;*
  - **sp_create_plan_guide** @name = N'Guide2', @stmt = N'SELECT TOP 1 * FROM Sales.SalesOrderHeader ORDER BY OrderDate DESC', @type = N'SQL', @module_or_batch = NULL, @params = NULL, @hints = N'OPTION (MAXDOP 1)';

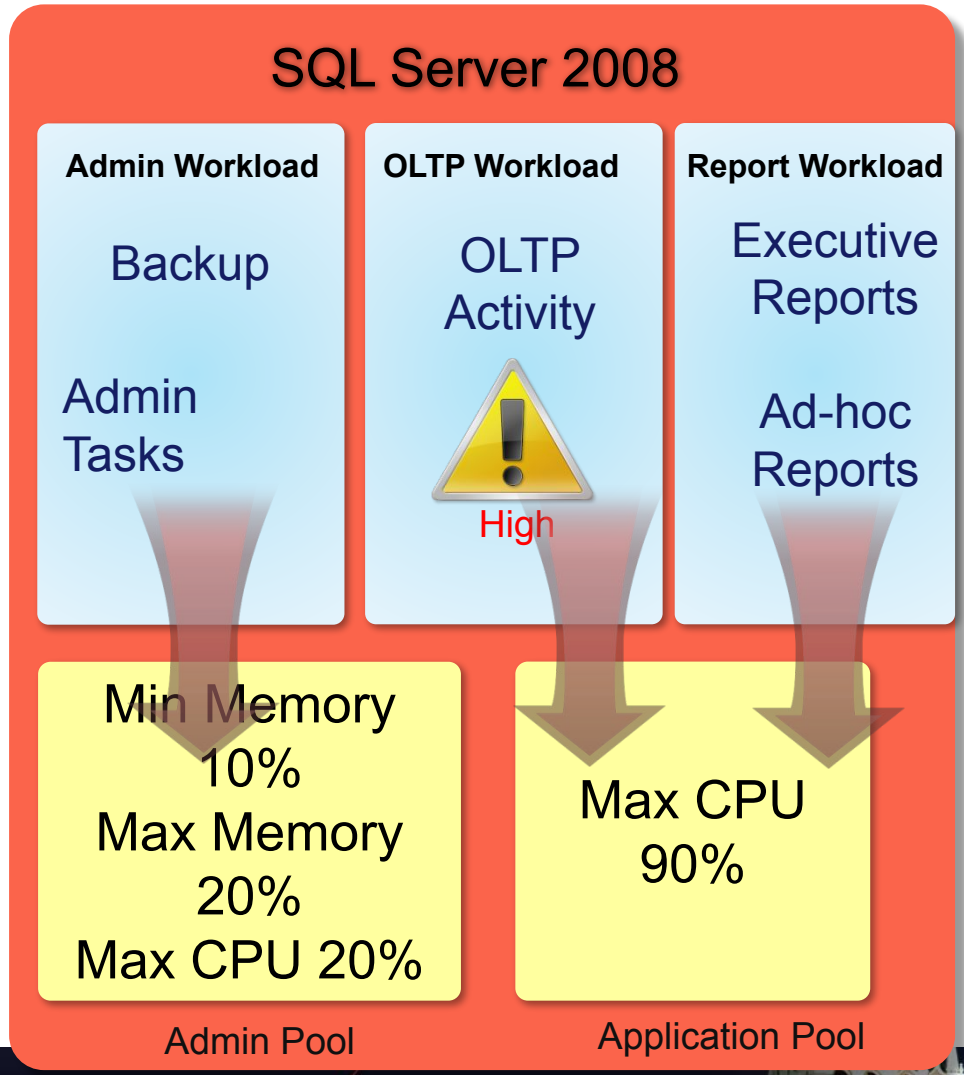
Optimize for Unknown

- OPTIMIZE FOR UNKNOWN
  - Указатель велит оптимизатору рассматривать запрос, как если бы ему не передавали параметров
  - Помогает решить проблему создания некачественных планов для определенных значений параметров
  - Пример
    - *@p1=1, @p2=9998,*
    - *SELECT * FROM t WHERE col > @p1 or col2 > @p2 ORDER BY col1 OPTION (OPTIMIZE FOR (@p1* ***UNKNOWN****, @p2* ***UNKNOWN****))*

Управление эскалацией

- Перед отключением эскалации нужно проверить действительно ли она – причина проблем
- Эскалацию отключать следует на уровне объекта или таблицы
- Разрешите выполнить эскалацию на уровень секции
  - *ALTER TABLE T1 SET (LOCK_ESCALATION = AUTO);*
  - *Может приводить к невиданным ранее Deadlock'ам!*
- При эскалации на уровне секции эскалация на уровне таблицы не происходит

Resource Governor



Преимущества

- Прогнозируемое качество услуг
- Предотвращает run-away запросы
- Контролирует «недисциплинированные» приложения
- Обеспечение сценариев DW & Консолидации

SQL Server 2008 RG

- Группы нагрузки связываются с Ресурсным пулом
- Динамическое изменение групп и пулов
- Мониторинг ресурсов в реальном времени
- До 20 ресурсных пулов (**64 в SQL Server 2012**)

Extended Events (XEvent)

- Высокопроизводительный и расширяемый механизм трассировки событий
- Динамический сбор данных при возникновении событий
- Интегрирован с ETW (Event Tracing for Windows)
  - Позволяет выяснить взаимосвязь между событиями Windows и приложениями третьих фирм
- SQL Server способен создавать сотни перехватываемых событий
- Позволяет (в том числе) набирать статистику ожиданий на уровне сессии/индивидуальной команды

План

- Особенности OLTP нагрузки
- Принципы построения OLTP приложений
- Что определяет и что мешает масштабируемости
- SQL Server 2008 R2 - обеспечение производительности и масштабируемости
- SQL Server 2012 - обеспечение производительности и масштабируемости
- SQL Server 2012 Distributed replay для тестирования (регулярного, при обновлении, проверке совместимости,…)
- Вертикальное масштабирование - "тяжелое железо"
- Заключение

SQL Server 2012 – Availability groups

- Распределение нагрузки по нескольким серверам
  - Оперативная обработка
  - Отчетность
  - Администрирование (резервные копии, DBCC,…)

Расширение числа Online операций в SQL Server 2012

Online операции

- Больше операций от прикладного слоя, меньше блокировок, выше производительность

- Добавление not-NULL поля с указанным DEFAULT
  - ALTER TABLE <mytable> ADD COLUMN <column> <type> DEFAULT <default_value>
- Операции CREATE, ALTER и DROP INDEX поддерживают ONLINE режим для таблиц/индексов с BLOB типами данных

Максимальное число секций

- SQL Server 2005 поддерживает до 1000 секций
  - 1000 секций позволяют
    - Секция каждый час => 1 месяц (24 x 31 = 744)
    - Секция каждый день => 2 года (365 + 366 = 731)
- SQL Server 2008 R2/2012 поддерживает 15000 секций
  - Секция каждый час => 1 год (24 x 366 = 8784)
  - Секция каждый день => 40 лет (40 * 365 + 10 = 14 610)

15,000 секций в SQL Server 2012

- Портирована в предыдущие версии 2008 SP2 и 2008 R2 SP1 но
  - По умолчанию присутствует в SQL Server 2012
  - Имея >1,000 секций в SQL Server 2012 поддерживаются
    - database mirroring
    - log shipping
    - репликация
    - SSMS
  - Повышение производительности – оптимизация использования памяти
  - Изменен алгоритм сбора статистики при более чем 1000 секций – sampling
  - Оптимизирован алгоритм блокировок. Нет длительных задержек на schema stability lock
- Не поддерживается
  - Более 1000 секций на таблицу\индекс на x86 системах
  - Не выравненные индексы на таблице с более чем 1000 секциями

План

- Особенности OLTP нагрузки
- Принципы построения OLTP приложений
- Что определяет и что мешает масштабируемости
- SQL Server 2008 R2 - обеспечение производительности и масштабируемости
- SQL Server 2012 - обеспечение производительности и масштабируемости
- SQL Server 2012 Distributed replay для тестирования (регулярного, при обновлении, проверке совместимости,…)
- Вертикальное масштабирование - "тяжелое железо"
- Заключение

Тестирование приложений

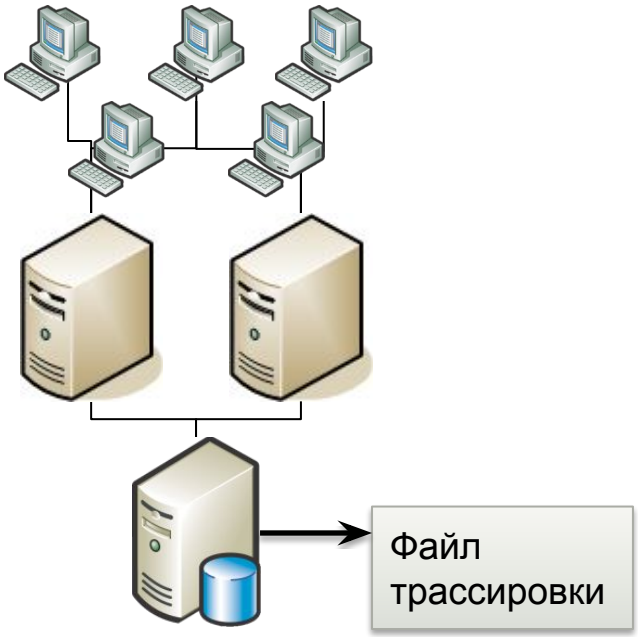
- Более тщательное тестирование производительности\ функциональности
- Способно определить потенциальные несовместимости версий, не распознаваемые Upgrade Advisor:
  - Тестирование ad-hoc T-SQL, создаваемых прикладным слоем
  - Тестирование фактического исполнения T-SQL, не только синтаксис
  - Тестирование конфигурации разграничения доступа
    - Может включать настройки уровня операционной системы
  - Определение различий в планах, длительности и результатах
  - Определение использования недокументированных объектов/возможностей
  - Определение очень редких, но возможных случаев, когда новая версия не отрабатывает запрос или меняет поведение запроса, но (пока) не документирована

- Не является обязательным для всех приложений, но рекомендуется для
  - Критичных приложений
  - Сложных приложений
  - Приложений, в которых запросы в основном создаются вне SQL Server и/или динамически

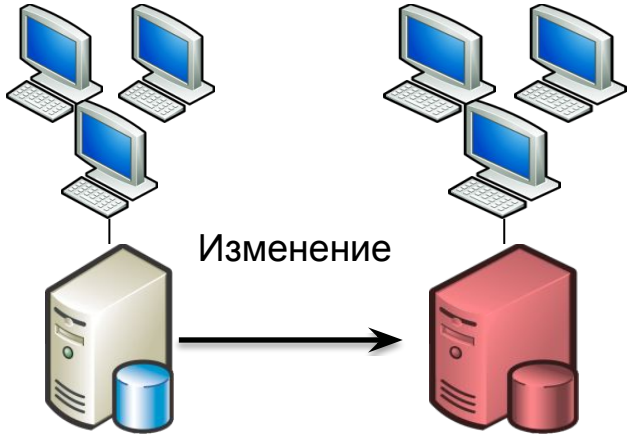
Тестирование приложений

Контроль качества/Пром. среда

Тестовая среда



Файл трассировки



Изменение

Перехват

Воспроизведение (до)

Воспроизведение (после)

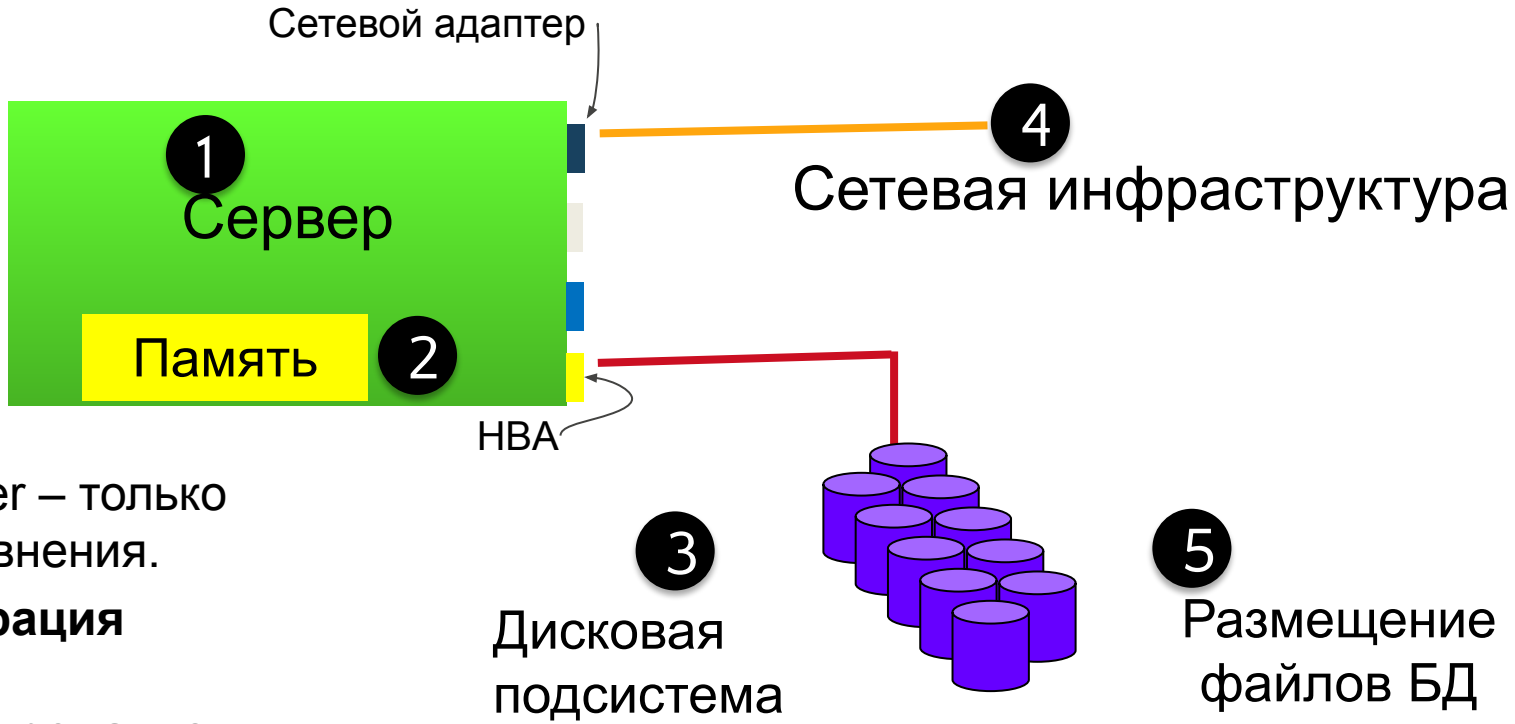
Сравнение Отчет

План

- Особенности OLTP нагрузки
- Принципы построения OLTP приложений
- Что определяет и что мешает масштабируемости
- SQL Server 2008 R2 - обеспечение производительности и масштабируемости
- SQL Server 2012 - обеспечение производительности и масштабируемости
- SQL Server 2012 Distributed replay для тестирования (регулярного, при обновлении, проверке совместимости,…)
- Вертикальное масштабирование - "тяжелое железо"
- Заключение

Основные компоненты

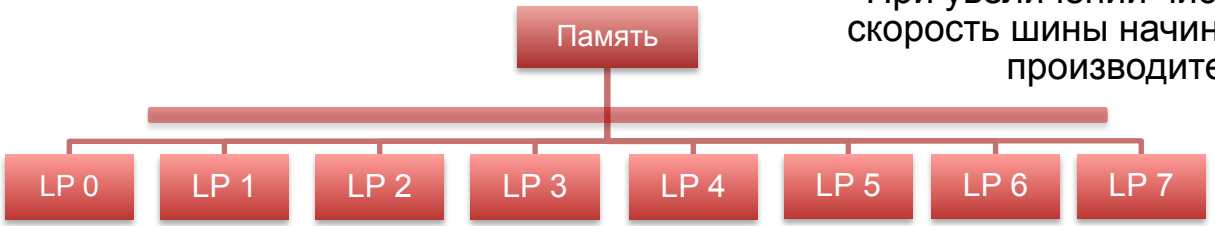
Задача – построить сбалансированную систему



SQL Server – только часть уравнения.

Конфигурация должна масштабироваться.

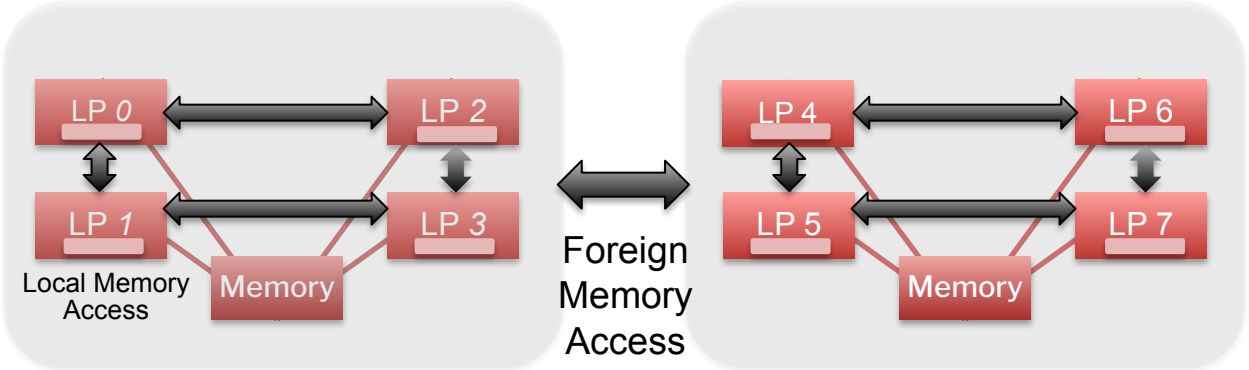
Концепция - NUMA



При увеличении числа процессоров, скорость шины начинает ограничивать производительность

Symmetric Multiprocessor Architecture

Поддержка более > 64 процессоров использует NUMA



Non-Uniform Memory Access

Foreign memory access > local memory access

Конфигурация дисковой подсистемы

Тенденция

- За последние 10 лет емкость дисков выросла в 100 раз
- При этом время доступа снизилось только в 10 раз
- Конфигурация нагруженных систем должна учитывать не только емкость, но интенсивность и характер нагрузки
- Solid State Disks становятся все более популярными

Конфигурация

- Масштабирование посредством увеличения числа HBA, дисков
- При использовании рекомендованного RAID 10 используйте HBA, способные выполнять одновременное чтение с дисков зеркала
- Для балансировки нагрузки следует использовать multipathing
- HBA Queue Depth – значение умолчания - 32 часто слишком мало
- Конфигурация должна обеспечивать малое время отклика < 10 msec

Для OLTP важно IOPs

Для аналитических систем важно MB\Sec

Один процессор способен потребить 200 MB\sec

Сетевая инфраструктура

Тенденции

- Gigabit – теперь стандарт
- Сетевые карты 10GBit Ethernet доступны для использования – особенно востребованы для iSCSI
- Полоса пропускания часто не является проблемой
  - Узким местом часто является нехватка мощности для параллельной обработки сетевых прерываний

Конфигурация

- Используйте Windows Server 2008 R2
- Предлагает распределенную обработку DPC на множестве процессоров
- Рекомендуется по одной сетевой карте для NUMA узла; максимально 4 - 8 ядер на сетевую карту
- Используйте Adapter teaming

Используйте Windows Server 2008 R2 для доступа к новому функционалу

Что мы можем – SQL Server в крупнейших инсталляциях

Категория	Метрика
Самая большая БД	80 TB
Самая большая таблица	20 TB
Суммарно данных у одного заказчика	2.5 PB
Максимальное число транзакций в секунду на одну БД	36,000
Самая производительная дисковая подсистема в промышленной эксплуатации	18 GB/sec
Самый быстрый куб «реального времени»	15 sec latency
Скорость загрузки 1TB	20 minutes
Самый большой куб	4.2 TB

Заключение

- SQL Server и Windows вместе обеспечивают масштабируемую среду для поддержания самых нагруженных OLTP приложений
- Качественное проектирование (и тестирование) – путь к надлежащей масштабируемости

Спасибо,
Вопросы