Курс дистанционного обучения

Ускорение и оптимизация систем на 1С:Предприятие 8.3

Подготовка на 1С:Эксперт по технологическим вопросам

Проект Курсы-по-1С.рф
2016

Оглавление

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Введение

В данном методическом пособии представлены материалы курса **Оптимизация и ускорение 1С:Предприятие 8 и подготовка к 1С:Эксперт по технологическим вопросам**.

Информация о курсе

Довольно часто специалисты сталкиваются с тем, что вся система или отдельные операции начинают работать медленно. В таких случаях одного только умения программировать недостаточно, без специальных знаний и навыков достаточно сложно найти причину замедления.

Апгрейд оборудования чаще всего не решает проблему, обычно он лишь устраняет симптомы, а не причину, и со временем замедление проявляется снова.

В данном курсе рассматриваются вопросы оптимизации и ускорения различных операций в системе 1С:Предприятие 8, разбираются как платные, так и бесплатные инструменты для расследования и анализа проблем производительности.

Рассмотрены вопросы настройки кластера серверов и сервера СУБД (на примере MS SQL Server), создание регламентных заданий по обслуживанию базы данных, инструменты автоматизированного тестирования и многое другое.

Курс ориентирован на работу в клиент-серверном режиме, поэтому для успешного прохождения, необходимо обладать основными знаниями о клиент-серверной архитектуре, принципах работы реляционных СУБД и иметь опыт написания клиент-серверного кода.

Схема работы с курсом

Если вы хотите получить максимальный эффект от обучения, тогда нужно следовать некоторым рекомендациям:

Занимайтесь каждый день. Лучше заниматься каждый день понемногу, чем 1 раз в неделю, но много. Рекомендуется выделять ежедневно 2-3 часа на занятия. Теорию, представленную в методическом пособии, можно загрузить на телефон или планшет и изучать в свободные минуты, например, в дороге. Лучше не планировать параллельное изучение других курсов, т.к. материала много и потребуется полная отдача.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Конспектируйте основные моменты. В обязательном порядке заведите конспект, это сильно повышает эффективность обучения и помогает запомнить больше материала. Конспект желательно просматривать время от времени.

Повторяйте за преподавателем. Пробуйте воспроизвести действия из курса на своей базе.

Применяйте знания на практике. Сразу применяйте полученные знания в своей работе, не нужно ждать, пока вы пройдете весь курс.

Повторяйте сложный материал. Информация, изложенная в курсе, довольно непривычна для рядового программиста 1С, поэтому она может показаться сложной для понимания с первого раза, это нормальная ситуация. Если что-то непонятно, попробуйте прочитать или пересмотреть урок еще несколько раз. Если остались вопросы, задавайте их в Мастер-группе.

Выполняйте практические задания. Обязательно выполняйте практические задания. В курсе есть обязательные и необязательные задания. Необязательные задания нужно выполнять, если вы хотите получить максимум от курса или если планируете сдавать экзамен 1С:Эксперт.

Пересматривайте курс после прохождения. Данный курс писался с прицелом на постоянное использование. Информации много, но, благодаря упорядоченным разделам, вы всегда можете «освежить» знания по любой из тем. Используйте это методическое пособие как справочник, и вы обретете незаменимого помощника для решения проблем производительности.

Используйте возможности Мастер-группы. Задавайте возникающие у вас вопросы и читайте вопросы других участников. Часто в Мастер-группе разбираются темы и вопросы которые не рассмотрены в курсе, что сильно повышает эффективность обучения.

Программы, необходимые для прохождения курса

Обязательные:

- Клиент-серверный вариант 1С:Предприятие 8.3
- MS SQL Server, желательно версия 2012 или выше
- SQL Profiler (входит в состав MS SQL Server, но не входит в состав SQL Server Express)

Желательные, но не обязательные:

- Конфигурация «Центр управления производительностью (ЦУП)», желательно версия 2.0.15 или выше
- Конфигурация «Тест-центр», желательно версия 2.0.15 или выше

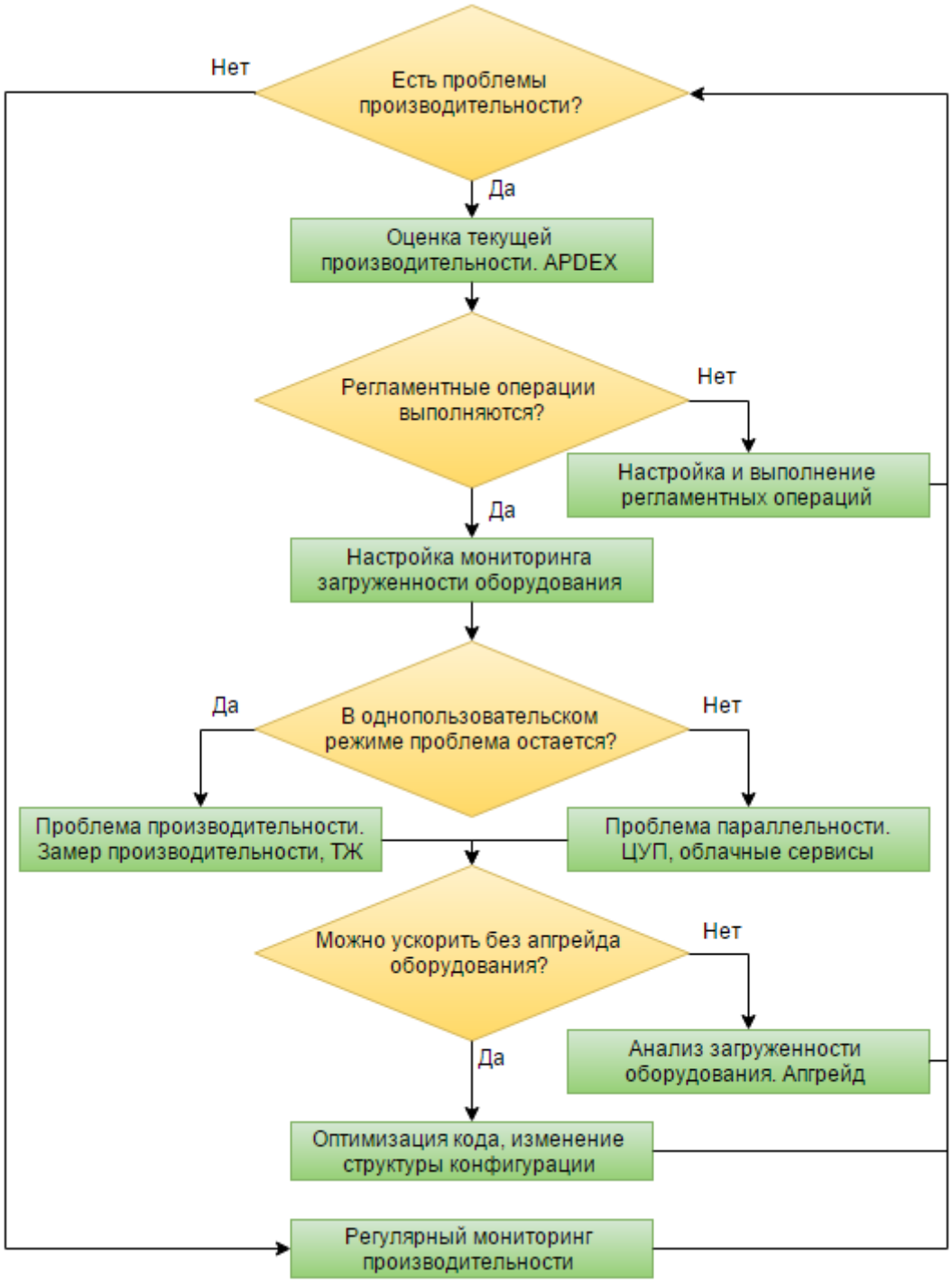
Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Общая схема расследования проблем производительности

Процесс повышения производительности и стабильности работы системы можно разделить на несколько этапов. Упрощенно все этапы представлены в виде схемы.



Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

1. Необходимо определить текущую производительность с помощью методики APDEX. Таким образом можно выделить отдельные операции, которые требуют ускорения, измерить производительность этих операций и понять текущую ситуацию.
2. Необходимо выяснить, выполняются ли регламентные операции на сервере СУБД. Если нет, то сначала нужно их настроить и выполнить, иначе нет смыла приступать к дальнейшим действиям.
3. Нужно настроить мониторинг оборудования и собрать данные о загруженности серверов.
4. Далее необходимо понять, какая проблема имеет место: проблема производительности или параллельности работы. И уже в зависимости от этого строить дальнейший план оптимизации и использовать соответствующие инструменты.
5. Нужно выяснить, есть ли возможность ускорить систему программно, с помощью оптимизации кода и настроек. Если код уже оптимизирован и дальнейшая программная оптимизация не представляется возможной, тогда стоит рассмотреть возможность апгрейда оборудования.

Данная схема является очень упрощенной, но порядок действий именно такой. При этом алгоритм действий на схеме зациклен, здесь нет завершения, т.к. процесс мониторинга производительности должен выполняться постоянно, пока используется информационная система.

В процессе изложения курса будет подробно разобран каждый из этапов оптимизации.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

APDEX

Занятие 1

Понятие индекса производительности

Для того чтобы оптимизировать систему, необходимо знать, какие именно операции нуждаются в ускорении, насколько их нужно ускорить, как оценить это ускорение, что оптимизировать в первую очередь, как оценить удовлетворенность пользователей скоростью работы системы и т.д. Чтобы ответить на все вышеперечисленные вопросы, и была придумана методика APDEX.

APDEX (Application Performance Index) – индекс производительности приложения. Открытый международный стандарт, разработанный с целью формирования объективной оценки показателей производительности корпоративных информационных систем.

Проще говоря, APDEX – это индекс, отражающий удовлетворенность пользователей скоростью работы информационной системы. Важно понимать, что речь идет только о скорости, а не о функциональности или удобстве работы. Суть методики состоит в том, что определяются ключевые операции (являются индикаторами скорости работы системы), для каждой из них устанавливается целевое время (время выполнения операции которое полностью устраивает пользователя, обозначается буквой **T**), фиксируется время выполнения каждой из ключевых операций и рассчитывается индекс производительности – так можно понять, насколько довольны пользователи скоростью работы системы.

При появлении проблем производительности, первое что нужно сделать, это рассчитать APDEX, т.е. сначала необходимо зафиксировать текущее состояние системы и только потом приступать непосредственно к оптимизации.

Методика APDEX

Методика APDEX состоит из следующих шагов:

- Определение списка ключевых операций
- Определение целевого времени ключевых операций
- Определение приоритета ключевых операций
- Сбор информации о реальном времени выполнения ключевых операций
- Вычисление APDEX.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Определение списка ключевых операций

Ключевая операция (КО) – это такая операция в информационной системе, которая критически важна для нормального функционирования организации.

Например, в торговой организации проведение документа по продаже товара будет являться ключевой операцией, ведь если документ будет проводиться медленно, то это скажется на бизнесе, покупатели не любят долго ждать. Типичными ключевыми операциями являются: проведение документов, формирование отчетов, выполнение различных обработок и открытие форм.

Под заказчиком не обязательно понимается организация, которая заказала проект по оптимизации, в случае оптимизации собственной конфигурации заказчиком может являться конечный пользователь или начальник отдела, направления, т.е. тот, кто оценивает эффект от оптимизации.

Список ключевых операций всегда составляется вместе с заказчиком.

Ключевой операцией не всегда является та операция, которая выполняется множество раз в течение дня, это может быть и редкая, но критически важная операция. Например, расчет себестоимости, восстановление последовательности и т.д.

Важно понимать, что если выполняется групповая обработка чего-либо, то вся групповая обработка будет являться одной КО, при этом отдельные действия, выполняемые в этой обработке, не являются КО. Например, надо перепровести 5 000 документов за 1 час, здесь ключевой операцией будет являться не проведение каждого отдельного документа, а выполнение всей обработки. Соответственно, и целевое время для данной операции будет 1 час, а не 0,72 секунд на один документ. Это имеет большое значение для расчета APDEX, т.к. выполнение обработки мы можем ускорить, например, распараллеливанием, а ускорить проведение одного документа до 0,72 сек. будет гораздо сложнее для эксперта и дороже для заказчика.

Возможна ситуация, когда документ может содержать как 10 строк, так и 1 000. Естественно, требовать одинакового времени проведения в этом случае будет неправильно. В такой ситуации необходимо разбить одну КО на несколько и к каждой из них предъявить разные требования по времени выполнения. Например, документ, в котором содержится до 50 строк – это одна ключевая операция с одним целевым временем, от 50 до 150 строк – другая КО с другим целевым временем и так далее. Все эти пороги необходимо обязательно согласовать с заказчиком или пользователями.

Также возможна ситуация, когда ключевой является какая-либо длительная и редкая, но важная для бизнеса операция, например, расчет себестоимости. Если КО выполняется 1 раз в месяц, нет возможности собрать 100 замеров и рассчитывать APDEX по формуле, ведь это займет очень много времени. В этом случае нужно просто ускорить операцию, чтобы она укладывалась в целевое время.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Определение целевого времени

Целевое время – это время выполнения операции, которое полностью устроит заказчика.

В формуле расчета APDEX целевое время обозначается буквой T.

Желательно проставить целевое время самостоятельно и отдать на утверждение заказчику, т.к. часто заказчик затрудняется определить целевое время или делает это очень долго.

Рекомендуется использовать следующие значения для наиболее типичных КО:

Ключевая операция	Целевое время (сек.)
Проведение документа	3
Формирование отчета	5
Открытие формы	1

Как правило, выполнение операций за указанное в таблице время не вызывает дискомфорта у пользователей.

Для специфичных операций конкретной конфигурации целевое время необходимо обязательно получить от заказчика.

Возможно, заказчик захочет установить слишком жесткие требования, например, проведение документа за 0,3 секунды, в этом случае, скорее всего, возникло недопонимание. Задача 1С:Эксперта – выяснить истинные цели заказчика и причины таких требований.

Возможно, что в действительности он хочет групповой обработкой перепровести 12 000 документов за 1 час. В этом случае необходимо объяснить заказчику, что ключевой операцией здесь является групповое перепроведение, а не проведение каждого отдельного документа.

В редких случаях заказчик может необоснованно настаивать на слишком малом времени выполнения ключевой операции.

Например, заказчик требует, чтобы документ, независимо от количества строк, проводился за 0,5 сек., тогда необходимо разъяснить, что добиться такого эффекта без серьезных финансовых затрат будет крайне сложно. Когда речь заходит про деньги, заказчик, как правило, выдвигает более реалистичные требования.

Следует понимать, что целевое время не является чем-то неизменным, в процессе работ по оптимизации оно может изменятся в любую сторону, но только по взаимной договоренности.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Например, целевое время операции 2 секунды, APDEX операции равен 0.84, проведен уже большой объем работ, и программная оптимизация не представляется возможной, можно ускорить операцию только апгрейдом оборудования. При этом если увеличить целевое время до 3 секунд, то APDEX будет равен 0,85, что и требуется. В данном случае разумнее увеличить целевое время на 1 секунду, чем покупать дорогостоящее оборудование. Здесь, как и всегда, все зависит от конкретной ситуации.

Определение приоритета

Приоритет ключевой операции – это степень важности ключевой операции с точки зрения деятельности организации. Приоритет должен быть уникальным, т.е. не может быть двух операций с приоритетом 1.

В последовательности шагов методики APDEX, определить приоритет следует сразу после определения ключевых операций, но на практике приоритет чаще всего определяется после выяснения целевого времени.

Обычно приоритет - это порядок, в котором будет производится оптимизация системы.

Часто заказчик затрудняется проставить приоритеты и утверждает, что для него одинаково важны все операции и нужно оптимизировать все сразу. Тем не менее, необходимо понять, с чего начинать оптимизацию. Надо разъяснить заказчику, что в любом случае нужно с чего - то начать и установить порядок. Как правило оптимизацию проводит только один человек, и он не может параллельно работать сразу над несколькими операциями.

На практике в системах с высокой нагрузкой нет возможности ежедневно обновлять конфигурацию. Как правило, в одно обновление попадает оптимизация сразу нескольких операций.

При этом не всегда операции оптимизируются по приоритету, все зависит от ситуации.

Приоритет	Ключевая операция	APDEX
1	Проведение Реализация товаров и услуг	0,83
2	Проведение Поступление товаров и услуг	0,65

Например, в случае, описанном в таблице, разумнее будет приступить к оптимизации второй операции, т.к. у первой операции APDEX близок к норме, при этом вторая операция выполняется очень плохо.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Сбор информации о реальном времени выполнения

После определения целей и приоритетов необходимо собрать информацию о фактическом времени выполнения КО. Для этого можно использовать как инструменты фирмы 1С, например, подсистему «Оценка производительности» из Библиотеки стандартных подсистем (БСП), так и альтернативные инструменты, например, облачный сервис контроля производительности по методике APDEX.

При желании вы можете написать свой собственный инструмент для измерения времени выполнения операций.

Вычисление APDEX

На основе собранных данных вычисляется APDEX. Для расчета необходимо выбрать период, за который будет рассчитано значение. В указанных выше инструментах есть возможность просматривать APDEX за произвольный период времени. Чаще всего APDEX рассматривается минимум в разрезе дня.

Рассчитанное значение APDEX нигде не хранится, т.к. при наличии исходных данных его всегда можно рассчитать заново, при необходимости меняя целевое время.

Индекс производительности рассчитывается по следующей формуле:

APDEX = (NS + NT / 2) / N

Где:

NS – число выполнений со временем от 0 до T
NT – число выполнений со временем от T до 4T
N – общее число выполнений данной операции

Чем больше число выполнений операции, тем точнее APDEX, желательно чтобы замеров было более 100, иначе рассчитанное значение может быть непоказательным.

APDEX может принимать значения от 0 до 1. Если APDEX = 0, значит пользователи не могут работать, т.к. система не отвечает на запросы. Если APDEX = 1, значит пользователи полностью довольны скоростью работы системы.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Диапазон значений APDEX	Производительность (мнение пользователей)
0,00 – 0,49	неприемлемо
0,50 – 0,69	очень плохо
0,70 – 0,84	плохо
0,85 – 0,94	хорошо
0,95 – 1,00	отлично

Например, необходимо подсчитать APDEX ключевой операций «Формирование отчета Остатки по складам» за 1 неделю. Целевое время операции (T) равно 1 секунде.

За этот период операция выполнялась 181 раз, 122 раза операция выполнялась не более 1 секунды и 47 раз от 1 до 4 секунд.

Подставим значения в формулу:

APDEX = (122 + 47 / 2) / 181 = 0,80

Значение 0,80 соответствует оценке «плохо», т.е. пользователи недовольны скоростью формирования этого отчета.

Как только APDEX стал равен значения 0,85 или выше, цель ускорения достигнута.

Зная APDEX каждой ключевой операции, можно получить APDEX всей информационной системы, для этого используется приведенная выше формула, но в качестве переменных выступают суммы значений параметров N, NS и NT. Для приведенной ниже таблицы общий APDEX рассчитывается так: APDEX = (333 + 47 / 2) / 392 = 0,90.

Ключевая операция	T	N	NS	NT	APDEX
Формирование отчета *Остатки по складам*	1	181	122	47	0,80
Проведение документа *Перемещение со склада*	3	211	211	0	1
Общая производительность системы		**392**	**333**	**47**	**0,90**

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Иногда требуется понять, как оптимизация одной ключевой операции повлияет на общий APDEX всей информационной системы. В этом случае можно использовать следующую формулу:

DeltaApdex = (NF + NT / 2) / Nall

Где:

NT - число выполнений операции с временем от T до 4T
NF - число выполнений операции с временем больше, чем 4T
Nall - число выполнений всех операций.

Недостатки методики APDEX

Как и в любой другой методике, у методики APDEX есть свои минусы. Методика плохо работает, если число замеров мало. За минимальное значение, которое обеспечивает корректную оценку, можно взять 100 замеров. Также наличие резких «всплесков» в замерах, слабо отображается в APDEX. Допустим, было 100 замеров некоторой операции. Целевое время равно 3 секундам. 85 операций было выполнено за время от 1 до 3 секунд, остальные операции – за время свыше 600 секунд. В результате получаем следующий APDEX:

APDEX = (NS + NT / 2) / N = (85+0 / 2) / 100 = 0,85

Получается, что значение APDEX соответствует оценке хорошо, но при этом 15 операций выполнились за совершенно неприемлемое время.

При использовании методики следует понимать, что главным все-таки является мнение пользователей, и если они не довольны скоростью работы, значит, либо было ошибочно выбрано время T, либо неправильно встроен код замера. Так же возможно надо поднимать минимальный порог APDEX, например, производить оптимизацию, пока APDEX не достигнет значения 0,90 или выше.

Также очень полезно строить различные отчеты на основании «сырых» замеров и смотреть, как распределяются времена в зависимости от дня недели, пользователя, рабочих часов и т.д., иногда таким образом можно определить некоторые неявные закономерности.

В любом случае необходимо измерять и собирать длительность выполнения ключевых операций. Имея данные о времени выполнения ключевых операций, можно рассчитать индекс производительности по любой формуле.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 2

Подсистема «Оценка производительности»

Видеоурок. Внедрение подсистемы в конфигурацию

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса.

Изучив этот урок, Вы узнаете:

- Как внедрить подсистему «Оценка производительности» в конфигурацию
- На что обращать внимание при объединении
- Как отобразить все элементы подсистемы замеров в интерфейсе.

Видеоурок. Настройка ключевых операций

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как создать свои ключевые операции в конфигураторе
- Как настроить приоритет и целевое время ключевых операций.

Видеоурок. Замер времени на клиенте

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как включить функцию замеров времени
- Как замерить время операции, выполняемой пользователем на клиенте
- Как можно создать ключевую операцию во время замера
- Надо ли в этом случае прописывать код окончания замера.

Видеоурок. Замер времени на сервере

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- Как измерить время операции, выполняемой на сервере
- Как измерить время одной операции, выполняемой внутри другой
- Как зафиксировать замер вручную.

Видеоурок. Возможные ошибки при встраивании замера

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Типичные ошибки при встраивании замеров
- Как правильно измерить время групповой операции.

Видеоурок. Обработка «Оценка производительности»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как рассчитать APDEX на основании выполненных замеров
- Как посмотреть динамику значений APDEX
- Как быстро поменять приоритет ключевой операции.

Видеоурок. Использование подсистемы в типовых конфигурациях

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Какие операции считают ключевыми разработчики типовых конфигураций
- Каким образом можно измерить время выполнения в типовых конфигурациях.

Видеоурок. Недостатки подсистемы. Ошибки

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Возможные ошибки при использовании подсистемы «Оценка производительности» из БСП

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Недостатки подсистемы. Исправление ошибок

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Способы исправления ошибок подсистемы «Оценка производительности» из БСП

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 3

Сервис APDEX

Описание и схема работы сервисов

Как альтернатива платным инструментам от 1С (в частности корпоративный инструментальный пакет) в курсе рассматриваются и бесплатные инструменты, в частности, облачные сервисы контроля производительности gilev.ru. Выбор в пользу сервисов сделан именно ввиду бесплатности использования их основного функционала (там есть и платные функции, но они в курсе не рассматриваются). Таким образом, даже не имея корпоративного инструментального пакета, можно расследовать и решать проблемы производительности.

Для использования сервисов необходимо скачать его клиентскую часть, настроить сбор данных, после чего данные отправляются в облако и анализируются. Для просмотра и анализа собранных данных можно подключиться к сервису через браузер или использовать тонкий клиент 1С с подключением через веб-сервер. При использовании браузера, клиентские лицензии будут использоваться с сервера, а их число ограничено, поэтому возможна ситуация, когда соединение будет принудительно завершено. Предпочтительнее использовать тонкий клиент и свою собственную лицензию, тогда никаких проблем с подключением не будет.

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Сервис использует только техническую информацию: контексты вызова кода, тексты SQL запросов, планы запросов, трассировки SQL и т.д. Данные по хояйственным операциям, пользователям и т.д. сервис не собирает.

Код клиентской части сервиса полностью открыт.

Видеоурок. Регистрация в облаке gilev.ru

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как зарегистрироваться для использования сервисов
- На что обращать внимание при регистрации
- Какие требования предъявляются к имени учетной записи.

Видеоурок. Установка сервиса APDEX

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как внедрить сервис APDEX в конфигурацию
- Какие объекты не нужно переносить при объединении
- Особенности интеграции сервиса при использовании платформы 8.3.

Видеоурок. Настройка сервиса APDEX

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как правильно настроить сервис
- Как проверить соединение с ЦОД gilev.ru
- Как настроить периодичность выгрузки данных в облако
- Как выгрузить данные в облако оперативно.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Автоматический замер времени проведения документов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как измерить время проведения автоматически
- Как устроен механизм автоматического замера проведений
- Что не фиксируется при автоматическом замере.

Видеоурок. Замер времени произвольной операции

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как измерить время выполнения любой операции

Видеоурок. Регистрация текста запроса

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как текст запроса измеряемой операции

Видеоурок. Замер времени открытия формы

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как измерить время открытия любой формы

Видеоурок. Просмотр данных APDEX

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- Какими способами можно подключиться к сервису APDEX
- Как настроить целевое время и приоритет ключевых операций

Видеоурок. Динамика APDEX и отчеты сервиса

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как посмотреть изменение значения APDEX в динамике
- Какие отчеты можно сформировать для анализа замеров.

Видеоурок. Отправка данных в сервис по почте

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как отправить данные в сервис, если у сервера 1С нет доступа в интернет
- В каком виде необходимо оформлять письмо с данными для сервиса.

Видеоурок. Недостатки сервиса APDEX

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как порядок подписок на события влияет на работу сервиса
- Какие настройки необходимо сделать для использования сервиса

Расчет целевого времени «от обратного»

Возможна ситуация, когда заказчик затрудняется определить целевое время для ключевой операции, в этом случае можно рассчитать целевое время «от обратного».

Сначала необходимо собрать оценки пользователей о текущей скорости выполнения операции по школьной пятибалльной шкале. Для этого можно выбрать ключевого пользователя и выяснить у него, какую оценку он бы поставил этой операции, как она выполняется сейчас, до оптимизации.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Следует уточнить, что если операцией является проведение, то и оценку нужно ставить именно проведению, а не заполнению или открытию документа. Если пользователь недоволен заполнением или открытием, то их нужно вынести в отдельные ключевые операции. Не нужно просить пользователя воспроизвести операцию и поставить оценку, он должен опираться на свой предыдущий опыт работы с этой операцией.

Далее, по оценке нужно определить текущий APDEX, используя таблицу соответствия, описанную выше, где «отлично» соответствует оценке 5, а «неприемлемо» оценке 1.

После этого включаем замеры операции, желательно произвести не менее 100 замеров.

Теперь необходимо подобрать такое значение T, при котором рассчитанный APDEX будет равен APDEX из оценки пользователя.

В итоге необходимо убедиться, что полученное целевое время устраивает заказчика.

Например, выполнение обработки является ключевой операцией, необходимо определить целевое время от обратного. Пользователи ставят оценку 3, что соответствует APDEX = 0,77.

Далее необходимо замерить время выполнения данной обработки. Желательно выполнить 100 замеров. Далее меняем значение целевого времени до тех пор, пока рассчитанный APDEX не станет равен 0,77.

Видеоурок. Автоматический расчет целевого времени «от обратного»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как с помощью обработки «Оценка производительности», зная APDEX, рассчитать целевое время
- Может ли сервис APDEX рассчитать целевое время «от обратного».

Сравнение подсистемы из БСП и сервиса APDEX

Каждый из описанных инструментов имеет свои сильные и слабые стороны. Для одного, возможно, эти недостатки будут несущественными, для другого они будут критичными.

Для удобства сравнения основные плюсы и минусы инструментов сведены в таблицу.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

	Подсистема «Оценка производительности»	Сервис APDEX
Плюсы	• Встроена в типовые конфигурации • Не требует доступа в Интернет • Экспорт данных в ЦКК (конфигурация центр контроля качества)	• Легкая интеграция • Автоматический замер проведения всех документов • Доступ к данным APDEX через Интернет
Минусы	• Совпадение метаданных при объединении • Требуется исправление кода для устранения ошибок • Внесение кода замера для каждой ключевой операции	• Требуется отправлять данные в облако • Ограниченное число подключений при использовании браузера • Особенности работы с подписками на события

APDEX. Итоги

У многих после прочтения данной главы может возникнуть вопрос: «Зачем нужен APDEX? При необходимости пользователь сам скажет, какая операция у него работает медленно».

Методика APDEX предоставляет следующие возможности:

- **Оценить состояние системы в любой момент времени и сразу узнать о проблемах**
  Обычно пользователи не сообщают о проблемах сразу, они терпят до тех пор, пока ситуация не станет критичной, а замедление – слишком сильным. Не нужно ждать, пока ответственный пользователь позвонит в IT-отдел и расскажет о проблеме. Обычно этого не происходит, и о проблемах производительности программисты узнают от руководства. Необходимо иметь собственные объективные данные о скорости работы системы, которые можно получить, не обращаясь к пользователям.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

С помощью сервиса APDEX можно посмотреть историю значений APDEX для операции, увидеть, как работала операция по состоянию на любой момент времени, например, можно сравнить, как быстро операция выполнялась неделю назад и как стала выполняться сейчас. Это позволяет обнаружить постепенное замедление выполнения операции и вовремя принять меры (пока все не стало слишком плохо). Эта возможность также позволяет обнаружить причину замедления выполнения операции, когда замедление возникло после определенного обновления.

- **Оценить объем работ для оптимизации**
  При заключении договора по оптимизации методика APDEX помогает примерно оценить объемы работ.

- **Определить приоритеты**
  С помощью методики APDEX можно понять, в каком порядке приступать к оптимизации операций.

- **Формализовать критерии выполнения работ по оптимизации**
  При заключении договора по оптимизации APDEX служит критерием успешного выполнения работ, как только APDEX достигает значения 0,85, операция считается достаточно оптимизированной.

Если в информационной системе еще не используется методика APDEX, то автоматические замеры необходимо внедрить в обязательном порядке. Для этого можно использовать один из рассмотренных инструментов или написать собственный. Главное, замеры должны производиться на регулярной основе, даже если на текущий момент проблем производительности не наблюдается.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Настройка сервера СУБД

Занятие 4

Перед тем как приступать непосредственно к оптимизации кода, необходимо убедиться, что сервер СУБД настроен оптимально с точки зрения производительности. Настройка будет рассмотрена на примере MS SQL Server.

В идеале нужно стремиться к тому, чтобы сервер СУБД работал на отдельном компьютере, на котором другие серверные роли не установлены. Такая архитектура сразу убирает большой пласт всевозможных проблем. Крайне не рекомендуется, чтобы компьютер с сервером СУБД выполнял роль терминального, почтового, веб-сервера или прочие серверные роли.

Можно устанавливать сервер приложений 1С и сервер СУБД на один компьютер, но только при относительно небольшой нагрузке на систему, в среднем около 100 одновременно работающих пользователей. Этот показатель зависит от мощности оборудования и от конкретной конфигурации. Если оборудование достаточно мощное, то ничто не мешает совмещать роли СУБД и 1С на одном сервере, даже если в базе работает 300 и более пользователей. Как проверить что оборудование справляется с нагрузкой, будет рассмотрено в отдельном разделе.

При выборе версии СУБД лучше выбирать старшую версию, например, лучше установить SQL Server 2012, чем 2008, а 2014 – лучше, чем 2012. В каждой новой версии исправляются ошибки и оптимизируются механизмы, иногда разница в скорости выполнения одних и тех же операций может быть очень существенна. Естественно, необходимо следить за тем, чтобы платформа поддерживала указанную версию СУБД.

Также следует учитывать ограничения различных версий ОС. Например, нет смысла ставить 64 ГБ памяти на сервер, где используется Windows Server 2008 R2 Standard x64. Данная ОС не может использовать более 32 ГБ оперативной памяти.

Очень важной является своевременная установка обновлений и Service Pack как для СУБД, так и для операционной системы сервера. Это может сильно повлиять на производительность и стабильность работы системы в лучшую сторону.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Общие сведения о файле данных и журнале транзакций

Цель данного раздела – дать упрощенное представление о назначении и принципах работы файла данных и журнала транзакций в СУБД MS SQL Server. Здесь не рассматриваются детали реализации и тонкости работы механизма фиксации изменений MS SQL Server, т.к. это выходит за рамки данного курса.

Любая база данных MS SQL Server состоит минимум из двух файлов: файла данных и файла журнала транзакций.

Файл данных содержит в себе непосредственно данные, здесь хранится содержимое таблиц базы данных. Файл данных имеет расширение .mdf или .ndf, при этом у одной базы данных файлов данных может быть несколько. Данные в нем находятся в страницах размером 8 КБ. Страница – это минимальная порция данных, которую может прочитать сервер СУБД.

Журнал транзакций содержит в себе информацию, необходимую для отката или фиксации транзакций: что было до транзакции, и что стало после. Например, у товара был изменен артикул с «Т001» на «Т011». Тогда в файле данных значение поля артикул в таблице товаров просто изменится с «Т001» на «Т011». В файле данных не сохранится информация о том, какой артикул был до изменения.

При этом в журнал транзакций будет записана примерно следующая информация: Транзакция с ID=46288032, в такой-то момент времени изменила в таблице товаров реквизит «Артикул» со значения «Т001» на «Т011».

Если говорить упрощенно, журнал хранит всю историю изменения данных, но не сами данные. Благодаря журналу можно восстановить базу в то состояние, в котором она находилась на определенный момент времени. Подробнее об этом будет рассказано в разделе, посвященном резервному копированию.

Журнал транзакций имеет расширение .ldf и в базе данных файлов журнала также может быть несколько.

Рассмотрим, каким образом происходит изменение данных в СУБД:

1. Пользователь меняет в справочнике какой-либо реквизит и нажимает кнопку *Сохранить*
2. Открывается транзакция
3. Идет поиск страницы данных, содержащей строку для изменения
4. Найденная страница извлекается из файла данных на диске и помещается в кэш данных в оперативной памяти. **Кэш данных** – область в оперативной памяти, где хранятся страницы данных, прочитанные с диска. Не путать с **процедурным кэшем**, в котором хранятся планы запросов

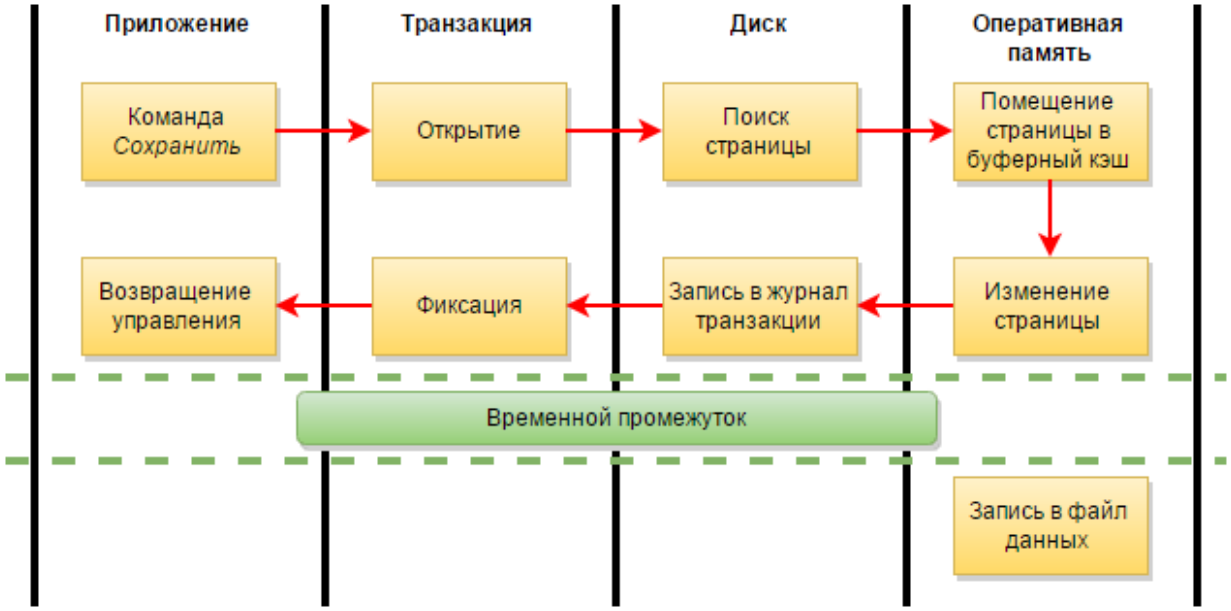
Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

5. В странице меняется значение поля, при этом страница остается в памяти
6. В журнал транзакций на диске вносится запись: что, как, когда и какой транзакцией было изменено. Пока информация не записана в журнал транзакций, инструкция SQL не считается исполненной.
7. Транзакция фиксируется
8. Управление возвращается пользователю
9. Измененная страница записывается в файл данных на диске

Схематично этот процесс представлен на рисунке.



Из схемы видно, что сначала идет запись в журнал транзакций, и только через некоторое время страница попадает в файл данных, но так происходит не всегда. Сервер СУБД не гарантирует, что при фиксации транзакции измененные данные сразу попадают в файл данных, но он гарантирует, что информация о транзакции записана в файл журнала транзакций. Когда измененные данные попадут в файл данных, зависит от настроек СУБД и от того, есть ли у сервера ресурсы, чтобы выполнить запись в файл данных во время транзакции, не вызывая при этом задержек. Если такая возможность есть, данные попадают в файл данных прямо во время транзакции, в ином случае – через некоторый промежуток времени.

Если во время транзакции произошел сбой, например, отключение электропитания, а измененная страница в этот момент находилась в памяти и не была сохранена в файл данных, и при этом имеется запись в журнале транзакций, то при следующем запуске сервер СУБД сверит информацию из файла данных с журналом транзакций и при необходимости внесет изменения в файл данных либо для фиксации транзакции, либо для ее отката.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Есть некоторые отличия в работе дисков при записи информации в файл данных и журнал транзакций. Информация в файле данных структурирована, и при изменении данных необходимо найти определенное место, куда будут внесены эти изменения, т.е. используется случайный доступ к диску. Когда идет запись в журнал транзакций, информация записывается последовательно, т.к. здесь нет необходимости соблюдать какую-либо структуру. Необходимо помнить, что обычные диски работают с последовательной записью значительно быстрее, чем со случайным доступом, исключением являются SSD диски.

Представим ситуацию: один из пользователей формирует «тяжелый» отчет, а остальные в этот момент проводят документы. Если и файл данных, и журнал транзакций находятся на одном диске, то жесткий диск вынужден постоянно переключаться между массовыми операциями чтения данных при построении отчета и операциями записи в журнал транзакций при проведении документов.

В результате таких действий, работа дисковой подсистемы будет малоэффективной. Размещение журнала транзакций и файла данных на разных физических дисках решает подобные проблемы.

Лучше всего и для журнала транзакций, и для файла данных подходят дисковые массивы RAID10, обеспечивающие высокую надежность и быстродействие.

Также следует упомянуть о служебной базе данных MS SQL Server под названием TempDB. Данная база предназначена для хранения различных временных данных, результатов промежуточных расчетов, временных таблиц и т.д. Служебная база, так же, как и любая другая, имеет файл данных и файл журнала транзакций, для TempDB это файлы Tempdb.mdf и Tempdb.ldf. На сервере СУБД база TempDB всегда одна и используется всеми остальными базами, установленными на этом сервере. Таким образом, нагрузка на нее может быть очень высокой, в связи с чем рекомендуется выносить базу TempDB на отдельный физический диск. Если позволяют ресурсы сервера, можно разместить TempDB на RAM диске.

Есть рекомендация делать несколько файлов данных и даже несколько файлов журнала транзакций, разносить их по разным физическим дискам и таким образом увеличить скорость чтения и записи. Такая операция имеет смысл только в очень нагруженных базах, где работают тысячи пользователей одновременно. Если сделать это на обычных базах, эффект, скорее всего, вообще не будет заметен. Такую тонкую настройку стоит делать там, где борьба идет за миллисекунды.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Настройка использования памяти MS SQL Server

Если на компьютере расположен только сервер СУБД, тогда дополнительно ничего настраивать не нужно, MS SQL Server сам возьмет у операционной системы максимально возможное количество памяти, при этом ОС не отдаст памяти больше, чем ей нужно для нормальной работы.

Ситуация меняется, если на сервере помимо СУБД запущены другие ресурсоемкие приложения, например, сервер 1С. В этом случае необходимо ограничить максимальный объем используемой MS SQL Server памяти, иначе СУБД может занять столько памяти, что не позволит другим приложениям полноценно работать.

Рассмотрим распространенную ситуацию, когда сервер 1С и сервер СУБД расположены на одном компьютере. В этом случае нужно учесть потребности в памяти для ОС и для сервера приложений, а остальное отдать серверу СУБД.

В итоге получаем следующую формулу:

Память для MS SQL Server = Память всего – Память для ОС – Память для сервера 1С

Например, на сервере установлено 64 ГБ оперативной памяти, необходимо понять, сколько памяти выделить серверу СУБД, чтобы хватило серверу 1С.

Для нормальной работы ОС в большинстве случаев более чем достаточно 4 ГБ, обычно 2-3 ГБ.

Чтобы определить, сколько памяти требуется серверу 1С, необходимо посмотреть, сколько памяти занимают процессы кластера серверов в разгар рабочего дня. Этими процессами являются *ragent*, *rmngr* и *rphost,* подробно данные процессы рассматриваются в разделе, который посвящен кластеру серверов. Снимать данные нужно именно в период пиковой рабочей активности, когда в базе работает максимальное количество пользователей. Получив эти данные, необходимо прибавить к ним 1 ГБ – на случай запуска в 1С «тяжелых» операций.

Например, были получены данные, что *ragent* занимает 0,1 ГБ, *rmngr* – 0,4 ГБ, первый процесс *rphost* – 4 ГБ, второй процесс *rphost* – 4,5 ГБ. Нехитрыми вычислениями получаем, что сервер 1С занимает 9 ГБ + 1 ГБ (запас по памяти) = 10 ГБ.

Подставим значения в формулу:

Память для MS SQL Server = 64 – 4 – 10 = 50 ГБ

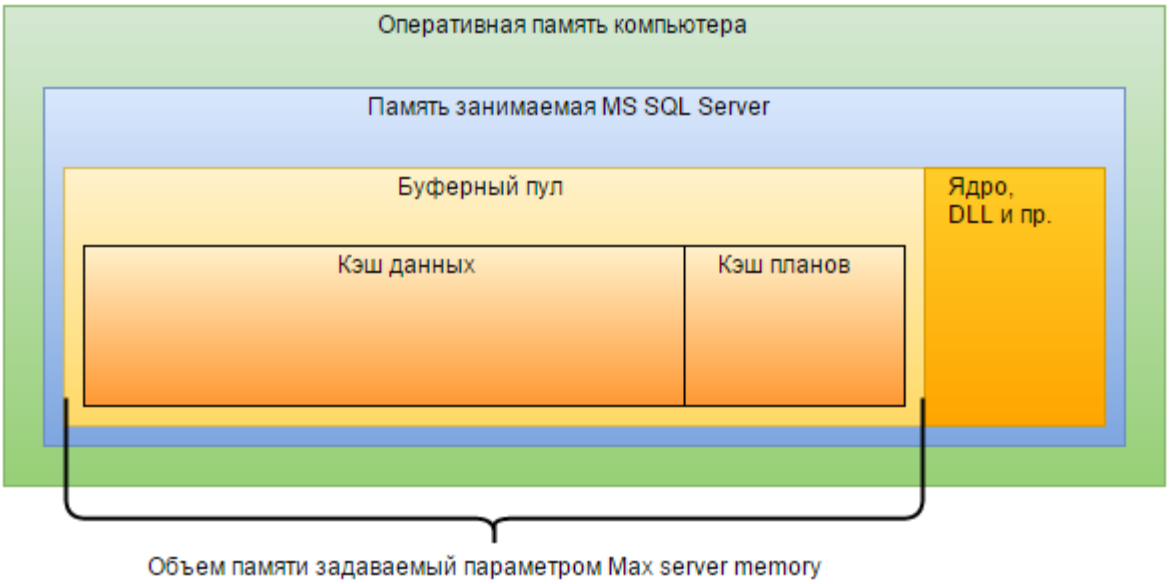
Из расчетов следует, что для рассмотренного примера серверу можно выделить максимум 50 ГБ оперативной памяти.

Максимальный объем памяти, выделяемый серверу MS SQL Server, устанавливается параметром **Max server memory** (Максимальный размер памяти сервера).

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

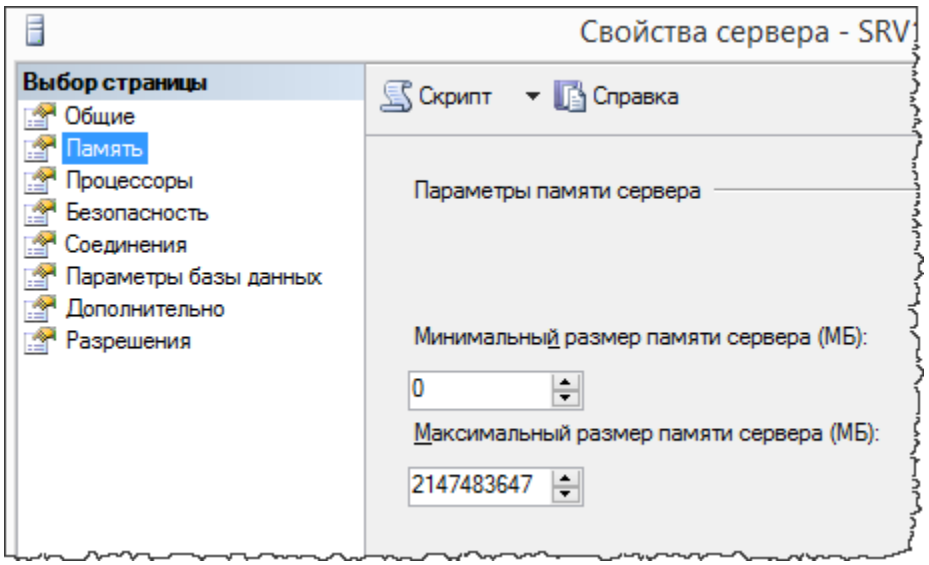
Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Следует учитывать, что мы ограничиваем только размер буферного пула, в котором хранятся кэш данных, кэш планов и прочие кэши. Память, которая выделяется под ядро и прочие компоненты MS SQL Server, в буферный пул не входит.



В общем случае объем памяти, которую занимает MS SQL Server, может превышать значение, заданное параметром *Max server memory*, и это нормальная ситуация.

Чтобы задать максимальный размер буферного пула, необходимо в Management Studio через контекстное меню открыть свойства сервера и на странице *Память* указать параметр *Максимальный размер памяти сервера*.



Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

При желании можно указать и *Минимальный размер памяти сервера* – это память которая гарантировано будет выделена под сервер СУБД, но обычно этот параметр не используется.

Для того чтобы узнать полный объем памяти, занимаемой MS SQL Server с учетом всех компонентов, необходимо выполнить запрос:

```
SELECT cntr_value/1024 as Mb
FROM
      master..sysperfinfo
WHERE
      counter_name = 'Total Server Memory (KB)'
```

Эту же информацию можно получить с помощью счетчика системного монитора *SQLServer:Memory Manager\Total Server Memory (KB)*, в русской локализации счетчик называется *SQLServer:Memory Manager\Общая память сервера (KB)*.

Работа со счетчиками системного монитора подробно разобрана в разделе, посвященном мониторингу оборудования.

Настройка

Видеоурок. Перенос журнала транзакций на другой диск

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как перенести файл журнала транзакций с помощью Management Studio.

Видеоурок. Перенос базы TempDB на другой диск

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение базы TempDB
- Как перенести файлы базы TempDB

Видеоурок. Параметр «Max degree of parallelism»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- На что влияет данный параметр
- Как указать количество процессоров для выполнения одного запроса

Видеоурок. Настройка авторасширения

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что такое авторасширение и зачем оно нужно
- Как настроить мгновенную инициализацию файлов.

Видеоурок. Настройка протокола Shared Memory

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Принцип и назначение работы протокола Shared Memory
- Как включить протокол Shared Memory.

Настройка сервера СУБД. Итоги

Для более оптимальной работы системы, необходимо настроить сервер СУБД определенным образом. Чем больше размер базы данных, тем большее значение имеет настройка сервера СУБД. Особенно важным является местоположение файла данных и файла журнала транзакций, по возможности необходимо разместить их на разных физических дисках. Базу TempDB также желательно разместить на отдельном физическом диске, можно даже на RAM диске.

В том случае, если сервер 1С и СУБД расположены на одном компьютере, необходимо включить протокол Shared Memory. Это несколько ускорит работу за счет обмена информацией через память, а не по сетевому протоколу.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Регламентные операции

Занятие 5

Основные регламентные операции

Для нормальной работы информационной системы ее необходимо обслуживать. В частности, под обслуживанием понимается регулярное выполнение регламентных операций СУБД и 1С. Понятие регламентной операции довольно широко, ниже дано определение регламентной операции в контексте повышения производительности системы.

Регламентная операция – это действие, направленное на поддержание производительности системы на высоком уровне, которое требуется выполнять с определенной периодичностью.

Говоря простым языком, это то, что необходимо регулярно выполнять, чтобы все работало быстро. Если перестать выполнять регламентные операции, то производительность начнет постепенно падать.

Регламентные операции необходимо выполнять как на сервере СУБД, так и на сервере 1С.

Основные регламентные операции сервера СУБД:

- Обновление статистики
- Дефрагментация
- Реиндексация.

Основные регламентные операции 1С:

- Сдвиг границы рассчитанных итогов
- Пересчет итогов
- Тестирование и исправление.

Рассмотрим каждую из этих операций подробно.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Регламентные операции СУБД

Понятие статистики

Статистика – это объект, содержащий статистические сведения о распределении значений в одном или нескольких столбцах таблицы.

Обновление статистики является одной из важнейших регламентных операций, она крайне необходима для быстрой работы запросов.

СУБД использует статистику для оценки числа строк, которые вернет запрос, т.е. еще до выполнения запроса СУБД может предположить, сколько строк вернется, что позволяет выполнять запрос максимально быстро.

Часть СУБД, которая отвечает за то, насколько быстро будет выполнен запрос, называется **оптимизатором**.

Оптимизатор, опираясь на статистику, наличие индексов и множество других данных, выбирает наиболее быстрый способ выполнить запрос, этот способ называется **планом запроса**.

План запроса определяет, как именно будет выполнен запрос, будут использоваться индексы или нет, в каком порядке соединять таблицы и прочее.

Например, с помощью статистики оптимизатор может сделать правильный выбор в пользу поиска по индексу вместо просмотра всей таблицы (что потребляет больше ресурсов) и благодаря этому повысить скорость работы запроса.

Очень подробно про работу оптимизатора и план запроса будет рассказано в соответствующей главе.

В основном, статистика создается по тем столбцам таблицы, по которым построен индекс, но это не всегда так. При необходимости, статистика может быть создана по любому столбцу непосредственно в момент выполнения запроса. В случае автоматического создания статистики по произвольной колонке имя объекта статистики будет начинаться с префикса «_WA». Если статистика создается по индексу, то имя объекта статистики совпадает с именем индекса.

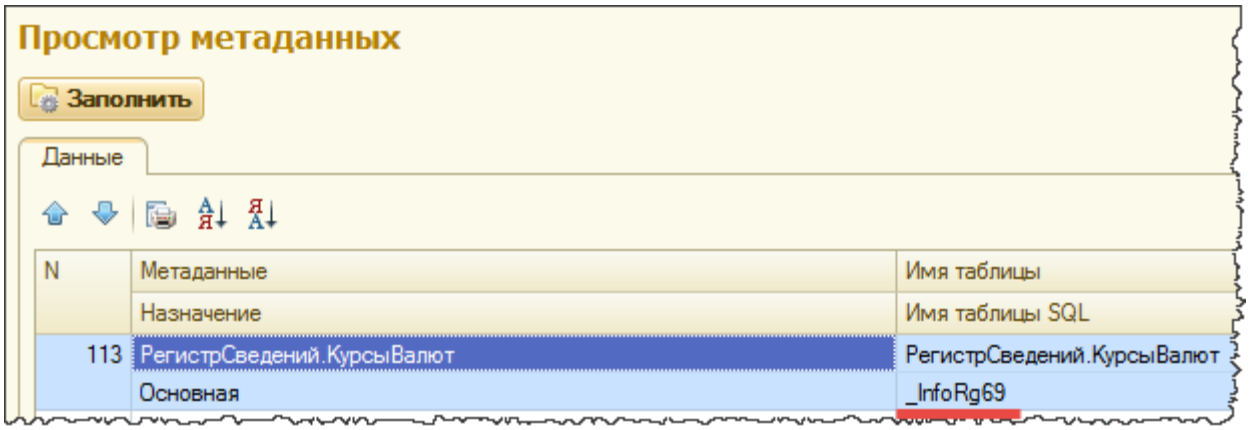
Рассмотрим статистику для регистра сведений «Курсы валют».

В первую очередь необходимо узнать имя таблицы СУБД, в которой хранятся данные этого регистра сведений. Это можно сделать с помощью обработки *Структура хранения метаданных.*

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

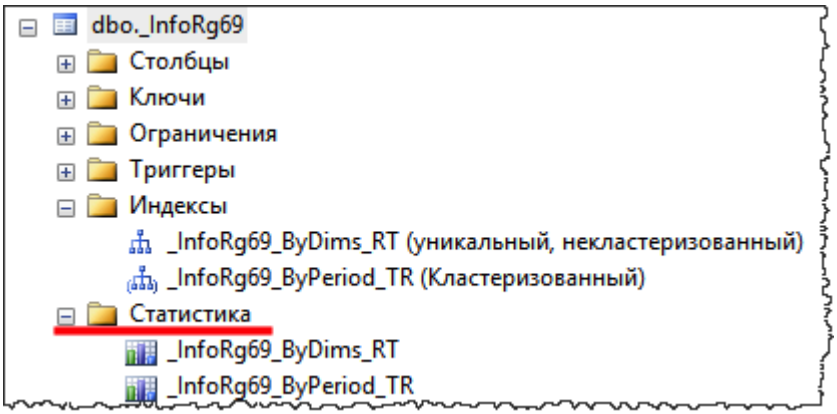
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



Видно, что Данные регистра хранятся в таблице «_InfoRg69».

С помощью Management Studio можно посмотреть, какая статистика создана для данной таблицы.



На рисунке видно, что было создано 2 объекта статистики, – по одному для каждого индекса. Рассмотрим статистику для индекса по периоду. Необходимо двойным кликом открыть элемент статистики и посмотреть ее состав на странице *Подробности*.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Имя таблицы: dbo._InfoRg69

Имя статистики: _InfoRg69_ByPeriod_TR

```
Статистика по INDEX "_InfoRg69_ByPeriod_TR".
-----------------------------------------------------------------------------------

Имя                              Обновлен                       Строки
-----------------------------------------------------------------------------------
_InfoRg69_ByPeriod_TR            окт 11 2015 10:15PM            23

Общая плотность                  Средняя длина                  Столбцы
-----------------------------------------------------------------------------------
0.05882353                       8                              _Period
0.04347826                       24                             _Period, _Fld70RRef

Шаги гистограммы
RANGE_HI_KEY                     RANGE_ROWS                     EQ_ROWS
-----------------------------------------------------------------------------------
01.01.08 0:00:00                 0                              1
05.02.08 0:00:00                 0                              1
14.02.08 0:00:00                 0                              1
20.02.08 0:00:00                 0                              2
01.03.08 0:00:00                 0                              1
11.03.08 0:00:00                 0                              2
13.03.08 0:00:00                 0                              1
18.03.08 0:00:00                 0                              2
19.03.08 0:00:00                 0                              1
27.03.08 0:00:00                 0                              1
08.04.08 0:00:00                 1                              1
16.04.08 0:00:00                 0                              2
21.05.08 0:00:00                 1                              1
```

Статистика хранится в MS SQL Server в виде гистограммы содержащей информацию о плотности распределения данных. В данном уроке рассмотрим только основные элементы статистики.

RANGE_HI_KEY – это ключевые значения, между которыми вычисляется количество записей таблицы, другими словами – это шаг гистограммы. Шагов не может быть больше 200, даже если в таблице содержатся миллиарды записей. Следовательно, чем больше таблица и чем больше в ней различных значений по выбранном столбцу, тем хуже статистика отражает реальную ситуацию.

Если индекс, по которому строится статистика, состоит из нескольких полей, то гистограмма строится только для первого поля. В данном случае индекс состоит из полей «_Period» и «_Fld70RRef», но гистограмма есть только по полю «_Period».

RANGE_ROWS – это количество строк таблицы, значение которых находится в пределах шага гистограммы, исключая граничные значения.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Например, для 11 шага гистограммы RANGE_ROWS равен 1, это значит, что в промежутке между 27.03.08 0:00:00 и 08.04.08 0:00:00 есть только 1 значение. В этом легко убедиться, посмотрев содержимое регистра сведений «Курсы валют».

Период	Валюта
01.01.2008	USD
05.02.2008	USD
14.02.2008	EUR
20.02.2008	USD
20.02.2008	EUR
01.03.2008	EUR
11.03.2008	USD
11.03.2008	EUR
13.03.2008	EUR
18.03.2008	USD
18.03.2008	EUR
19.03.2008	USD
27.03.2008	USD
02.04.2008	EUR
08.04.2008	EUR
16.04.2008	USD
16.04.2008	EUR
05.05.2008	USD
21.05.2008	Рубли
22.08.2012	USD
22.08.2012	EUR
08.09.2012	USD
08.09.2012	EUR

EQ_ROWS – это количество строк, значение которых равно границе шага гистограммы. Например, для 4 шага гистограммы EQ_ROWS равно 2, это значит, что в регистре есть 2 записи, где период равен граничному значению 20.02.08 0:00:00. В этом можно убедиться, посмотрев содержимое регистра сведений на приведенном выше скриншоте.

Используя статистику и некоторые другие сведения, СУБД выбирает наиболее оптимальный план выполнения для данного запроса с учетом его конкретных параметров и условий. Тот же запрос, но с другими значениями параметров, может выполняться по-другому.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Автоматическое обновление статистики

Когда распределение данных в таблице сильно меняется, например, если произошло массовое удаление строк или загрузка большого объема новых данных, необходимо обновить статистику.

Статистика может быть обновлена как автоматически, самим MS SQL Server, так и с помощью настроенного администратором регламентного задания. Возможность автоматического обновления статистики настраивается для каждой базы данных отдельно. По умолчанию автоматическое обновление статистики включено, и это правильно.

Автоматическое обновление статистики означает, что оптимизатор может обновить статистику до компиляции плана во время выполнения запроса. Это произойдет только в том случае, если оптимизатор посчитает, что затраты на обновление статистики будут меньше, чем выигрыш в скорости работы запроса.

У автоматического обновления статистики есть 3 основных недостатка:

- Общее время выполнения запроса увеличится на время обновления статистики
- Автообновление зависит от количества строк в таблице
- СУБД может не обновить статистику тогда, когда ее действительно нужно обновить.

Разберем каждый из этих недостатков подробнее.

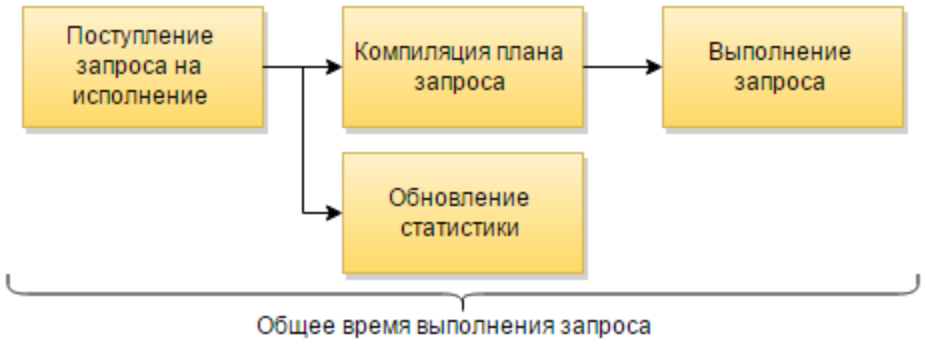
Общее время выполнения запроса увеличивается на время обновления статистики

Рассмотрим схему выполнения запроса с автоматическим обновлением статистики и без него.

Частично данная проблема решается асинхронным обновлением статистики, по умолчанию оно выключено. Если асинхронное обновление включено, то СУБД, обнаружив при выполнении запроса устаревшую статистику, продолжает выполнение, но параллельно запускается процесс обновления статистики, и уже следующий запрос к этой таблице будет выполнен с актуальной статистикой.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



Если используется ежедневное обновление статистики через регламентное задание, тогда автоматическое обновление почти никогда не будет срабатывать. Это значит, что включение возможности асинхронного обновления не имеет большого смысла.

Автообновление зависит от количества строк в таблице

Момент, когда необходимо выполнить обновление статистики, напрямую зависит от того, сколько строк в таблице и сколько строк было изменено. До MS SQL Server 2008 включительно автообновление срабатывало, только когда в таблице менялось минимум 20% строк. Для маленьких таблиц это оправдано, но для больших таблиц обновление было редким. Например, в таблице 1 млн. строк, тогда, чтобы сработало автообновление статистики, в таблице должно измениться 200 тыс. строк с момента последнего обновления статистики, и пока 200 тыс. строк не изменятся, запросы будут работать с неактуальной статистикой.

В MS SQL Server 2012 и выше механизм автообновления статистики был изменен. Теперь, в зависимости от количества строк, в таблице меняется порог для автоматического обновления статистики. Для небольших таблиц порог по-прежнему будет равен примерно 20%, но когда размер таблицы превысит 25000 строк, начнет действовать динамическое изменение порога срабатывания, и при увеличении количества строк пороговое значение становится все ниже и ниже. Например, в таблице со 100 тыс. строк порог снижен до 10%, а в таблице с 1 млн. строк – равен примерно 1%.

СУБД может не обновить статистику тогда, когда ее действительно нужно обновить

Бывают ситуации, когда необходимо обновить статистику, даже если изменилось относительно небольшое число строк. Например, есть таблица с 1 млн. строк, в которой присутствует один столбец, по которому построен индекс. В этом столбце значения во всех строках одинаковы, и поэтому индекс не используется. Если планируется вернуть всю таблицу, то самый быстрый способ - это просмотреть всю таблицу, т.е. выполнить сканирование. Далее в таблицу добавляют новую строку, в которой значение в индексируемой колонке отличается, и начинают строить запросы с отбором по новому значению.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Из-за того, что изменилась всего 1 строка, автообновления не происходит. По причине того, что новое значение не отображено в статистике, СУБД по-прежнему «думает», что выгоднее использовать сканирование всего миллиона строк вместо поиска по индексу одной строки, и запросы с отбором по новому значению работают медленно.

Использование регламентных операций для обновления статистики решает все вышеописанные проблемы и дает гарантию, что статистика будет всегда актуальной.

Обновление статистики не блокирует работу пользователей, но оказывает некоторую нагрузку на оборудование. Рекомендуется выполнять обновление статистики ежедневно в нерабочее время. В редких случаях может потребоваться обновлять статистику по некоторым таблицам несколько раз в день, например, когда в течение дня в таблице меняется большой объем данных.

Очистка процедурного кэша

Процедурный кэш – это место в оперативной памяти, где хранятся планы запросов.

Построение плана запроса – довольно затратная операция, поэтому, построив для запроса план, сервер сохраняет его в процедурном кэше, и если запрос с таким же текстом снова поступает на выполнение, то для него план уже не формируется, а берется из кэша.

Во многих источниках можно встретить информацию о том, что после обновления статистики обязательно необходимо выполнять очистку процедурного кэша. Якобы в кэше остался план, который сформирован при старой статистике, поэтому, когда статистика обновилась, нужно обязательно очистить кэш, чтобы скомпилировался новый план.

На текущий момент эта рекомендация уже устарела. Для MS SQL Server 2005 и выше, после обновления статистики, СУБД сама при следующем исполнении запросов (если статистика не была обновлена ранее принудительно) перекомпилирует те планы, которые зависели от старой статистики. Именно по этой причине нет необходимости очищать процедурный кэш после обновления статистики и делать это регламентной операцией.

Иногда бывает полезно знать, как выполняется запрос с «холодным» (не заполненным) процедурным кэшем. Например, это может пригодится при отладке запросов.

Очистить процедурный кэш можно командой DBCC FREEPROCCACHE. Следует учитывать, что эта команда очищает кэш для всех баз, расположенных на данном сервере. Это особенно важно, если тестовые базы расположены вместе с рабочими, а вы хотите почистить кэш только для тестовой базы. Очистка кэша на рабочем сервере в разгар рабочего дня может сильно замедлить работу системы на некоторое время. Замедление произойдет потому, что все запросы всех баз начнут перекомпилироваться, и служебные запросы в том числе. Перекомпиляция большого числа запросов даст кратковременную, но ощутимую нагрузку на сервер СУБД. По этой причине крайне не рекомендуется очищать процедурный кэш на рабочем сервере в рабочее время.

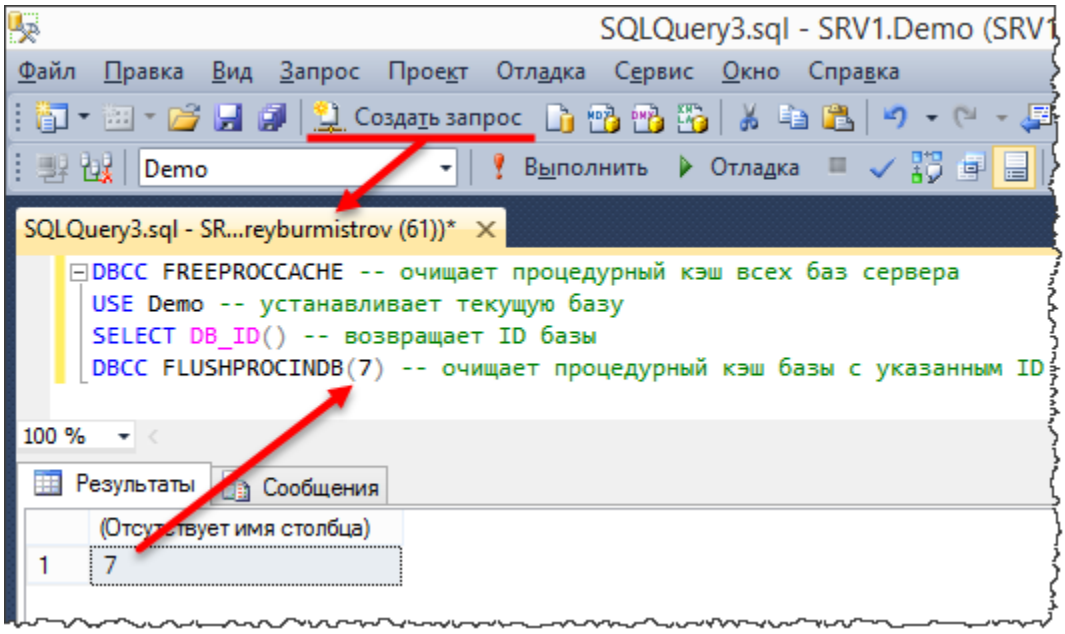
Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Для очистки процедурного кэша только одной базы можно использовать недокументированную команду DBCC FLUSHPROCINDB(ID). В качестве параметра ID необходимо подставить значение, полученное запросом «SELECT DB_ID()». Чтобы получить ID нужной базы, необходимо перед запросом дописать команду «USE ИмяБазыSQL».

Для выполнения всех вышеперечисленных команд следует использовать Management Studio.

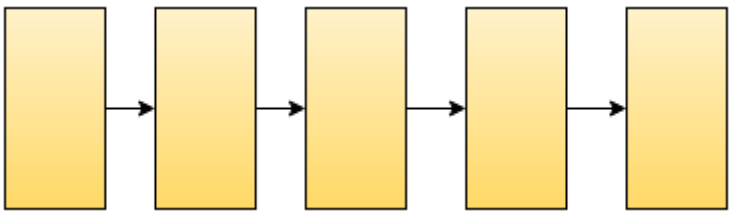


Занятие 6

Дефрагментация индексов

Удобнее всего рассматривать фрагментацию на примере.

Допустим, что мы создали таблицу и заполнили ее первоначальными данными. В этом случае страницы индекса будут следовать на диске друг за другом.



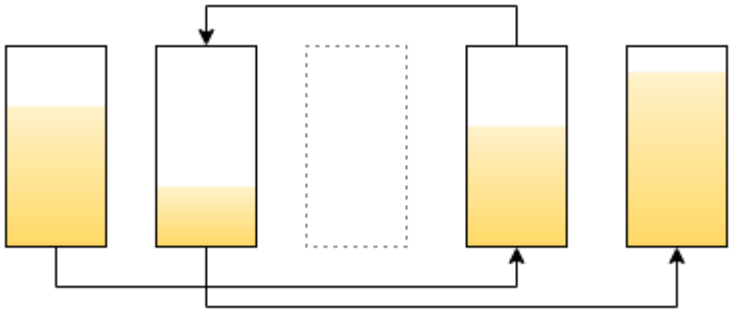
Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

При чтении индекса головка диска будет производить последовательное чтение, что гораздо быстрее произвольного.

Далее с базой данных идет активная работа, выполняется много операций удаления, изменения и вставки. При этом страницы индексов видоизменяются, и новая страница может быть создана далеко от предыдущей страницы, внутри страниц так же образуются незаполненные зазоры. Все вместе это приводит к фрагментации индексов.



Фрагментация может быть внутренней, когда страница заполнена не полностью, и в ней есть пустое место, и внешней, когда пустое место появляется между страницами. MS SQL Server хранит информацию в страницах по 8 КБ, и минимальным объемом читаемых данных так же является страница. MS SQL Server может за один раз прочитать больше одной страницы, но меньше он прочитать не может, даже если требуется всего 1 КБ данных.

Из-за внешней фрагментации головке диска приходится делать гораздо больше перемещений, что замедляет чтение данных, в результате чего запросы выполняются медленнее.

Из-за внутренней фрагментации, приходится считывать больше страниц, что приводит к повышенному расходу оперативной памяти. Например, нужно прочитать 7 КБ данных. Из-за фрагментации 4 КБ данных расположены на 1 странице и 3 КБ на другой, придется прочитать 2 страницы, что займет 16 КБ в оперативной памяти. Если бы внутренней фрагментации не было, то все 7 КБ располагались бы на 1 странице и потребовалось бы только 8 КБ оперативной памяти.

Если база расположена на SSD дисках, то внешняя фрагментация не играет роли, но из-за внутренней фрагментации, дефрагментацию индексов все равно необходимо выполнять, но реже.

Дефрагментация (реорганизация) индексов – это процесс устранения внутренней и внешней фрагментации.

Для дефрагментации используется команда ALTER INDEX ALL ON ИмяТаблицы REORGANIZE.

Некоторые источники предлагают для дефрагментации использовать команду DBCC INDEXDEFRAG, но эта команда является устаревшей, и в будущих версиях MS SQL Server она будет удалена.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Microsoft рекомендует делать дефрагментацию в том случае, если фрагментация индекса более 5 % и не превышает 30 %. В случае, если фрагментация выше, рекомендуется сделать перестроение индекса, т.е. реиндексацию. На практике нижний порог можно смело увеличить до 10 %.

Посмотреть степень фрагментации индекса можно либо в свойствах индекса на странице *Фрагментация*, либо с помощью системного представления sys.dm_db_index_physical_stats.

Пример использования данного представления приведен в скрипте дефрагментации и реиндексации.

Выполнение дефрагментации не блокирует работу пользователей, но создает некоторую нагрузку на оборудование, поэтому не рекомендуется выполнять дефрагментацию в рабочее время.

Дефрагментацию рекомендуется выполнять скриптом минимум один раз в неделю.

Реиндексация таблиц

Реиндексация таблиц – это операция пересоздания индексов.

При высокой степени фрагментации эффективнее будет создать индекс заново, чем делать дефрагментацию.

Для выполнения реиндексации используется команда ALTER INDEX ALL ON ИмяТаблицы REBUILD.

Пример использования команды приведен в скрипте для дефрагментации/реиндексации.

До версии MS SQL Server 2005 включительно выполнение данной команды блокировало работу пользователей, и, следовательно, ее не стоило выполнять в рабочее время. Начиная с версии MS SQL Server 2008 появился параметр ONLINE, который позволяет выполнять реиндексацию, практически не мешая работе пользователей. Следует отметить, что данный параметр доступен только в версиях SQL Server Enterprise Edition, Developer Edition и Evaluation Edition.

Данный параметр можно использовать с некоторыми ограничениями: например, если в таблице есть поле неограниченной длины или типа «ХранилищеЗначений», то перестроить ее индекс онлайн уже не получится, будет выдана ошибка.

Чтобы избежать ошибок и быть уверенным, что все индексы без исключения будут перестроены, не рекомендуется использовать онлайновое перестроение индексов без контроля исполнения. Для баз с режимом работы 24/7 можно написать скрипт таким образом, чтобы он перестраивал онлайн те индексы, которые поддерживают такое перестроение. Индексы, не поддерживающие онлайновое перестроение, можно перестраивать отдельным скриптом во время технологических окон, которые должны быть выделены для обслуживания системы.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Реиндексация является самой ресурсоемкой регламентной операцией. Кроме того, она обладает следующими особенностями:

- Во время выполнения занимает дополнительное место на диске
- Может быть ускорена с помощью базы TempDB
- Может быть ускорена с помощью многопроцессорной обработки
- Перестроение кластерного индекса не вызывает перестроения некластерных индексов
- На маленьких таблицах фрагментация не устраняется.

Разберем эти особенности подробнее.

Перестройка индексов занимает дополнительное место на диске.

Прежде чем делать реиндексацию на рабочей базе, необходимо проверить, сколько времени и дискового пространства займет данная операция на копии базы. Особенно это актуально для баз большого размера. Также перед реиндексацией желательно сделать резервную копию базы данных.

Реиндексация может быть ускорена с помощью базы TempDB.

Если TempDB и файл данных рабочей базы находятся на разных физических дисках, и при этом на диске с базой TempDB есть запас свободного места, равный размеру самой большой таблицы базы, тогда можно ускорить перестроение индексов, используя параметр SORT_IN_TEMPDB. Ускорение достигается за счет того, что часть данных для выполнения операции будет сохранена в базе TempDB, что позволяет немного разделить нагрузку между дисками. В представленном скрипте данный параметр по умолчанию включен.

Реиндексация может быть ускорена с помощью многопроцессорной обработки.

Можно ускорить перестроение индексов, используя многопроцессорную обработку, для этого необходимо указать параметр MAXDOP. Параметр задает число процессоров, задействованных при выполнении одной индексной операции. Если параметр MAXDOP не задан, то он определяется параметром конфигурации MAX DEGREE OF PARALLELISM. Если инструкции по настройке СУБД из предыдущего раздела были выполнены, то этот параметр должен принимать значение 1.

Чтобы ускорить перестроение индекса, рекомендуется установить MAXDOP=0, тогда СУБД будет автоматически определять, сколько процессов можно использовать. Данный параметр действует только при выполнении текущей инструкции и не переопределяет параметр базы данных MAX DEGREE OF PARALLELISM.

Перестроение кластерного индекса не вызывает перестроения некластерных индексов

Есть мнение, что при перестроении кластерного индекса, некластерные индексы этой таблицы перестроятся автоматически, т.е. перестраивать их не нужно. Но это верно только в том случае, если кластерный индекс не является уникальным. Подробно структура и виды индексов будут рассмотрены в соответствующем разделе. В базах 1С 99% кластерных индексов являются уникальными, поэтому необходимо перестраивать все индексы в любом случае.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

На маленьких таблицах фрагментация не устраняется

При выполнении дефрагментации/реиндексации можно заметить, что некоторые индексы по-прежнему фрагментированы. Как правило, это наблюдается у таблиц малого размера.

Для таблиц малого размера нет смысла делать дефрагментацию или реиндексацию, т.к. из-за особенностей хранения страниц в таких таблицах полностью устранить фрагментацию в них не удастся. Это не является проблемой, т.к. фрагментация маленького индекса не будет оказывать существенного влияния на производительность. Например, в скрипте для дефрагментации и реиндексации стоит фильтр на дефрагментацию только тех таблиц, в которых более 128 страниц.

Как уже было сказано, рекомендуется делать дефрагментацию/реиндексацию минимум 1 раз в неделю, но для больших баз с высоким уровнем доступности это может быть невыполнимо. В этом случае частота реиндексации подбирается индивидуально.

Для выполнения реиндексации на больших таблицах удобно использовать праздничные и выходные дни. Также следует помнить о том, что план обслуживания можно разбить на части, например, часть индексов можно перестроить в режиме онлайн в рабочие дни, а часть в автономном режиме в выходные дни. Для оценки времени на выполнение реиндексации можно использовать копию рабочей базы.

После выполнения дефрагментации/реиндексации нет необходимости обновлять статистику, т.к. при дефрагментации плотность данных не меняется, а при реиндексации индекс создается заново, и статистика по нему пересоздается автоматически.

Видеоурок. Настройка регламентной операции обновления статистики

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как создать план обслуживания
- Как обновить статистику одновременно для нескольких таблиц
- Как настроить расписание выполнения регламентной операции.

Видеоурок. Настройка регламентной операции дефрагментации/реиндексации

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как выполнить дефрагментацию/реиндексацию с помощью скрипта
- Какие настройки скрипта необходимо указать.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Перенос плана обслуживания

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Каким образом импортировать/экспортировать планы обслуживания.

Видеоурок. Оповещения по e-mail

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как настроить компонент Database Mail
- Как настроить автоматическую отправку писем о выполнении регламентных операций
- Как отправлять письма нескольким пользователям.

Видеоурок. Возможные проблемы при настройке оповещений

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Наиболее частые проблемы при настройке оповещений и способы их решения.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 7

Регламентные операции 1С

Принцип получения итогов

Во многих источниках говорится о необходимости ежемесячно сдвигать границу рассчитанных итогов, но далеко не везде объясняется, зачем это нужно делать. Также зачастую многие специалисты путают операции сдвига границы рассчитанных итогов и пересчета итогов.

Для того чтобы понять, в каком случае двигать границу итогов, а в каком их пересчитывать, необходимо иметь представление, каким образом платформа хранит и рассчитывает эти итоги.

Все регистры на уровне СУБД состоят из нескольких таблиц. Например, регистр накопления состоит из таблицы движений (основная таблица) и таблицы итогов. Если вид регистр накопления «Остатки», то в таблице итогов хранятся остатки, если вид регистра «Обороты», то хранятся обороты. В дальнейшем для примера будем использовать регистр остатков с измерениями «Склад», «Товар» и одним ресурсом «Количество».

Данные в таблице движений хранятся с периодичностью по регистратору, а в таблице итогов – с периодичностью по месяцам. Представим ситуацию, что за март и апрель 2015 года было проведено 2 приходных документа и один расходный, в этом случае таблицы регистра накопления примут следующий вид.

Ниже представлена таблица движений, цветом разделены данные по месяцам.

Период	Регистратор	Склад	Товар	Количество
15.03.2015 18:35:04	Поступление товара 000000001 от 15.03.2015 18:35:04	Средний	Сапоги	1
15.03.2015 18:35:04	Поступление товара 000000001 от 15.03.2015 18:35:04	Средний	Туфли	20
19.03.2015 12:40:57	Расход товара 000000002 от 19.03.2015 12:40:57	Средний	Туфли	1
19.03.2015 12:40:57	Расход товара 000000002 от 19.03.2015 12:40:57	Средний	Сапоги	1
12.04.2015 11:45:40	Поступление товара 000000003 от 01.04.2015 11:45:40	Малый	Сапоги	2
18.04.2015 11:45:40	Поступление товара 000000003 от 01.04.2015 11:45:40	Средний	Туфли	2
…	…	…	…	…

Все движения позднее апреля отображаются как многоточия.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Таблица итогов.

Период	Склад	Товар	Количество
01.04.2015	Средний	Сапоги	0
01.04.2015	Средний	Туфли	19
01.05.2015	Средний	Сапоги	0
01.05.2015	Средний	Туфли	21
01.05.2015	Малый	Сапоги	2
01.11.3999	Средний	Сапоги	0
01.11.3999	Средний	Туфли	37
01.11.3999	Малый	Сапоги	2

Итоги за месяц хранятся на первое число следующего месяца, т.е. остаток на конец марта записан на 1 апреля. В данном примере граница итогов рассчитана по апрель включительно на 01.05.2015. За май и последующие месяцы итогов нет, хотя движения в этих периодах есть.

Оранжевым цветом отмечены текущие итоги, они хранятся на дату 01.11.3999. Текущие итоги также иногда называют актуальными итогами. Они обновляются при записи любого движения, т.е. это итоги на самый последний момент времени. В данном примере для товара «Туфли» текущий остаток равен 37, т.к. за май и июнь по нему были движения, которые в таблицу итогов не попали, потому что итоги рассчитаны только по апрель.

При изменении остатка за март «задним» числом будут пересчитаны все итоги по данным значениям измерений за апрель, май и все последующие месяцы, если бы они были в таблице итогов, а также текущие итоги.

Когда идет обращение к виртуальной таблице остатков, предварительно выполняется уточняющий запрос для получения настроек регистра: можно ли использовать итоги, на какую дату они рассчитаны, включены ли текущие итоги и т.д. Исходя из результатов уточняющего запроса, а также параметров виртуальной таблицы, особенно параметра даты получения остатков, строится итоговый текст запроса получения остатков.

Допустим, необходимо получить остаток по товару «Туфли» на складе «Средний», граница рассчитанных итогов – 01.05.2015. В этом случае возможны различные варианты, в зависимости от даты, на которую необходимо получить остаток.

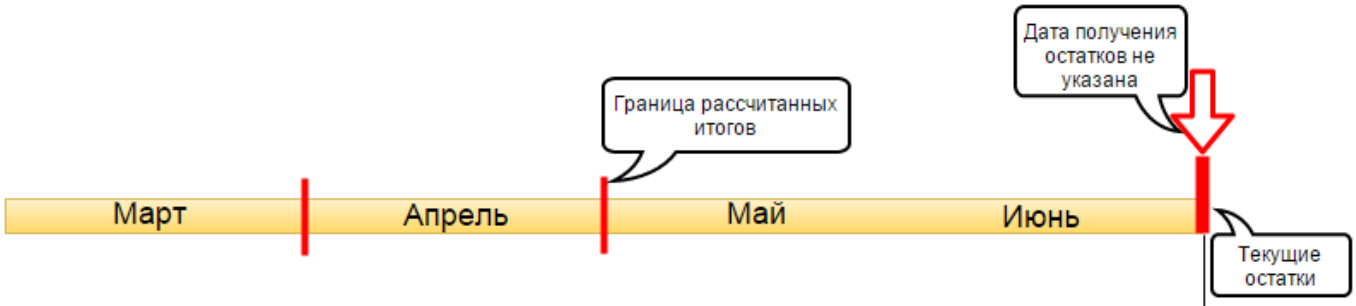
Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Вариант 1: дата не указана или указано первое число месяца, по которому рассчитаны итоги.

В этом случае будет обращение только к таблице итогов, таблица движений не используется. Если дата не указана, то будут использованы текущие итоги.



Период	Регистратор	Склад	Товар	Количество
15.03.2015 18:35:04	Поступление товара 000000001	Средний	Сапоги	1
15.03.2015 18:35:04	Поступление товара 000000001	Средний	Туфли	20
19.03.2015 12:40:57	Расход товара 000000002	Средний	Туфли	1
19.03.2015 12:40:57	Расход товара 000000002	Средний	Сапоги	1
12.04.2015 11:45:40	Поступление товара 000000003	Малый	Сапоги	2
12.04.2015 11:45:40	Поступление товара 000000003	Средний	Туфли	2
...	...	...	...	...

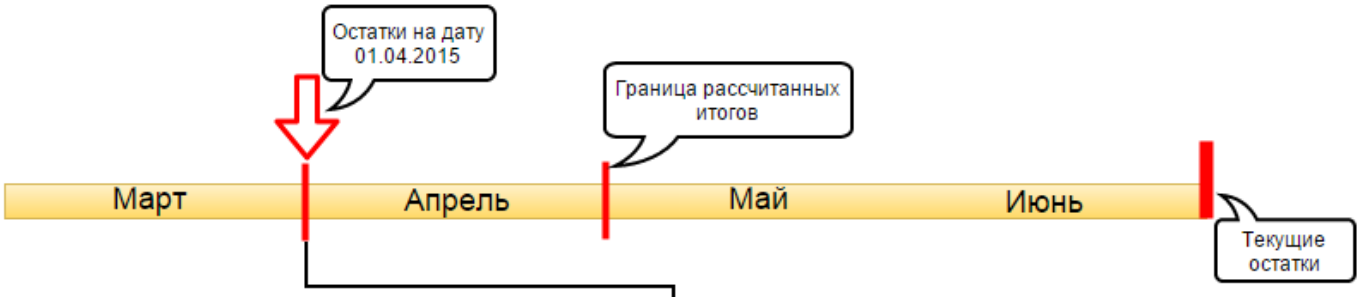
Период	Склад	Товар	Количество
01.04.2015	Средний	Сапоги	0
01.04.2015	Средний	Туфли	19
01.05.2015	Средний	Сапоги	0
01.05.2015	Средний	Туфли	21
01.05.2015	Малый	Сапоги	2
01.11.3999	Средний	Сапоги	0
01.11.3999	Средний	Туфли	37
01.11.3999	Малый	Сапоги	2

Если же дата указана и это первое число месяца, который входит в период рассчитанных итогов, тогда из таблицы итогов будет выбрана строка итогов за соответствующий месяц.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



Период	Регистратор	Склад	Товар	Количество
15.03.2015 18:35:04	Поступление товара 000000001	Средний	Сапоги	1
15.03.2015 18:35:04	Поступление товара 000000001	Средний	Туфли	20
19.03.2015 12:40:57	Расход товара 000000002	Средний	Туфли	1
19.03.2015 12:40:57	Расход товара 000000002	Средний	Сапоги	1
12.04.2015 11:45:40	Поступление товара 000000003	Малый	Сапоги	2
12.04.2015 11:45:40	Поступление товара 000000003	Средний	Туфли	2
...	...	...	...	...

Период	Склад	Товар	Количество
01.04.2015	Средний	Сапоги	0
01.04.2015	Средний	Туфли	19
01.05.2015	Средний	Сапоги	0
01.05.2015	Средний	Туфли	21
01.05.2015	Малый	Сапоги	2
01.11.3999	Средний	Сапоги	0
01.11.3999	Средний	Туфли	37
01.11.3999	Малый	Сапоги	2

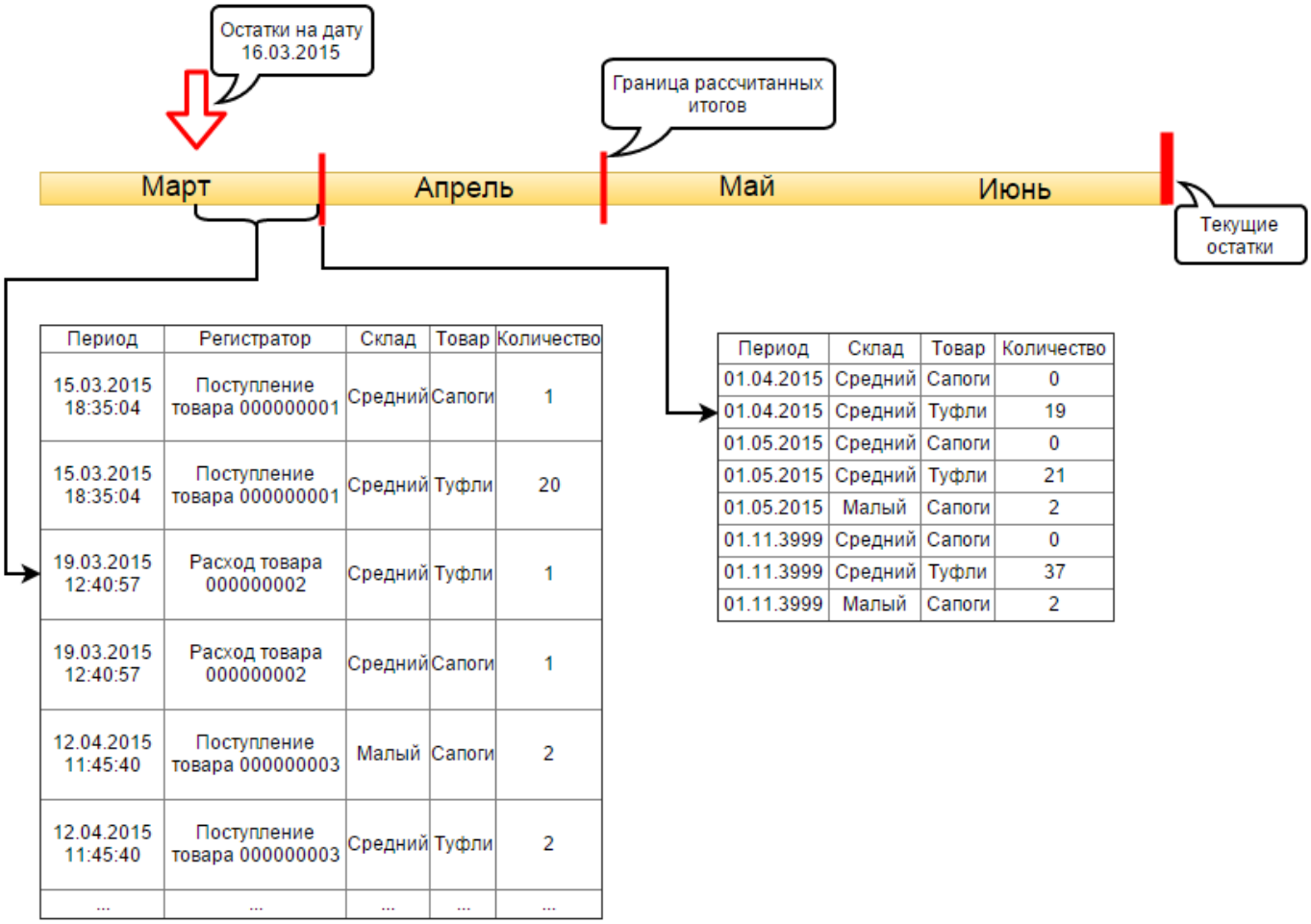
Вариант 2: дата находится в пределах рассчитанных итогов и отличается от первого дня месяца.

В этом случае будет получен остаток из таблицы итогов на первое число следующего от указанной даты месяца и учтены движения с указанной даты по конец месяца.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



Период	Регистратор	Склад	Товар	Количество
15.03.2015 18:35:04	Поступление товара 000000001	Средний	Сапоги	1
15.03.2015 18:35:04	Поступление товара 000000001	Средний	Туфли	20
19.03.2015 12:40:57	Расход товара 000000002	Средний	Туфли	1
19.03.2015 12:40:57	Расход товара 000000002	Средний	Сапоги	1
12.04.2015 11:45:40	Поступление товара 000000003	Малый	Сапоги	2
12.04.2015 11:45:40	Поступление товара 000000003	Средний	Туфли	2
...	...	...	...	...

Период	Склад	Товар	Количество
01.04.2015	Средний	Сапоги	0
01.04.2015	Средний	Туфли	19
01.05.2015	Средний	Сапоги	0
01.05.2015	Средний	Туфли	21
01.05.2015	Малый	Сапоги	2
01.11.3999	Средний	Сапоги	0
01.11.3999	Средний	Туфли	37
01.11.3999	Малый	Сапоги	2

Для расчета остатка в данном случае к значению из таблицы итогов (19 шт.) будут прибавлены расходы (1 шт.) и вычтены приходы (приходов указанного товара на данный склад с 16 по 31 марта не было).

В результате расчетов 19 + 1 получаем остаток на 16.03.2015 равный 20.

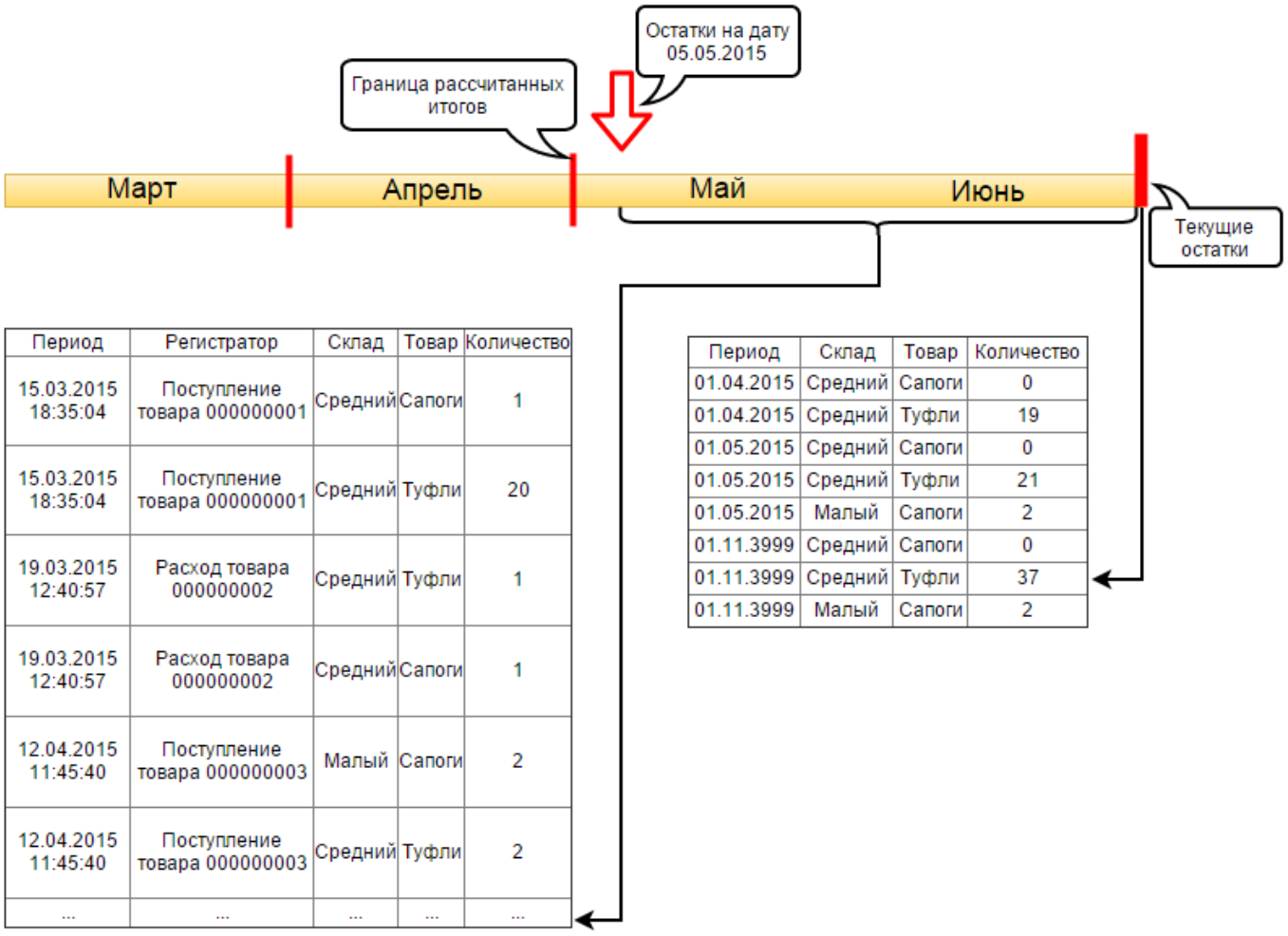
Вариант 3: дата остатков выходит за границу рассчитанных итогов.

В этом случае сначала из таблицы итогов будет получен текущий остаток, а потом учтены движения от даты получения остатка до текущих остатков.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



Период	Регистратор	Склад	Товар	Количество
15.03.2015 18:35:04	Поступление товара 000000001	Средний	Сапоги	1
15.03.2015 18:35:04	Поступление товара 000000001	Средний	Туфли	20
19.03.2015 12:40:57	Расход товара 000000002	Средний	Туфли	1
19.03.2015 12:40:57	Расход товара 000000002	Средний	Сапоги	1
12.04.2015 11:45:40	Поступление товара 000000003	Малый	Сапоги	2
12.04.2015 11:45:40	Поступление товара 000000003	Средний	Туфли	2
...	...	...	...	...

Период	Склад	Товар	Количество
01.04.2015	Средний	Сапоги	0
01.04.2015	Средний	Туфли	19
01.05.2015	Средний	Сапоги	0
01.05.2015	Средний	Туфли	21
01.05.2015	Малый	Сапоги	2
01.11.3999	Средний	Сапоги	0
01.11.3999	Средний	Туфли	37
01.11.3999	Малый	Сапоги	2

В данном примере итоги рассчитаны только за апрель, а остаток требуется в мае. Из-за того, что граница итогов не сдвинута, приходится просматривать таблицу движений почти за 2 месяца, что может сильно замедлить запрос.

Необходимо ежемесячно сдвигать границу рассчитанных итогов, иначе получение остатков может замедлиться из-за обращения к таблице движений и чтения большого числа строк.

Конечно, было бы оптимальнее получать ближайший к требуемой дате рассчитанный остаток, а не брать всегда текущие итоги, но разработчики платформы по какой-то причине решили этого не делать. Вполне возможно, что когда вы будете читать эти строки, в платформу уже внесут это изменение.

Подробнее с устройством регистра накопления и механизмом получения итогов можно ознакомиться в дополнительных материалах курса.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Зачем пересчитывать итоги

Внимательный читатель заметил, что в таблице итогов присутствуют строки, где количество равно нулю. Это происходит в том случае, если количество приходов и расходов по набору значений измерений одинаково, т.е. списали все, что поступило в этом месяце. Платформа намеренно не удаляет нулевые записи, т.к. операция удаления отнимает больше ресурсов, чем операция обновления. К тому же в любой момент могут появиться движения по этому набору измерений, а обновить строку гораздо «дешевле» для сервера, чем вставлять новую.

Количество строк с нулевыми значениями ресурсов постоянно увеличивается, что замедляет запросы к таблице итогов, потому что приходится сначала эти строки прочитать, а потом откинуть.

Если запустить процесс пересчета итогов, то таблица итогов обновится на основании данных таблицы движений, при этом строки с нулевыми значениями ресурсов будут удалены. Так будет выглядеть таблица итогов после пересчета.

Период	Склад	Товар	Количество
01.04.2015	Средний	Туфли	19
01.05.2015	Средний	Туфли	21
01.05.2015	Малый	Сапоги	2
01.11.3999	Средний	Туфли	37
01.11.3999	Малый	Сапоги	2

Следует различать операции установки границы рассчитанных итогов и пересчет итогов. В первом случае нулевые итоги не удаляются, даже наоборот, появляются новые записи в таблице итогов на следующий месяц, и если в прошлом месяце были нулевые итоги, то они переносятся в следующий месяц. Если делать пересчет итогов, то никаких новых записей не появляется, вместо этого из таблицы итогов удаляются записи, у которых значения всех ресурсов равны нулю.

В большинстве случаев нет необходимости пересчитывать итоги за все время, как правило, достаточно ежемесячно пересчитывать только текущие итоги. Запросы к остаткам в большинстве случаев строятся на даты рядом с текущей, поэтому чаще всего используются именно текущие итоги.

Здесь есть один интересный нюанс. У регистров накопления и бухгалтерии есть метод *ПересчитатьИтогиЗаПериод(Дата1, Дата2)*. Данный метод, как несложно догадаться, пересчитывает итоги только за указанный период, но у этого метода есть одна интересная особенность.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Допустим, есть база с данными за 10 лет, была произведена свертка, и были оставлены все документы только за последний год. В результате в таблице итогов регистров будут записи с нулевыми итогами за последние 9 лет. Большая часть таблицы будет занята бесполезной информацией.

Теперь допустим, что программист решил с помощью метода *ПересчитатьИтогиЗаПериод()* выполнить пересчет итогов, чтобы свернуть эти строки. После пересчета выяснится, что нулевые итоги за 9 лет не удалились. Это произошло потому, что данный метод находит самую раннюю дату движения в таблице движений регистра, и если она старше даты, указанной в первом параметре, то берется самая ранняя дата движения в качестве параметра. Получается, что если пересчитывать итоги с 01.01.2003 г., а первое движение было только в 2014 году, то данный метод возьмет в качестве первого параметра 2014 год.

Если выполнить полный пересчет итогов, то нулевые строки будут удалены в любом случае. Поэтому после свертки базы рекомендуется выполнять полный пересчет итогов по регистрам.

Крайне редко, но возможна ситуация, когда в отчетах итоговый остаток отличается от оборотов, т.е. если посчитать все движения, то должен быть один остаток, а в отчете другой. В этом случае рекомендуется также сделать полный пересчет итогов по данному регистру.

Для оборотного регистра накопления проблема нулевых значений в таблице итоговых оборотов также актуальна и решается аналогично – пересчетом итогов.

Если по регистру включен разделитель итогов, то пересчет итогов сворачивает строки таблицы итогов с различными значениями разделителя. Назначение и принцип работы разделителя итогов подробно рассматриваются в отдельной главе.

Видеоурок. Установка границы рассчитанных итогов в режиме 1С:Предприятие

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что такое минимальный период рассчитанных итогов
- Установка границы рассчитанных итогов в режиме 1С:Предприятие.

Видеоурок. Установка границы рассчитанных итогов с помощью встроенного языка

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- Каким образом можно установить границу рассчитанных итогов с помощью встроенного языка.

Видеоурок. Пересчет итогов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как узнать количество нулевых строк в таблице итогов по каждому регистру
- Как пересчитать итоги в режиме 1С:Предприятие
- Особенности пересчета итогов из конфигуратора.

Видеоурок. Тестирование и исправление

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как часто необходимо выполнять операцию тестирования и исправления
- Особенности выполнения данной операции.

Регламентные операции. Итоги

К основным регламентным операциям относятся:

- Обновление статистики
- Дефрагментация и реиндексация индексов
- Сдвиг границы рассчитанных итогов
- Пересчет итогов.

Тестирование и исправление можно выполнять раз в квартал или в полугодие. Регламентные операции должны быть настроены для каждой рабочей базы, это является залогом оптимальной работы. Без регулярного обслуживания начнется постепенная деградация производительности. После настройки регламентных операций в обязательном порядке необходимо настроить уведомления о выполнении регламентных операций.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Мониторинг загруженности оборудования

Занятие 8

Объекты мониторинга

Обязательным условием процесса оптимизации является анализ загруженности оборудования. В идеале, мониторинг загруженности оборудования должен быть настроен с момента начала работы системы, но, как показывает практика, этого обычно никто не делает. В большинстве случаев от администраторов сложно получить информацию о том, как были нагружены диски 2 недели назад на определенном сервере. Для выполнения работ по оптимизации необходимо иметь фактические данные о загруженности различных аппаратных компонентов сервера на любой момент времени. Необходимо подчеркнуть, что любые заявления администратора о загрузке оборудования должны быть подкреплены соответствующими логами.

Ниже перечислены объекты мониторинга оборудования в порядке убывания приоритета:

- Сервер СУБД
- Сервер приложений 1С:Предприятие
- Терминальный сервер (при наличии)
- Веб-сервер (при наличии)

Самыми важными в этом списке являются серверы СУБД и 1С. Нагрузку на терминальный сервер следует отслеживать в обязательном порядке, т.к. возможна ситуация, когда система работает быстро, но интерфейс на терминальном сервере прорисовывается медленно. В результате APDEX хороший, а пользователи недовольны скоростью работы системы.

Базовые счетчики производительности

Для мониторинга производительности оборудования используются определенные показатели, которые собираются с помощью счетчиков. Ниже приведен базовый набор счетчиков, которые должны собираться на всех серверах в обязательном порядке.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

В ОС Windows для сбора данных используется Системный монитор, он же Performance Monitor. Названия счетчиков приведены на английском языке, так, как они называются в системном мониторе, в русской локализации названия будут на русском языке. Если используется другая ОС или средства виртуализации, то там также можно будет получить такую информацию, используя соответствующие (предназначенные для данной ОС) инструменты и счетчики.

	Счетчик	**Описание**	**Критерий нормы**
Диск	**PhysicalDisk(*)\ Avg. Disk Queue Length**	Очередь к дискам	Не более 2 на 1 физический диск, работающий параллельно
	PhysicalDisk(*)\ Avg. Disk Write Time	Среднее время записи на диск	Не более 15 мс.
	PhysicalDisk(*)\ Avg. Disk Read Time	Среднее время чтения с диска	Не более 15 мс.
Процессор	**Processor(_Total)\ % Processor Time**	% загруженности процессоров	Не более 80 %
	System\ Processor Queue Length	Очередь к процессорам	Не более 2 на 1 ядро
Память	**Memory\ Available Mbytes**	Доступная оперативная память	Не менее 400 – 500 МБ
Сеть	**Network Adapter(*)\ Bytes Total/sec**	Скорость передачи данных по сети	Не более 65% от пропускной способности сетевого адаптера

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Разберем каждый из счетчиков подробнее.

Avg. Disk Queue Length – это средняя длина очереди к диску. Другими словами, сколько запросов на чтение или на запись ожидает доступа к диску. В нормальных условиях это значение не должно превышать 2 на каждый физический диск, работающий параллельно. Например, если у вас 2 диска работают параллельно, то очередь не должна превышать 4.

Все данные дисковых счетчиков необходимо собирать в разрезе логических дисков.

Довольно часто у многих специалистов возникает вопрос, как трактовать значение счетчика очереди к диску, если используется RAID-массив. В этом случае нужно понять, сколько дисков в данном массиве работает параллельно. Для каждого типа RAID будет свое количество параллельно работающих дисков. Рассмотрим наиболее популярные типы RAID. Допустим, есть 12 дисков, объединенных в RAID различной конфигурации.

RAID10 – 6 дисков работает параллельно, 6 используются для зеркалирования, следовательно, максимально допустимая очередь – 12.

RAID5 – 11 дисков работает параллельно, 1 используется для контроля четности, максимально допустимая очередь – 22.

RAID0 – 12 дисков работает параллельно, максимально допустимая очередь – 24.

Очень подробно все возможные виды RAID с их плюсами и минусами описаны в данной статье https://ru.wikipedia.org/wiki/RAID

Для высоконагруженных систем обычно используются RAID10 и, как правило, это оправдано.

Avg. Disk Write Time – среднее время записи на диск. Запись на диск не должна превышать 15 миллисекунд. Обычно это время гораздо меньше, т.к. запись считается выполненной, как только данные попадают в дисковый кэш.

Avg. Disk Read Time – среднее время чтения с диска. Чтение с диска также не должно превышать 15 миллисекунд.

Важно помнить, что показатели диска нужно снимать отдельно для каждого логического диска, а не для физического диска, это очень важно для правильной трактовки данных.

В данных счетчиках везде указано среднее значение и возникает вопрос, среднее относительно чего? При настройке счетчиков указывается периодичность получения данных, и среднее значение считается за этот интервал. Например, указана периодичность сбора данных 5 секунд, значит, будет отображено среднее значение счетчика за 5 секунд.

Исходя из 3-х вышеописанных счетчиков можно делать выводы о состоянии дисковой подсистемы. Если показатели превышены длительное время, это говорит о том, что диски возможно перегружены.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Особое внимание нужно обратить на те диски, на которых находятся файлы данных и файлы логов (журналов транзакций) вашей базы, а также база TempDB.

% Processor Time – процент загруженности процессора. При необходимости можно посмотреть загрузку с детализацией по ядрам процессора. Загрузка процессора не должна превышать 80% на протяжении длительного времени. Не стоит ждать, когда загрузка достигнет 90% или 100%, если процессор загружен выше 80% длительное время, это уже говорит о том, что, заменив процессор, можно ускорить работу системы. Это не значит, что нужно сразу делать апгрейд, вполне возможно, что после программной оптимизации нагрузка на оборудование снизится.

Processor Queue Length – длина очереди процессора. Это текущая длина очереди процессора, измеряемая числом ожидающих потоков. Все процессоры используют одну общую очередь, в которой потоки ожидают получения циклов процессора. Этот счетчик не включает потоки, которые выполняются в настоящий момент. Длительное время существующая очередь длиной больше двух потоков на 1 ядро процессора обычно свидетельствует о перегруженности процессора. Этот счетчик отражает текущее значение и не является средним значением по некоторому интервалу времени.

В некоторых источниках можно встретить информацию о том, что для мониторинга загруженности процессора полезно использовать счетчик «\System\Threads» (количество потоков). Например, этот счетчик указан вместе с двумя вышеописанными в верном ответе на один из билетов 1С:Профессионал по технологическим вопросам. Фактически данный счетчик не несет практически никакой ценной информации для анализа. Большое количество потоков еще не говорит о том, что процессор перегружен, вполне возможно, что процессор успешно справляется с нагрузкой.

Available Mbytes – объем физической оперативной памяти, доступной для использования. В системе обязательно должно быть свободно как минимум 400–500 МБ, иначе операционная система может начать активно использовать своп, что сильно увеличит нагрузку на диск и приведет к резкой деградации производительности.

В некоторых источниках можно встретить информацию, что для анализа загруженности памяти следует использовать счетчик **Pages/sec**, который показывает интенсивность обмена между диском и памятью, но это не совсем так. Счетчик **Pages/sec** не дает однозначного ответа, хватает памяти или нет, высокое значение данного счетчика может быть вызвано, например, копированием большого числа файлов, поэтому однозначно судить о нагрузке на память только по этому счетчику нельзя.

Bytes Total/sec – скорость передачи данных по сети. Счетчик отображает скорость получения и отправки данных в разрезе сетевых карт.

Вышеописанные счетчики дают общую информацию о загруженности оборудования, без сильной детализации. Данный список далеко не полный, его можно и нужно дополнять по вашему желанию и исходя из ваших потребностей и вашей ситуации. Например, если видны задержки по диску, можно добавить дополнительные дисковые счетчики.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Если в системе оборудование действительно перегружено, то это будет хорошо заметно: значения счетчиков в этом случае превышают нормы не на 1-2 пункта, а в разы или десятки раз.

Счетчики производительности для сервера СУБД

При установке MS SQL Server добавляет свои собственные счетчики производительности. Поэтому для сервера СУБД к базовому набору можно добавить еще несколько счетчиков, с помощью которых можно определить, хватает ли памяти MS SQL Server.

Счетчик	**Описание**	**Критерий нормы**
Process("sqlservr")\% Processor Time	Загруженность процессора процессом MS SQL Server	
SQLServer:Buffer Manager\ Buffer cache hit ratio	Процент чтения страниц из кэша	Более 90 %
SQLServer:Buffer Manager\ Page life expectancy	Среднее время нахождения страницы в буфере	Не менее 300 сек.

Счетчик **Process("sqlservr")\% Processor Time** показывает, насколько сильно MS SQL Server нагружает процессор. Эти данные могут быть особенно полезны, если на компьютере установлено несколько ресурсоемких приложений и при этом необходимо выявить, которое из них нагружает процессор.

Buffer cache hit ratio – процент чтения страниц данных из кэша в оперативной памяти. Счетчик показывает, насколько часто сервер СУБД находит данные в памяти, а значит, ему не приходится обращаться за ними на диск. Показатель должен быть выше 90%. Если памяти мало, то сервер СУБД постоянно будет читать данные с диска, что сильно замедлит его производительность.

Page life expectancy – среднее время нахождения страницы в буферном пуле. Счетчик показывает, как долго данные лежат в буфере. Чем выше значение счетчика, тем дольше данные находятся в буфере и тем лучше для производительности.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Если значения счетчика часто опускается ниже 300 сек. даже на короткий промежуток времени, это может указывать на проблему с памятью. Вовсе не обязательно при проблемах с памятью сразу обновлять сервер, чаще всего проблема в коде, например, в неоптимальных запросах, которые из-за отсутствия индекса читают слишком большой объем данных.

Счетчики производительности для сервера 1С

Для сервера приложений также можно добавить несколько счетчиков производительности, которые могут быть полезны.

Счетчик	Описание
Process("ragent*")\% Processor Time	Загруженность процессора с разбивкой по процессам сервера 1С
Process("rmngr*")\% Processor Time	
Process("rphost*")\% Processor Time	
Process("ragent*")\ Working Set - Private	Объем оперативной памяти, занимаемый процессами сервера 1С
Process("rmngr*")\ Working Set - Private	
Process("rphost*")\ Working Set - Private	

Первый счетчик показывает, насколько сильно загружает процессор каждый из процессов сервера 1С, что может быть полезно при поиске виновника высокой загруженности процессора на сервере приложений.

Working set Private – отображает частный рабочий набор, занимаемый процессами сервера 1С. Можно считать, что данный счетчик отображает объем оперативной памяти, занимаемый процессами.

Оба счетчика не имеют каких-либо критериев нормы, их значения можно рассматривать только относительно.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Например, на компьютере установлены и сервер 1С, и сервер СУБД, при этом загрузка процессора 95%, необходимо выяснить, виноват ли в этом сервер приложений или сервер СУБД. С помощью счетчика загруженности процессора в разрезе процессов эту задачу можно легко выполнить.

Занятие 9

Мониторинг загруженности оборудования для Windows

Видеоурок. Добавление счетчиков вручную

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Какими правами необходимо обладать для использования системного монитора
- Как создать группу сборщиков данных
- Как добавить определенные счетчики

Видеоурок. Добавление счетчиков шаблоном

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Различные способы создания группы сборщиков данных
- Возможные ошибки при создании группы сборщиков данных.

Видеоурок. Добавление счетчиков bat-файлом

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Различные способы создания группы сборщиков данных
- Возможные ошибки при создании группы сборщиков данных.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Настройка группы сборщиков данных

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как указать формат и периодичность записи логов
- Как запустить и завершить счетчик в строго заданное время
- С какой периодичностью перезапускать сборщик данных
- Прочие настройки логов системного монитора.

Видеоурок. Настройка автозапуска после перезагрузки

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как настроить автозапуск bat файлом
- Как сделать автозапуск планировщиком заданий.

Видеоурок. Просмотр и анализ графиков загруженности оборудования

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как посмотреть данные о загруженности из нескольких файлов
- Сколько времени должны быть превышены счетчики.

Видеоурок. Пример анализа загруженности оборудования

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- На что обращать внимание при анализе графиков
- Настройки отображения графиков, упрощающие анализ.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Просмотр дисковой активности

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как понять, какой процесс загружает диски
- Как увидеть объем записываемых/читаемых этим процессом данных.

Видеоурок. Сравнение производительности разных дисков

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Какую утилиту можно использовать для сравнения производительности дисков
- Какие настройки необходимо задать для тестирования дисков.

Видеоурок. Сравнение производительности 1С в разных условиях

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Какой инструмент можно использовать для тестирования скорости работы 1С
- Как оценивать результаты тестирования.

Мониторинг загруженности оборудования для Linux

Рекомендуется осуществлять постоянный мониторинг и запись основных показателей загруженности оборудования во время работы системы. Для операционной системы Linux следует использовать утилиту vmstat.

Запустите команду со следующими параметрами: vmstat -n 15 480 > StatResult.txt
где:

- n – предотвращает повторный вывод заголовка
- 15 – информация будет собираться раз в 15 секунд
- 480 – команда будет выполнена 480 раз
- > StatResult.txt – результат работы утилиты будет сохранен в файл StatResult.txt

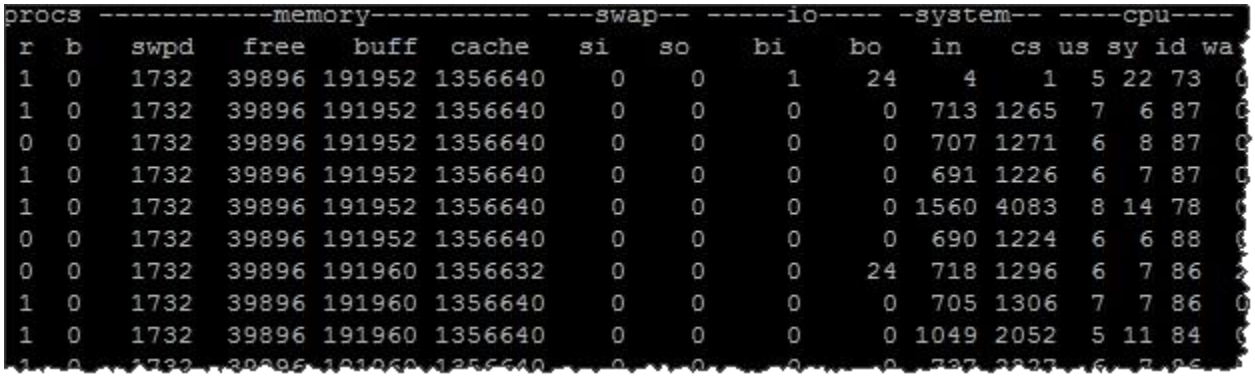
Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

В итоге статистика по загруженности оборудования будет собрана за 2 часа работы системы. Если надо собирать данные за полный рабочий день, вместо 480 укажите соответствующее число.

Результат работы утилиты выглядит следующим образом:



```
procs -----------memory---------- ---swap-- -----io---- -system-- ----cpu----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa
 1  0   1732  39896 191952 1356640    0    0     1    24    4    1  5 22 73  0
 1  0   1732  39896 191952 1356640    0    0     0     0  713 1265  7  6 87  0
 0  0   1732  39896 191952 1356640    0    0     0     0  707 1271  6  8 87  0
 1  0   1732  39896 191952 1356640    0    0     0     0  691 1226  6  7 87  0
 1  0   1732  39896 191952 1356640    0    0     0     0 1560 4083  8 14 78  0
 0  0   1732  39896 191952 1356640    0    0     0     0  690 1224  6  6 88  0
 0  0   1732  39896 191960 1356632    0    0     0    24  718 1296  6  7 86  2
 1  0   1732  39896 191960 1356640    0    0     0     0  705 1306  7  7 86  0
 1  0   1732  39896 191960 1356640    0    0     0     0 1049 2052  5 11 84  0
```

Полученный файл StatResult.txt необходимо проанализировать с помощью любого табличного редактора.

Первую строку после заголовка нужно удалить, т.к. это усредненные статистические данные, собранные с начала работы системы.

Нам нужна следующая информация:

- Значение колонки us
- Значение колонки sy
- Значение колонки r
- Значение колонки free
- Значение колонки b.

Ниже в таблице приведены описания колонок и предельные значения для каждой из них. При превышении этих значений следует рассмотреть вопрос об увеличении производительности соответствующей аппаратной компоненты.

Счетчик	**Описание**	**Критерий нормы**
Значение колонки us	Загрузка процессора пользовательскими процессами	Не более 70
Значение колонки sy	Загрузка процессора системными процессами	Не более 35

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Значение колонки r	Очередь к процессорам	Не более 2 на одно ядро
Значение колонки free	Объем свободной оперативной памяти	Больше нуля
Значение колонки b	Очередь к дискам	Не более 2 на один физический диск, работающий параллельно

Для анализа загруженности сети следует использовать утилиту PktStat.

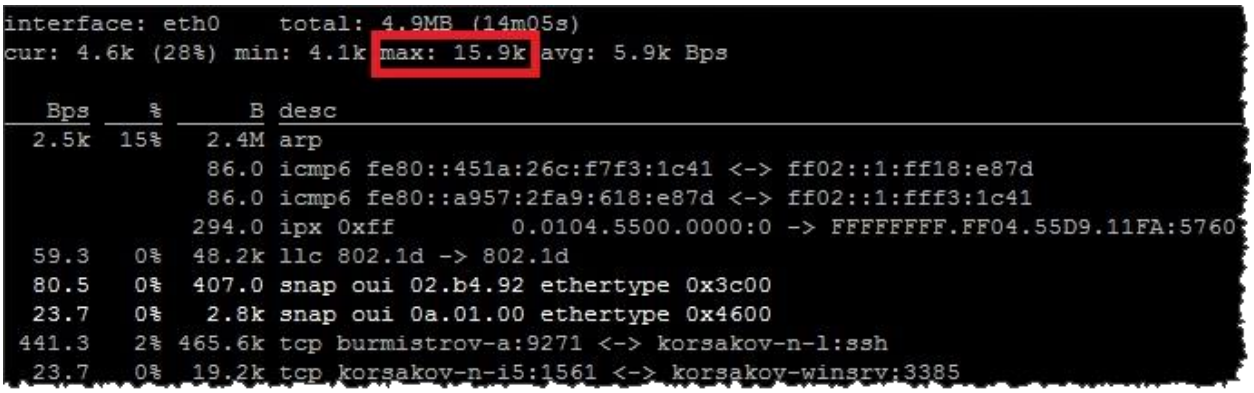
PktStat необходимо запускать только под пользователем root.

Запустите команду со следующими параметрами: Pktstat -T –B –i eth0
где:

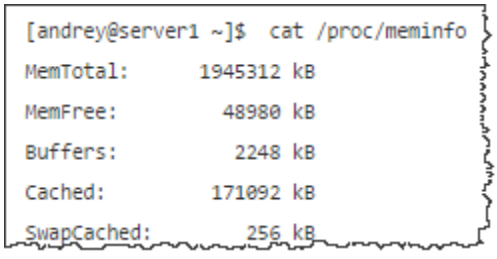
- T – будет отображена общая статистика, включая максимальное и среднее значение
- B – значения будут отражены в байтах
- 480 – команда будет выполнена 480 раз
- i eth0 – будет показана статистика по сетевому интерфейсу eth0.

Нас интересует только один показатель – максимальное количество переданных байт.

Результат работы утилиты выглядит следующим образом:



Для того чтобы оперативно посмотреть состояние памяти, можно использовать команду «cat /proc/meminfo».



Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Наибольший интерес здесь представляет значение *MemFree* – объем свободной оперативной памяти.

Для определения средней нагрузки процессора, можно использовать команду «cat /proc/loadavg».

Результатом работы команды являются 5 чисел, например: 0.20 0.18 0.12 1/80 11206.

Для нас интересны только первые 3 числа, это средняя загрузка процессора за последние 1, 5 и 15 минут. В данном примере процессор за последнюю минуту был загружен в среднем на 20 %, за 5 минут – на 18 %, за 15 минут – на 12 %.

Загрузка отображается для всего процессора с учетом всех ядер. Например, на сервере 1 ядро, тогда нагрузка 1,5 за 15 минут, говорит, что процессор перегружен, но если на сервере 4 ядра, тогда значение 1,5 уже не будет говорить о проблеме, т.к. каждое ядро нагружено на 37,5 % (1,5 / 4 * 100).

Занятие 10

Рекомендации по оборудованию

Включение режима Turbo Boost

По умолчанию процессоры работают на номинальной, а не максимально возможной частоте, это может отрицательно сказаться на производительности высоконагруженных серверов. Для повышения рабочей частоты процессора используется технология **Turbo Boost**.

Turbo Boost – технология компании Intel для автоматического увеличения тактовой частоты процессора свыше номинальной, если при этом не превышаются ограничения на температуру и ток в составе расчетной мощности. Это приводит к увеличению производительности однопоточных и многопоточных приложений. Фактически это технология «саморазгона» процессора. У компании AMD подобная технология называется Turbo CORE.

Для работы данного режима в BIOS должна быть выставлена соответствующая настройка, которая в каждой версии BIOS называется по-своему. Например, настройка может называться следующим образом: Intel Turbo Boost, TurboMode Tech, Turbo CPB и т.д.

Одной настройки в BIOS недостаточно. Необходимо в операционной системе выставить настройку «Схема электропитания» в значение «Максимальная производительность». Эта

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

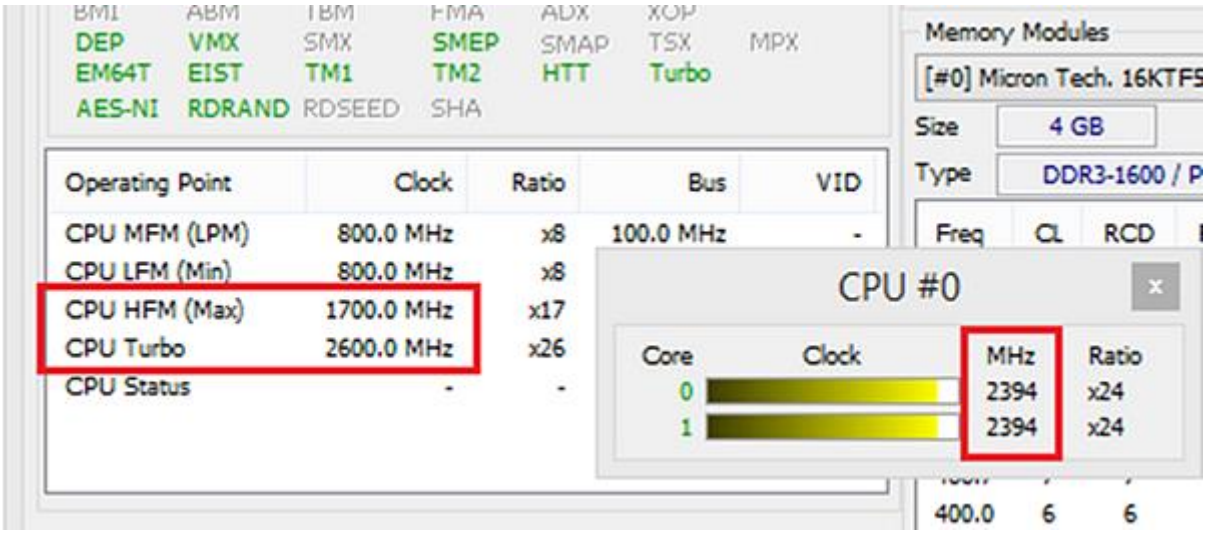
Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

настройка находится в *Панели управления*, пункт *Электропитание*. Неоптимальная настройка данного параметра встречается практически на каждом проекте по оптимизации.

Если выбрана схема электропитания «Сбалансированный» или «Экономичный», то ОС может принудительно занижать рабочую частоту процессора, что отрицательно влияет на производительность.

Чтобы проверить, включен ли режим Turbo Boost, следует использовать утилиту HWiNFO. Утилита полностью бесплатна в использовании, при загрузке скачивайте версию с разрядностью, соответствующей разрядности вашей ОС.

Очень редко, но были зафиксированы случаи, когда запуск данной утилиты приводил к перезагрузке сервера, поэтому лучше перестраховаться и запускать ее в нерабочее время.



Если режим Turbo Boost включен, то это будет видно по следующим признакам:

- Зеленым цветом будет подсвечен индикатор Turbo
- Текущая частота ядер установится между номинальной и максимальной в режиме Turbo
- График текущей частоты ядер процессора зафиксируется в определенном положении, если режим Turbo выключен, график постоянно будет меняться.

Необходимо включить режим Turbo Boost на всех серверах информационной системы, включая терминальный сервер.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Виртуальные машины

При работе с виртуальными машинами необходимо снимать счетчики загруженности оборудования как с физической машины, так и с виртуальной. Только таким образом можно увидеть полную картину по загрузке оборудования.

Вполне возможна ситуация, когда данные счетчиков виртуальной машины будут превышены, в то время как на физической машине оборудование не загружено. В этом случае стоит рассмотреть вопрос выделения виртуальной машине дополнительных ресурсов физической машины.

Возможна и обратная ситуация, когда на виртуальной машине показатели не превышены, но все работает медленно, и при этом превышены счетчики на физической машине. Здесь уже стоит рассмотреть вопрос о переносе виртуальной машины на другое оборудование, снятии с физической машины других нагрузок или об апгрейде физической машины.

В большинстве случаев значения счетчиков коррелируют между собой. В любом случае без данных с той и другой стороны нельзя делать выводы о состоянии оборудования.

Виртуальные машины очень удобны для организации сложных IT-структур, отказоустойчивости, безопасности, распределения нагрузки и т.д. Но у них есть и минусы, например, влияние на производительность. Виртуальные машины – это удобство, а не производительность, к тому же введение дополнительного звена может затруднить расследование сложных случаев.

Если у вас используются виртуальные машины, то должны быть выполнены следующие рекомендации:

- **Подключать диски только как внешние.** Не нужно использовать виртуальные диски, даже если они расположены на SSD дисках. Это может сильно замедлить работу, особенно если на виртуальных дисках расположена база данных. Если СУБД расположена на виртуальной машине, то база данных должна размещаться на физическом носителе. Не стоит забывать, что скорость ввода/вывода зависит не только от дисков, но и от контроллера и канала передачи данных.
- **Выключить создание снимков (Snapshot) для виртуальных машин.** Если вдруг во время работы начнется процесс создания слепка с виртуальной машины, то это крайне негативно скажется на производительности.
- **Выделить фиксированное количество ресурсов.** Нужно сразу жестко задать, сколько памяти и процессорных мощностей может использовать данная виртуальная машина. Не должно быть динамического распределения ресурсов, иначе это не только может негативно сказаться на производительности, но также может стать причиной неработоспособности программной лицензии 1С.
- **Отключить дедупликацию памяти.** Если у вас используется виртуальная машина EXSi, тогда необходимо отключить данный механизм, иначе возможны сложно диагностируемые проблемы с производительностью.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Типичные проблемы с оборудованием

В данном разделе приведен список наиболее частых проблем с оборудованием, встречающихся на проектах по оптимизации.

Многоядерный процессор с низкой частотой.

При использовании такого процессора, процессы хорошо распараллеливаются, но каждый отдельный процесс выполняется медленно. Именно по этой причине необходимо уделять внимание не только количеству ядер, но и частоте.

Например, сервер приложений 1С очень чувствителен к частоте процессора, особенно если используются управляемые формы, т.к. почти вся работа выполняется на сервере приложений. Хотя в 8.3.7 часть нагрузки по отрисовке управляемых форм перенесли на клиент.

Не стоит также впадать и в другую крайность: выбирать процессор с высокой частотой, но с малым количеством ядер.

Бытует мнение, что сервер 1С не умеет распределять нагрузку по ядрам процессора, но это не так. Каждый сеанс на сервере приложений работает на одном ядре, т.е. внутри сеанса распределения нагрузки по ядрам практически нет. Например, если на четырехъядерном процессоре запустить в 1С бесконечный цикл, то процессор будет загружен на 25%, если во второй сессии запустить еще один бесконечный цикл, то процессор будет загружен на 50% и т.д.

Но один рабочий процесс (*rphost*) сервера 1С обслуживает множество сеансов, поэтому сам сервер 1С является многопоточным и может распределять нагрузку по всем ядрам процессора.

В результате, если пожертвовать числом ядер в пользу частоты, то каждый отдельный сеанс будет работать быстро, но при большом числе сеансов может образоваться очередь к процессору. Если же пожертвовать частотой в пользу ядер, то очередь к процессору будет минимальна, но каждый отдельный сеанс будет выполнятся медленнее. Именно по данной причине здесь, как и в любой другой сфере, нужно искать компромисс и избегать крайностей.

На данный момент при выборе процессора для сервера приложений желательно выбирать процессор с частотой минимум 3 ГГц.

Выключен режим Turbo Boost.

Несмотря на то что данная технология появилась достаточно давно, до сих пор практически на каждом проекте приходится включать данный режим, т.к. по умолчанию он либо выключен в BIOS, либо стоит режим энергосбережения.

Один сервер совмещает в себе множество ролей.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Также довольно популярна ситуация, когда на один дорогой и мощный сервер устанавливается все, что только можно установить. Если сервер совмещает в себе роли сервера СУБД, сервера 1С, терминального сервера, почтового сервера, FTP – сервера и т.д., то даже самый мощный сервер может быть перегружен. Например, у процессора будет уходить много времени на контекстные переключения между задачами.

Та же ситуация и с дисковыми массивами: даже самый дорогой и быстрый диск может «проседать», если «навесить» на него слишком большое количество задач. Или другая ситуация: диски справляются с нагрузкой, но контроллер или интерфейс становится узким местом. Все это нужно учитывать при выборе оборудования.

Дорогие серверы – это не бесконечная мощность, у них также имеется свой предел. Поэтому при закупке аппаратного обеспечения стоит сразу учитывать, что рано или поздно этот предел будет достигнут.

В идеале нужно стремится к тому, чтобы СУБД и сервер 1С стояли на отдельных компьютерах с выделенными только для них дисками. Если нагрузка на систему относительно небольшая и/или компьютер имеет большой запас мощности, тогда можно роли СУБД и 1С совместить, но ничего другого на сервере не должно быть установлено. Особенно не рекомендуется устанавливать на сервер антивирус.

Параметры оборудования на различных проектах

Для того чтобы **приблизительно** понимать, какое оборудование нужно под вашу нагрузку, можно использовать данные с проектов ЦКТП http://v8.1c.ru/expert/cts/serv.html

Для окончательного выбора оборудования следует провести нагрузочное тестирование, т.к. очень многое зависит не только от числа пользователей, но и от используемой конфигурации и характера работы. Одна и та же конфигурация может по-разному загружать одно и то же оборудование в разных компаниях, в зависимости от специфики работы пользователей и бизнес-процессов этих организаций. Если у вас от 100 до 200 человек в базе (или менее), то выбор сервера можно произвести и без предварительного нагрузочного тестирования, просто выбрав подходящий по характеристикам сервер, при этом ориентируясь на конфигурации серверов с похожей нагрузкой. Однако, если планируется работа большего числа пользователей, выполнять выбор сервера без предварительного тестирования не рекомендуется.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Мониторинг загруженности оборудования. Итоги.

Состояние серверов нужно отслеживать в обязательном порядке и в регулярном режиме. При необходимости вы должны иметь возможность получить информацию о том, как сильно было нагружено оборудование, например, 2 недели назад в 10:32. При этом не стоит опираться только на слова системного администратора, надо в обязательном порядке смотреть на логи самостоятельно. Если текущей информации недостаточно, добавьте дополнительные счетчики.

После настройки счетчиков настройте автозапуск на случай перезагрузки сервера. В обязательном порядке включите на всех серверах режим Turbo Boost для максимальной производительности процессора.

Серверу 1С и СУБД стоит уделить особое внимание и добавить для них отдельные счетчики производительности по процессам.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Расследование причин медленной работы

Занятие 11

Общие принципы расследования и анализа проблем производительности

Прежде чем перейти непосредственно к процессу расследования проблем производительности, необходимо описать общие принципы, которых необходимо придерживаться.

Принцип № 1. Сначала анализ, потом оптимизация

Для того чтобы решить какую-либо проблему, необходимо понять ее причину. К сожалению, при возникновении проблем с производительностью об этом почти всегда забывают. Вместо анализа причины замедления большинство специалистов стремится сразу угадать решение проблемы, т.е. они действуют «методом тыка».

Перебор разных рецептов обычно выражается в виде изменения различных настроек ОС и СУБД, апгрейде оборудования, переходе на другую СУБД и т.д. Как правило, такие хаотичные действия крайне редко приносят положительный результат, и даже если кому-то в похожей ситуации удалось решить проблему, то нет никаких гарантий, что это же решение поможет другому. Симптомы проблем порой похожи, но могут быть десятки или сотни различных причин, и в каждом случае нужно искать свое решение.

Решение проблем производительности похоже на лечение болезни. Доктор не может выписать лекарства, пока не соберет все анализы и не удостоверится в верности диагноза. Причем иногда на анализ может уйти больше времени, чем на лечение. То же самое справедливо и для проектов по оптимизации.

Зачастую именно выявление причины медленной работы является более трудоемкой операцией, чем устранение этой причины.

Принцип № 2. 20 % действий решают 80 % проблем

Закон Паретто гласит, что 20 % усилий дают 80 % результата, и это справедливо для всех сфер деятельности, в том числе и для проектов по оптимизации.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Не нужно оптимизировать каждую выполняющуюся в системе операцию, необходимо решить только самые серьезные проблемы, которые вносят наибольший вклад в замедление системы. Все инструменты анализа производительности ранжируют проблемы в порядке их вклада в замедление системы. Поэтому всегда в первую очередь необходимо приступать к решению самых «тяжеловесных» проблем.

Часто встречается ситуация, когда решение одной большой проблемы автоматически решает несколько мелких, т.к. они были лишь следствием большой. Не нужно распыляться, концентрируйтесь на серьезных проблемах, множество мелких могут потом исчезнуть автоматически.

Принцип № 3. После решения одной проблемы может возникнуть другая

Будьте готовы к тому, что после решения одной проблемы в системе может появиться другая, которой раньше не было. Например, в результате анализа в системе были обнаружены взаимные блокировки. Специалист провел оптимизацию, но взаимоблокировки остались. Специалист решил, что он что-то сделал неправильно, и начал тратить время и искать другие пути решения. А на самом деле после устранения одной взаимоблокировки стала проявляться другая, условия для возникновения которой появились после устранения первой взаимоблокировки. Такая ситуация встречается часто, при необходимости нужно сделать несколько итераций оптимизации.

Принцип № 4. Все лгут

Следует помнить, что люди зачастую предоставляют недостоверные сведения.

Конечно, в большинстве случаев это делается ненамеренно, однако нельзя исключать и намеренную дезинформацию.

Типичный пример: программист просит пользователя повторно выполнить ключевую операцию и сказать, стала ли она выполняться быстрее. Пользователю лень тратить свое драгоценное время, тем более что он 3 часа назад уже выполнял эту операцию, и она работала медленно, поэтому он, ничего не предпринимая, через несколько минут может сообщить, что все работает медленно, как и раньше. Пользователь не знает, что программист провел оптимизацию за эти 3 часа, а программист будет думать, что оптимизация была неэффективна.

Если надо проверить, как быстро операция выполняется под определенным пользователем, необходимо либо встроить в код замер, либо подключиться удаленно к компьютеру пользователя и замерить время секундомером.

Критически важные вещи, от которых зависит ход дальнейшего расследования, необходимо обязательно проверять самостоятельно. Например, не нужно верить системным администраторам на слово, когда спрашиваете у них о загруженности оборудования, лучше получить у них доступ к логам системного монитора, чтобы в дальнейшей работе опираться на четкие и проверяемые данные.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Принцип № 5. Данные о скорости должны быть объективными

Эффект от оптимизации надо оценивать не по субъективным впечатлениям, а по замерам времени. Для этого необходимо использовать методику APDEX. При наличии замеров времени, не будет никаких споров и сомнений, стало быстрее или нет, ведь APDEX показывает информацию, основанную на объективных замерах. APDEX нужно внедрять до каких-либо работ по оптимизации, чтобы потом было с чем сравнить.

Общий порядок действий таков: выполняем оптимизацию, смотрим APDEX, если лучше не стало, снова оптимизируем, и так до тех пор, пока APDEX не войдет в норму.

Занятие 12

Основные причины медленной работы

Причин медленной работы может быть великое множество, но большая их часть одинакова и повторяется на всех проектах.

В подавляющем большинстве случаев причина проблем кроется в коде, а не в аппаратом обеспечении. Наиболее частые проблемы это:

- Неоптимальные запросы
- Ожидания на избыточных блокировках
- Взаимные блокировки.

Необходимо выяснить, какие из этих проблем наблюдаются в исследуемой базе, именно этому и будет посвящена данная глава.

Возможна ситуация, когда все действия в системе выполняются медленно, в том числе открытие форм, меню, файлов, отрисовка интерфейса и т.д.

Наиболее распространенные причины возникновения данной ситуации это:

- Сильная фрагментация памяти на сервере 1С. Помогает перезапуск службы сервера 1С
- На сервере 1С и/или СУБД установлен антивирус
- Неправильная настройка RAID на сервере СУБД. Например, отключено кэширование или один из дисков вышел из строя
- Оборудование перегружено

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

- Выполняются ресурсоемкие регламентные задания, например, обновление полнотекстового поиска. Следует учитывать, что регламентные задания могут выполняться и в других базах, но на том же оборудовании, где расположена рабочая база. Необходимо отключить все регламентные операции, которые не используются в вашей базе. Например, во многих базах не используется полнотекстовый поиск, но по умолчанию он включен и отнимает ресурсы сервера
- Выполнение ресурсоемких операций на тестовых базах для разработчиков, которые расположены на том же оборудовании, что и рабочая база
- Выполнение резервного копирования, установка обновлений, прочие действия системных администраторов, которые могут сильно загрузить оборудование
- Если используется терминальный сервер, то, возможно, перегружен именно он. В данном случае отрисовка интерфейса может выполняться медленно, при этом операции в базе будут выполняться быстро.

Также в системе могут быть и другие причины медленной работы, хотя и встречаются они гораздо реже. Рассмотрим некоторые из них:

- Замедление только под определенным пользователем 1С. Возможно, проблема в настройке прав пользователя – обычно наблюдается, когда используется RLS
- Замедление только на определенном компьютере. Возможно, проблема в сетевом адаптере или настройках брандмауэра на данном компьютере
- Замедление из-за ошибок платформы. Крайне сложно диагностируется, информацию об ошибках платформы можно посмотреть здесь: www.partners.v8.1c.ru и здесь: https://bugboard.v8.1c.ru/catalog.html

Проблемы производительности и параллельности

Все причины медленной работы можно условно поделить на 2 группы: проблемы производительности и проблемы параллельности. Каждая группа имеет свои отличительные особенности.

Проблемы производительности:

- Легко воспроизводятся в однопользовательском режиме, также обычно воспроизводятся на копии базы. Это правило не всегда строго выполняется, но в большинстве случаев все же проблему можно воспроизвести
- Легко расследуется. Опять же, не всегда, но в большей части случаев причину подобных проблем довольно легко диагностировать
- Частая причина проблем производительности – это неоптимальные запросы
- Инструменты для расследования: замер производительности, SQL Profiler и технологический журнал

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Проблемы параллельности:

- Проявляются только при многопользовательской работе
- Довольно сложно воспроизводятся
- Сложно расследовать без специальных инструментов
- Часто при устранении одной проблемы возникает другая
- Инструменты для расследования: ЦУП, облачные сервисы, SQL Profiler и технологический журнал.

Для расследования необходимо понять, с какой именно группой мы имеем дело. Если пользователи получают ошибки по таймауту или сообщения о взаимоблокировках, тогда все понятно – это проблемы параллельности. Если таких ошибок нет, то это еще не значит, что в системе нет ожиданий на блокировках. Наличие ожиданий можно выяснить только специальными инструментами.

Если проблема проявляется даже в однопользовательском режиме, тогда, скорее всего, это проблема производительности. Для воспроизведения проблемы необходимо найти в регистре замеров APDEX такое выполнение вашей ключевой операции, которое длилось дольше всего. Далее нужно попробовать повторить эту операцию, желательно под тем же пользователем, в тех же условиях, с теми же параметрами, на том же оборудовании и т.д. Если медленное выполнение воспроизвелось, то это проблема производительности, если нет, то это, скорее всего, проблема параллельности.

Расследование проблем производительности

Видеоурок. Включение отладки на сервере

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как включить отладку на сервере через реестр
- Как включить отладку с помощью обработки
- Можно ли оставить отладку постоянно включенной.

Видеоурок. Замер производительности

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- На что обращать внимание при выполнении замера
- Что не фиксируется замером производительности.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Нет лидера в замере производительности

Возможна ситуация, когда в замере производительности нет какого-то одного ярко выраженного узкого места и время выполнения как бы «размазано» по коду, каждая строка кода отнимает по чуть-чуть.

Такие операции оптимизировать гораздо сложнее, при этом можно использовать следующие приемы:

- Разбить одну сложную операцию на несколько более простых
- Создать регистры с промежуточными технологическими данными, которые будут хранить уже обработанные данные
- Часть простых операций выполнять заранее (например, регламентным заданием) и заполнять подготовленные технологические регистры. Из сложной операции исключить выполненные заранее простые операции.

Например, в ключевой операции выполняются сложные расчеты, при этом необязательно заставлять пользователя ждать, пока эти расчеты завершатся. Можно выполнить расчеты в фоновом задании, которое будет запускаться внутри операции, а если что-то пошло не так, уведомлять пользователя о проблеме. Если операция каждый раз что-то вычисляет на основе одних и тех же данных, можно сделать регистр для хранения промежуточных расчетов и заполнять его регламентными заданиями.

Занятие 13

Расследование проблем параллельности

К проблемам параллельности обычно относят избыточные ожидания на блокировках и взаимные блокировки. Чем больше пользователей одновременно работает с базой, тем больше вероятность возникновения проблем параллельности.

Проблемы параллельности диагностируются и расследуются гораздо сложнее, чем проблемы производительности, и без специальных инструментов решать такие задачи крайне сложно.

В качестве основных инструментов в рамках данного курса рассматриваются две разработки:

- ЦУП. Инструмент от 1С, является платным и входит в состав КИП
- Облачные сервисы Гилева. Являются бесплатной альтернативой ЦУП.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

В данном разделе эти инструменты будут рассматриваться с позиции выявления симптомов проблем. В другой главе эти же инструменты будут рассматриваться с точки зрения анализа выявленных проблем.

ЦУП

Общая информация о ЦУП

Центр управления производительностью (ЦУП) – это инструмент мониторинга и анализа производительности клиент-серверных информационных систем на платформе 1С:Предприятие 8. ЦУП предназначен для оценки производительности системы, сбора подробной технической информации об имеющихся «узких местах» и анализа этой информации с целью дальнейшей оптимизации.

С помощью ЦУП можно выявлять и решать следующие проблемы:

- Неоптимальные запросы
- Излишние блокировки
- Взаимоблокировки на уровне сервера СУБД (только для MS SQL Server).

ЦУП не позволяет решать:

- Проблемы стабильности работы
- Проблемы на уровне кода (например, обработка гигантской таблицы значений)
- Взаимоблокировки на уровне сервера 1С.

Следует учитывать, что полноценно ЦУП может работать только с MS SQL Server, с другими СУБД нормальная работа возможна только при сборе оперативных данных, с учетом некоторых ограничений.

ЦУП – это обычная конфигурация 1С, но использовать ее рекомендуется только в клиент-серверном варианте. Можно, конечно, поставить и файловый вариант, но работа в этом случае будет нестабильная и медленная.

Сам по себе ЦУП не решает никаких проблем, это просто инструмент, которым необходимо уметь пользоваться и правильно интерпретировать его данные. Без необходимых знаний и навыков ЦУП будет бесполезен.

Если у службы безопасности возникнут вопросы относительно использования ЦУП, то можно убедиться, что нет никаких поводов для беспокойства. Код конфигурации полностью открыт.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Для своей работы ЦУП собирает исключительно служебную информацию: логи платформы, трассировки MS SQL Server, данные системного монитора и информацию из кластера серверов 1С.

ЦУП не обращается к содержимому таблиц информационной базы, не собирает данные о сетевом окружении, используемом оборудовании и т.д.

Как и любая другая конфигурация, ЦУП привязан к версии платформы, это связано с тем, что формат технологического журнала, который использует ЦУП, может меняться в зависимости от релиза. Перед началом использования обязательно посмотрите, с каким релизом совместима ваша версия ЦУП.

Подготовка к настройке ЦУП

Если было принято решение установить ЦУП, то необходимо запастись временем и терпением, т.к. настройка ЦУП достаточно сложна и является нетривиальной задачей.

Для того чтобы минимизировать возможные ошибки при настройке, необходимо выполнить несколько подготовительных действий, которые впоследствии помогут сэкономить много времени и нервных клеток.

Синхронизация времени

Необходимо в обязательном порядке синхронизировать время на сервере СУБД и сервере 1С.

Время играет важную роль при анализе данных, т.к. ЦУП сопоставляет данные логов 1С с данными трассировок СУБД, в том числе и по времени, и если оно будет различаться, то анализ может быть не выполнен или выполнен некорректно.

Свободное место на диске

При сборе аналитических показателей ЦУП получает довольно подробные логи сервера приложений 1С. Размер логов зависит от настроек ЦУП, количества проблем в базе и нагрузки на систему. В некоторых случаях рост логов может достигать 1ГБ в минуту, в связи с чем необходимо заранее позаботиться о свободном месте на диске, который выбран для хранения файлов технологического журнала. В качестве места хранения логов ни в коем случае не указывайте системный диск, потому что при его заполнении весь сервер придет в неработоспособное состояние.

Лицензии

При своей работе ЦУП использует фоновые задания и COM-соединения. Все это необходимо учитывать, если у вас ограниченное число лицензий.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Ошибки

К сожалению, даже если все настроено верно, в коде ЦУП довольно много ошибок, которые мешают его нормальной работе. Наиболее популярные ошибки и способы их решения описаны в приложенном файле «Ошибки в ЦУП и способы решения». Ошибки могут проявиться как на этапе настройки, так и на этапе анализа.

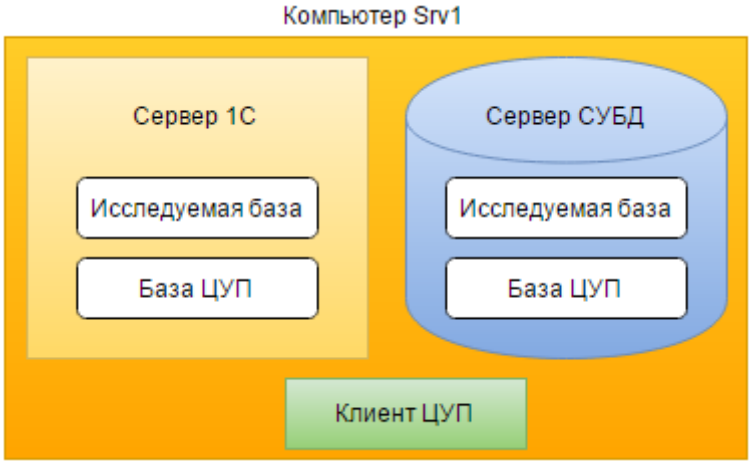
Месторасположение базы ЦУП

Есть несколько вариантов месторасположения базы ЦУП, рассмотрим наиболее распространенные.

Вариант 1. Серверы СУБД и 1С исследуемой базы расположены на одном компьютере

Это самый простой вариант, в этом случае базу ЦУП рекомендуется расположить на этом же сервере. Здесь также возможны варианты в зависимости от того, где будет запускаться клиент ЦУП.

Удобнее всего клиентское приложение ЦУП запускать также на этом сервере. Данный вариант наиболее прост в настройке и минимизирует возможные ошибки.

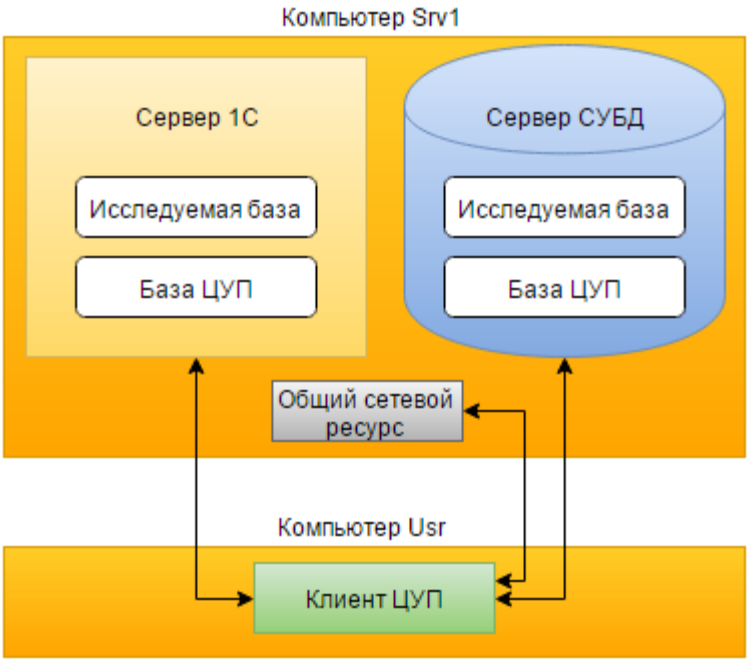


Если клиент ЦУП расположен на другом компьютере, то пользователь ОС, под которым запущен клиент ЦУП, должен иметь доступ по сети к общим каталогам сервера, чтобы считывать логи и трассировки.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



Если сервер входит в домен, а компьютер с клиентом ЦУП - нет, тогда клиент ЦУП не сможет получить доступ к «расшаренным» каталогам сервера, что является необходимым условием для нормальной работы ЦУП. Для решения этой проблемы можно включить клиентский компьютер в домен и заходить на него под доменным пользователем, но такое решение доступно не всегда.

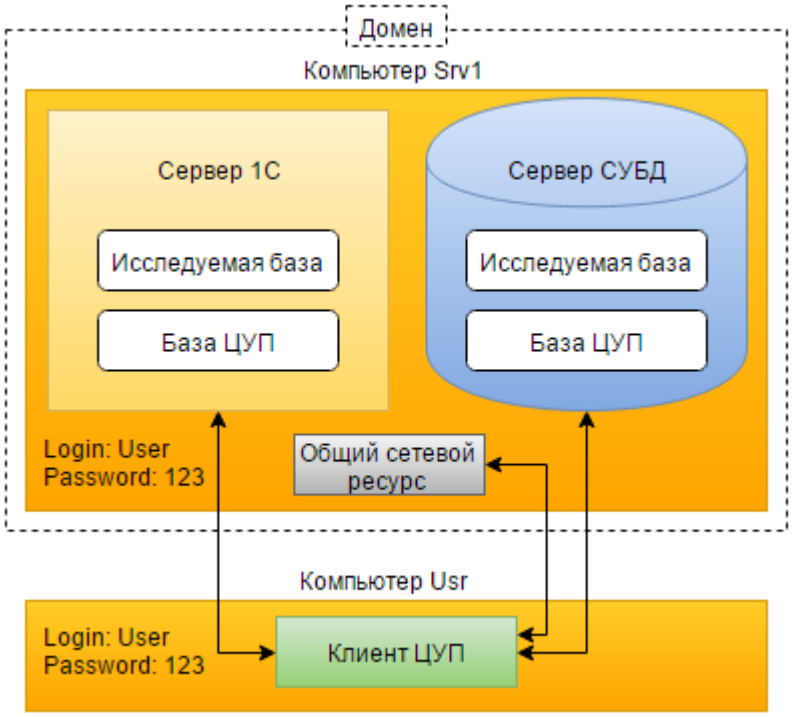
В качестве альтернативы можно создать на сервере локального пользователя, логин и пароль которого будут совпадать с логином и паролем локального пользователя ОС на клиентском компьютере, под которым запущен клиент ЦУП. На сервере для созданного локального пользователя нужно разрешить сетевой доступ к общему ресурсу.

Обратите внимание: не нужно ничего запускать на сервере под локальным пользователем, нужно просто дать этому пользователю права на чтение сетевого каталога.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

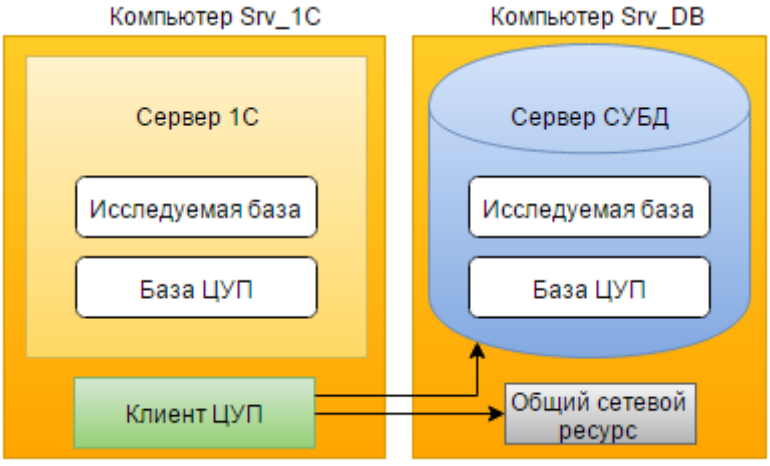
Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



Подробно о правах доступа при настройке ЦУП будет рассказано в отдельном уроке.

Вариант 2. Серверы СУБД и 1С исследуемой базы расположены на разных компьютерах

В данном случае базу ЦУП рекомендуется расположить на том же сервере, где находится исследуемая база. Клиента ЦУП желательно запускать на компьютере, где находится сервер 1С, тогда настройка несколько упрощается.



Здесь также возможен вариант, когда клиент ЦУП располагается на отдельном компьютере.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Все, что было сказано насчет домена в первом варианте, относится также и ко второму варианту, но возможен и более экзотический случай, когда один из серверов в домене, а другой нет. Например, сервер СУБД находится в домене, а сервер 1С – нет. В таком случае рекомендуется включить серверы в один домен и запускать службу сервера 1С под доменным пользователем.

Возможны и другие варианты расположения, например, базу ЦУП можно расположить на сервере 1С, который отличается от сервера 1С исследуемой базы. Такой вариант гораздо сложнее в настройке, что увеличивает вероятность возникновения ошибок. Именно поэтому данный вариант в курсе не рассматривается.

Рекомендуется устанавливать базу ЦУП на тот же сервер 1С, на котором установлена исследуемая база.

Минус описанных вариантов в том, что при анализе собранных данных ЦУП будет нагружать оборудование, на котором расположена исследуемая база. Степень нагрузки зависит от количества собранных данных, но обычно нагрузка не настолько критична, чтобы мешать работе пользователей. В крайнем случае анализ собранных данных можно запустить в ночное время, когда в исследуемой базе никто не работает.

Занятие 14

Видеоурок. Настройка прав для каталогов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Где и какие каталоги необходимо создать для настройки ЦУП
- Кому и какие права на каталоги необходимо установить.

Видеоурок. Настройка разрешений для СУБД

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Какие разрешения необходимо установить на сервере СУБД для нормальной работы ЦУП.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Мастер настройки ЦУП

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что такое мастер настройки и для чего он нужен
- Особенности использования мастера настройки
- Как запустить процесс настройки ЦУП
- Первые шаги мастера настройки.

Видеоурок. Шаг «Центральный сервер»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Имя какого сервера надо указывать на данном шаге
- Какой порт указывать при настройке.

Видеоурок. Шаг «COM-Соединитель»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Зачем нужен COM-Соединитель
- Как зарегистрировать COM-компоненту.

Видеоурок. Шаг «Кластер»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Какой кластер необходимо указывать на данном этапе
- Каким образом пройти аутентификацию в кластере.

Видеоурок. Шаг «Информационная база»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- Какими правами должен обладать пользователь, под которым ЦУП обращается к исследуемой базе.

Видеоурок. Шаг «Типы показателей»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Какие типы показателей бывают
- Какие показатели указывать.

Видеоурок. Шаг «Показатели 1С:Предприятия»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что означает флаг «processadmin».

Видеоурок. Шаг «Показатели ОС»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Возможные ошибки на данном шаге и способы их исправления.

Видеоурок. Шаг «Технологический журнал»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что такое каталог настроек технологического журнала (ТЖ)
- Что такое сетевой и локальный каталог сбора данных ТЖ
- Типичные ошибки при настройке ТЖ.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Шаг «Трассировки»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Какой каталог необходимо указать
- Относительно какого компьютера указывается путь.

Видеоурок. Шаг «Сервер COM-Соединитель»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Основные ошибки при прохождении данного шага.

Видеоурок. Шаг «Сервер ЦУП (Трассировки)»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Какие ошибки возможны на данном этапе
- Можно ли пропустить данный шаг.

Видеоурок. Сценарий «Мониторинг»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что такое сценарий работы ЦУП
- Зачем нужен сценарий «Мониторинг»
- Как включить запись показателей
- Как использовать закладки.

Видеоурок. Сценарий «Просмотр»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- Назначение данного сценария
- Изменение масштаба просмотра и использование закладок.

Видеоурок. Создание собственных сценариев

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как устроены сценарии работы
- Как создать собственный сценарий.

Видеоурок. Сценарий «Регламентный мониторинг»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение и принцип работы регламентного мониторинга
- Состав сценария регламентного мониторинга.

Видеоурок. Оперативные показатели

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Состав и назначение оперативных показателей
- Источники данных для оперативных показателей.

Видеоурок. Аналитические показатели

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Состав и назначение аналитических показателей.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Сбор оперативных показателей

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Какие показатели необходимо собирать
- Как долго собирать показатели.

Видеоурок. Симптомы неоптимальных запросов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как анализировать полученные показатели
- Как на графиках ЦУП выглядят медленные запросы.

Видеоурок. Симптомы ожиданий на блокировках

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как на графиках ЦУП выглядят ожидания на блокировках.

Видеоурок. Симптомы взаимоблокировок

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как с помощью ЦУП можно понять, что в системе есть взаимные блокировки.

Видеоурок. Проверка работоспособности ЦУП

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как убедиться в том, что ЦУП настроен верно.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Формирование логов при отключенном ЦУП

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что делать в том случае, если ЦУП отключен, но его логи продолжают собираться.

Занятие 15

Облачная система контроля производительности gilev.ru

Общие сведения о системе

Облачная система контроля производительности предназначена для сбора данных и анализа проблем производительности.

Эта система состоит из отдельных сервисов, каждый из которых решает только свою узкоспециализированную задачу.

В курсе рассматриваются следующие сервисы:

- Сервис анализа неоптимальных запросов
- Сервис анализа ожиданий на блокировках
- Сервис анализа взаимных блокировок
- Сервис анализа событий технологического журнала (чтобы определить наличие взаимоблокировок).

Задачи и у сервисов, и у ЦУП одинаковые, но схема и принципы работы различаются.

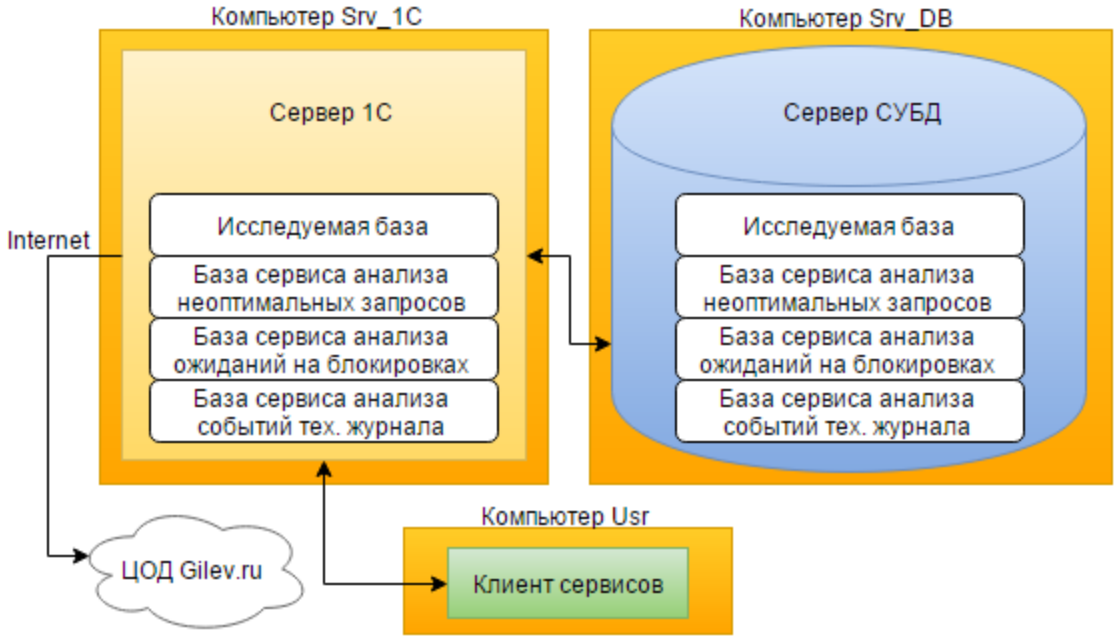
Каждый сервис состоит из двух частей, причем одна часть расположена на клиенте, а вторая в «облаке».

Желательно создавать базы сервисов на том же сервере 1С, на котором располагается исследуемая база.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



Для корректной работы сервисов время на сервере 1С и на сервере СУБД должно быть синхронизировано, иначе нельзя будет сопоставить логи платформы с трассировками MS SQL Server.

Клиентская часть сервиса является обычной конфигурацией и устанавливается в отдельно созданную базу. База сервиса обязательно должна быть клиент-серверной.

В отличие от ЦУП, абсолютно неважно, где находится клиент для подключения к базе сервиса, которая расположена на сервере 1С. Клиентская часть обращается только к серверу 1С, который и выполняет всю работу по настройке и чтению логов.

В облачной системе контроля производительности для каждой проблемы предназначен свой сервис, и для каждого сервиса нужно создавать свою информационную базу. Эти базы не хранят собранные данные, они просто обрабатывают их и отправляют в облачную часть, поэтому сами базы сервисов занимают очень мало места.

В целях минимизации нагрузки на систему каждый сервис собирает минимально возможное, но достаточное для анализа количество данных. При этом сервисы спроектированы таким образом, чтобы количество действий при их настройке было минимальным.

Использование сервисов, описанных в курсе, бесплатно, в них есть отдельные платные функции, но они необязательны.

Сервисы изначально проектировались для работы в режиме 24/7 в высоконагруженных системах, поэтому при сборе данных они оказывают гораздо меньшую нагрузку на систему, чем ЦУП, а при анализе данных не оказывают нагрузку вовсе, т.к. анализ происходит в «облаке».

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Сервис анализа неоптимальных запросов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

В этом уроке рассматриваются:

- Описание сервиса
- Установка и настройка сервиса
- Расширенные настройки.

Видеоурок. Сбор и анализ данных с помощью сервиса

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как часто сервис отправляет данные
- Как посмотреть данные сервиса.

Видеоурок. Сервис анализа ожиданий на блокировках

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Описание сервиса
- Установка и настройка сервиса.

Видеоурок. Сбор и анализ данных сервиса блокировок

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как проверить работоспособность сервиса
- Как часто данные выгружаются в облачную часть сервиса
- Как посмотреть длительность ожидания на определенной таблице или строке кода.

Видеоурок. Сервис анализа событий технологического журнала

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- Описание сервиса
- Установка и настройка сервиса.

Видеоурок. Подключение сервисов через тонкий клиент

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Преимущества работы сервиса через тонкий клиент.

Видеоурок. Ошибка SHOWPLAN permission denied

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Описание и причины возникновения ошибки
- Способы решения данной проблемы.

Расследование причин медленной работы. Итоги

Наиболее частые проблемы производительности – это неоптимальные запросы, избыточные блокировки и взаимные блокировки. В первую очередь необходимо понять, какие из этих проблем наблюдаются в исследуемой базе. Перед тем как что-то оптимизировать, необходимо выяснить, с какими именно проблемами мы имеем дело. Никогда не следует пытаться угадать решение, в первую очередь необходимо провести тщательное расследование, которое, конечно, займет какое-то время, но сделает дальнейшую работу гораздо более эффективной.

Для расследования и анализа проблем можно использовать различные инструменты, например: замер производительности из конфигуратора, ЦУП, облачную систему контроля производительности и прочие инструменты, которые будут разобраны далее в курсе.

Основными инструментами являются ЦУП и облачная система контроля производительности. Они оба рассмотрены в курсе, и каждый из них имеет свои сильные и слабые стороны.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

	ЦУП	Сервисы
Плюсы	• Показывает данные онлайн • Анализирует все 3 основные проблемы производительности в одной базе • Не требует подключения к интернету	• Бесплатны • Просты в настройке • Оказывают минимальную нагрузку на систему • Доступ к данным через интернет • Удобный интерфейс • Анализ взаимоблокировок для MS SQL 2014
Минусы	• Платный • Оказывает сильную нагрузку на систему • Большое число ошибок • Сложен в настройке • Не анализирует взаимоблокировки MS SQL 2014 (по состоянию на начало 2016 года)	• Необходимо передавать данные по интернету • Для каждой проблемы свой сервис • Нет онлайн мониторинга

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Неоптимальные запросы

Занятие 16

Неоптимальные запросы – это наиболее частая причина проблем производительности. Кроме того, неоптимальные запросы очень часто влекут за собой другие причины проблем, такие как избыточные блокировки и взаимные блокировки. Также по вине медленных запросов возможна чрезмерная загруженность оборудования, в частности, дисков. Перед тем как принимать решение о покупке более мощной дисковой подсистемы, стоит проверить, можно ли уменьшить нагрузку на диски путем исправления медленных запросов. Именно из-за того, что неоптимальные запросы часто являются первопричиной других проблем, их необходимо оптимизировать в первую очередь. В этом случае велика вероятность, что после этого часть других проблем исчезнет.

Для того чтобы писать оптимальные запросы и исправлять неоптимальные, необходимо знать, как устроены объекты метаданных на уровне СУБД.

Общие сведения о таблицах и индексах

Таблицы объектов метаданных

Каждый объект метаданных представляет собой одну или несколько таблиц на уровне СУБД.

В самом простом случае один объект метаданных соответствует одной таблице в СУБД. Например, документ без табличной части хранится как одна таблица, но если создать одну табличную часть, то в СУБД добавится еще одна таблица, и в результате этому документу будут соответствовать две таблицы СУБД.

Для того чтобы понять, из каких таблиц СУБД состоит объект метаданных, следует использовать функцию *ПолучитьСтруктуруХраненияБазыДанных(, Истина)*. Второй параметр обязательно должен быть в значении *Истина*, иначе будут отражены не все данные.

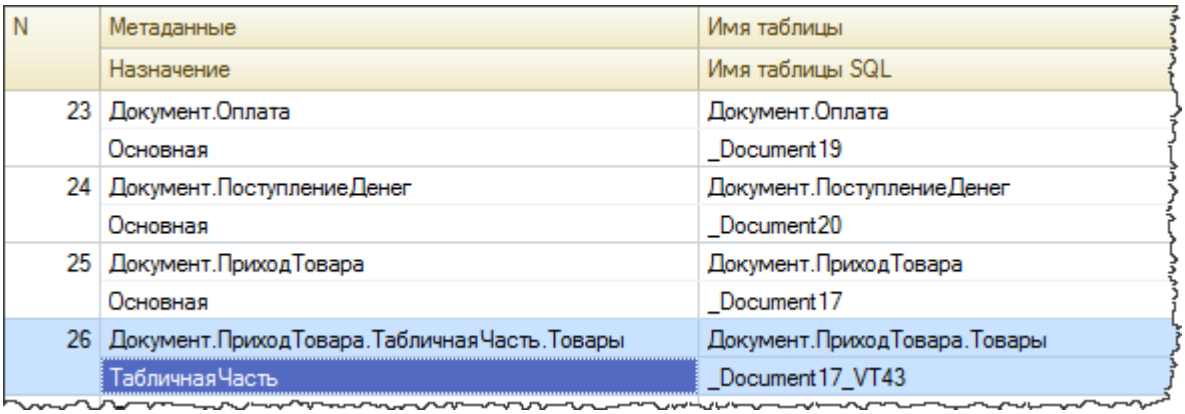
В курсе для этой цели используется обработка «Структура хранения метаданных», в которой с помощью указанной выше функции в удобном виде отображается информация о структуре хранения объектов конфигурации.

Ниже приведен фрагмент результата, полученного с помощью этой обработки.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



N	Метаданные	Имя таблицы
	Назначение	Имя таблицы SQL
23	Документ.Оплата	Документ.Оплата
	Основная	_Document19
24	Документ.ПоступлениеДенег	Документ.ПоступлениеДенег
	Основная	_Document20
25	Документ.ПриходТовара	Документ.ПриходТовара
	Основная	_Document17
26	Документ.ПриходТовара.ТабличнаяЧасть.Товары	Документ.ПриходТовара.Товары
	ТабличнаяЧасть	_Document17_VT43

Из таблицы видно, что документы без табличной части состоят из одной таблицы с названием _Document<N>, где N – произвольное число. Документы с одной табличной частью состоят из двух таблиц, при этом имя таблицы СУБД, в которой содержится табличная часть документа, состоит из имени основной таблицы СУБД и символов _VT<M>, где M – произвольное число.

В процессе оптимизации постоянно приходится работать с текстом запроса на языке SQL и, зная основные правила именования таблиц, можно понять, к какому именно объекту 1С относится таблица СУБД.

Рассмотрим основные типы объектов метаданных и соответствующие им имена таблиц СУБД.

Объект метаданных	Имя таблицы	Описание
Константа	_Const<N>	Начиная с версии 8.2.14 каждая константа хранится в отдельной таблице. До 8.2.14 все константы хранились в одной таблице _Consts
Справочник	_Reference<N>	Основная таблица справочника
	_Reference<N>_VT<M>	Табличная часть справочника
Документ	_Document<N>	Основная таблица документа
	_Document<N>_VT<M>	Табличная часть документа
Регистр сведений	_InfoRg<N>	Основная таблица регистра сведений
	_InfoRgSF<N>	Таблица итогов среза первых. Используется только для периодического регистра сведений в версии платформы 8.3, при установленном флаге «Разрешить итоги: Срез первых»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

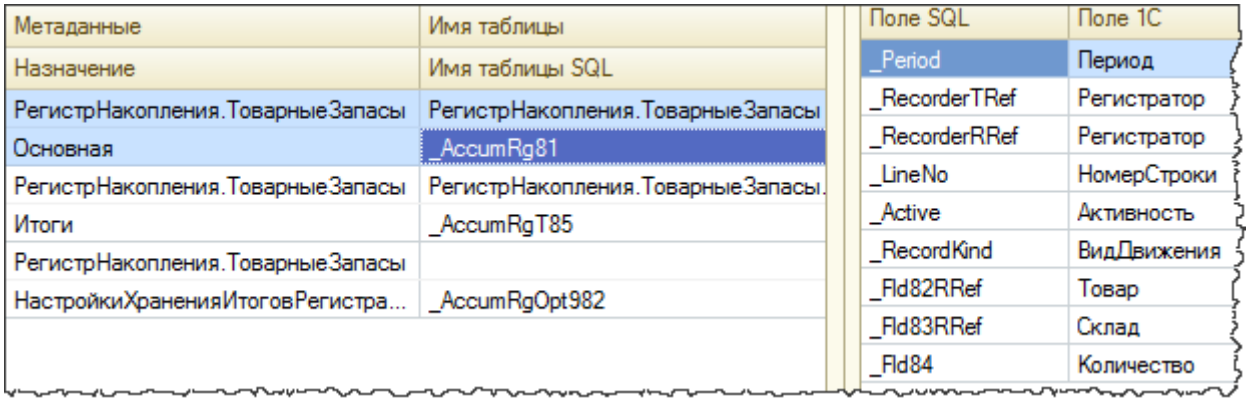
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

	_InfoRgSL<N>	Таблица итогов среза последних. Используется только для периодического регистра сведений в версии платформы 8.3, при установленном флаге «Разрешить итоги: Срез последних»
Регистр накопления	_AccumRg<N>	Таблица движений
	_AccumRgT<N>	Таблица итогов. Только для остаточных регистров
	_AccumRgTn<N>	Таблица оборотов. Только для оборотных регистров
Регистр бухгалтерии	_AccRg<N>	Таблица движений
	_AccRgAT0<N>	Таблица итогов по счету
Регистр расчета	_CRg<N>	Таблица движений
	_CRgActP<N>	Таблица фактических периодов действия. Только если у регистра установлен флаг «Период действия»

Здесь приведены лишь основные таблицы, полный список можно посмотреть на диске ИТС в разделе «Размещение данных 1С:Предприятие 8».

С помощью обработки можно посмотреть, какие реквизиты каким полям таблицы СУБД соответствуют. Если реквизит простого или ссылочного типа, то ему соответствует одно поле в таблице. Если реквизит составного типа, то ему соответствует несколько полей.



На рисунке представлены поля таблицы и соответствующие им реквизиты. Реквизит «Регистратор» является составным, поэтому для него отражается 2 поля, в одном из которых содержится тип документа регистратора, а в другом – GUID (значение ссылки) документа регистратора.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Используя данную информацию и имея только запрос на языке SQL, можно понять, к каким реквизитам и объектам метаданных идет обращение.

Основные сведения о временных таблицах

Временные таблицы – это объекты СУБД, никаких временных таблиц на сервере 1С нет, и не стоит их путать с таблицами значений. Вызывает вопрос лишь конкретное, физическое расположение временных таблиц в СУБД: либо на жестком диске, либо в оперативной памяти. Попробуем в этом разобраться. Все временные таблицы относятся к базе данных TempDB, что вовсе не значит, что они обязательно будут записываться на диск.

Правильный ответ на этот вопрос звучит так: все временные таблицы по умолчанию создаются в оперативной памяти, а именно – в буферном кэше. Конечно, есть исключения. Например, если таблица слишком большая, то сервер может принять решение сбросить ее на диск. Также возможна ситуация, когда сервер по каким-либо причинам решил отдать память под другие данные, в этом случае таблица тоже будет записана на диск.

Почему временная таблица создается именно в памяти? Все очевидно – дело в производительности. Думаю, не стоит объяснять, что чтение из оперативной памяти гораздо быстрее чтения с диска, даже если этот диск SSD.

Проведем эксперимент и проверим, где же создается временная таблица.

Проведем эксперимент и проверим, где создается временная таблица. Напишем следующий запрос в консоли:

```
ВЫБРАТЬ 1 КАК Поле1 ПОМЕСТИТЬ ВТ
```

Запускаем трассировку SQL Profiler с событием SQL:BatchComplited, выполняем запрос в консоли и получаем следующий текст SQL запроса:

```
INSERT INTO #tt1 (_Q_001_F_000) SELECT 1.0
```

Здесь мы видим только заполнение временной таблицы, т.к. код создания временных таблиц в нашей трассировке не отображается.

Чтобы узнать, где создается временная таблица, необходимо понять, откуда читаются данные, с диска или из памяти. Для этого используем показатель physical reads (количество физических чтений), т.е. сколько страниц данных было прочитано с диска для выполнения запроса. Чтобы получить значение этого показателя, необходимо выполнить создание и чтение временной таблицы в Management Studio.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Создаем новый запрос и пишем следующее:

```
create table #tt1 (_Q_001_F_000 int); -- создаем локальную временную таблицу tt1
INSERT INTO #tt1 (_Q_001_F_000) SELECT 1.0 -- заполняем таблицу
set statistics io on; -- включаем вывод статистики ввода/вывода
select * from #tt1 -- читаем данные из таблицы
set statistics io off; -- выключаем вывод статистики
drop table #tt1 -- удаляем таблицу
```

После выполнения данного кода на закладке «Сообщения» получим следующий текст:

```
(строк обработано: 1) Таблица
«#tt1_____000000000066".
Число просмотров 1, логических чтений 1, физических чтений 0, упреждающих чтений 0,
lob логических чтений 0, lob физических чтений 0, lob упреждающих чтений 0.
```

Самое важное здесь то, что данные с диска не читались, т.к. число физических чтений равно 0, при этом есть 1 логическое чтение, т.е. данные были прочитаны только из памяти. Отсюда можно сделать вывод, что временные таблицы в большинстве случаев создаются и хранятся в оперативной памяти, исключения уже описаны выше.

Индексация временных таблиц

На дисках ИТС и на тренинге 1С:Эксперт говорится о том, что нужно индексировать поля условий и соединений во временных таблицах. Эта рекомендация настолько очевидна, что уже практически никто не подвергает ее сомнению. Но, как показывает практика, чаще всего индексы во временных таблицах не используются либо используются, но запрос выполняется еще медленнее, чем без них. Даже больше: польза от индексов на временных таблицах – скорее исключение, чем правило.

Это не значит, что во временных таблицах индексы не нужно использовать, это значит, что необходимо анализировать каждый конкретный случай, и случаев, когда индекс не нужен, значительно больше. Именно поэтому на курсах по оптимизации постоянно делается акцент на то, что с каждой проблемой надо разбираться отдельно, а не слепо выполнять рекомендации, не понимая, почему эти рекомендации именно такие.

Давайте рассмотрим ситуацию с индексацией на примере. Создадим временную таблицу с одним числовым полем и значениями от 1 до 1 млн. Это можно сделать с помощью следующего пакетного запроса:

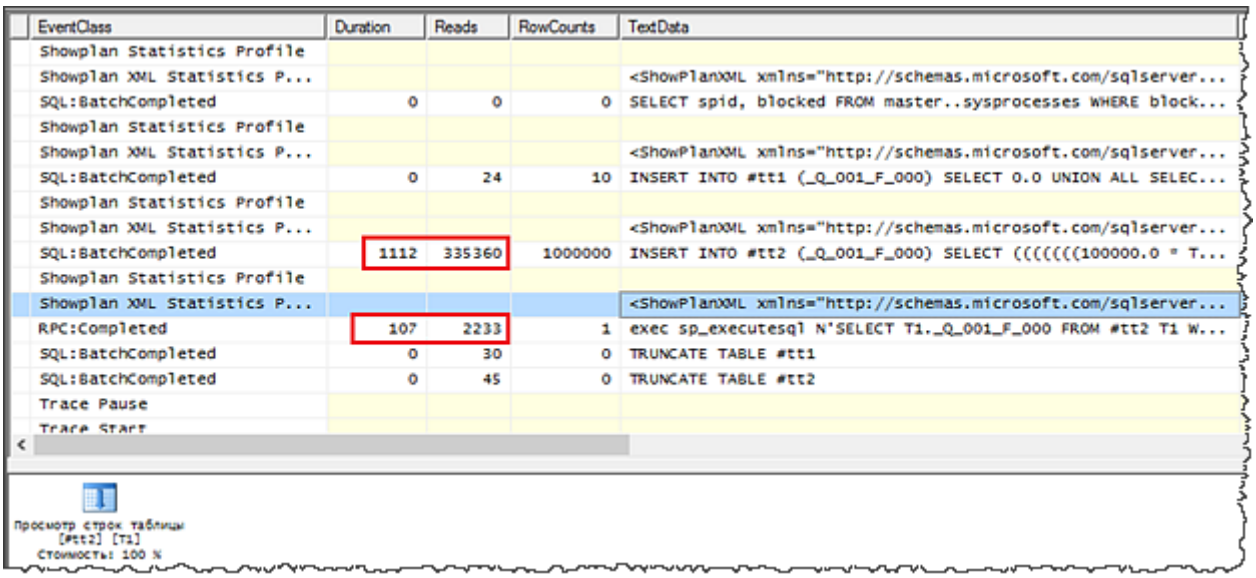
Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

```
ВЫБРАТЬ 0 КАК Цифра ПОМЕСТИТЬ ВТ_Цифры ОБЪЕДИНИТЬ ВСЕ ВЫБРАТЬ 1 ОБЪЕДИНИТЬ ВСЕ
ВЫБРАТЬ 2 ОБЪЕДИНИТЬ ВСЕ ВЫБРАТЬ 3 ОБЪЕДИНИТЬ ВСЕ ВЫБРАТЬ 4 ОБЪЕДИНИТЬ ВСЕ ВЫБРАТЬ 5
ОБЪЕДИНИТЬ ВСЕ ВЫБРАТЬ 6 ОБЪЕДИНИТЬ ВСЕ ВЫБРАТЬ 7 ОБЪЕДИНИТЬ ВСЕ ВЫБРАТЬ 8
ОБЪЕДИНИТЬ ВСЕ ВЫБРАТЬ 9
;
////////////////////////////////////////////////////////////////////////////
ВЫБРАТЬ 100000 * Таб6.Цифра + 10000 * Таб5.Цифра + 1000 * Таб4.Цифра + 100 *
Таб3.Цифра + 10 * Таб2.Цифра + Таб1.Цифра + 1 КАК Число ПОМЕСТИТЬ ВТ_Числа ИЗ
ВТ_Цифры КАК Таб1, ВТ_Цифры КАК Таб2, ВТ_Цифры КАК Таб3, ВТ_Цифры КАК Таб4, ВТ_Цифры
КАК Таб5, ВТ_Цифры КАК Таб6
;
////////////////////////////////////////////////////////////////////////////
ВЫБРАТЬ ВТ_Числа.Число ИЗ ВТ_Числа КАК ВТ_Числа ГДЕ ВТ_Числа.Число = 777
```

Весь запрос выполняется в среднем за 1.2 секунды. Если посмотреть трассировку SQL Profiler, то мы увидим следующее:



EventClass	Duration	Reads	RowCounts	TextData
Showplan Statistics Profile				
Showplan XML Statistics P...				<ShowPlanXML xmlns="http://schemas.microsoft.com/sqlserver...
SQL:BatchCompleted	0	0	0	SELECT spid, blocked FROM master..sysprocesses WHERE block...
Showplan Statistics Profile				
Showplan XML Statistics P...				<ShowPlanXML xmlns="http://schemas.microsoft.com/sqlserver...
SQL:BatchCompleted	0	24	10	INSERT INTO #tt1 (_Q_001_F_000) SELECT 0.0 UNION ALL SELEC...
Showplan Statistics Profile				
Showplan XML Statistics P...				<ShowPlanXML xmlns="http://schemas.microsoft.com/sqlserver...
SQL:BatchCompleted	1112	335360	1000000	INSERT INTO #tt2 (_Q_001_F_000) SELECT (((((((100000.0 * T...
Showplan Statistics Profile				
Showplan XML Statistics P...				<ShowPlanXML xmlns="http://schemas.microsoft.com/sqlserver...
RPC:Completed	107	2233	1	exec sp_executesql N'SELECT T1._Q_001_F_000 FROM #tt2 T1 W...
SQL:BatchCompleted	0	30	0	TRUNCATE TABLE #tt1
SQL:BatchCompleted	0	45	0	TRUNCATE TABLE #tt2
Trace Pause				
Trace Start				

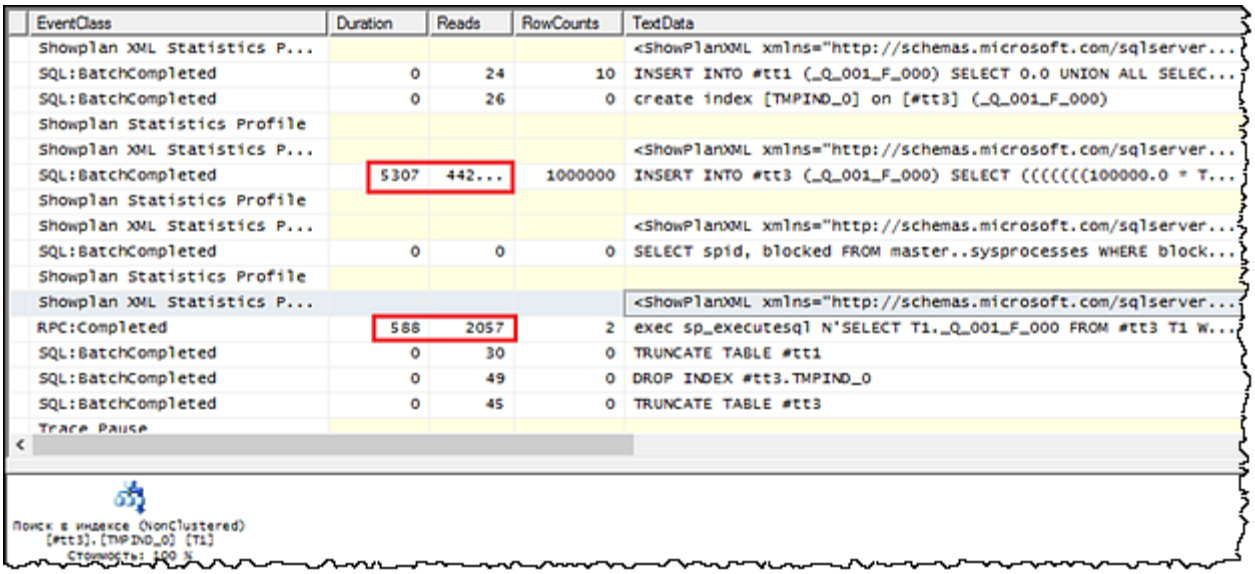
Просмотр строк таблицы
[#tt2] [T1]
Стоимость: 100 %

На создание таблицы уходит 1.1 секунды и еще 0.1 секунды на сканирование всей таблицы, чтобы вернуть нам 1 строку. Давайте посмотрим, что изменится, если добавить индекс в таблицу ВТ_Числа. В нашем примере запрос стал выполняться в среднем за 6 секунд.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



EventClass	Duration	Reads	RowCounts	TextData
Showplan XML Statistics P...				<ShowPlanXML xmlns="http://schemas.microsoft.com/sqlserver...
SQL:BatchCompleted	0	24	10	INSERT INTO #tt1 (_Q_001_F_000) SELECT 0.0 UNION ALL SELEC...
SQL:BatchCompleted	0	26	0	create index [TMPIND_0] on [#tt3] (_Q_001_F_000)
Showplan Statistics Profile				
Showplan XML Statistics P...				<ShowPlanXML xmlns="http://schemas.microsoft.com/sqlserver...
SQL:BatchCompleted	5307	442...	1000000	INSERT INTO #tt3 (_Q_001_F_000) SELECT ((((((100000.0 * T...
Showplan Statistics Profile				
Showplan XML Statistics P...				<ShowPlanXML xmlns="http://schemas.microsoft.com/sqlserver...
SQL:BatchCompleted	0	0	0	SELECT spid, blocked FROM master..sysprocesses WHERE block...
Showplan Statistics Profile				
Showplan XML Statistics P...				<ShowPlanXML xmlns="http://schemas.microsoft.com/sqlserver...
RPC:Completed	588	2057	2	exec sp_executesql N'SELECT T1._Q_001_F_000 FROM #tt3 T1 W...
SQL:BatchCompleted	0	30	0	TRUNCATE TABLE #tt1
SQL:BatchCompleted	0	49	0	DROP INDEX #tt3.TMPIND_0
SQL:BatchCompleted	0	45	0	TRUNCATE TABLE #tt3
Trace Pause				

Время создания таблицы увеличилось с 1 секунды до 5.3 секунды, при этом даже поиск по индексу в таблице все равно происходит медленнее, чем сканирование: 0.5 секунды против 0.1 секунды без индекса. Единственное, в чем этот запрос выигрывает – немного меньше логических чтений: 2057 против 2233 при сканировании. Поэтому следует использовать индексирование только в том случае, если вы видите от этого явный положительный эффект, т.е. если запрос начинает работать быстрее.

Для данного примера использовался MS SQL Server 2008, на других версиях СУБД результат по времени может отличаться, но в данном случае все равно везде затраты на создание индекса будут больше чем выигрыш от его использования.

Работа с инструментом SQL Profiler будет рассмотрена в отдельном разделе курса.

Удаление временных таблиц

Если временную таблицу использовать только в одном пакетном запросе, то она «живет», пока выполняется этот пакетный запрос. Это значит, что менеджер временных таблиц (МВТ) был создан неявно. Как только пакетный запрос завершается, неявный МВТ закрывается, и автоматически следует команда «Truncate table», которая очищает созданную таблицу. Таблица на сервере не удаляется, она остается, просто в ней очищаются данные, и это сделано намеренно. Если платформе требуется создать другую временную таблицу, она сначала ищет, есть ли уже существующая таблица с подходящим набором колонок. Если такая таблица есть, то используется она и новая таблица не создается.

Если временная таблица проиндексирована, то сначала будет очищен индекс и только потом таблица.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Если явное объявление МВТ не используется, то обычно нет необходимости использовать команду УНИЧТОЖИТЬ. Такая необходимость может возникнуть в следующих случаях:

- Создается очень большая временная таблица. Например, у вас есть большой пакетный запрос, в самом начала создается очень большая временная таблица которая используется только один раз. После использования эта таблица уже не нужна, но пока все запросы не выполнятся, она будет занимать память. В этом случае сразу после использования временную таблицу лучше удалить, чтобы освободить ресурсы сервера.
- Необходимо создать в пакетном запросе новую временную таблицу с именем уже существующей временной таблицы

Здесь главное – понимать, что таблица все равно будет удалена при завершении пакетного запроса.

Ситуация меняется, если Вы явно используете менеджер временных таблиц (МВТ), т.е. создаете соответствующий объект.

Такая таблица будет удалена в любом из следующих вариантов:

- В запросе использована команда УНИЧТОЖИТЬ
- Вызван метод *МенеджерВременныхТаблиц.Закрыть()*
- Объект *МенеджерВременныхТаблиц* перестал существовать, например, завершилась работа процедуры или функции, которая породила этот объект, или пользователь закрыл программу.

Если Вы используете объект МВТ, то временные таблицы рекомендуется удалять одним из первых двух методов, как только в них отпала необходимость, иначе они будут занимать память сервера СУБД, пока процедура или функция не закончит работу. Если же код, в котором был создан МВТ, завершается как раз выполнением запроса, тогда, конечно, МВТ можно не удалять, т.к. сработает 3-е условие.

Подведем итог: если явное объявление МВТ не используется, то явно удалять временную таблицу не требуется, она будет удалена после завершения пакетного запроса. Если явное объявление МВТ используется, то рекомендуется удалить таблицу вручную, например, в запросе командой УНИЧТОЖИТЬ, либо методом *МВТ.Закрыть()*.

Минусы временных таблиц

Идеальных инструментов не бывает, тем более в мире 1С. Давайте рассмотрим, какие проблемы может принести активное использование временных таблиц.

Чрезмерное разрастание базы TempDB.

Если Вы активно используете временные таблицы, то у Вас может довольно сильно разрастаться база TempDB, и «в один прекрасный день» она может занять все свободное место на диске.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Размер TempDB автоматически только увеличивается, но не уменьшается. Внутри файла место может как заниматься, так и освобождаться, но сам размер файла только увеличивается.

Типичная ситуация: установили обновление, и через несколько дней TempDB разрослась до неимоверных размеров, а потом выясняется, что разработчики переписали запросы с использованием временных таблиц, причем таблицы эти внушительного размера и их много. Для уменьшения размера базы TempDB необходимо выполнить следующие команды:

```
dbcc shrinkfile (tempdev, ЖелаемыйРазмерФайлаДанныхМб)
dbcc shrinkfile (templog, ЖелаемыйРазмерФайлаЛоговМб)
```

Чрезмерное упрощение запросов

Часто временные таблицы используются как раз для упрощения сложных запросов, чтобы серверу было легче подобрать оптимальный план, и это правильно. Но можно встретить и другую крайность: запрос, который вполне можно было бы написать без временных таблиц и который работал бы быстро и оптимально, все равно пишут с использованием временных таблиц. Если можно написать оптимальный запрос без использования временных таблиц, то лучше обойтись без них. Запрос с оптимальным кодом без временных таблиц в любом случае будет работать быстрее и использовать меньше ресурсов, чем запрос с временными таблицами.

Занятие 17

Основные сведения об индексах

Индекс – это объект базы данных, предназначенный для ускорения поиска данных.

Индексы оказывают огромнейшее влияние на скорость выполнения запросов, поэтому данной теме необходимо уделить самое пристальное внимание. Грамотное использование индексов может ускорить запросы не просто в разы, а в сотни, иногда даже в тысячи раз. Такого ускорения невозможно добиться аппаратными средствами.

Для грамотного использования индексов необходимо знать и понимать их устройство и принцип работы.

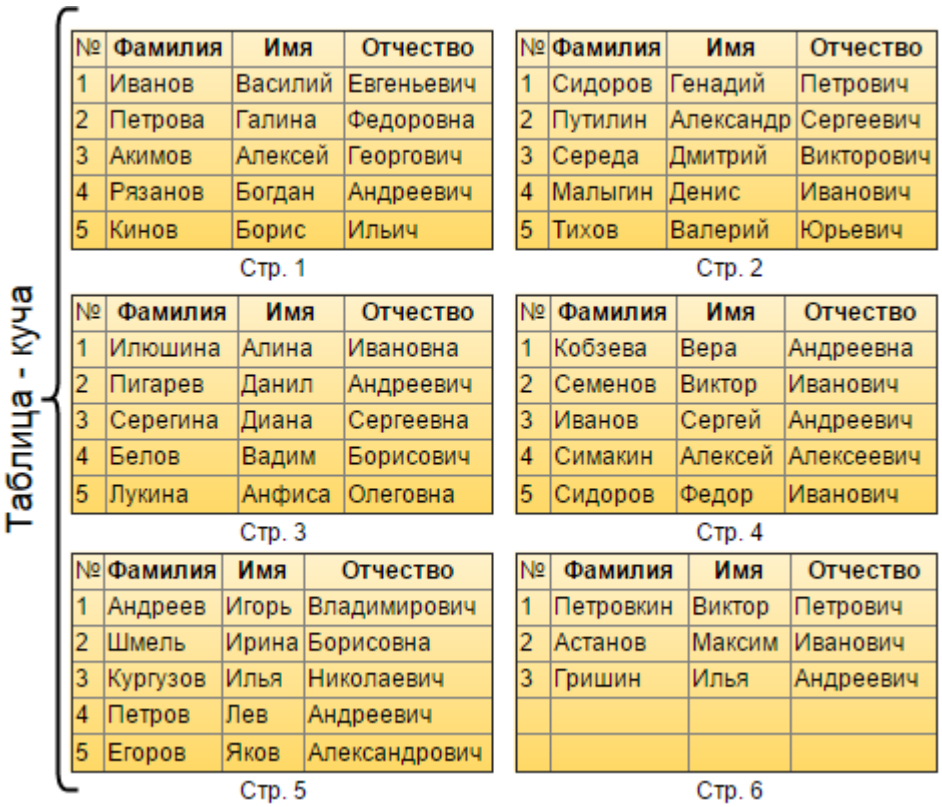
Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Таблица без индекса

Рассмотрим, как выполняется поиск, если в таблице нет ни одного индекса. Таблица без индексов – это куча, неупорядоченный набор данных, который хранится в восьмикилобайтных страницах на диске. При вставке новой строки она обычно добавляется в самый конец, на самую последнюю страницу.

Допустим, есть база, в которой хранятся ФИО сотрудников. В таблице три колонки и нет ни одного индекса.



Единственный способ поиска в такой таблице – это перебор всех записей (сканирование). Если требуется найти имена всех сотрудников с фамилией Гришин, то придется просмотреть все записи в таблице, т.е. прочитать все 6 страниц данных. Если таблица маленькая, то чтение будет выполнено быстро и проблем не возникнет. Но если записей в таблице не 28, как в примере, а 280 млн., тогда просмотр всех записей будет выполнятся очень долго.

Для ускорения поиска следует создать индекс по тому полю, по которому задается условие поиска.

Можно привести следующую аналогию: таблица – это книга, а индекс – это оглавление. Если бы не было оглавления, пришлось бы пролистать все страницы для поиска нужной главы.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

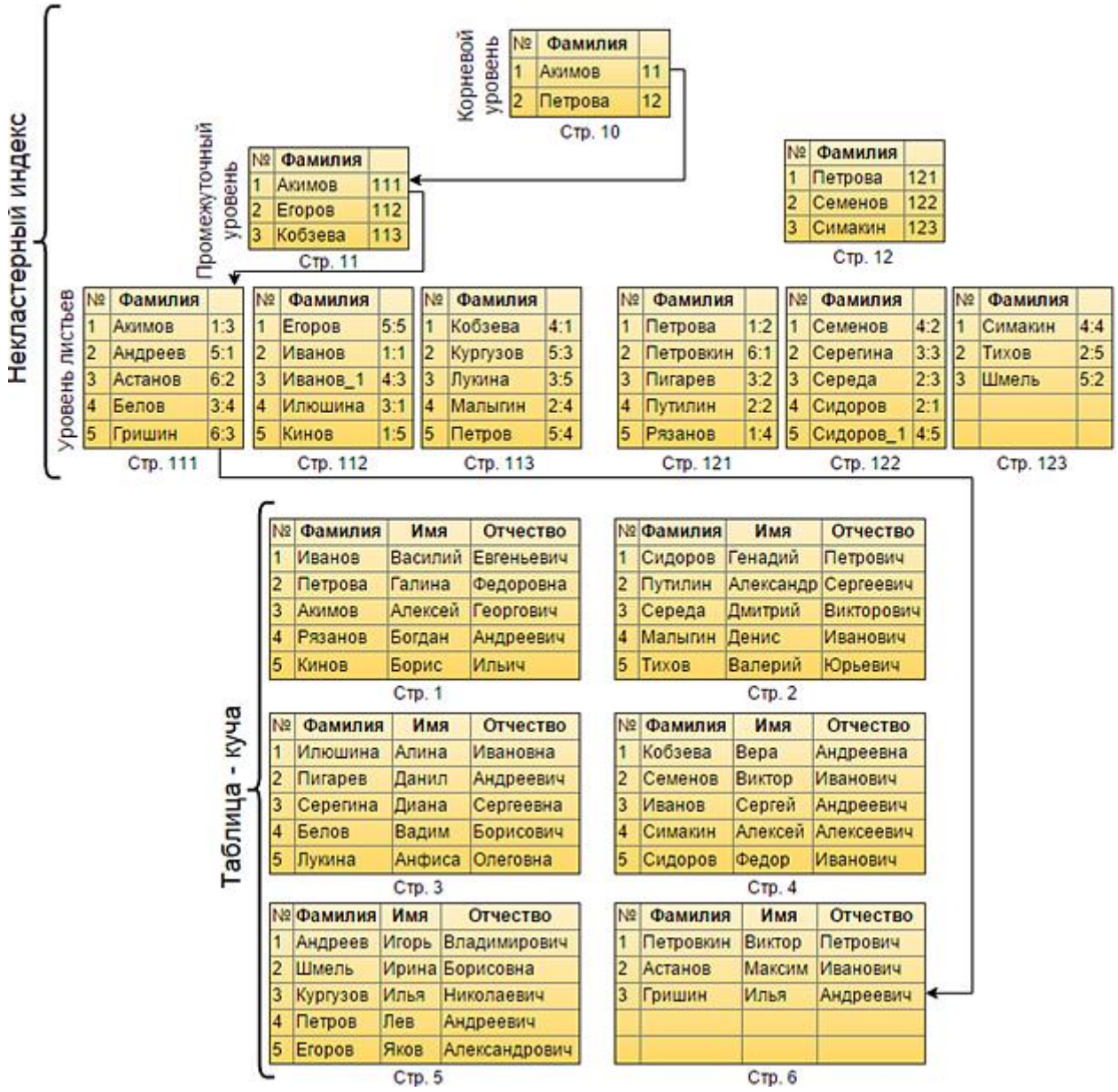
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Таблицы-кучи в базах 1С используются редко, исключением являются временные таблицы и регистры расчета. Если у временной таблицы не создавать индекс, то это будет типичная таблица-куча.

Некластерный индекс

В приведенном примере, ускорения поиска по фамилии необходимо создать простой некластерный индекс. Отличия кластерных и некластерных индексов будут рассмотрены чуть позже. После создания индекса в базе данных СУБД будет два объекта: таблица и индекс. Это разные объекты с разной структурой.



Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Индекс представляет собой сбалансированное B-дерево. Сбалансированность дерева означает, что длина любых двух путей от корня до листьев различается не более, чем на единицу, другими словами, поиск любого значения по индексу будет проходить почти с одинаковой скоростью.

Индекс состоит из нескольких уровней. Первый уровень называется корневым, он всегда состоит из одной страницы. Далее следуют промежуточные уровни, их может быть несколько, а может и вовсе не быть. Самый нижний уровень – это уровень листьев, который содержит адрес строки данных в таблице. Адрес данных состоит из номера файла данных, номера страницы и номера записи в странице. В большинстве случаев используется один файл данных, следовательно номер файла данных будет одинаковым, поэтому в целях упрощения в примере номер файла данных не указывается. Например, на приведенном выше рисунке, ключу индекса Акимов соответствует адрес 1:3, это значит, что данные расположены на первой странице в третьей строке.

Индекс всегда хранится в отсортированном виде для ускорения поиска, при этом сама таблица-куча остается неотсортированной, т.к. некластерный индекс – это отдельный от таблицы объект.

Следует помнить, что некластерный индекс не хранит в себе всю строку данных, там хранится только значение ключа индекса (в примере это фамилия) и адрес строки данных в таблице.

Разберем, как происходит поиск по индексу на примере. Допустим, что необходимо найти имена всех сотрудников с фамилией Гришин, поиск выполняется следующим образом:

1. Считывается страница корневого уровня индекса. Значение «Гришин» находится после значения «Акимов», но до значения «Петров», значит, надо перейти на 11-ю страницу.
2. Считывается 11 страница в промежуточном уровне, и определяется адрес следующей страницы – страница 111.
3. Считывается 111 страница на уровне листьев, и определяется адрес строки данных: страница 6, строка 3.
4. В таблице данных считывается страница 6, и из строки номер 3 получается значение колонки «Имя».

Если бы в запросе не требовалось получить имя, а нужна была только фамилия, тогда бы 4 пункт не выполнялся, т.к. фамилия уже содержится в самом индексе и является его ключом, поэтому нет смысла обращаться к таблице.

Благодаря своей древовидной структуре, поиск значений по индексу производится гораздо быстрее, чем поиск сканированием. В данном примере было считано 4 страницы, при сканировании пришлось бы считать 6 страниц. Чем больше таблица, тем сильнее будет разница в скорости между поиском по индексу и сканированием.

При внимательном рассмотрении схемы индекса можно заметить, что там присутствуют значения Иванов_1 и Сидоров_1, которых нет в таблице. Дело в том, что индекс всегда уникальный, и даже если значения ключа повторяются, то СУБД добавляет к ключу приставку, чтобы сохранить уникальность. Эта приставка пользователю не видна, она используется только внутренними механизмами СУБД.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

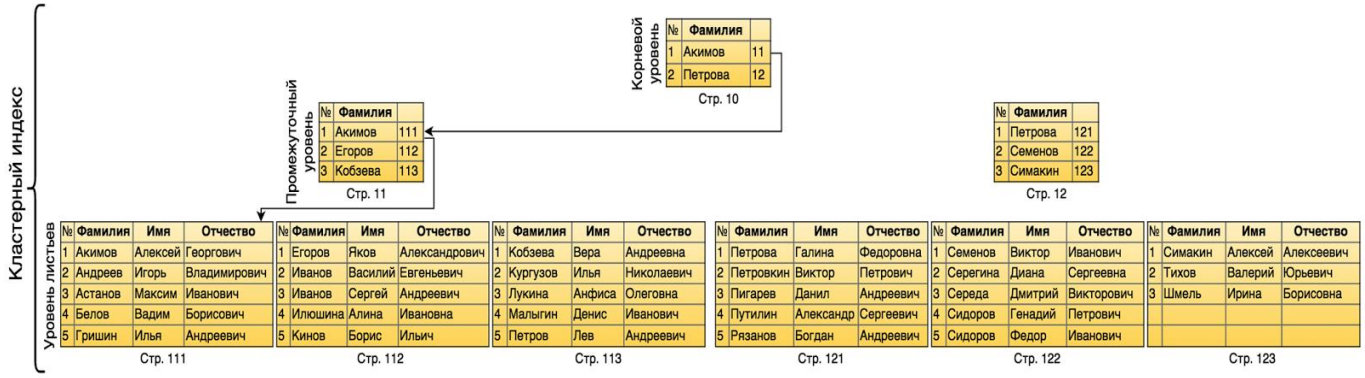
Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Кластерный индекс

По способу организации таблицы индексы могут быть кластерными и некластерными. Выше рассмотрен типичный некластерный индекс. Важно помнить, что некластерный индекс является отдельным объектом и на уровне листьев хранит ссылку на строку данных.

Кластерный индекс – это смесь таблицы и индекса, где на уровне листьев в отсортированном виде содержатся сами данные.

Допустим, в таблице сотрудников из нашего примера был удален некластерный индекс и создан кластерный по фамилии, тогда схема индекса будет выглядеть следующим образом.



(изображение можно увеличить без потери качества)

Изображение является слишком широким что бы поместиться на странице, поэтому оно разделено на две части.

Само понятие «Кластер» говорит о том, что объект состоит из нескольких сущностей, объединенных в одну. В данном случае имеется ввиду объединение таблицы и индекса.

При создании кластерного индекса таблица сортируется по ключу кластерного индекса, т.е. меняется ее структура и таблица становится упорядоченной.

Очень часто возникает путаница между понятиями кластерный индекс и кластеризованная таблица, поэтому здесь нужно уделить отдельное внимание терминологии.

Принято говорить, что некая таблица имеет кластерный индекс, но это лишь устоявшийся оборот речи. Кластерный индекс, кластеризованная таблица и таблица, имеющая кластерный индекс – все это один и тот же объект. Обычно под кластерным индексом подразумевают поля, которые являются ключами индекса и содержаться в корневом и промежуточных уровнях, а под кластеризованной таблицей – поля на уровне листьев того же индекса, т.е. это разные части одного объекта.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

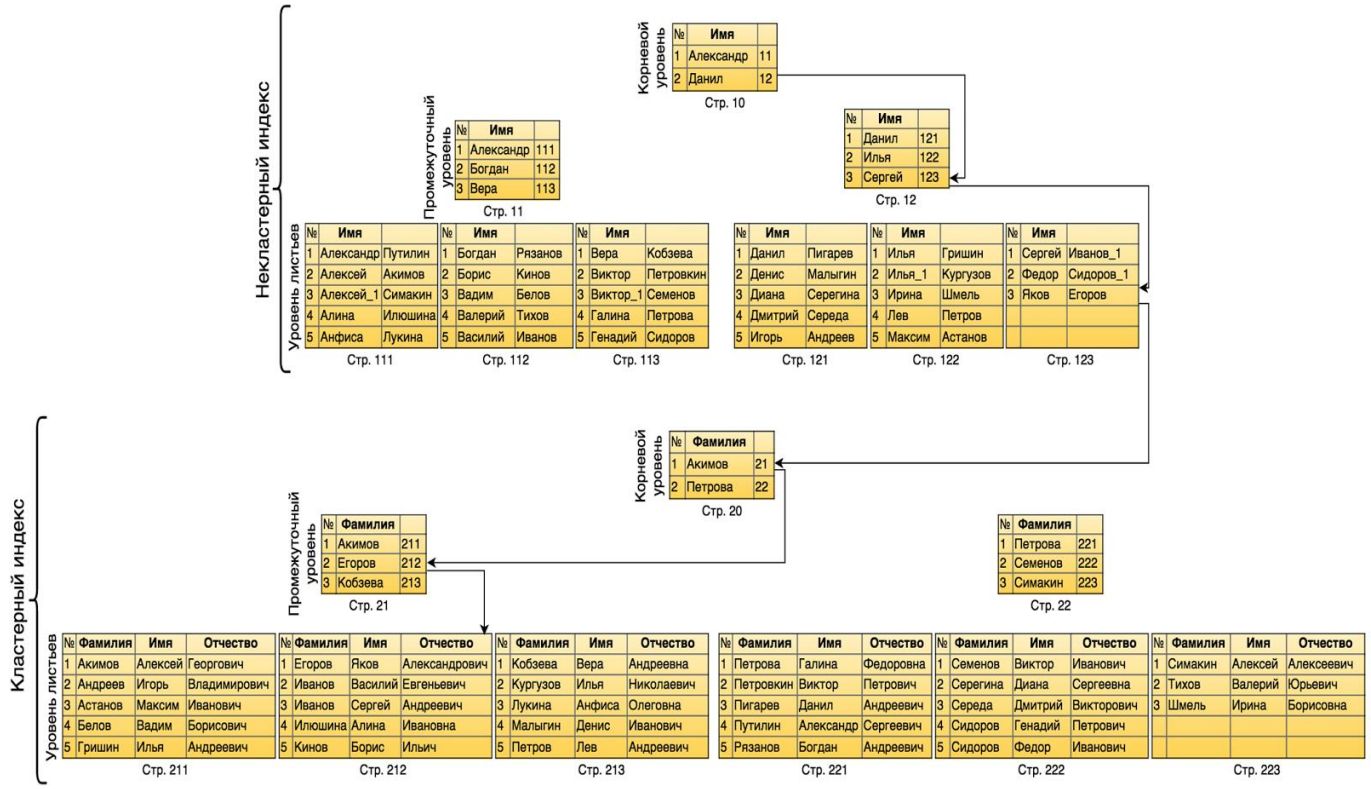
У таблицы может быть только один кластерный индекс, т.к. по его ключу сортируются данные, а существование второго кластерного индекса предполагало бы одновременную сортировку таблицы данных по разным полям, что невозможно.

Поиск в кластерном индексе осуществляется почти так же, как и в некластерном: идет поиск по дереву сверху вниз, отличие заключается лишь в том, что на уровне листьев содержатся все данные строки.

Кластерные индексы очень удобны в том случае, если в таблицу вставляются данные, у которых значение ключа индекса постоянно возрастает. Например, для документов, которые всегда проводятся текущей датой, было бы идеально сделать кластерный индекс по дате.

Некластерный индекс поверх кластерного

Рассмотрим, что произойдет, если для таблицы со списком сотрудников, имеющей кластерный индекс по полю «Фамилия», создать некластерный индекс по полю «Имя».



(изображение можно увеличить без потери качества)

Как видно из рисунка, был создан отдельный некластерный индекс, т.е. на уровне СУБД будет 2 разных объекта. Обратите внимание, что на уровне листьев в некластерном индексе будет храниться не адрес данных, а значение ключа кластерного индекса, именно эта особенность отличает некластерный индекс для таблицы-кучи и некластерный индекс поверх кластерного.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Рассмотрим процесс поиска данных с использованием некластерного индекса поверх кластерного.

Допустим, в запросе необходимо выбрать отчество всех сотрудников с именем Яков. В этом случае СУБД выполнит следующую последовательность действий:

1. Условие в запросе по полю «Имя», следовательно, можно использовать некластерный индекс по имени. Считывается страница корневого уровня некластерного индекса. Значение «Яков» ниже, чем «Данил», поэтому надо перейти на страницу 12.
2. Считывается страница 12. «Яков» ниже, чем «Сергей», значит, надо перейти на страницу 123
3. Считывается страница 123. Из найденной строки считывается ключ кластерного индекса «Егоров», значит, надо перейти в кластерный индекс и продолжить поиск там по значению «Егоров».
4. Считывается страница корневого уровня кластерного индекса. «Егоров» находится между «Акимов» и «Петрова», значит, надо считать страницу 21
5. Считывается страница 21. Строка со значением «Егоров» находится на 212 странице.
6. Считывается страница 212. Значение колонки «Отчество» соответствующей строки возвращается в результирующий набор запроса.

Процесс поиска зависит от текста запроса: если бы требовалось вернуть только фамилию или имя, тогда бы не было обращения к кластерному индексу, т.к. эти данные есть в некластерном индексе.

Как видно из схемы, некластерный индекс поверх кластерного занимает довольно много места из-за того, что приходится хранить значение ключа кластерного индекса. Поиск по такому индексу может отнимать довольно много времени, ведь сначала поиск идет по дереву некластерного индекса, а потом по дереву кластерного. Именно поэтому СУБД в таких случаях может выбрать сканирование таблицы вместо поиска по индексу. С другой стороны, некластерный индекс поверх кластерного дает возможность искать по тем полям, которые не входят в состав кластерного или находятся в нем не на первом месте.

Индексы именно такого типа (некластерный поверх кластерного) – самые распространенные в базах 1С.

Простые и составные индексы

По своему составу индексы делятся на простые и составные (композитные).

Простой индекс состоит из одного поля. Например, все индексы, описанные выше, являются простыми, т.к. состоят из одного поля. Составной индекс, как несложно догадаться, состоит из нескольких полей.

Не следует путать простые/составные и кластерные/некластерные индексы.

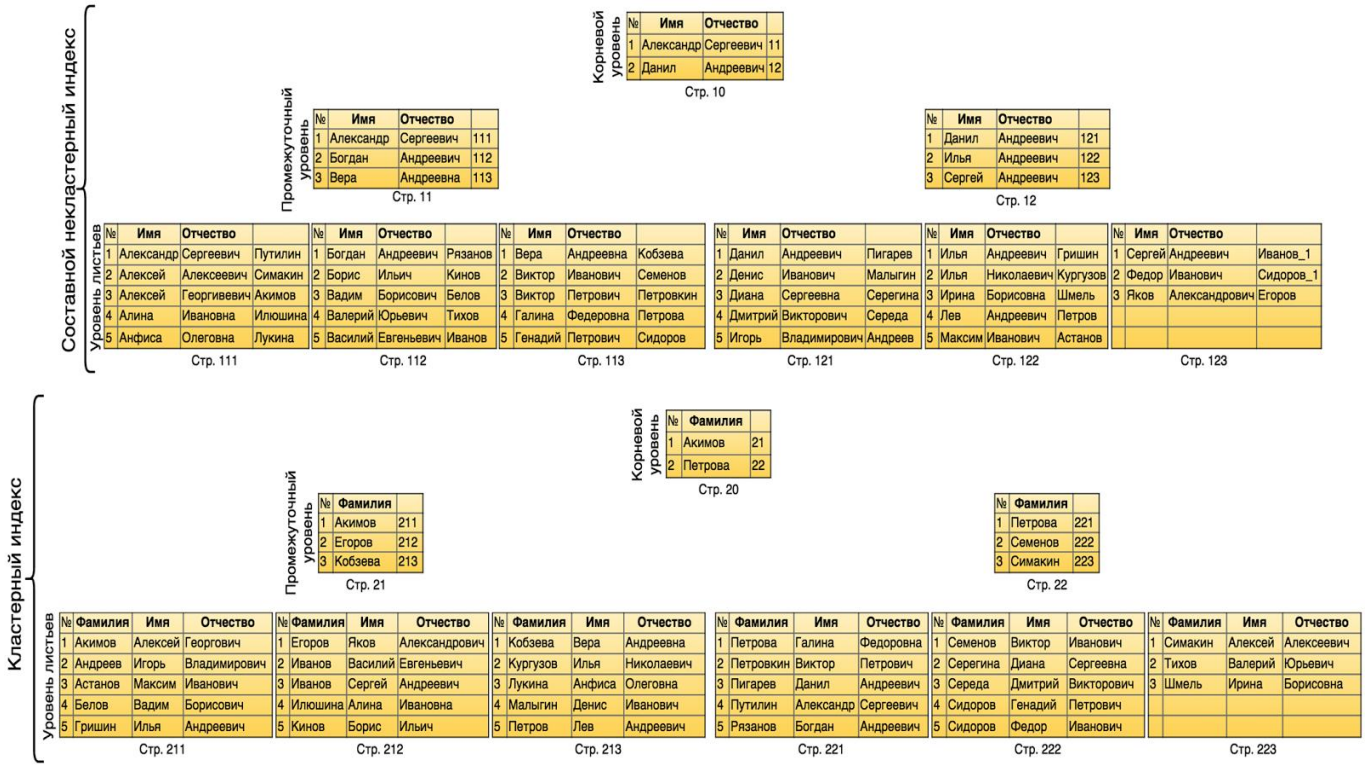
Простой индекс может быть как кластерным, так и некластерным, составной индекс также может быть как кластерным, так и некластерным.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Допустим, по таблице сотрудников, в которой создан кластерный индекс по полю «Фамилия», был создан составной некластерный индекс, состоящий из двух полей Имя+Отчество.



(изображение можно увеличить без потери качества)

В составном индексе не требуется уникальность каждого отдельного поля, главное, чтобы комбинация полей была уникальна. Если же комбинация полей будет совпадать, то СУБД снова добавит к одному из полей приставку, чтобы обеспечить уникальность.

У составного индекса есть одна важная особенность: поиск может осуществляться только в том порядке, в котором следуют поля в индексе, т.е. слева направо. Например, индекс Имя+Отчество может использоваться, только если условие задано по полю «Имя» или сразу по двум полям: «Имя» и «Отчество», но если задать условие только по отчеству, то данный индекс не будет использован, т.к. колонка «Имя» как бы мешает использовать условие по колонке «Отчество», ведь сортировка идет сначала по имени. В результате вместо поиска по индексу СУБД выполнит сканирование таблицы, т.е. произведет полный перебор данных.

Разберем другой пример: допутим, есть составной индекс из трех полей A+B+C. Если задать условие поиска по полю A или A и B, или по всем трем полям, то будет выполнен поиск по индексу, но если задать условие толькойкй по B или только по C, или по B и C, то будет выполнено сканирование, т.к. поле A предшествует полю B и его нет в условии. Подробнее об этом будет рассказано в разделе об основных причинах медленной работы запроса.

В базах 1С большая часть индексов является составными.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

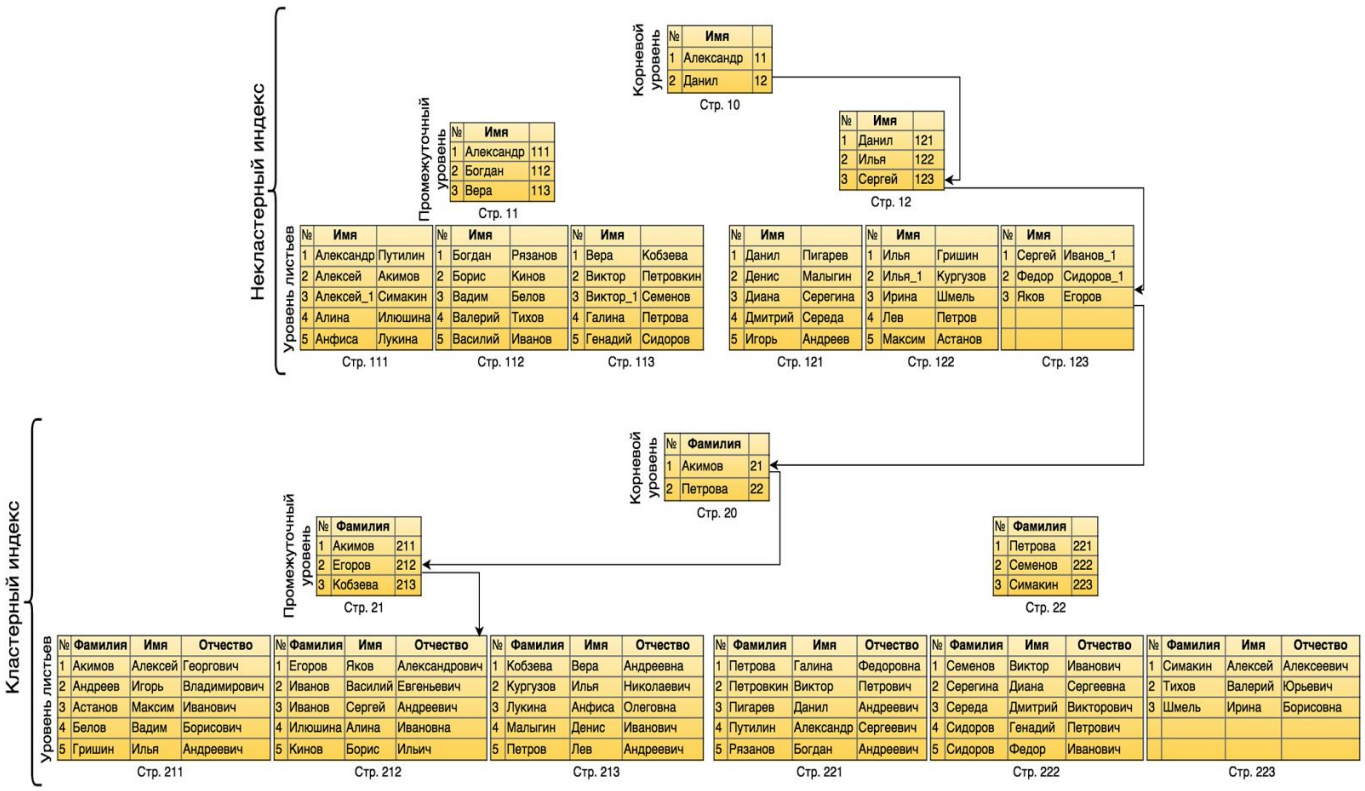
Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

На диске ИТС сказано, что в составном индексе на первом уровне дерева находятся значения первого поля индекса, на втором уровне - второго поля и так далее. Такое определение не соответствует действительному устройству индекса, возможно оно было приведено в целях упрощения понимания строения составного индекса. В действительности составной индекс устроен именно так, как показано на рисунке выше, т.е. все поля составного индекса всегда присутствуют на всех его уровнях.

Покрывающий индекс

Покрывающий индекс – это индекс, который содержит в себе все поля, возвращаемые в запросе.

Выше уже было сказано, что набор полей, возвращаемых в запросе, очень сильно влияет на то, какие индексы будут использованы при выполнении запроса. Допустим, что в таблице сотрудников есть некластерный индекс по полю Имя и кластерный по полю Фамилия.



(изображение можно увеличить без потери качества)

Пользователь выполняет запрос:

```
ВЫБРАТЬ Имя, Отчество
ИЗ Таблица.Сотрудники
ГДЕ Имя = «Яков»
```


Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Условие задано по имени, следовательно для поиска будет использоваться некластерный индекс по полю Имя.

В данном случае требуется вернуть два поля: Имя и Отчество, но в некластерном индексе хранится только имя, а отчество находится в кластерном индексе, и так как мы не можем получить все возвращаемые поля запроса из индекса по полю Имя, то для данного запроса он не является покрывающим.

Если в секции ВЫБРАТЬ возвращать только Имя или Имя и Фамилию, тогда индекс будет покрывающим, т.к. Имя – это ключ самого некласттерного индекса, а Фамилия – это ключ кластерного индекса, который выступает в роли адреса.

Если сделать запрос с условием по фамилии, то будет использован кластерный индекс. В кластерном индексе хранятся все поля таблицы, поэтому кластерный индекс всегда будет покрывающим.

Индексы платформы

Платформа 1С автоматически создает индексы для всех объектов метаданных. Состав индексов зависит от объекта и от версии платформы.

Рассмотрим основные индексы для наиболее распространенных объектов метаданных.

Объект метаданных	Имя таблицы	Индекс	Тип и условие создания индекса
Справочник	Reference	Ссылка	Кластерный. Всегда
		Код+Ссылка	Если длина кода > 0
		Наименование+Ссылка	Если длина наименования > 0
Документ	Document	Ссылка	Кластерный. Всегда
		Дата+Ссылка	Всегда
		Номер+Ссылка	Если длина номера > 0

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Регистр сведений непериодический	InfoRg	Измерение1+…+ИзмерениеN	Кластерный, если регистр независимый. Если количество измерений > 0
		Регистратор+НомерСтроки	Кластерный, если регистр подчинен регистратору. Всегда
Регистр сведений периодический	InfoRg	Период+Измерение1+…+ИзмерениеN	В 8.2 кластерный, в 8.3 некластерный. Всегда
		Измерение1+…+ИзмерениеN+Период	В 8.3 кластерный, в 8.2 некластерный. Если количество измерений > 0
		Регистратор+НомерСтроки	Если регистр подчинен регистратору
Регистр накопления остатков	AccumRg	Период+Регистратор+НомерСтроки	Кластерный. Всегда
		Регистратор+НомерСтроки	Всегда
	AccumRgT	Период+Измерение1+…+ИзмерениеN +Splitter	Кластерный. Всегда. Splitter добавляется только при включенном разделении итогов
Регистр накопления оборотов	AccumRg	Период+Регистратор+НомерСтроки	Кластерный. Всегда
		Регистратор+НомерСтроки	Всегда
	AccumRgTn	Период+Измерение1+…+ИзмерениеN +Splitter	Кластерный. Всегда. Splitter добавляется только при включенном разделении итогов

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Регистр бухгалтерии	AccRg	Период+Регистратор+НомерСтроки	Кластерный. Всегда
		Регистратор+НомерСтроки	Всегда
	AccRgAT0	В версии 8.2: Период+Счет+Измерение1+… +ИзмерениеN В версии 8.3: Счет+Период+Измерение1+… +ИзмерениеN	Кластерный. Всегда
Регистр расчета	CRg<N>	ПериодРегистрации+Регистратор+ НомерСтроки	Всегда
		Регистратор+НомерСтроки	
		ПериодРегистрации+Измерение1	Если Измерение1 является базовым
		Измерение1+ПериодРегистрации	

В последней колонке указан тип индекса и условия его создания. Если тип индекса не указан, значит, он некластерный. Как видно из таблицы, почти для всех основных объектов, за исключением регистров расчета, создаются кластерные индексы.

Если в базе используются разделители областей данных, тогда к каждому индексу первым полем добавится поля значения разделителя данных.

В данной таблице приведены лишь основные индексы, создаваемые платформой. С полным списком индексов можно ознакомиться в разделе «Индексы таблиц базы данных» на диске ИТС.

В таблице описаны индексы, создаваемые платформой автоматически, однако при необходимости можно создать собственные индексы. Перед тем как создавать индекс, имеет смысл убедиться, что подобный индекс уже не создан платформой. Также основные индексы необходимо знать для оптимального написания запросов.

Полный состав индексов базы данных всегда можно посмотреть с помощью обработки «Структура хранения метаданных».

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Минусы индексов

У любого индекса есть 3 главных минуса.

1. **Занимает место**. В больших таблицах с большим числом индексов их совокупный размер может превышать размер самой таблицы.
2. **Требует регулярного обслуживания**. Индексы нуждаются в дефрагментации, реиндексации и обновлении статистики. В больших таблицах эти операции могут выполнятся очень долго.
3. **Замедляет операции вставки, изменения и удаления**. Чем больше индексов у таблицы, тем медленнее выполняются операции изменения строки, т.к. надо поддерживать порядок сортировки в индексе и поддерживать сбалансированность дерева.
   Например, в таблицу добавляют новую строку с фамилией Алаев. Чтобы сохранить сортировку, СУБД вставит новую строку между строками со значениями между Акимов и Андреев, при необходимости (если на требуемой страинце нет места) перестроит (расщепит) страницу и перенесет часть данных на другие или вновь созданные страницы. Процесс расщепления страниц довольно затратный, и чем больше размер индекса, тем ветвистее дерево индекса и тем медленнее этот процесс. Для кластерного индекса процесс расщепления страницы будет еще более медленным, т.к. на уровне листьев хранятся значения всех столбцов таблицы, следовательно придется переносить больше данных.

Перед тем как создавать индекс, необходимо взвесить все «за» и «против» и только после этого принимать решение. Создавать индекс для каждой колонки таблицы – крайне плохая идея.

Создание собственных индексов

Видеоурок. Свойство «Индексировать»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как создать собственный индекс с помощью конфигуратора
- Как посмотреть состав получившегося индекса
- Какие индексы получаются для разных объектов метаданных
- Можно ли индексировать ресурс.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Индексировать с дополнительным упорядочиванием

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Чем индексация с дополнительным упорядочиванием отличается от обычной.

Видеоурок. Создание индекса для первого измерения

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Нужно ли создавать индекс для первого измерения
- На что нужно обращать внимание при создании индекса по первому измерению.

Видеоурок. Свойство «Ведущее»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Влияние данного свойства на состав индекса
- В каких случаях индекс не создается, даже если свойство установлено.

Видеоурок. Создание индексов через Management Studio

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как создать индекс, используя Management Studio
- В каких случаях может понадобиться создавать индекс средствами MS SQL Server.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

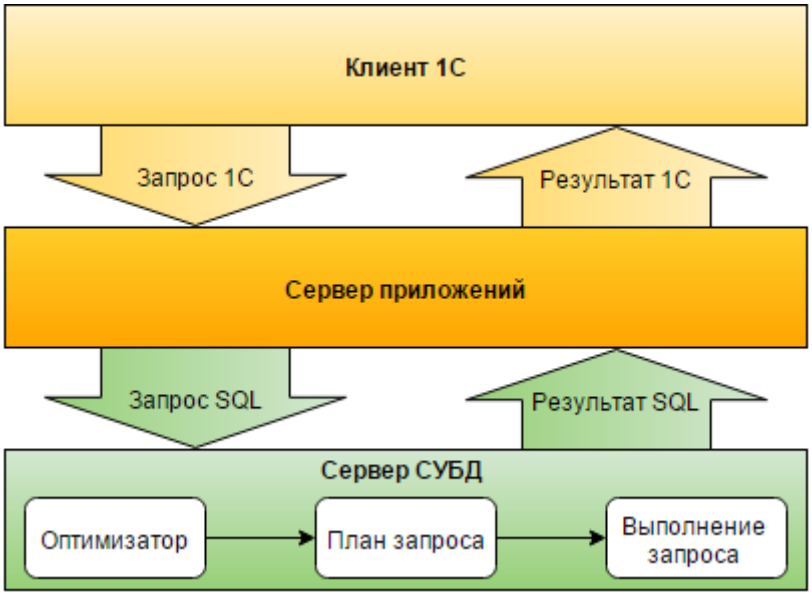
Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 18

Основные сведения о планах запроса

Схема работы запроса в 1С

Давайте подробно рассмотрим схему выполнения запроса, сформированного в 1С в клиент-серверном варианте.



Выполнение запроса в 1С можно разбить на несколько этапов.

1. Текст запроса на языке 1С передается с клиента на сервер приложений 1С. Клиенты не обращаются к серверу СУБД напрямую, вся работа идет посредством сервера приложений.
2. Сервер приложений модифицирует запрос, написанный на языке 1С, в ту форму, которая будет понятна серверу СУБД, т.е. запрос 1С преобразуется в запрос на языке SQL. На этом этапе вместо имени объекта 1С подставляется имя соответствующей ему таблицы СУБД. Итоговый текст запроса SQL зависит не только от того, к каким объектам метаданных идет обращение, но и от выбранных условий, и даже от значений параметров запроса. Например, если при обращении к виртуальной таблице остатков регистра накопления указать дату начала месяца, то запрос SQL будет только к таблице итогов, а если указать любую другую дату, то этот же запрос будет обращаться уже к двум таблицам: к таблице итогов и таблице движений.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

3. Итоговый текст запроса на языке SQL передается на исполнение в СУБД. Обычно итоговый текст запроса SQL не зависит от типа и версии СУБД, так что можно считать, что запросы 1С транслируются в SQL всегда одинаково.

4. Сервер СУБД начинает выполнение запроса. Выполнение запроса также состоит из нескольких этапов. Ниже в упрощенной форме описаны только основные из них.

   4.1. Сначала начинает работать **оптимизатор запроса**, задача которого – определить, каким именно образом будет выполняться запрос. Запрос содержит информацию о том, какие данные нужны пользователю, но в нем нет информации о том, как эти данные получить. У оптимизатора очень сложная задача: используя данные о статистике, наличии индексов и многих других факторов, он должен за минимальное время найти наиболее быстрый способ выполнить запрос.
   Допустим, нужно соединить 8 таблиц, при этом в один момент времени можно соединить только 2 таблицы, таким образом, число возможных комбинаций соединений будет довольно большим, и это не считая того, что таблицы можно соединить тремя разными способами. Пример с таблицами – лишь малая часть работы оптимизатора, который является одним из самых сложных и дорогих компонентов любой СУБД. Далее мы будем очень подробно изучать его работу и влияние на производительность запросов.

   4.2. Результатом работы оптимизатора является план запроса. **План запроса** – это инструкции для СУБД, как именно выполнять запрос. Другими словами, план запроса – это пошаговый алгоритм получения данных. Приведем пример: прочитать полностью таблицу 1, соединить ее с таблицей 3, то, что получилось, отфильтровать и соединить с таблицей 2.

   4.3. Сервер СУБД выполняет план запроса. Если оптимизатор сгенерировал неоптимальный план, то и запрос будет выполнятся неоптимально.
   Для упрощения понимания плана запроса лучше всего использовать аналогию с навигатором GPS. Указание в навигаторе адреса места назначения – это запрос, т.е. пользователь говорит, что именно ему нужно. Далее навигатор рассчитывает наиболее быстрый маршрут с учетом пробок, аварий и прочих данных о дорожной ситуации – это работа оптимизатора запроса. Итоговый маршрут, появившийся на экране, – это план запроса. Если навигатор ошибся и выбрал неверный маршрут, например, через пробку (неоптимальный план запроса), то дорога вместо запланированных 15 минут может занять 2 часа (медленный запрос).

5. Результат, полученный сервером СУБД, возвращается на сервер приложений 1С. На сервере приложений результат запроса может быть дополнительно обработан, например, в том случае, если в запросе используется ключевое слово ИТОГИ, т.к. эта опция есть только в языке запросов 1С и отсутствует в «чистом» SQL.

6. Результат запроса, обработанный сервером приложений 1С, возвращается на клиентское приложение 1С.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Формирование плана запроса

Неоптимальный план запроса – это главная причина медленной работы запросов. Нельзя повлиять на план запроса напрямую (невозможно исправить код СУБД), но можно создать такие условия, чтобы оптимизатор не ошибался и всегда компилировал оптимальный план. По сути, оптимизация запросов заключается в том, чтобы упростить работу оптимизатора, а для этого, необходимо знать основные принципы его работы.

Оптимизатор строит план за ограниченное время. Никому не нужен план, который компилировался 5 минут, когда запрос при этом выполнялся за 5 секунд. Для пользователя лучше, чтобы план компилировался 5 миллисекунд и запрос при этом выполнился за 6 секунд. Поэтому оптимизатор не ищет всегда самый лучший план, он выбирает лучший план из тех, которые он успел сформировать за отведенное время.

Если оптимизатор не успевает найти самый лучший план за отведенное время, то у выбранного плана появляется свойство «Reason for Early Termination» со значением «Time Out», это так называемый «Timeout Warning». Это вовсе не значит, что получившийся план будет плохим, просто он будет лучшим из тех, что оптимизатор успел рассмотреть за отведенное время.

Обычно причиной такого таймаута является слишком сложный запрос с большим количеством таблиц, соединений и подзапросов. Для оптимизатора чем проще запрос, тем он лучше, тем быстрее для него можно скомпилировать план. Следовательно, чем больше и сложнее запрос, тем больше вероятность того, что оптимизатор выберет для него неоптимальный план, и тем больше шансов, что запрос будет выполняться медленно.

Можно подумать, что таймаут – это некое пороговое время, за которое оптимизатор должен успеть скомпилировать план, однако это не так. Построение плана – довольно сложный процесс с большим количеством операций. Для поиска наилучшего плана запрос подвергается определенным преобразованиям (предварительной оптимизации), и чем сложнее запрос, тем больше таких преобразований требуется. Таймаут для формирования плана – это количество операций преобразования запроса. Это значит, что время компиляции плана будет различным для разных запросов. Планы простых запросов будут компилироваться быстро, планы сложных запросов – медленно.

Компиляция плана – довольно сложная задача, которая отнимает много ресурсов у сервера СУБД, и именно поэтому планы запросов кэшируются. Если запрос выполняется в первый раз, то для него компилируется план, если запрос выполняется повторно, то план уже не компилируется, а берется из кэша. Этот кэш можно принудительно очистить, выполнив на сервере MS SQL команду *DBCC FREEPROCCACHE*. Однако делать это в рабочее время крайне не рекомендуется, иначе планы всех запросов, в том числе и служебных, начнут перекомпилироваться, что на некоторое время приведет к увеличению нагрузки на сервер и деградации производительности.

Существуют запросы, для которых возможен только один план выполнения, такой план называют тривиальным. **Тривиальный план** – это единственно возможный план выполнения.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Типичный пример такого запроса – выбор всех полей из таблицы без условий.

```
ВЫБРАТЬ * ИЗ ИмяТаблицы
```

Такой запрос можно выполнить максимально быстро, только просканировав всю таблицу, в этом случае оптимизатор даже не будет искать другие варианты.

Видеоурок. Просмотр плана запроса в SQL Profiler

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как, используя инструменты MS SQL Server, получить план запроса
- Какие события необходимо регистрировать в SQL Profiler.

Видеоурок. Просмотр плана запроса в консоли запросов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как получить план с помощью консоли запросов
- Откуда консоль запросов получает план
- Какие исправления необходимо внести в консоль запросов для нормальной работы во всех конфигурациях.

Видеоурок. Оператор Table Scan

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение и принцип работы данного оператора
- В каких случаях используется данный оператор.

Видеоурок. Оператор Clustered Index Scan

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- Назначение и принцип работы данного оператора
- В каких случаях используется данный оператор.

Видеоурок. Оператор Index Scan

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение и принцип работы данного оператора
- В каких случаях используется данный оператор.

Видеоурок. Оператор Constant Scan

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение и принцип работы данного оператора
- В каких случаях используется данный оператор
- Отличие данного оператора от операторов сканирования.

Видеоурок. Оператор Index Seek

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение и принцип работы данного оператора
- В каких случаях используется данный оператор.

Видеоурок. Оператор Clustered Index Seek

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение и принцип работы данного оператора
- В каких случаях используется данный оператор.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Оператор Sort

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение и принцип работы данного оператора
- В каких случаях используется данный оператор.

Видеоурок. Оператор Compute Scalar

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение и принцип работы данного оператора
- В каких случаях используется данный оператор.

Видеоурок. Оператор Nested Loops

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение и принцип работы данного оператора
- В каких случаях используется данный оператор.

Видеоурок. Оператор Merge Join

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение и принцип работы данного оператора
- В каких случаях используется данный оператор.

Видеоурок. Оператор Hash Join

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- Назначение и принцип работы данного оператора
- В каких случаях используется данный оператор.

Видеоурок. Логические и физические операторы

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Чем логические операторы отличаются от физических.

Видеоурок. Логический оператор Semi Join

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Понятие полусоединения
- В каком случае получается полусоединение
- Какие логические операторы обозначают полусоединение.

Занятие 19

Видеоурок. Отображение плана запроса

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Основные виды отображения плана запроса
- Плюсы и минусы разных видов отображения.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Порядок выполнения операторов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каком порядке выполняются операторы
- В каком направлении передаются данные, полученные операторами.

Видеоурок. Стоимость операторов и плана запроса

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что такое стоимость, и в чем она измеряется
- Всегда ли стоимость адекватно отображает ситуацию.

Видеоурок. Основные свойства операторов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как посмотреть информацию об операторах
- Каким свойствам стоит уделить особое внимание.

Видеоурок. Чтение графического плана запроса

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Каким образом читать графический план
- На что обращать внимание при чтении графического плана.

Видеоурок. Чтение текстового плана

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как читать текстовый план
- Отличия между текстовым и графическим планом.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Чтение плана запроса с разыменованием полей

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как выглядит текст и план типичного запроса с разыменованием полей.

Видеоурок. Чтение плана запроса с соединением нескольких таблиц

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как выглядит текст и план типичного запроса с соединением нескольких таблиц.

Видеоурок. Чтение плана запроса с использованием составного типа

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как выглядит текст и план типичного запроса с обращением к реквизиту поля составного типа.

Видеоурок. Чтение плана запроса с TOP и Nested Loops

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как обрабатывается конструкция запроса ВЫБРАТЬ ПЕРВЫЕ.

Видеоурок. Предварительная оптимизация запроса

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Каким основным преобразованиям подвергается запрос перед выполнением.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Признаки неоптимального плана. Оператор Nested Loops

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каких случаях использование Nested Loops является неоптимальным
- На какие виды проблем может указывать данный оператор
- В каком случае без данного оператора нельзя обойтись.

Видеоурок. Признаки неоптимального плана. Оператор Scan

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каких случаях использование оператора Scan является оптимальным
- В каких случаях использование оператора Scan указывает на проблему.

Видеоурок. Признаки неоптимального плана. Конструкция Seek… Where

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение данной конструкции
- На что может указывать эта конструкция в плане запроса.

Видеоурок. Признаки неоптимального плана. Оператор Hash Aggregate

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение данного оператора
- В каких случаях данный оператор используется в плане.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Признаки неоптимального плана. Оператор Key Lookup

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение оператора
- На какую проблему указывает данный оператор
- Возможные варианты решения проблемы.

Видеоурок. Признаки неоптимального плана. Оператор Table Spool

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение данного оператора
- В каких случаях этот оператор используется в плане запроса.

Видеоурок. Порядок анализа плана запроса

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каком порядке анализировать план запроса
- На что обращать особое внимание при анализе
- Как ориентироваться на стоимость текстового плана запроса.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

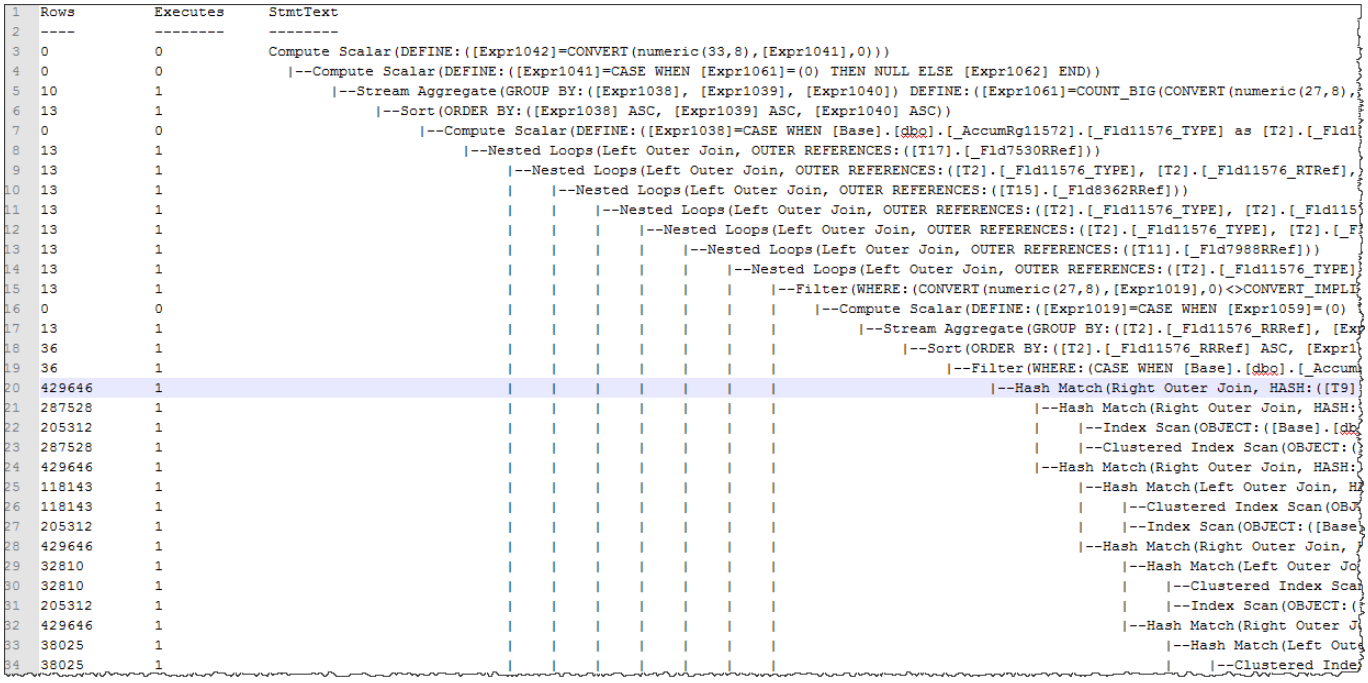
Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 20

Основные причины медленной работы запросов

Зачастую причину медленной работы запроса можно понять просто посмотрев текст запроса, даже не прибегая к анализу плана. Обычно быстрее будет проанализировать сам запрос на наличие узких мест, т.к. анализ плана запроса иногда представляется довольно затруднительным.

Ниже в качестве примера приведен фрагмент плана запроса с одного из проектов по оптимизации. План довольно большой и запутанный.



Rows	Executes	StmtText
0	0	Compute Scalar(DEFINE:([Expr1042]=CONVERT(numeric(33,8),[Expr1041],0)))
0	0	|--Compute Scalar(DEFINE:([Expr1041]=CASE WHEN [Expr1061]=(0) THEN NULL ELSE [Expr1062] END))
10	1	|--Stream Aggregate(GROUP BY:([Expr1038], [Expr1039], [Expr1040]) DEFINE:([Expr1061]=COUNT_BIG(CONVERT(numeric(27,8),
13	1	|--Sort(ORDER BY:([Expr1038] ASC, [Expr1039] ASC, [Expr1040] ASC))
0	0	|--Compute Scalar(DEFINE:([Expr1038]=CASE WHEN [Base].[dbo].[_AccumRg11572].[_Fld11576_TYPE] as [T2].[_Fld1
13	1	|--Nested Loops(Left Outer Join, OUTER REFERENCES:([T17].[_Fld7530RRef]))
13	1	|--Nested Loops(Left Outer Join, OUTER REFERENCES:([T2].[_Fld11576_TYPE], [T2].[_Fld11576_RTRef],
13	1	|--Nested Loops(Left Outer Join, OUTER REFERENCES:([T15].[_Fld8362RRef]))
13	1	|--Nested Loops(Left Outer Join, OUTER REFERENCES:([T2].[_Fld11576_TYPE], [T2].[_Fld115
13	1	|--Nested Loops(Left Outer Join, OUTER REFERENCES:([T2].[_Fld11576_TYPE], [T2].[_F
13	1	|--Nested Loops(Left Outer Join, OUTER REFERENCES:([T11].[_Fld7988RRef]))
13	1	|--Nested Loops(Left Outer Join, OUTER REFERENCES:([T2].[_Fld11576_TYPE]
13	1	|--Filter(WHERE:(CONVERT(numeric(27,8),[Expr1019],0)<>CONVERT_IMPLI
0	0	|--Compute Scalar(DEFINE:([Expr1019]=CASE WHEN [Expr1059]=(0)
13	1	|--Stream Aggregate(GROUP BY:([T2].[_Fld11576_RRRef], [Exp
36	1	|--Sort(ORDER BY:([T2].[_Fld11576_RRRef] ASC, [Expr1
36	1	|--Filter(WHERE:(CASE WHEN [Base].[dbo].[_Accum
429646	1	|--Hash Match(Right Outer Join, HASH:([T9]
287528	1	|--Hash Match(Right Outer Join, HASH:
205312	1	|--Index Scan(OBJECT:([Base].[db
287528	1	|--Clustered Index Scan(OBJECT:(
429646	1	|--Hash Match(Right Outer Join, HASH:
118143	1	|--Hash Match(Left Outer Join, H
118143	1	|--Clustered Index Scan(OBJ
205312	1	|--Index Scan(OBJECT:([Base
429646	1	|--Hash Match(Right Outer Join,
32810	1	|--Hash Match(Left Outer Jo
32810	1	|--Clustered Index Sca
205312	1	|--Index Scan(OBJECT:(
429646	1	|--Hash Match(Right Outer J
38025	1	|--Hash Match(Left Out
38025	1	|--Clustered Inde

(изображение можно увеличить без потери качества)

Анализ такого плана займет длительное время без каких-либо гарантий на успех. В данном случае гораздо легче проанализировать текст запроса.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Сам запрос выглядит следующим образом:

```
ВЫБРАТЬ
	ДенежныеСредстваКПоступлениюБезналичныеОбороты.Документ.Контрагент.Партнер,
	МЕСЯЦ(ДенежныеСредстваКПоступлениюБезналичныеОбороты.Период) КАК Месяц,
	ГОД(ДенежныеСредстваКПоступлениюБезналичныеОбороты.Период) КАК Год,
	СУММА(ДенежныеСредстваКПоступлениюБезналичныеОбороты.СуммаПриход) КАК
СуммаПриход
ИЗ
	РегистрНакопления.ДенежныеСредстваКПоступлениюБезналичные.Обороты(,, Месяц,
Документ.Контрагент.Партнер = &Партнер) КАК
ДенежныеСредстваКПоступлениюБезналичныеОбороты
СГРУППИРОВАТЬ ПО
	ДенежныеСредстваКПоступлениюБезналичныеОбороты.Документ.Контрагент.Партнер,
	МЕСЯЦ(ДенежныеСредстваКПоступлениюБезналичныеОбороты.Период),
	ГОД(ДенежныеСредстваКПоступлениюБезналичныеОбороты.Период)
```

Опытный разработчик сразу поймет в чем дело: в параметрах виртуальной таблицы идет обращение через 2 точки, ситуация также усугубляется тем, что измерение *Документ* имеет составной тип.

В результате на анализ ушло всего несколько минут, и причина медленной работы стала очевидной. Анализ плана того же запроса занял бы гораздо больше времени, и именно поэтому оптимизация всегда начинается с анализа текста запроса.

Причин неоптимальной работы запросов может быть великое множество, но основные причины давно известны и достаточно хорошо изучены. Ниже приведен список причин, встречающихся наиболее часто:

- Невыполнение регламентных заданий на сервере СУБД
- Соединение с подзапросами
- Не используются индексы
- Неосторожное использование сортировки
- Подзапрос в условии соединения
- Неосторожное использование полей составного типа
- Фильтрация виртуальных таблиц без использования параметров
- Запросы в цикле.

Далее будет подробно рассмотрена каждая из вышеперечисленных причин.

Приведенные причины являются основными, но не единственными. Некоторые другие возможные причины замедления запросов также будут рассмотрены в курсе.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Невыполнение регламентных операций

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Признаки неактуальной статистики
- Как неактуальная статистика может повлиять на план
- Действительно ли статистика критична только для сложных запросов.

Видеоурок. Соединение с подзапросами

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Почему соединение с подзапросом может привести к замедлению
- Особенности работы разных СУБД с подобными запросами.

Видеоурок. Использование временных таблиц

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как с помощью временных таблиц избавиться от соединения с подзапросами
- Стоит ли везде и всегда использовать временные таблицы
- Как ориентироваться на стоимость текстового плана.

Видеоурок. Индексация временных таблиц

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Всегда ли используется индекс у временной таблицы
- Отличие индексов временных таблиц в 8.2 и в 8.3.

Видеоурок. Соединение с виртуальными таблицами

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- Почему соединение с виртуальными таблицами может привести к замедлению
- Как оптимизировать такие запросы

Видеоурок. Подзапрос в условии соединения

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Почему подзапрос в условии соединения лучше не использовать
- Способы оптимизации таких запросов.

Видеоурок. Подзапросы в условиях и вложенные подзапросы

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каких случаях подзапросы не являются проблемой
- Можно ли использовать вложенные подзапросы
- Можно ли использовать подзапросы в параметрах виртуальной таблицы.

Видеоурок. Несоответствие индексов и условий

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Типичные ошибки при написании условий в запросах
- Правила использования составного индекса.

Видеоурок. Несоответствие индексов и условий. Регистр накопления

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Особенности оптимизации для регистра накопления
- Правило выбора первого измерения для регистра накопления.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Условия, не позволяющие использовать индекс. ИЛИ

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Когда условие ИЛИ не является проблемой
- В каких случаях ИЛИ не позволяет использовать индекс
- Как оптимизировать запросы с условием ИЛИ.

Видеоурок. Условия, не позволяющие использовать индекс. Вычисление

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Почему вычисление над полями условий замедляет запрос
- Как можно решить данную ситуацию.

Видеоурок. Условия, не позволяющие использовать индекс. НЕ В

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Почему нежелательно использовать в запросах данное условие
- Как оптимизировать запросы с условием НЕ В.

Видеоурок. Условия, не позволяющие использовать индекс. Функции

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Можно ли применять функции к полям условий и соединений
- Чем заменить использование функций работы с датами
- Чем заменить функции работы со строками

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

- Как ускорить выполнение, если нет возможности заменить функцию
- Можно ли использовать условие ПОДОБНО.

Видеоурок. Условия, не позволяющие использовать индекс. Вхождение полей в разные списки

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каком случае условия вхождения в списки могут замедлить запрос
- Как оптимизировать такие запросы.

Видеоурок. Условия, не позволяющие использовать индекс. Вхождение в список с большим числом элементов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Почему при большом числе элементов в списке возможно замедление запроса
- Как оптимизировать такие запросы.

Видеоурок. Непокрывающие индексы

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- К чему приводит наличие непокрывающего индекса
- От чего зависит итоговый план при наличии непокрывающего индекса.

Видеоурок. Покрывающий индекс с дополнительным упорядочиванием

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- К чему приводит наличие непокрывающего индекса
- Как создать покрывающий индекс с помощью дополнительного упорядочивания
- От чего зависит план запроса при наличии непокрывающего индекса.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Селективность

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что такое селективность, и как она влияет на план запроса
- Когда нужен индекс по неселективным полям
- Влияние селективности на порядок полей в индексе.

Видеоурок. Определение недостающих индексов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Различные способы определения недостающих индексов
- Наиболее эффективный способ определения недостающих индексов.

Видеоурок. Сортировка по полю, которое не входит в индекс

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Почему такой вид сортировки снижает скорость работы запроса
- Симптомы проблемы в плане запроса
- Каким образом ускорить сортировку.

Занятие 21

Видеоурок. ВЫБРАТЬ ПЕРВЫЕ и сортировка

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каких случаях совместное использование конструкций ВЫБРАТЬ ПЕРВЫЕ и УПОРЯДОЧИТЬ приводит к проблемам
- Варианты решения данной проблемы.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Когда сортировка не влияет на производительность

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каких случаях сортировка не оказывает влияния на производительность
- Особенности сортировки ссылочных значений.

Видеоурок. Внутреннее устройство полей составного типа

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как реализованы поля составного типа на уровне СУБД
- Сколько колонок создается для полей составного типа, и что в них хранится.

Видеоурок. Неявное образование полей составного типа

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Когда составной тип может образовываться неявно.

Видеоурок. Обращение к реквизитам поля составного типа

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Почему не рекомендуется получать данные через точку от полей составного типа
- Как преобразуется запрос при обращении к реквизитам поля составного типа
- Как можно оптимизировать такие запросы.

Видеоурок. Смешивание простых и ссылочных типов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Когда смешивание типов приводит к проблемам
- Как оптимизировать такие запросы.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Минимум и максимум от полей составного типа

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Приводят ли данные операции к сканированию индекса по полю составного типа
- Способы оптимизации таких запросов в зависимости от типа данных.

Видеоурок. Составной тип и RLS

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Может ли применение RLS к полям составного типа замедлить запрос
- Способы оптимизации таких запросов.

Видеоурок. Когда составной тип не приводит к проблемам

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каких случаях использование составного типа не приводит к проблемам.

Видеоурок. Определяемые типы

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Каково влияние определяемых типов на производительность.

Видеоурок. Фильтрация виртуальных таблиц

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Почему нужно использовать параметры виртуальных таблиц для фильтрации
- Будет ли запрос всегда работать неоптимально, если не использовать параметры для фильтрации виртуальных таблиц.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Запрос в цикле. Метод НайтиПо

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Почему даже быстрые запросы в цикле – это плохо
- Как оптимизировать большую часть запросов в цикле.

Видеоурок. Запрос в цикле. Обращение к реквизитам

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Почему не рекомендуется обращаться к реквизитам элемента в цикле
- В каком случае обращение к реквизитам не является проблемой.

Видеоурок. Запрос в цикле. Вывод ссылки на экран

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Почему при отображении ссылки появляется запрос в цикле
- Как оптимизировать такие запросы.

Видеоурок. Запрос в цикле. Коррелированные запросы

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что такое коррелированный запрос
- Каким образом выполняются такие запросы
- Как оптимизировать коррелированные запросы.

Видеоурок. Запрос в цикле. Намеренное использование

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каком случае можно намеренно использовать запрос в цикле.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 22

Другие возможные причины медленной работы запросов

Видеоурок. Большой объем выборки данных

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Причины получения большого объема выборки данных
- Как можно исправить такую ситуацию.

Видеоурок. Обращение к полю через несколько точек

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- К чему приводит злоупотребление разыменованием полей.

Видеоурок. Получение ссылки от поля ссылочного типа

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Влияет ли на производительность получение поля *Ссылка* от полей ссылочного типа
- Как это влияет на текст SQL запроса.

Видеоурок. ОБЪЕДИНИТЬ и ОБЪЕДИНИТЬ ВСЕ

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Есть ли разница в производительности при использовании этих команд
- Какую из команд предпочтительнее использовать, если результат запроса будет одинаковым.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Запросы с RLS

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как понять, что запрос выполняется с RLS
- Способы оптимизации таких запросов

Видеоурок. Универсальные запросы

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Стоит ли писать универсальные запросы на все случаи жизни
- Могут ли такие запросы привести к деградации производительности.

Полезная информация по запросам

Видеоурок. Особенности работы с виртуальной таблицей остатков

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как параметры виртуальной таблицы остатков влияют на скорость выполнения запроса.

Видеоурок. Особенности работы с виртуальной таблицей среза первых/последних

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как в 8.3 ускорить получение данных из виртуальной таблицы среза первых/последних
- Какие есть ограничения при использовании таблиц итогов регистров сведений.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Особенности выполнения пакетных запросов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Особенности выполнения пакетных запросов
- Как понять, какая часть пакетного запроса выполняется медленно.

Видеоурок. Особенности объектного чтения данных

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Каким образом происходит объектное чтение данных
- Есть ли разница при чтении объектов с табличной частью и без нее.

Видеоурок. Анализ больших запросов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Приемы анализа больших запросов.

Видеоурок. Анализ запросов СКД

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Приемы анализа запросов СКД.

Видеоурок. Метод запроса Выполнить

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Принцип работы метода *Выполнить*
- В какой момент данные с сервера СУБД передаются серверу 1С
- Где сервер 1С хранит промежуточный результат запроса.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Метод запроса Выбрать

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Принцип работы метода *Выбрать*
- Чем метод *Выбрать* отличается от метода *Выгрузить*.

Видеоурок. Метод запроса Выгрузить

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Принцип работы метода *Выгрузить*
- Особенности работы метода на 32-разрядном сервере 1С.

Видеоурок. Как выполнить запрос «как в первый раз»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как очистить кэш данных
- Как очистить кэш планов для определенной базы и для всего сервера СУБД.

Видеоурок. Хранение временных таблиц

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В какой базе хранятся временные таблицы
- Где физически располагаются временные таблицы
- На что влияет использование больших временных таблиц.

Видеоурок. Сжатие базы TempDB

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Зачем может потребоваться сжатие базы TempDB

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

- Какой командой можно выполнить сжатие базы
- Можно ли выполнить сжатие базы в рабочее время.

Видеоурок. Кто сейчас выполняет долгий запрос

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как с помощью консоли узнать, кто сейчас выполняет долгий запрос.

Видеоурок. Что не влияет на производительность запроса

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Какие вычисления не оказывают влияния на скорость работы запроса
- Оказывает ли влияние на скорость работы запроса порядок таблиц и условий в запросе.

Видеоурок. Рекомендации по написанию запросов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Основные рекомендации по написанию запросов.

Занятие 23

Анализ и оптимизация медленных запросов

Видеоурок. Анализ с помощью сервиса. Запросы без контекста

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- Как с помощью сервиса анализировать неоптимальные запросы
- В каком случае может не отображаться контекст
- Как при отсутствии контекста узнать, откуда выполнялся код.

Видеоурок. Анализ с помощью сервиса. Запросы с контекстом

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример анализа запроса с контекстом.

Видеоурок. Анализ с помощью ЦУП. Настройка и сбор данных

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как настроить показатели для сбора данных.

Видеоурок. Анализ с помощью ЦУП. Анализ

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каких разрезах можно провести анализ проблем производительности с помощью ЦУП
- Как в ЦУП группировать одинаковые запросы
- Просмотр текстов и планов запросов в ЦУП.

Видеоурок. Анализ с помощью SQL Profiler

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как анализировать запросы через SQL Profiler
- На какие свойства событий необходимо обращать особое внимание.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Сохранение трассировки в таблицу

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как можно сохранить трассировку в таблицу базы данных
- Каким образом можно использовать эту таблицу.

Видеоурок. Шаблоны SQL Profiler

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как ускорить настройку SQL Profiler
- Разные способы создания шаблонов.

Видеоурок. Сопоставление плана и текста запроса в SQL Profiler

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как точно определить план для текста запроса.

Видеоурок. Сохранение трассировки в файл

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как сохранить трассировку в отдельный файл
- Нужен ли установленный MS SQL Server для открытия этого файла.

Видеоурок. Ошибка Trace skipped records

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каком случае можно встретить в SQL Profiler данную ошибку
- Как исправить эту ошибку.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Различные события SQL Profiler

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- События, которые могут представлять интерес, но обычно не используются.

Видеоурок. Анализ запроса с помощью технологического журнала

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как найти медленные запросы с помощью технологического журнала
- В каких случаях для анализа медленных запросов используется технологический журнал.

Видеоурок. Анализ неоптимального запроса. Пример 1

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Видеоурок. Анализ неоптимального запроса. Пример 2

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Видеоурок. Анализ неоптимального запроса. Пример 3

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Видеоурок. Анализ неоптимального запроса. Пример 4

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Анализ неоптимального запроса. Пример 5

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Неоптимальная работа запросов. Итоги

Неоптимальные запросы являются наиболее популярной проблемой производительности. Они также могут являться первопричиной других проблем, поэтому оптимизация в большинстве случаев начинается именно с ускорения запросов. Следует учитывать, что разные запросы вносят разный вклад в снижение производительности. В системе может быть всего несколько запросов, которые создают 80% проблем.

Чтобы найти медленные запросы, необходимо использовать специальные инструменты, но сами по себе эти инструменты бесполезны. Собрав данные о запросах, необходимо уметь их интерпретировать, чтобы понять причину неоптимальности.

Для ускорения запросов необходимо знать принципы работы оптимизатора, уметь читать и понимать план запроса, понимать, как выполняются операторы, а главное – почему план получился именно таким.

План запроса состоит из операторов, каждый из которых выполняет определенную роль. В курсе рассмотрены лишь основные операторы, описание всех операторов можно найти здесь: https://technet.microsoft.com/ru-ru/library/ms191158(v=sql.105).aspx

Зачастую причину неоптимальности запроса можно понять по тексту запроса.

Основные причины медленной работы запроса достаточно хорошо изучены, в первую очередь необходимо искать именно их. Другие причины встречаются реже, но, зная принципы работы оптимизатора и умея читать план запроса, можно разобраться в любом запросе и найти причину его замедления.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Ожидания на блокировках

Занятие 24

Данный раздел посвящен анализу проблем избыточных блокировок, которые практически неминуемо есть в любой системе, работающей в многопользовательском режиме. Важно понимать, что блокировки сами по себе не являются проблемой, это механизм защиты данных. Проблемой являются излишние блокировки.

Для решения проблем с излишними блокировками требуются обширные теоретические знания о транзакциях, типах и принципах работы блокировок, особенностях хранения метаданных на уровне СУБД и многом другом. Важно не просто знать, когда и какие блокировки накладываются, важнее понимать, почему происходит именно так, а не иначе.

Без понимания теории анализ проблем с блокировками может быть очень затруднителен и привести к нарушению в логике работы системы, например, к возможности появления отрицательных остатков.

Транзакции

Понятие транзакции

Транзакция – это набор действий, который может быть либо полностью выполнен, либо полностью не выполнен. Если хотя бы одно из действий не выполняется, тогда вся транзакция (все предыдущие действия) отменяется, т.е. транзакция имеет смысл, только если выполнены полностью все действия, в нее входящие. Транзакция работает по принципу «либо все, либо ничего».

Типичный пример, когда транзакция просто необходима, – это перемещение товара с одного склада на другой. Этот процесс можно представить как следующий набор действий:

- Прочитать остаток на первом складе
- Списать товар с первого склада
- Сохранить новый остаток на первом складе
- Увеличить остаток на втором складе
- Сохранить новый остаток на втором складе.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Если хотя бы один из этапов завершится с ошибкой, логика работы системы будет нарушена, поэтому необходимо, чтобы либо все операции были выполнены, либо все не выполнены.

Транзакционные блокировки могут существовать только в рамках транзакции, при этом объектные блокировки от транзакций никак не зависят. Подробнее тема блокировок будет рассмотрена чуть позже.

Свойства транзакции

Наиболее распространенным набором требований к транзакциям считается набор ACID (Atomicity, Consistency, Isolation, Durability). Данный именной набор и определяет основные свойства транзакции.

- Атомарность (Atomicity)
- Согласованность (Consistency)
- Изолированность (Isolation)
- Надежность (Durability).

Разберем каждое из свойств подробнее.

Атомарность или неделимость – гарантирует, что в транзакции будут выполнены либо все действия, либо ни одного, т.е. транзакция не может быть зафиксирована частично. Благодаря данному свойству нельзя, например, при перемещении списать товар с одного склада и не оприходовать на другой. Если одно из действий в транзакции завершится с ошибкой, то произойдет «откат» и система вернется в исходное состояние.

Согласованность – гарантирует, что после окончания транзакции система останется в согласованном состоянии.

Данное свойство является довольно спорным и неоднозначным, т.к. во многом согласованность должны обеспечить программисты, пишущие прикладной код. Если при перемещении списать товар с одного склада, но не написать код, который делает оприходование на другой склад, то данные будут не согласованы. Описанная в примере несогласованность – это не вина СУБД, это вина разработчика.

В данном случае свойство согласованности транзакции должно обеспечить обработку кода именно так, как предполагал разработчик.

Свойство согласованности может не соблюдаться внутри транзакции, пока она выполняется, главное, чтобы оно соблюдалось в момент завершения транзакции. Данное свойство также является необходимым условием для обеспечения надежности.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изолированность – гарантирует, что пока одна транзакция выполняется, другие транзакции не могут повлиять на ее результат. Например, благодаря изолированности невозможна ситуация, когда в промежутке между списанием с первого склада и оприходованием на второй, другая транзакция спишет весь товар с первого склада.

Свойство изолированности полностью соблюдается только для уровня изоляции Serializable, т.к. только с данным уровнем невозможна проблема фантомного чтения. Об уровнях изоляции транзакций и проблемах параллельной работы подробнее будет рассказано в отдельном разделе.

Надежность – гарантирует, что сбой после успешно завершенной транзакции не отменяет результат этой транзакции. Например, если после перемещения товара, которое завершилось успешно, неожиданно выключили электропитание, то после восстановления работы системы перемещенный товар будет находиться на новом складе. Другими словами, если транзакция успешно завершена, то, что бы потом ни произошло, результат транзакции не изменится.

Данное свойство в определенной степени зависит от используемого оборудования. Например, сейчас многие диски используют кэширование не только на чтение, но и на запись, и при записи данных они немного «обманывают» систему. СУБД получает подтверждение об успешном выполнении записи, когда данные попали в кэш, а не когда были записаны на диск, таким образом, повышается быстродействие.

Но между записью в кэш и записью на диск есть небольшой промежуток времени, за который теоретически может произойти сбой, и тогда данные не будут записаны, хотя СУБД «думает», что они уже зафиксированы. Для предотвращения таких ситуаций многие современные диски снабжаются аккумуляторными батареями, а сами серверы и сетевое оборудование – системами бесперебойного питания.

Когда транзакция открывается

В MS SQL Server абсолютно любой запрос всегда выполняется в транзакции. Это неявная транзакция, открываемая самой СУБД, при этом неважно, какая версия платформы используется, какой режим блокировки, также неважно, откуда выполняется запрос: из консоли запросов, отчета или из модуля проведения документа. В том, что любой запрос выполняется в транзакции, легко убедиться, используя SQL Profiler. Если у события плана запроса заполнено свойство TransactionID, значит, запрос с этим планом выполнялся в транзакции. Если у нескольких событий одинаковое значение TransactionID, значит, они выполнялись в одной транзакции.

Например, выполним в консоли запрос:

```
ВЫБРАТЬ ПЕРВЫЕ 1
      Тест.Наименование
ИЗ    Справочник.Тест КАК Тест
```


Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Будет получена следующая трассировка:

EventClass	Duration	Reads	Row...	TextData	TransactionID
Trace Start					
Showplan Statistics Profile					1438931
Showplan XML Statistics Profile				<ShowPlanXML xmlns="http://schemas.mi...	1438931
SQL:BatchCompleted	0	0	0	SELECT spid, blocked FROM master.dbo....	
Showplan Statistics Profile					1438934
Showplan XML Statistics Profile				<ShowPlanXML xmlns="http://schemas.mi...	1438934
SQL:BatchCompleted	1	2	1	SELECT TOP 1 T1._Description FROM dbo...	
Trace Stop					

По трассировке видно, что у событий плана запроса одинаковые ID транзакции, но у самого события запроса ID транзакции никогда не отображается. При этом явных событий начала и окончания транзакции в трассировке также нет, потому что данную транзакцию открыла сама СУБД, а не платформа 1С.

При этом само приложение, в нашем случае это 1С, может посылать команды в СУБД для открытия транзакции. Если транзакция открыта по инициативе платформы, то в трассировке будут отображаться моменты начала и завершения транзакции.

Например, если обратиться в коде к реквизиту объекта, у которого есть табличная часть (например, Док.Дата), то платформа даст СУБД команду на открытие транзакции, и в результате будет получена следующая трассировка.

EventClass	Duration	Reads	Ro...	TextData	Transaction...
Trace Start					
Showplan Statistics Pro...					1523754
Showplan XML Statistics...				<ShowPlanXML xmlns="http://schemas....	1523754
SQL:BatchCompleted	0	0	0	SELECT spid, blocked FROM master.db...	
SQL:BatchCompleted	0	0	0	BEGIN TRANSACTION	1523757
Showplan Statistics Pro...					1523757
Showplan XML Statistics...				<ShowPlanXML xmlns="http://schemas....	1523757
RPC:Completed	7	4	1	exec sp_executesql N'SELECT T1._Ver...	1523757
Showplan Statistics Pro...					1523757
Showplan XML Statistics...				<ShowPlanXML xmlns="http://schemas....	1523757
RPC:Completed	1	6	0	exec sp_executesql N'SELECT T2._IDR...	1523757
Showplan Statistics Pro...					1523757
Showplan XML Statistics...				<ShowPlanXML xmlns="http://schemas....	1523757
RPC:Completed	3	12	0	exec sp_executesql N'SELECT T3._Lin...	1523757
SQL:BatchCompleted	0	0	0	COMMIT TRANSACTION	
Trace Pause					

В трассировке будут видны явные команды платформы на открытие и фиксацию транзакции, при этом ID транзакции для всех событий в рамках этой транзакции будет одинаковым.

При выполнении запроса транзакция открывается всегда, но когда говорят, что действие выполняется в транзакции, то обычно имеют ввиду транзакцию, которая открывается именно по инициативе платформы.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Если говорят, что действие выполняется вне транзакции, то имеют в виду: «вне транзакции, порожденной платформой». В дальнейшем в материалах курса будем придерживаться этой терминологии.

Транзакция по инициативе платформы может быть открыта как явно (вручную), так и неявно (автоматически).

Рассмотрим ситуации, когда транзакция открывается платформой автоматически.

- **При изменении данных.** Например, при записи, удалении, проведении или обновлении транзакция будет открыта автоматически. Элемент не может быть изменен вне транзакции. Важно помнить, что транзакция открывается при возникновении события на сервере, а не на клиенте
- **При чтении наборов записей регистра методом Прочитать()**
- **При обращении через точку к реквизитам объектов, которые содержат табличную часть.** Например, если написать Дата = ДокументСсылка.Дата
- **При выполнении метода ПолучитьОбъект() или Прочитать() для объектов, которые содержат табличную часть**
- **При обращении к виртуальной таблице «ОстаткиИОбороты» с указанием периодичности, отличной от значения «Период»**. В данном случае есть несколько моментов, которые необходимо прояснить.
    - Если периодичность не указывать или указать значение «Период», тогда транзакция не будет открыта
    - В транзакции будет выполнен не основной запрос, который обращается к таблицам регистров, а лишь заполнение вспомогательной временной таблицы, которая будет автоматически создана для выполнения данного запроса. Основной запрос, который обращается к физическим таблицам, все равно будет выполнен вне транзакции, следовательно, не будет блокировать данные.

Разработчик никак не может повлиять на транзакции, которые платформа открывает неявно.

Также платформа позволяет открывать транзакции явным образом с помощью команды *НачатьТранзакцию()*. Это может потребоваться в следующих случаях:

- Необходимо объединить несколько зависимых действий в одну транзакцию
- Необходимо прочитать данные и быть уверенным, что, пока идет чтение, эти данные не изменятся
- Нужно быстро записать большой объем данных. Гораздо быстрее в одной транзакции записать 1 000 элементов, чем 1 000 раз открыть транзакцию, записать элемент и закрыть транзакцию.

Чтобы понять, выполняется предопределенная процедура в транзакции или нет, необходимо вызывать справку в синтаксис-помощнике по этой процедуре. В описании будет указано, выполняется ли процедура в транзакции.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

СправочникОбъект.<Имя справочника> (CatalogObject.<Имя справочника>).ПередЗаписью (BeforeWrite)

Синтаксис:

ПередЗаписью(<Отказ>)

Параметры:

<Отказ>

Тип: Булево.

Признак отказа от записи элемента. Если в теле процедуры-обработчика установить данному параметру значение Истина, запись элемента выполнена не будет и будет вызвано исключение.

Значение по умолчанию: Ложь.

Описание:

Возникает перед выполнением записи элемента справочника. Процедура-обработчик вызывается после начала транзакции записи, но до начала записи элемента справочника.

В процессе отладки также можно определить, что код выполняется внутри транзакции, для этого следует использовать команду *ТранзакцияАктивна()*.

Вложенные транзакции

Транзакции в 1С могут быть вложенными, но работать они будут все равно как одна транзакция. Обычно вложенные транзакции открываются неявно, например, когда при проведении документа происходит еще и запись других объектов, например, регистров.

При отмене вложенной транзакции внешняя транзакция также будет отменена, поэтому можно считать, что платформа позволяет написать вложенную транзакцию, но на уровне СУБД все равно будет только та транзакция, которая была открыта первой.

Можно написать следующий код:

```
НачатьТранзакцию();
      Процедура1();
      НачатьТранзакцию();
            Процедура2();
      ОтменитьТранзакцию();
ЗафиксироватьТранзакцию();
```

Данный код будет работать без ошибок, платформа поддерживает такое написание, но активной все равно будет только первая транзакция. При отмене транзакции в 5 строке будет отменена как процедура 2, так и процедура 1.

Видеоурок. Обработка исключений в транзакциях. Пример 1

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Обработка исключений в транзакциях. Пример 2

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Видеоурок. Обработка исключений в транзакциях. Пример 3

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Видеоурок. Обработка исключений в транзакциях. Пример 4

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Видеоурок. Обработка исключений в транзакциях. Пример 5

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Видеоурок. Обработка исключений в транзакциях. Пример 6

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Занятие 25

Блокировки

Понятие блокировки

Блокировка – это информация о том, что ресурс захвачен для выполнения какого-либо действия. Сами по себе блокировки не являются проблемой, они необходимы в многопользовательских системах, иначе будет нарушена логика работы приложения.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Если бы в системе не было блокировок, это было бы аналогично тому, как 2 пользователя пытаются одновременно внести данные в одну ячейку Excel. Сохранится только информация, внесенная последним пользователем, при этом каждый из них подумает, что именно его данные были сохранены. Механизм блокировок используется во всех СУБД, поддерживаемых платформой 1С, в том числе Oracle и PostgreSQL.

Блокировки могут быть как необходимыми, так и избыточными. Необходимые блокировки нужны для защиты данных, избыточные блокировки для защиты данных не используются. Оптимизация заключается в том, чтобы отличить избыточные блокировки от необходимых и устранить избыточные.

Блокировка может быть избыточной в следующих категориях:

- **По данным.** Блокируются излишние данные, которые можно не блокировать
- **По времени.** Блокируются необходимые данные, но длительность этой блокировки слишком большая. Чем короче по времени транзакция, тем лучше. Ни в коем случае нельзя вызывать модальные окна внутри транзакции, иначе она будет длиться, пока модальное окно не закроется
- **По времени и по данным.** Комбинация двух вышеописанных категорий. Блокируются лишние данные на слишком долгий срок.

Платформа работает как с объектными, так и с транзакционными блокировками. Оба типа блокировок предназначены для обеспечения многопользовательской работы, но имеют различный смысл. Далее будет подробно разобрана работа блокировок различных типов и видов.

Объектные блокировки

Объектная блокировка – это блокировка объектных данных на уровне платформы 1С.

Такая блокировка нужна, чтобы оповещать пользователей о том, что объектная сущность (документ, бизнес-процесс, справочник и т.д.) была захвачена другим пользователем. К необъектным данным (регистрам, последовательностям, константам и пр.) объектная блокировка неприменима. Пользователи могут одновременно открыть один документ, но благодаря объектным блокировкам записать измененный документ сможет только один из пользователей, у других возникнет ошибка.

Объектные блокировки не связаны с транзакциями и вообще никак не зависят от СУБД, они существуют исключительно на уровне платформы 1С.

В подавляющем большинстве случаев объектные блокировки используются при работе с интерфейсом и, как правило, не приводят к проблемам производительности.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Такие блокировки никогда не бывают избыточными по данным, т.к. они всегда блокируют какой-то один конкретный объект. Но такая блокировка может быть избыточной по времени. Например, пользователь открыл документ, изменил в нем какой-либо реквизит и ушел на обед, оставив документ открытым и не сохранив его. В этом случае другой пользователь не сможет этот объект изменить и будет ждать, пока первый пользователь не сохранит или отменит свои изменения. При этом открытый документ можно изменить или даже удалить программно, о чем будет рассказано в соответствующем видеоуроке.

Есть два вида объектных блокировок:

- **Пессимистическая**. Данная блокировка гарантирует, что если объект изменился, но еще не записан, то никто другой не сможет его изменить
- **Оптимистическая**. Данная блокировка гарантирует, что будет записана именно та версия данных, которую мы прочитали, и пользователь не сможет затереть изменения другого пользователя.

Ниже подробно рассмотрены механизмы работы объектных блокировок.

Видеоурок. Пессимистическая объектная блокировка

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Принцип работы пессимистической блокировки
- Когда устанавливается пессимистическая блокировка.

Видеоурок. Оптимистическая объектная блокировка

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Принцип работы оптимистической блокировки
- Когда устанавливается оптимистическая блокировка.

Видеоурок. Объектные блокировки и защита данных в СУБД

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каком случае объектные блокировки защищают данные, а в каком – нет
- Можно ли удалить объект, пока он редактируется другим пользователем.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 26

Транзакционные блокировки

Транзакционные блокировки – это блокировки, которые могут существовать только внутри транзакции, они предназначены для защиты данных.

Транзакционные блокировки делятся на два вида:

- **Блокировки СУБД.** Присутствуют в системе всегда, независимо от того, какой режим блокировки используется в конфигурации. Такие блокировки работают на уровне сервера баз данных. Блокировки есть у всех СУБД, даже «версионники» (Oracle и PostgreSQL) используют блокировки при изменении данных
- **Управляемые блокировки 1С**. Присутствуют в системе только в том случае, если транзакция работает в управляемом режиме блокировки данных. Такие блокировки существуют только на сервере приложений и обслуживаются собственным менеджером блокировок сервера 1С.

Транзакционные блокировки не могут существовать вне транзакции, это относится как к блокировкам СУБД, так и к блокировкам 1С. Следовательно, если вы выполняете запрос вне транзакции, тогда и блокировок он никаких не накладывает. Например, такие действия как выполнение отчетов, открытие форм списков документов и т.д. не блокируют данные, т.к. выполняются вне транзакции (если транзакция явно не была прописана в коде). Но следует помнить, что иногда транзакция открывается автоматически, например, при обращении в коде к реквизиту, который имеет составной тип.

Когда говорят об оптимизации излишних блокировок, имеют в виду именно транзакционные блокировки, далее речь пойдет именно о них.

Типы блокировок СУБД

Для выполнения различных задач существуют разные типы блокировок, которые защищают данные в разной степени.

Например, в MS SQL Server есть 3 основных типа/режима блокировок:

- **S (shared)** – разделяемая блокировка, устанавливается при чтении данных. В 1С версии 8.2 и ниже при выполнении запросов в транзакции будет установлена S блокировка. Платформой 8.3 при чтении S блокировка не ставится (об этом будет рассказано чуть позже)

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

- **X (exclusive)** – эксклюзивная блокировка. Данную блокировку также называют монопольной или исключительной. Монопольная блокировка ставится при любом изменении данных. В 1С X блокировки ставятся при использовании методов, приводящих к модификации данных, таких как Записать(), Провести(), Удалить() и т.д.
- **U (update)** – блокировка обновления. Данная блокировка используется в том случае, если данные необходимо сначала прочитать, а потом записать. Таким образом мы говорим другим транзакциям, что прочитанные данные потом будут изменены. Когда данные будут записываться, U блокировка будет преобразована в X. U блокировка нужна для предотвращения возможной взаимной блокировки. В 1С U блокировка возникает в автоматическом режиме блокировки данных, если в тексте запроса используется опция «ДЛЯ ИЗМЕНЕНИЯ». В тексте SQL запроса опция «ДЛЯ ИЗМЕНЕНИЯ» преобразуется в хинт (подсказку) UPDLOCK, таким образом СУБД понимает, что надо поставить U блокировку.

Всего в СУБД MS SQL сервер используется 22 типа различных блокировок.

Совместимость блокировок СУБД

Различные типы блокировок могут по-разному взаимодействовать между собой. Например, одни и те же данные одновременно может читать неограниченное число пользователей, но менять эти данные в определенный момент времени может только кто-то один.

Ниже приведена таблица совместимости блокировок СУБД.

	S	U	X
S	+	+	-
U	+	-	-
X	-	-	-

Данная таблица построена по простому принципу: читать одни и те же данные можно параллельно, но менять их можно только последовательно.

X блокировка не совместима ни с чем: пока данные кто-то меняет, другим их нельзя ни изменить, ни прочитать.

S блокировка совместима только с другими S блокировками и с U блокировкой. Читать данные можно параллельно.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

U блокировка совместима только с S блокировкой, т.к. U блокировка – это то же чтение, с той лишь разницей, что потом данные будут изменены. U блокировки между собой не совместимы, т.к. нельзя одновременно захватить данные, с правом дальнейшей модификации.

При этом порядок установки блокировок не играет никакой роли. Неважно, что было установлено сначала – S или X блокировки, они все равно будут несовместимы.

Блокировки будут всегда совместимы, если выполняется **любое** из двух условий:

- **Блокировки установила одна транзакция.** Внутри одной транзакции все блокировки совместимы, например, можно сначала поставить S блокировку на таблицу, а потом на ту же таблицу наложить X блокировку
- **Блокируются разные ресурсы.** Если идет блокировка разных строк или таблиц, то мы работаем с независимыми блоками данных. В этом случае тип блокировки роли не играет, т.к. конфликта в любом случае не будет.

Блокировки будут несовместимы, если выполняются **все** перечисленные ниже условия:

- **Блокировки установлены разными транзакциями**
- **Блокируется один и тот же ресурс**
- **Блокировки имеют несовместимые типы.**

Реализация блокировок в СУБД

Все блокировки в СУБД обрабатывает специальный механизм, который называется **менеджер блокировок**. Он хранит все блокировки, отслеживает их совместимость, организует очередь блокировок и т.д.

Блокировки в СУБД реализованы в виде служебной таблицы, которая хранится в оперативной памяти.

Очень упрощенно эту таблицу можно представить следующим образом.

Соединение	ID ресурса	Тип блокировки	Состояние блокировки
54	1000099	X	GRANT
77	3000088	S	GRANT
54	3000088	X	WAIT
21	3000088	S	GRANT

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Можно посмотреть примерный состав этой таблицы, используя системное представление sys.dm_tran_locks.

По третьей строке таблицы из нашего примера видно, что при попытке изменить ресурс, который уже занят для чтения, идет ожидание на блокировке. В то же время, установка блокировки совместимой с уже имеющейся, проходит успешно (4-я строка)

Блокировки намерения

Помимо блокировки самих данных в MS SQL существуют так называемые блокировки намерения.

Блокировка с намерением ставится на более высоком уровне, чтобы дать понять остальным транзакциям, что есть блокировка на более низком уровне. Например, при установке S блокировки на одну строку таблицы, будет также установлена разделяемая блокировка намерения IS на страницу и таблицу. Такие блокировки не ставятся на строку, только на страницу или таблицу.

Блокировки намерения обозначаются с буквой I в начале. Например, IX – это блокировка намерения обновления, она ставится на таблицу и страницу в том случае, если на строку накладывается X блокировка. IS – это разделяемая блокировка намерения, ставится на страницу и таблицу в том случае, если на строку наложена S блокировка.

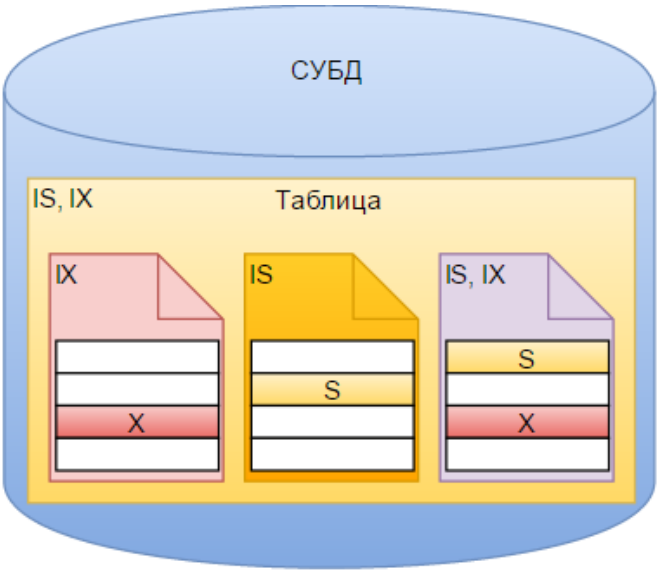
Блокировки намерения повышают скорость работы с блокировками и не позволяют захватить ресурс высокого уровня, пока есть блокировка на низком уровне.

Например, есть таблица с 1 млн. строк. Одна транзакция заблокировала последнюю строку таблицы S блокировкой, а другая хочет заблокировать всю таблицу X блокировкой. Если бы не было блокировок намерения, то вторая транзакция стала бы блокировать все строки большой таблицы, пока не дошла до последней строки и только тогда стала бы в очередь. Это было бы слишком долго и накладно для сервера. Благодаря блокировкам намерения вторая транзакция, перед тем как блокировать таблицу, смотрит, есть ли на этой таблице блокировка намерения, и, если есть, и она не совместима, тогда вторая транзакция сразу становится в очередь.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



Например, если транзакция меняет одну строку таблицы, то будет наложена IX блокировка на таблицу, IX блокировка на страницу и X блокировка на изменяемую строку.

Блокировки намерения почти всегда совместимы друг с другом, ничто не мешает читать одну строку на одной странице, и менять другую строку на той же странице. В таком случае на странице будет установлено 2 блокировки намерения IS и IX.

Полную таблицу совместимости блокировок, включая блокировки намерения, можно посмотреть здесь: https://technet.microsoft.com/ru-ru/library/ms186396(v=sql.105).aspx

Гранулярность и эскалация блокировок

Гранулярность блокировки – это область блокируемых данных. Например: строка, страница или таблица. В большинстве случаев гранулярность блокировки равна строке, а если говорить точнее, то блокируется ключ индекса. Если индекс кластерный, то это и есть вся строка таблицы т.е. будут заблокированы все поля. Если индекс некластерный, то блокируется только ключ индекса, т.е. только те поля которые входят в состав индекса. Далее для упрощения будем говорить, что блокируется просто строка таблицы.

Чем больше блокировок в системе, тем больше объем служебной таблицы блокировок, тем больше требуется оперативной памяти, тем сложнее серверу СУБД все эти блокировки обслуживать. При большом числе блокировок в таблице сервер может решить, что проще заблокировать всю таблицу и хранить в таблице блокировок 1 строку. Этот процесс называется эскалацией блокировки.

Эскалация – это укрупнение гранулярности блокировки.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Эскалация предназначена для экономии аппаратных ресурсов, в первую очередь – оперативной памяти, а также для повышения скорости обработки блокировок. Гораздо проще обслужить 1 блокировку таблицы, чем 10 тыс. блокировок строк.

Конечно, при возникновении эскалации снижается параллельность работы, но при этом блокировка будет быстрее обработана и будет затребовано гораздо меньше аппаратных ресурсов.

Эскалация всегда идет сразу до уровня таблицы. Например, требуется заблокировать 30 тыс. строк в одной таблице. Если сервер СУБД принял решение об эскалации, то менеджер блокировок не будет пытаться блокировать страницы, он сразу заблокирует всю таблицу.

На низком уровне эскалация – это преобразование блокировки намерения таблицы в обычную блокировку, например, IX преобразуется в X.

В MS SQL Server при эскалации транзакция попытается заблокировать всю таблицу, но если строки таблицы кем-то заняты, то транзакция будет блокировать в таблице по 1 250 строк и каждый раз после этого пытаться снова блокировать всю таблицу, т.е. сервер будет повторять попытку эскалации через каждые 1 250 блокировок строк.

Эскалация на MS SQL Server происходит при выполнении одного из следующих условий:

- Одна инструкция Transact-SQL запросила более 5 000 блокировок на одну таблицу или индекс. Если при выполнении одного запроса будет наложено 4 000 блокировок на 1 индекс и 4 000 на другой, то эскалации не будет. Эскалация возможна, только если один запрос устанавливает более 5 000 блокировок на один объект. При этом порог в 5 000 означает, что сервер будет рассматривать возможность эскалации, но это не означает, что он обязательно ее сделает.
- Таблица блокировок превышает доступный объем памяти или заданные пороговые значения. Одна блокировка занимает 96 байт в памяти. По умолчанию эскалация происходит, если таблица блокировок займет более 24% памяти, занимаемой MS SQL Server. Впрочем, в документации нет определенности насчет этого процента, например, в другой статье указано, что порог по умолчанию равен 40%. Данный порог можно регулировать, меняя настройку сервера locks, и явно указать, сколько блокировок может храниться в памяти, но лучше оставить это значение по умолчанию.

Эскалация может приводить к избыточным блокировкам. Чтобы этого избежать, желательно не делать запись больших объемов данных, а записывать данные малыми порциями.

При использовании файлового режима работы минимальная гранулярность блокировки – это таблица, то есть, пока один документ проводится, другой пользователь не может записать документ того же типа, даже если данные не пересекаются. В файловом режиме абсолютно неважно, какой режим блокировки установлен в конфигурации – управляемый или автоматический, все равно блокируется вся таблица целиком, т.е. работа с одной таблицей всегда будет только последовательная.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Пример установки X блокировки

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каком случае устанавливается X блокировка, и как долго она длится
- Как быстро посмотреть, какие блокировки установлены сейчас в системе.

Видеоурок. Пример установки S блокировки

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каком случае ставится S блокировка.

Видеоурок. Пример установки U блокировки

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каком случае ставится U блокировка, и как ее намеренно установить
- Особенности установки U блокировки.

Видеоурок. Какие объекты блокируются

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Всегда ли блокируется строка таблицы целиком
- Можно ли пометить элемент на удаление, если он заблокирован S блокировкой.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 27

Уровни изоляции транзакции

Проблемы при параллельной работе

Без блокировок в многопользовательской системе были бы возможны следующие проблемы параллельной работы:

- Потерянное обновление (lost update)
- «Грязное» чтение (dirty read)
- Неповторяющееся чтение (non-repeatable read)
- Фантомное чтение (phantom reads).

Для решения данных проблем используются различные уровни изоляции транзакции.

Уровень изоляции транзакции – это уровень защиты данных, который обеспечивает данная транзакция. Чем выше уровень изоляции, тем сильнее защищены данные от возможных четырех проблем, описанных выше, но тем меньше параллельность работы пользователей. С другой стороны, чем ниже уровень изоляции, тем выше параллельность работы пользователей и общее быстродействие системы, но тем меньше защищены данные.

Для решения проблем параллельной работы существует 4 основных уровня изоляции транзакции, которые определены стандартом SQL-92.

- **Read uncommitted** (чтение неподтвержденных/незафиксированных данных)
- **Read committed** (чтение подтвержденных данных). Используется в MS SQL и Oracle по умолчанию
- **Repeatable read** (повторяющееся чтение)
- **Serializable** (упорядоченное чтение).

В каждой конкретной СУБД список поддерживаемых уровней изоляции и их реализация могут отличаться.

Уровень Read uncommitted хуже всего защищает данные, но обеспечивает самую высокую параллельность работы пользователей.

Serializable является самым сильным уровнем изоляции и обеспечивает максимальную защиту данных, но при этом сильно снижает параллельность работы.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Каждый последующий уровень изоляции включает в себя свойства предыдущих. Именно уровень изоляции транзакции и выполняемое действие определяют, какая именно блокировка будет наложена и как долго она будет держаться.

Помимо четырех описанных уровней изоляции транзакции в платформе 8.3 при работе в управляемом режиме блокировок используется еще один уровень изоляции – Read committed snapshot.

Какой именно уровень изоляции использовать, решают разработчики приложения, в случае 1С это решают разработчики платформы. Платформа 1С для разных действий в разных режимах использует различные уровни изоляции.

Для того чтобы лучше понять, что такое уровни изоляции транзакции, как они работают и почему платформа в разных случаях использует разные уровни, необходимо знать, какие проблемы параллельной работы они решают.

Потерянное обновление

Допустим, в системе нет вообще никаких блокировок, при этом две транзакции хотят обновить одну строку данных. В результате останется только то изменение, которое было выполнено последней транзакцией. При этом первая транзакция будет «думать», что ее изменения были успешно записаны.

Такая ситуация называется проблемой потерянного обновления.

Транзакция 1	Цена	Транзакция 2
Установить цену 50 руб.	50	
	70	Установить цену 70 руб.
Зафиксировать транзакцию	70	
	70	Зафиксировать транзакцию

Уровень изоляции Read uncommitted решает проблему потерянного обновления, и это единственная проблема, которую он решает.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Для решения используются X блокировки, которые ставятся перед изменением объекта. Когда СУБД «видит» что запрос выполняется с уровнем изоляции Read uncommitted и запрос модифицирует данные, то СУБД «знает» что надо поставить X блокировку, потому что разработчик хочет защитить данные от потерянного обновления. Если же запрос с уровнем изоляции Read uncommitted не изменяет данные, тогда СУБД никаких блокировок на данные не ставит.

X блокировки при любом уровне изоляции и при любом режиме работы всегда держатся до конца транзакции.

Транзакция 1	Цена	Транзакция 2
Установить X блокировку	0	
Установить цену 50 руб.	50	*Попытка установить X блокировку*
X блокировка держится до конца транзакции	50	*Ожидание…*
Зафиксировать транзакцию	50	*Ожидание…*
	50	*Установить X блокировку*
	70	**Установить цену 70 руб.**
	70	**Зафиксировать транзакцию**

X блокировка гарантирует, что одни и те же данные могут изменяться транзакциями только последовательно. Read uncommitted использует только X блокировки, S блокировки при чтении этот уровень изоляции не устанавливает.

Помимо X блокировок данный режим устанавливает служебные блокировки стабильности схемы Sch-S, которые не позволяют изменять структуру таблицы, пока выполняется запрос. Благодаря таким блокировкам нельзя удалить колонку из таблицы или поменять тип поля, пока выполняется запрос. Блокировки стабильности схемы не влияют на параллельность работы с данными.

Ни в 1С, ни где-либо еще проблема потерянного обновления невозможна, т.к. все СУБД, в том числе и «версионники», в любом случае ставят X блокировку при изменении данных.

Оптимистические объектные блокировки тоже защищают данные от похожей проблемы, но в случае объектных блокировок изменение могло «потеряться» по вине самого приложения 1С, а не по вине транзакции. Поэтому и решение этой проблемы обеспечивается самим приложением 1С (оптимистическими блокировками), а не X блокировкой СУБД.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

«Грязное» чтение

Проблема «грязного» чтения заключается в том, что транзакция может прочитать данные, которые еще не записаны другой транзакцией, что может привести к неверным решениям.

Транзакция 1	Остаток	Транзакция 2
Прочитать остаток	1	
Установить X блокировку	1	
Списать 1 шт.	0	
X блокировка держится до конца транзакции	0	**Прочитать остаток**
Отменить транзакцию	1	
	1	**Отменить списание (нет товара)**

Вторая транзакция прочитала данные, которые еще не записаны первой транзакцией, в результате чего не смогла списать товар по причине его нехватки. При этом первая транзакция откатилась, и товар стал доступен.

В 1С «грязное» чтение возможно только во время чтения данных вне транзакции при работе с версией 8.2, либо при работе в автоматическом режиме блокировок. При чтении данных вне транзакции запрос будет выполнен с уровнем изоляции Read uncommitted, который не защищает от «грязного» чтения. Все запросы, которые обеспечивают отображение элементов интерфейса (журналы документов, формы списков и т.д.) а также формирование отчетов – все они получают «грязные» данные.

Если бы эти запросы работали с более высоким уровнем изоляции, то любое формирование отчета или открытие списка документа блокировало бы данные, и многопользовательская работа была бы невозможна.

Платформа не использует Read uncommitted при изменении данных, он используется только при чтении вне явной транзакции. Для решения проблемы «грязного» чтения в 8.2 используется уровень изоляции Read committed. Этот режим решает и проблему потерянного обновления, и «грязного» чтения, т.к. каждый последующий уровень изоляции решает проблемы всех предыдущих. На данном уровне изоляции помимо X блокировок используются S блокировки. S блокировка держится только на время чтения строки. Как только строка прочитана, блокировка снимается, не дожидаясь конца запроса. Во всех источниках 1С написано, что S блокировка для Read Committed будет держаться до конца запроса, однако на практике это не так.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

На экзамене 1С:Эксперт лучше отвечать, что в режиме Read Committed S блокировка держится до конца запроса. В принципе, большой роли это не играет, и процесс оптимизации никак не меняется. Поэтому часто по ходу курса будет говориться, что для Read Committed блокировка при чтении держится до конца запроса, это упрощает понимание.

Транзакция 1	Остаток	Транзакция 2
Установить S блокировку	1	
Прочитать остаток	1	
Снять S блокировку	1	
Установить X блокировку	1	
Списать 1 шт.	0	*Попытка установить S блокировку*
X блокировка держится до конца транзакции	0	*Ожидание…*
Отменить транзакцию	1	*Ожидание…*
		Установить S блокировку
	1	**Прочитать остаток**
	1	*Снять S блокировку*
	1	*Установить X блокировку*
	0	**Списать 1 шт.**
	0	*X блокировка держится до конца транзакции*
	0	**Зафиксировать транзакцию**

Благодаря S блокировке транзакция не может прочитать незафиксированные («грязные») данные, таким образом, проблема исчезает. Следует обратить внимание, что S блокировка в данном режиме изоляции снимается, не дожидаясь окончания транзакции.

Read committed используется в 1С, только если конфигурация работает в управляемом режиме блокировок. В данном режиме выполняется как чтение в транзакции, так и запись.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Уровень изоляции Read committed snapshot

Если в версии платформы 8.3 используется управляемый режим блокировок и используется MS SQL 2005 или выше, то платформа будет работать с режимом изоляции Read committed snapshot (режим изоляции моментального снимка). Сокращенно данный режим называется RCSI (Read Committed Snapshot Isolation). Не следует путать данный режим с режимом Snapshot, он в 1С не используется. Следует учесть, что если у конфигурации будет стоять режим совместимости с 8.2 или 8.1, то RCSI использоваться не будет.

RCSI выполняет ту же задачу, что и Read committed: решает проблему «грязного» чтения, но делает это с помощью версионирования.

В данном режиме нет блокировки при чтении, но перед модификацией данных сначала создается копия изменяемой строки (версия данных до изменения) и помещается в базу TempDB, и там она меняется. В том случае, если вторая транзакция хочет прочитать данные, изменяемые первой транзакцией, вторая транзакция просто читает версию данных до изменения.

Транзакция 1	Остаток в TempDB	Остаток в базе	Транзакция 2
Прочитать остаток		1	
Установить X блокировку		1	
Сохранить версию строки	1	1	
Списать 1 шт.	0	1	
X блокировка держится до конца транзакции	0	1	**Прочитать остаток на момент до X блокировки**
Отменить транзакцию		1	
		1	*Установить X блокировку*
		0	**Списать 1 шт.**
		0	*X блокировка держится до конца транзакции*
		0	**Зафиксировать транзакцию**

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Как видно из схемы, при чтении не накладывается S блокировка, вместо этого читается старая версия данных, которая была до изменения первой транзакцией.

В результате достигается сразу два эффекта:

- Решается проблема «грязного» чтения, т.к. читаются только подтвержденные данные
- Повышается параллельность работы пользователей, т.к. пишущие транзакции не блокируют читающих, читающие просто читают версию данных до изменения.

Единственный минус данного уровня изоляции в том, что он увеличивает нагрузку на оборудование, т.к. приходится делать копии и хранить версии строк, особенно это сказывается на оперативной памяти и дисках.

Возможно, кто-то заметил, что если бы первая транзакция не была бы отменена, то на складе образовались бы отрицательные остатки. Это действительно так, но, чтобы этого не произошло, существуют управляемые блокировки, о которых будет рассказано чуть позже. Сейчас важно просто понять, как работают различные уровни изоляции.

Если конфигурация работает в управляемом режиме блокировок, а в качестве СУБД используется «версионник» (Oracle или PostgreSQL), тогда поведение будет аналогично уровню изоляции RCSI, т.е. «грязное» чтение там невозможно.

Можно привести следующее сравнение между режимами Read committed и Read committed snapshot.

	Read committed	**Read committed snapshot**
Когда используется	Управляемый режим блокировок	8.3 + управлямый режим блокировок + нет режима совместимости с 8.2 + MS SQL 2005 или выше (либо СУБД «версионник»)
Защита данных	С помощью S блокировки на время чтения	С помощью версионирования
Преимущества	Меньше нагрузка на оборудование	Лучшая параллельность (пишущие не блокируют читающих)
Недостатки	Меньшая параллельность (пишущие блокируют читающих)	Дополнительная нагрузка на оборудование (надо делать и хранить версии строк)

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

При работе с MS SQL Server в платформе 8.2, RCSI не используется. Если в конфигурации 8.2 используется управляемый режим блокировок, то есть возможность включить режим версионирования напрямую в СУБД. Для этого необходимо выполнить в Management Studio следующие команды:

```
ALTER DATABASE ИмяБазы
SET ALLOW_SNAPSHOT_ISOLATION ON
ALTER DATABASE ИмяБазы
SET READ_COMMITTED_SNAPSHOT ON
```

После этого поведение системы будет аналогично 8.3, при этом такое поведение сохранится независимо от обновления базы или платформы 1С.

Технически ничто не мешает включить этот режим в 8.2, работать он будет, и параллельность повысится, но здесь возникает проблема юридическая.

Формально это работа с базой в обход средств платформы, поэтому расценивается как нарушение лицензионного соглашения.

Неповторяющееся чтение

Проблема неповторяющегося чтения выглядит следующим образом:

- Транзакция 2 читает данные
- Транзакция 1 эти данные меняет
- Транзакция 2 снова читает данные и видит, что они самопроизвольно изменились.

Схема возникновения данной проблемы приведена ниже.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Транзакция 1	Остаток	Транзакция 2
Установить S блокировку	1	*Установить S блокировку*
Прочитать остаток	1	**Прочитать остаток**
Снять S блокировку	1	*Снять S блокировку*
Установить X блокировку	1	**Прочие действия…**
Списать 1 шт.	0	
X блокировка держится до конца транзакции	0	
Зафиксировать транзакцию	0	
	0	*Установить S блокировку*
	0	**Прочитать остаток**
	0	*Снять S блокировку*
	0	**Зафиксировать транзакцию**

С точки зрения второй транзакции остаток самопроизвольно изменился с 1 на 0.

Данную проблему также можно назвать «проблемой отрицательных остатков». Обычно никто не читает данные в транзакции два раза, чаще вместо второго чтения идет изменение данных на основании ранее прочитанного значения. Из-за того, что блокировка снимается сразу после чтения, данные остаются незащищенными, и другая транзакция может их в этот момент изменить.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Транзакция 1	Остаток	Транзакция 2
Установить S блокировку	1	*Установить S блокировку*
Прочитать остаток	1	**Прочитать остаток**
Снять S блокировку	1	*Снять S блокировку*
Установить X блокировку	1	**Прочие действия …**
Списать 1 шт.	0	*Попытка установить X блокировку*
X блокировка держится до конца транзакции	0	*Ожидание…*
Зафиксировать транзакцию	0	*Ожидание…*
	0	*Установить X блокировку*
	-1	**Списать 1 шт.**
	-1	*X блокировка держится до конца транзакции*
	-1	**Зафиксировать транзакцию**

Эту проблему решает уровень изоляции Repeatable Read. На этом уровне изоляции S блокировки ставятся до конца транзакции, таким образом, прочитанные однажды данные никто не сможет изменить до окончания транзакции. При этом S блокировка до конца транзакции держится только на тех записях, которые подходят под условия запроса. Если строка под условия не подходит, то блокировка снимается с нее до окончания выполнения запроса.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

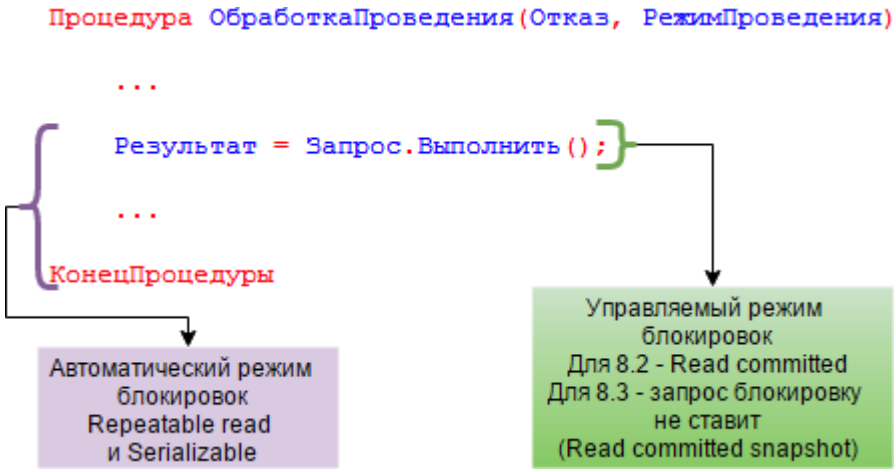
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Транзакция 1	Остаток	Транзакция 2
Установить S блокировку	1	Установить S блокировку
Прочитать остаток	1	**Прочитать остаток**
Попытка установить X блокировку	1	**Прочие действия…**
Ожидание…	1	**Прочитать остаток**
Ожидание…	1	**Зафиксировать транзакцию**
Установить X блокировку	1	
Списать 1 шт.	0	
X блокировка держится до конца транзакции	0	
Зафиксировать транзакцию	0	

В 1С уровень изоляции Repeatable read используется только в автоматическом режиме управления блокировками и только при чтении объектных сущностей (справочников, документов и т.д.).

Repeatable read очень похож на Read committed и отличается от него только длительностью блокировки. В Read committed блокировка строк держится до конца запроса, а в Repeatable read – до конца транзакции.



Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Фантомное чтение

Фантомное чтение – это чтение в транзакции данных, которых раньше не было, с точки зрения читающей транзакции они появились из ниоткуда (фантомы).

Проблема фантомного чтения выглядит следующим образом:

- Транзакция 2 читает все записи, где цена находится в диапазоне от 200 до 300
- Транзакция 1 добавляет новую запись, где цена равна 250
- Транзакция 2 повторно читает записи с ценой от 200 до 300 и обнаруживает новую строку (фантомную запись).

Эта проблема похожа на неповторяющееся чтение, но там менялись существующие данные, а здесь появились новые.

Схематически проблему фантомного чтения можно представить следующим образом:

Транзакция 1	Число товаров	Транзакция 2
		Установить S блокировку
	3	**Выбрать товары с ценой от 200 до 300 рублей**
Добавить новый товар с ценой 250 рублей	4	**Прочие действия…**
Зафиксировать транзакцию	4	**Выбрать товары с ценой от 200 до 300 рублей**
	4	**Зафиксировать транзакцию**

Проблему фантомного чтения решает самый строгий уровень изоляции – Serializable.

Для решения проблемы фантомов данный уровень устанавливает Range блокировки – блокировки диапазона. Range блокировки имеют разные типы в зависимости от выполняемых действий. RangeS – блокировка диапазона для чтения, RangeX – блокировка диапазона для изменения и т.д.

Блокировки диапазона записываются в следующем формате: RangeB1-B2, где B1 – это режим блокирования диапазона, а B2 – режим блокирования строки. Например, блокировка RangeS-S означает, что и диапазон, и строки заблокированы для чтения. Режим RangeS-U означает блокировку диапазона для чтения и блокировку строк для обновления.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Блокировка диапазона достигается за счет того, что до конца транзакции будут заблокированы все строки, которые обрабатывал запрос во время своего выполнения. Это значит, что если был выбран план со сканированием, то до конца транзакции будет заблокирована вся таблица, даже если запрос вернет всего одну строку.

Например, если по полю *Цена* есть индекс, то будет заблокирован диапазон строк со значениями от 200 до 300. Когда другая транзакция попытается вставить новое значение, попадающее в этот диапазон, она не сможет этого сделать и встанет в очередь.

Транзакция 1	Число товаров	Транзакция 2
	3	*Установить RangeS-S блокировку на диапазон от 200 до 300 и одну соседнюю запись снизу*
	3	**Выбрать товары с ценой от 200 до 300 рублей**
Попытка добавить новый товар с ценой 250 рублей	3	**Прочие действия…**
Ожидание…	3	**Выбрать товары с ценой от 200 до 300 рублей**
Ожидание…	3	**Зафиксировать транзакцию**
Добавить новый товар с ценой 250 рублей	4	
Зафиксировать транзакцию	4	

При блокировании диапазона значений ключа индекса есть одна особенность – блокируются не только значения ключа индекса, которые входят в диапазон, но и одно значение снизу, т.е. блокируется N+1 строка, где N – это число строк, подходящих под условие выборки. Это сделано для того, чтобы нельзя было вставить граничное значение.

Разберем эту ситуацию на примере. Допустим, индекс содержит следующие значения.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Значения ключа индекса
103
190
211
260
275
305
330

При этом выполняется запрос с условием на выборку товаров с ценой между 200 и 300. Жирным отмечены строки, которые попадают в выборку. Оранжевым закрашены строки, которые заблокированы, при этом видно, что заблокировано одно нижнее пограничное значение 305, которое не входит в выборку.

Если бы нижнее пограничное значение не блокировалось, а блокировались бы только значения до 275, то ничто не мешало бы другой транзакции вставить значение 281 или любое другое значение от 275 до 300 включительно. В результате проблема фантомного чтения не была бы решена.

Наверняка, многие слышали, что блокировки диапазона блокируют не одну, а две соседних строки. Одну сверху диапазона и одну снизу. Давайте разберемся, откуда появилось такое мнение.

В приведенном выше примере строка 190 не заблокирована, и, казалось бы, ничто не мешает другой транзакции вставить значение 200, что снова приведет к проблеме фантомного чтения. Однако следует понимать, что при вставке также будет заблокирована N+1 строка, где N – это количество вставляемых строк.

Схематически это можно представить так:

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Значения ключа индекса	
103	**Вставка значения 200**
190	200
211	**211 (ожидание)**
260	
275	
305	
330	

В результате все равно получится ожидание на ключе со значением 211, и фантомного чтения не будет. Обычно в целях упрощения просто говорят, что Range блокировки захватывают две соседние записи: одну сверху и одну снизу, хотя по факту захватывается только одна нижняя соседняя запись. Но она захватывается и при чтении, и при вставке нового значения.

Во всех остальных уровнях, отличных от Serializable, если строка не подходит под условие запроса, блокировка снимается с этой строки сразу, не дожидаясь окончания запроса. При работе с уровнем Serializable запрос блокирует и держит до конца транзакции все строки, которые он обработал, и неважно, подходят они под условие или нет. Например, если будет скан таблицы, в которой 1 млн. строк, и запрос ничего не вернет, то вся таблица будет заблокирована до окончания транзакции.

В нашем примере, если индекса по полю «Цена» нет, то поиск будет выполняться сканированием, следовательно, будет заблокирована вся таблица до конца транзакции. Именно поэтому при работе с уровнем Serializable особенно важны оптимальные запросы, иначе излишние блокировки будут не только по данным, но и по времени.

Как видно из примеров, данный уровень изоляции приводит к излишним блокировкам (блокируются соседние строки), и это не является виной программиста, просто так реализован механизм защиты от фантомов.

Serializable используется в 1С только в автоматическом режиме блокировок и только при чтении необъектных данных (регистров, последовательностей, констант и пр.), т.к. там проблема фантомного чтения может быть критичной. При этом, если в автоматическом режиме блокировок читается объектная сущность (справочник, документ и пр.), там фантомное чтение допускается, т.к. используется уровень изоляции Repeatable read. Для объектов фантомное чтение некритично, поэтому оно не является проблемой.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Сводная таблица уровней изоляции и проблем параллельной работы

В данной таблице представлена сводная информация о том, какой уровень изоляции какие проблемы решает.

	Блокировки при чтении	Потерянное обновление	«Грязное» чтение	Неповтор. чтение	Фантомное чтение
Read uncommitted	нет	+	-	-	-
Read committed snapshot	нет	+	+	-	-
Read committed	S на момент чтения строки	+	+	-	-
Repeatable read	S до конца транзакции, если строка подходит под условия	+	+	+	-
Serializable	S до конца транзакции, если строка обработана запросом	+	+	+	+

Видеоурок. Read uncommitted

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример работы уровня изоляции Read uncommitted
- Как по тексту SQL запроса понять, что используется этот уровень изоляции.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Read committed

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример работы уровня изоляции Read committed
- Как по тексту SQL запроса понять, что используется этот уровень изоляции.

Видеоурок. Read committed snapshot

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример работы уровня изоляции Read committed snapshot
- Можно ли по тексту SQL запроса понять, что используется этот уровень изоляции.

Видеоурок. Repeatable read

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример работы уровня изоляции Repeatable read
- В каком случае используется данный режим изоляции
- Как по тексту SQL запроса понять, что используется этот уровень изоляции.

Видеоурок. Serializable

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример работы уровня изоляции Serializable
- Пример блокировки диапазона строк.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 28

Режимы блокировок в 1С

Конфигурация может работать в 3 режимах управления блокировкой данных.

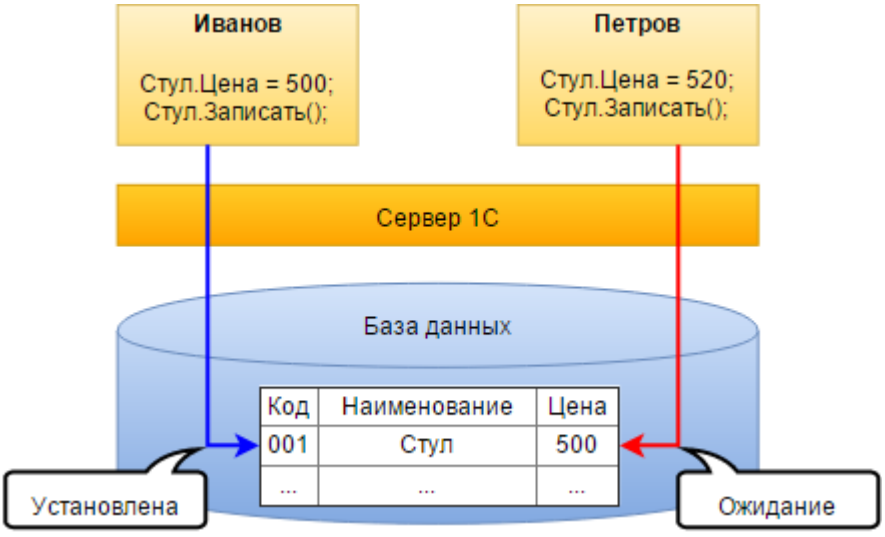
- Автоматический (устаревший)
- Управляемый (по умолчанию)
- Автоматический и управляемый.

Каждый из режимов имеет свои особенности, преимущества и недостатки.

Режим блокировки данных можно указать как для всей конфигурации, так и отдельно для каждого объекта метаданных. Режим задается свойством «Режим управления блокировкой данных». Если для конфигурации выбрано значение «Автоматический» или «Управляемый», то режим, установленный для объектов, будет игнорироваться. Если у конфигурации выбран смешанный режим «Автоматический и управляемый», то будет использоваться тот режим, который указан для объекта метаданных, открывшего транзакцию.

Автоматический режим блокировок

В автоматическом режиме блокировок платформа использует только блокировки СУБД. Соответственно, все ожидания на блокировках происходят на стороне СУБД.



Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

До версии 8.1 платформа работала только в автоматическом режиме блокировок, который имеет массу недостатков.

В данном режиме для запросов в транзакции используются уровни изоляции Repeatable read (для объектных данных) и Serializable (для необъектных).

Это высокие уровни изоляции, которые в силу особенностей своей реализации снижают параллельность работы.

Можно выделить следующие недостатки автоматического режима:

- **Присутствуют излишние блокировки по вине СУБД.** Это происходит при работе с уровнем изоляции Serializable. В этом случае будет блокировка лишней строки на границе диапазона. При этом блокировка держится до конца транзакции
- **Блокировка пустой таблицы.** Если идет обращение к таблице какой-нибудь необъектной сущности (например, регистра) и при этом таблица пустая, то будет заблокирована вся таблица. В пустой таблице любая вставляемая другой транзакцией запись будет соседней, следовательно, будет ожидание.
- **S блокировки держатся до конца транзакции.** При обращении к объектным данным до конца транзакции будут заблокированы только строки, которые возвращает запрос, потому что используется уровень изоляции Repeatable read. При чтении необъектных данных используется уровень изоляции Serializable и до конца транзакции блокируются все строки, которые запрос обрабатывает, и неважно, подошла строка под условие или нет.
- **При использовании СУБД «версионника» будет блокироваться вся таблица.** Если бы вся таблица не блокировалась, то в рассмотренном примере были бы возможны отрицательные остатки, т.к. поведение «версионника» аналогично работе Read committed snapshot. Поэтому использование конфигурации в автоматическом режиме блокировок на Oracle или PostgreSQL для многопользовательской работы не имеет большого практического смысла, необходимо использовать управляемый режим блокировок.

Реализация уровня Serializable у «версионников» отличается от MS SQL Server. Например, в MS SQL Server можно параллельно выполнить два запроса в транзакциях, где один запрос просто читает данные (S блокировка), а второй читает те же данные с опцией ДЛЯ ИЗМЕНЕНИЯ (U блокировка). S и U блокировки совместимы, поэтому конфликта не происходит. Но если попытаться сделать то же самое в PostgreSQL, то возникнет ожидание, т.к. чтение для обновления заблокирует всю таблицу даже для других читающих транзакций. При этом неважно, идет ли работа с одними и теми же данными или с разными, «версионники» в этом режиме будут блокировать всю таблицу целиком.

Рассмотрим параллельность работы PostgreSQL в автоматическом режиме. Все операции, кроме чтения, независимо от набора данных выполняются в транзакции по одной таблице последовательно. Параллельность возможна только при использовании разных таблиц.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Операция 1	Операция 2	Данные разные	Данные одинаковы
Запись	Запись	-	-
Запись	Чтение	-	-
Запись	Чтение ДЛЯ ИЗМЕНЕНИЯ	-	-
Чтение	Чтение	+	+
Чтение	Чтение ДЛЯ ИЗМЕНЕНИЯ	-	-
Чтение ДЛЯ ИЗМЕНЕНИЯ	Чтение ДЛЯ ИЗМЕНЕНИЯ	-	-

Данная таблица симметрична, т.е. неважно, какая из операций выполнялась первой, а какая второй.

Если идет работа с разными таблицами, то все операции будут выполнятся параллельно.

Управляемый режим блокировок

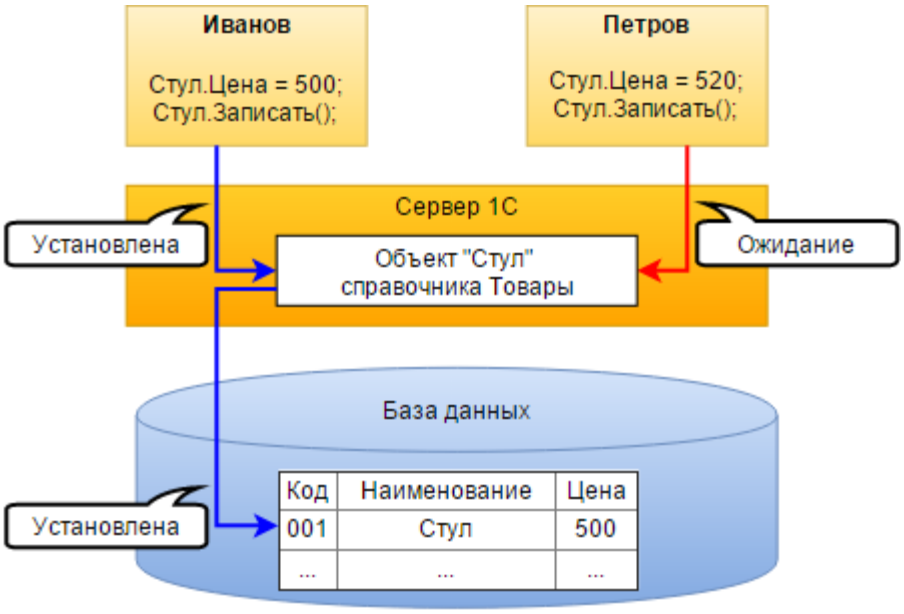
При включении данного режима уровень изоляции транзакции на СУБД снижается, и начинает работать собственный менеджер блокировок сервера 1С. Теперь блокировка будет ставиться сначала на сервере 1С и только потом на сервере СУБД. При этом, если сервер 1С обнаружит, что блокировки несовместимы, ожидание будет на самом сервере 1С. В данном случае СУБД даже не будет знать, что было какое-то ожидание.

Блокировки, которые ставятся на сервере 1С, называют управляемыми блокировками.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



Сервер 1С ставит блокировку на объекты 1С, в то время как блокировки СУБД ставятся на строки таблиц и индексов. Например, при сохранении элемента справочника с табличной частью на сервере 1С будет просто заблокирован объект элемента справочника, а на СУБД будут заблокированы строки в основной таблице справочника и в таблице, в которой хранится его табличная часть.

В управляемом режиме снижается уровень изоляции транзакции, что повышает параллельность работы пользователей. При этом для 8.2 используется уровень изоляции Read committed, а для 8.3 – Read committed snapshot.

Управляемый режим не имеет недостатков автоматического, поэтому сейчас является стандартом для разработки. Некоторые команды автоматического режима не работают в управляемом, например, опция запроса «ДЛЯ ИЗМЕНЕНИЯ» в управляемом режиме будет проигнорирована.

Минусы управляемого режима:

- **Повышается нагрузка на сервер 1С.** Включается собственный менеджер блокировок, который потребляет дополнительные ресурсы.
- **Необходимо дорабатывать конфигурацию.** Для работы в управляемом режиме необходимо дорабатывать конфигурацию, в связи с чем появляются дополнительные требования к квалификации разработчика.

Несмотря на незначительные минусы, управляемый режим значительно повышает параллельность работы пользователей и сейчас является режимом по умолчанию. Конфигурации на автоматических блокировках считаются устаревшими.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Смешанный режим «Автоматический и управляемый»

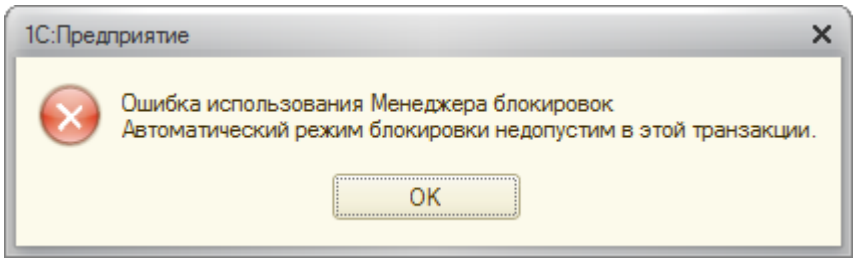
Данный режим доступен только в свойстве конфигурации, его нельзя установить для объктов метаданных. При его использовании режим работы определяется тем режимом, который установлен у объекта метаданных. При этом если в транзакции участвуют разные объекты, то будет использоваться режим того объекта, который открыл транзакцию.

В смешанном режиме при открытии явной транзакции можно использовать параметр РежимБлокировок. Например, если требуется явно начать транзакцию в автоматическом режиме, нужно написать: *НачатьТранзакцию(РежимУправленияБлокировкойДанных.Автоматический)*

В смешанном режиме итоговый режим работы определяется первой открытой транзакцией, т.е. если в одной транзакции открывается другая, то она все равно будет работать в том же режиме, что и первая.

Режим первой транзакции	Режим вложенной транзакции	Итоговый режим
Автоматический	Автоматический	Автоматический
Автоматический	Управляемый	Автоматический
Управляемый	Управляемый	Управляемый
Управляемый	Автоматический	Ошибка

Например, у документа стоит режим автоматический, а у регистра, который он двигает, – управляемый. В таком случае транзакция будет выполнена в автоматическом режиме блокировки данных. Если у документа режим управляемый, а у регистра автоматический, то будет вызвана исключительная ситуация.



При использовании смешанного режима менеджер блокировок 1С будет работать всегда. Даже если транзакция открыта в автоматическом режиме, управляемая блокировка на сервере 1С все равно будет ставиться. Данный режим можно использовать только как временный при переходе с режима автоматических блокировок, но обычно его не используют вообще.

При использовании смешанного режима возможны такие редкие и трудно диагностируемые проблемы, как распределенная взаимоблокировка. Данная проблема будет рассматриваться в соответствующем разделе.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 29

Управляемые блокировки

Зачем нужны управляемые блокировки

Прежде чем перейти к работе с управляемыми блокировками, необходимо понять, для чего же их придумали и какие проблемы они решают.

Представим, что в конфигурации изменили режим блокировок с автоматического на управляемый. При этом запросы стали выполняться с более низким уровнем изоляции, и блокировка при чтении либо не будет ставиться вообще (в 8.3), либо будет ставиться только на время запроса (8.2). Давайте посмотрим, к чему это может привести.

Транзакция 1	**Остаток**	**Транзакция 2**
Установить S блокировку	1	*Установить S блокировку*
Прочитать остаток	1	**Прочитать остаток**
Снять S блокировку	1	*Снять S блокировку*
Установить X блокировку	1	**Прочие действия …**
Списать 1 шт.	0	*Попытка установить X блокировку*
X блокировка держится до конца транзакции	0	*Ожидание…*
Зафиксировать транзакцию	0	*Ожидание…*
	0	*Установить X блокировку*
	-1	**Списать 1 шт.**
	-1	*X блокировка держится до конца транзакции*
	-1	**Зафиксировать транзакцию**

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Если остатки контролируются перед записью, то запрос снимает блокировку, не дожидаясь конца транзакции, а в это время другая транзакция может списать товар, и в результате образуются отрицательные остатки.

Чтобы этого избежать, необходимо поставить явную управляемую блокировку, которая будет держаться до конца транзакции. В этом случае списание будет идти последовательно.

Транзакция 1	Остаток	Транзакция 2
Установить исключительную управляемую блокировку	1	*Попытка установить исключительную управляемую блокировку*
Установить S блокировку	1	*Ожидание…*
Прочитать остаток	1	*Ожидание…*
Снять S блокировку	1	*Ожидание…*
Установить X блокировку	1	*Ожидание…*
Списать 1 шт.	0	*Ожидание…*
X блокировка держится до конца транзакции	0	*Ожидание…*
Зафиксировать транзакцию	0	*Ожидание…*
	0	*Установить исключительную управляемую блокировку*
	0	*Установить S блокировку*
	0	**Прочитать остаток**
	0	*Снять S блокировку*
	0	**Отменить транзакцию из-за нехватки товара**

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Типы управляемых блокировок

Есть всего два типа управляемых блокировок:

- **Разделяемая.** Аналог S блокировки, блокировка для чтения
- **Исключительная.** Аналог X блокировки, блокировка для изменения.

	Разделяемая	Исключительная
Разделяемая	**+**	**-**
Исключительная	**-**	**-**

Разделяемая блокировка используется крайне редко, в основном используется исключительная блокировка.

Когда устанавливается управляемая блокировка

Бытует ошибочное мнение, что управляемую блокировку всегда нужно ставить вручную, но это не так. Управляемая блокировка может быть установлена как явно (вручную), так и неявно (автоматически).

Неявно управляемая блокировка ставится в следующих случаях:

- При любом изменении данных: записи, проведении, удалении и т.д. В данном случае будет установлена исключительная блокировка
- При чтении регистров методом *Прочитать()*. Данный метод откроет транзакцию и устанавит разделяемую управляемую блокировку.

При модификации данных платформа автоматически устанавливает исключительную управляемую блокировку только на те данные, которые изменяются. Например, если проводится документ, то управляемая блокировка будет наложена на сам документ и на все объекты, которые изменяет этот документ при проведении, например, на регистры.

Явная управляемая блокировка ставится при использовании объекта *БлокировкаДанных*. Такая блокировка может потребоваться, если необходимо защитить данные, пока длится транзакция, например, при контроле остатков перед записью.

Не важно, какой тип управляемой блокировки используется, разделяемая или исключительная, она всегда держится до конца транзакции.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Управляемая блокировка **не ставится** в следующих случаях:

- При чтении запросом *Запрос.Выполнить()*
- При чтении объектных данных через объектную модель *Ссылка.ПолучитьОбъект()*, *Ссылка.Реквизит*, *Ссылка.Прочитать()*. При этом не важно, есть у объекта табличная часть или нет
- При использовании объекта *БлокировкаДанных* вне транзакции. Управляемые блокировки, так же, как и блокировки СУБД, могут существовать только в транзакции.

Явные управляемые блокировки

Явная управляемая блокировка требуется довольно редко, например, в том случае, если используется партионный учет и необходимо сначала получать остатки и только потом делать списание. Подобный случай уже был описан в примере выше.

Установка явных управляемых блокировок не накладывает никаких блокировок на сервере СУБД, блокировки ставятся только на сервере 1С.

Допустим, необходимо установить явную управляемую блокировку перед запросом контроля остатков при проведении документа. В этом случае необходимо написать следующий код:

```
// создаем объект явной управляемой блокировки
Блокировка = Новый БлокировкаДанных;
// указываем пространство блокировки (где блокируем)
ЭлементБлокировки = Блокировка.Добавить("РегистрНакопления.ТоварыНаСкладах");
// указываем тип блокировки (как блокируем)
ЭлементБлокировки.Режим = РежимБлокировкиДанных.Исключительный;
// указываем какие значения нужно заблокировать (что блокируем)
// можно указать источник данных если значений много
// в качестве источника можно указать табличную часть, таблицу значений
// набор записей или результат запроса
ЭлементБлокировки.ИсточникДанных = ДокументОбъект.ТабличнаяЧасть;
// указываем какой реквизит таб. части какому измерению регистра соответствует
ЭлементБлокировки.ИспользоватьИзИсточникаДанных("Номенклатура", "Номенклатура");
// указываем значение склада для измерения регистра «Склад»
ЭлементБлокировки.УстановитьЗначение("Склад", ДокументОбъект.Склад);
// блокировка устанавливается только при выполнении метода Заблокировать
Блокировка.Заблокировать();
// запрос к остаткам
Результат = ЗапросКонтрольОстатков.Выполнить();
…
```

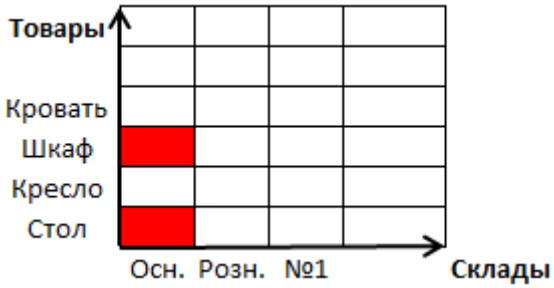
При выполнении этого кода на сервере 1С будет установлена исключительная блокировка на регистр *ТоварыНаСкладах*. При этом будут заблокированы все товары из табличной части на указанном складе. Если бы склад не был указан, то товары в регистре были бы заблокированы на всех складах.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Наглядно продемонстрировать, что именно будет заблокировано, можно с помощью координатной сетки, где в качестве осей выступают измерения регистра. Если в табличной части документа выбраны товары «Шкаф» и «Стол» и при этом используется склад «Основной», то получается следующее:

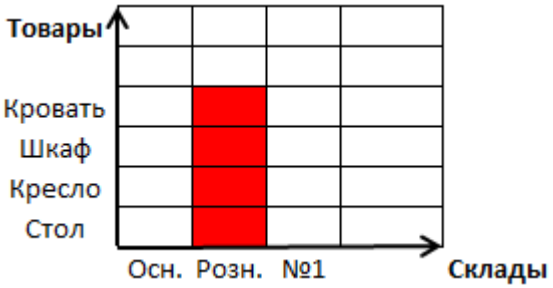


Из схемы видно, что блокировка устанавливается на пересечении значений измерений.

Красные клетки означают то, что было заблокировано, т.е. «Стол» можно будет списать с других складов, но со склада «Основной» не получится. В результате было заблокировано только то, что нужно, ничего лишнего.

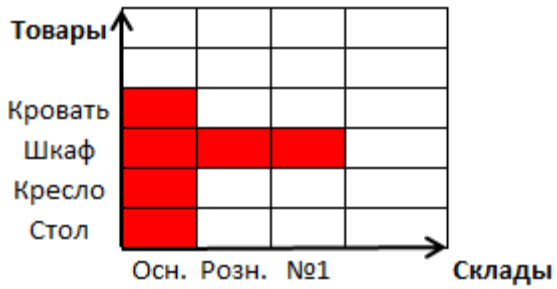
Рассмотрим несколько примеров управляемой блокировки.

```
// блокировка всех товаров по одному складу
// источник указан, но это не имеет смысла, т.к. не указаны поля
// поэтому подразумеваются все значения
…
ЭлементБлокировки.ИсточникДанных = Док.ТабличнаяЧасть;
ЭлементБлокировки.УстановитьЗначение(«Склад», СкладРозницаССылка);
…
```




```
// блокировка одного товара по всем складам
// и всех товаров на одном складе
ЭлементБлокировки.УстановитьЗначение(«Склад», СкладОсновнойССылка);
ЭлементБлокировки2 = Блокировка.Добавить("РегистрНакопления.ТоварыНаСкладах");
ЭлементБлокировки2.УстановитьЗначение(«Товар», ТоварШкафССылка);
```


Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



Поля блокировки данных

В версии 8.2 явную управляемую блокировку на объектные данные можно наложить только по ссылке. Например, если по какой-либо причине нужно заблокировать элемент справочника *Контрагенты*, то код будет выглядеть следующим образом:

```
Блокировка = Новый БлокировкаДанных;
ЭлементБлокировки = Блокировка.Добавить("Справочник.Контрагенты");
ЭлементБлокировки.Режим = РежимБлокировкиДанных.Исключительный;
ЭлементБлокировки.УстановитьЗначение("Ссылка", КонтрагентСсылка);
Блокировка.Заблокировать();
```

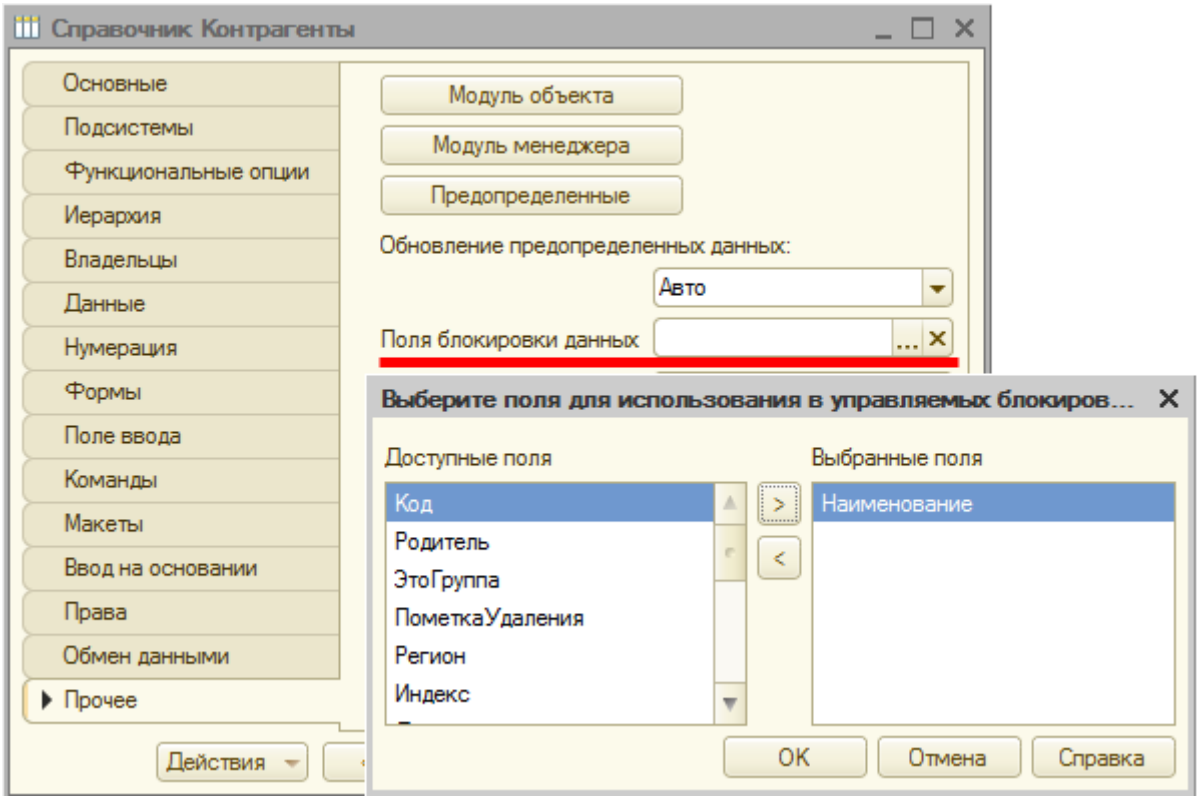
В версии 8.2 мы не можем сразу заблокировать всех контрагентов с каким-то условием по одному полю. Например, чтобы заблокировать всех контрагентов с определенным наименованием, пришлось бы сначала запросом получать ссылки на эти элементы и только потом устанавливать явную управляемую блокировку на каждый элемент.

В 8.3 для объектных данных появилось новое свойство – поля блокировки данных. С помощью этого свойства можно выбрать те поля, по которым может потребоваться установка блокировки.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



По выбранным полям можно будет ставить явную управляемую блокировку, т.е. вместо ссылки указывать значение реквизита.

```
…
ЭлементБлокировки.УстановитьЗначение("Наименование", «ИП Иванов»);
…
```

Если в справочнике есть несколько элементов с таким наименованием, то все они будут заблокированы.

В качестве полей блокировки данных не могут участвовать поля с типом:

- Строка неограниченной длины
- Хранилище значения
- Тип значения плана видов характеристик
- Составной тип, включающий один из вышеперечисленных типов.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 30

Пространства управляемых блокировок

При описании явной управляемой блокировки задается объект, который необходимо заблокировать. Например:

```
ЭлементБлокировки = Блокировка.Добавить("РегистрНакопления.ТоварыНаСкладах");
```

Данный параметр называется пространством блокировки. Обычно пространство блокировки совпадает с именем объекта метаданных, но это не всегда так.

Рассмотрим основные пространства, доступные для использования.

Имя пространства	Имя поля
Для объектных данных: ТипОбъекта.ИмяОбъекта Справочник.Товары, ПланСчетов.Основной …	Ссылка Поля блокировки данных (в 8.3)
РегистрСведений.ИмяРегистра.НаборЗаписей (только если регистр подчинен регистратору) РегистрНакопления.ИмяРегистра.НаборЗаписей РегистрБухгалтерии.ИмяРегистра.НаборЗаписей РегистрРасчета.ИмяРегистра.НаборЗаписей Последовательность.ИмяПоследовательности.НаборЗаписей	Регистратор
РегистрСведений.ИмяРегистра РегистрНакопления.ИмяРегистра	Период (если РС периодичекий) ИмяИзмерения
РегистрБухгалтерии.ИмяРегистра	Период ИмяИзмерения ВидДвижения Счет Субконто ВидСубконто

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

РегистрРасчета.ИмяРегистра	ПериодРегистрации ПериодДействия ИмяИзмерения

Полный список пространств и полей описан на диске ИТС в разделе «9.3.4 Работа с управляемыми блокировками средствами встроенного языка»

Как видно из таблицы, у некоторых необъектных данных есть 2 пространства блокировки, первое по имени объекта и второе – *НаборЗаписей*.

Пространство *НаборЗаписей* блокирует только записи конкретного регистратора. При этом разные пространства одного объекта не конфликтуют между собой.

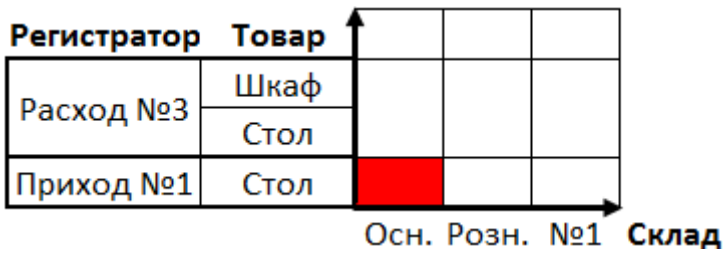
Допустим, в системе выполняется следующий код:

```
НачатьТранзакцию();

      Блокировка = Новый БлокировкаДанных;
      ЭлементБлокировки =
Блокировка.Добавить("РегистрНакопления.ТоварныеЗапасы.НаборЗаписей");
      ЭлементБлокировки.Режим = РежимБлокировкиДанных.Исключительный;
      ЭлементБлокировки.УстановитьЗначение("Регистратор",
Документы.ПриходТовара.НайтиПоНомеру("000000001"));
      Блокировка.Заблокировать();

ЗафиксироватьТранзакцию();
```

При этом в документе выбран товар «Стол» и склад «Основной».

В таком случае ничто не мешает параллельно списывать этот же товар с того же склада в другом документе. Дело в том, что указанным выше кодом были заблокированы не записи объекта регистра, а наборы записей, которые относятся только к указанному документу. Если попытаться в другой сессии снять с проведения документ приход товара с номером «000000001», вот тогда возникнет ожидание на блокировке, т.к. произойдет попытка очистки движений этого документа.

На схеме пространство блокировок *НаборЗаписей* будет выглядеть следующим образом:



На практике пространство *НаборЗаписей* используется крайне редко.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

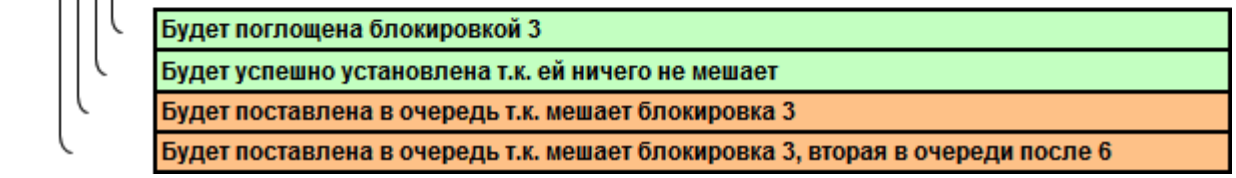
Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Реализация управляемых блокировок

За работу менеджера блокировок отвечает процесс сервера приложений под названием **rmngr**. Процесс rmngr работает не только с блокировками, он выполняет и другие функции, но управляемые блокировки – одна из главных его обязанностей. Менеджер блокировок для работы использует служебную таблицу, которая хранится в оперативной памяти сервера приложений.

Очень упрощенно таблицу можно представить следующим образом:

№	Соед.	Режим	Имя пространства блокировок	Поле 1		Поле 2	
				Имя	Значение	Имя	Значение
1	16	Разделяемый	Справочник.Контрагенты	Ссылка	ООО "Мебел		
2	21	Исключительный	РегистрНакопления.ТоварыНаСклада	Товар	Стул	Склад	Основной
3	39	Исключительный	РегистрНакопления.ТоварыНаСклада	Товар	Шкаф	Склад	Розница
4	39	Исключительный	РегистрНакопления.ТоварыНаСклада	Товар	Шкаф	Склад	Розница
5	41	Разделяемый	РегистрНакопления.ТоварыНаСклада	Товар	Стул	Склад	Розница
6	41	Исключительный	РегистрНакопления.ТоварыНаСклада	Товар	Шкаф	Склад	
7	50	Исключительный	РегистрНакопления.ТоварыНаСклада	Товар	Шкаф	Склад	Розница



Как только транзакция запрашивает блокировку, менеджер проверяет, заблокирован ли запрашиваемый ресурс, и если нет, то блокировка ставится. Если ресурс уже кем-то заблокирован, тогда проверяется совместимость запрашиваемой и установленной блокировок. Если режимы совместимы, то блокировка устанавливается, если же несовместимы, транзакция встает в очередь. При этом если менеджер видит, что от одного соединения идет попытка заблокировать один и тот же объект еще раз, то второй раз блокировка не накладывается.

Чем больше в системе блокировок, тем больше размер служебной таблицы и тем больше памяти потребуется менеджеру блокировок. Поэтому на управляемых блокировках тоже есть эскалация.

Эскалация управляемых блокировок

Пороги срабатывания эскалации на управляемых блокировках четко определены. Для 8.2 эскалация происходит при наложении более 20 тыс. элементов блокировок на одно пространство, в 8.3 – при наложении более 100 тыс. Если в одной транзакции наложить 90 тыс. блокировок, а потом в этой же транзакции наложить на это же пространство еще 5 тыс. и еще 6 тыс. блокировок, то в итоге получится, что на пространство установилось 101 тыс. блокировок, и сработает эскалация. Причем эскалация произойдет в момент наложения той порции блокировок, которая превышает порог эскалации, в данном случае – при наложении последних 6 тыс. блокировок.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Если в базе используются разделители областей данных, то эскалация происходит в пределах одной области данных. Запретить эскалацию или изменить порог срабатывания нельзя, эти алгоритмы жестко заданы в платформе.

Порог эскалации на сервере 1С довольно высокий, поэтому чаще можно встретить эскалацию на сервере СУБД, чем на сервере 1С.

Есть некоторые отличия в механизме эскалации 1С и СУБД.

В MS SQL Server если строки таблицы кем-то заняты, то сервер будет накладывать по 1 250 блокировок и каждый раз после этого пытаться снова заблокировать таблицу.

В сервере 1С, если строки таблицы кем-то заняты, то сессия, запрашивающая эскалацию, становится в очередь и ожидает, пока ресурс не освободится, чтобы заблокировать всю таблицу.

Эскалация возможна только для транзакционных блокировок, эскалации объектных блокировок не бывает.

Перевод конфигурации в управляемый режим

Вся суть перевода конфигурации на управляемые блокировки заключается в том, чтобы найти все места в коде, где идет чтение данных регистров с последующей записью.

Рассмотрим перевод конфигурации по шагам.

1. Нужно сделать копию рабочей базы и поставить у всей конфигурации режим управления блокировкой данных в значение «Управляемый». При этом в течение всего времени перевода желательно вносить минимальное число изменений в рабочую конфигурацию. Если эти доработки не касаются модификации данных, то их можно вносить, иначе лучше подождать.
2. В конфигурации необходимо найти все места, где данные регистров сначала считываются, а потом записываются. Типичный пример – контроль остатков перед записью движений.

   Если конфигурация полностью типовая, то сделать это довольно просто. Можно выполнить глобальный поиск по фразе «ДЛЯ ИЗМЕНЕНИЯ». Данную опцию ставят, чтобы избежать взаимоблокировки, и обычно ее ставят там, где данные регистра сначала считываются, а потом движения по этому регистру записываются.

   Если конфигурация типовая, но доработанная, или полностью нетиповая, тогда помимо поиска запросов с опцией «ДЛЯ ИЗМЕНЕНИЯ» желательно найти все запросы для каждого из регистров. В этом случае нельзя полностью полагаться на опцию «ДЛЯ ИЗМЕНЕНИЯ», т.к. в нетиповой конфигурации разработчики могли просто забыть ее поставить. Далее нужно проанализировать код, выполняется ли после чтения запись в эти регистры или нет. Если нет, то следует перейти сразу к пункту 4. Если запись после чтения выполняется, то следует перейти к пункту 3.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

3. Когда все места, где данные регистров сначала читаются, а потом записываются найдены, есть 2 варианта развития событий:
   - Сделать контроль остатков после записи движений. Сначала нужно сделать запись движений, а потом проверить, появились ли отрицательные остатки. Если появились, то отменяем проведение. Это так называемая «новая методика проведения», но ее можно использовать не всегда, а только если позволяет логика работы приложения. Данная методика имеет свои нюансы, которые будут рассмотрены в соответствующем разделе
   - Установить явную управляемую блокировку перед запросом контроля остатков. Этот вариант используется, если первый не подходит, например, при партионном учете.
4. После исправления всех найденных мест необходимо проверить, что отрицательные остатки не могут образоваться. Для проверки нужно создать два документа с одинаковыми данными, которые двигают регистр с контролем остатков. Один документ необходимо провести под отладкой и остановиться на точке останова в конце модуля проведения, а второй документ провести во второй сессии. Если все сделалано правильно, то второй документ не проведется, возникнет ожидание, следовательно, отрицательных остатков этот документ создать не сможет. Таким образом, желательно проверить все критичные для вашей системы документы, которые могут образовывать отрицательные остатки.
5. Необходимо выполнить объединение тестовой и рабочей конфигурации. В идеале перед обновлением рабочей базы можно смоделировать многопользовательскую работу в копии, написав минимальный сценарий.

Возможна ситуация, когда явная управляемая блокировка может потребоваться на регистре, который читается, но не записывается.

Допустим, в конфигурации есть два регистра: *ОстаткиНаСкладах* и *РезервыНаСкладах*.

В документе *Резерв* двигается только регистр *РезервыНаСкладах*, при этом перед списанием выполняется проверка, что резервируемое количество должно быть больше, чем свободный остаток (*ОстаткиНаСкладах .Остаток – РезервыНаСкладах.Остаток)*.

В документе *Списание* двигается только регистр *ОстаткиНаСкладах*, при этом перед списанием выполняется проверка, что списываемое количество должно быть больше, чем свободный остаток (*ОстаткиНаСкладах .Остаток – РезервыНаСкладах.Остаток)*.

В обоих документах используется старая методика контроля остатков, т.е. до записи движений. В данном случае не рассматривается, насколько правильно или неправильно с методологической точки зрения такое решение, просто оно есть, и необходимо оперативно перевести конфигурацию в управляемый режим.

Если в документе *Резерв* блокировать явной управляемой блокировкой только регистр *РезервыНаСкладах* и не блокировать регистр *ОстаткиНаСкладах*, то может получиться ситуация, когда параллельно с проведением документа *Резерв* документ *Списание* спишет товар с регистра *ОстаткиНаСкладах*. Оба документа проведутся параллельно, и в результате получится, что мы зарезервировали уже списанный товар. Чтобы этого не случилось, нужно в каждом из документов ставить явную исключительную управляемую блокировку на оба регистра.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Возможно, здесь было бы правильнее изменить саму методику списания, но иногда на это просто нет времени, а заказчик требует оперативно перейти на управляемые блокировки.

При переводе конфигурации в управляемый режим очень важно понимать, что вы делаете и зачем вы это делаете. Не нужно действовать наугад или блокировать что-то на всякий случай. Если сомневаетесь, проведите различные опыты и эксперименты, спешка в таком деле нежелательна.

Перевод конфигурации на управляемые блокировки может занимать от нескольких дней до 2-3 недель. Полностью типовые конфигурации переводятся быстро, самописные – медленнее. Также все зависит от опыта и квалификации разработчика, но, несмотря на все трудозатраты, оно того стоит.

Видеоурок. Пример неявной управляемой блокировки

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример установки управляемой блокировки при записи документа
- Как узнать, что была установлена управляемая блокировка.

Видеоурок. Пример явной управляемой блокировки

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример установки явной блокировки на регистр накопления
- Ставится ли в этом случае блокировка на СУБД.

Связь уровня изоляции, блокировок, запросов и режимов работы 1С

Что, когда и насколько блокируется

В данной таблице показано, в каком случае какая блокировка (1С или СУБД) устанавливается и как долго она держится.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Режим	Действие в транзакции	Блокировка			Длительность блокировки
		1С	СУБД		
			8.2	8.3	
Автоматический	Запрос.Выполнить()		**S**		Объектные данные блокируются до конца транзакции, только если запрос возвращает эту строку, иначе блокировка снимается, не дожидаясь конца запроса. Необъектные данные блокируются до конца транзакции, если запрос обработал строку, не важно, вернет он ее или нет. X блокировка всегда держится до конца транзакции.
	Ссылка.ПолучитьОбъект() Ссылка.Реквизит Ссылка.Прочитать()		**S**		
	НаборЗаписей.Прочитать()		**S**		
	Любое изменение данных Записать(), Провести()…		**X**		
Управляемый	Запрос.Выполнить()		**S**		После прочтения строка освобождается сразу, не дожидаясь окончания запроса. При этом не важно, вернет запрос эту строку или нет.
	Ссылка.ПолучитьОбъект() Ссылка.Реквизит Ссылка.Прочитать() НаборЗаписей.Прочитать()		**S**		
	НаборЗаписей.Прочитать()	**P**			Всегда до конца транзакции
	Любое изменение данных Записать(), Провести()…	**И**	**X**		
	Блокировка.Заблокировать()	**P, И**			

Данные в таблице требуют некоторых пояснений.

- В управляемом режиме при использовании уровня изоляции Read Committed при чтении ставится S блокировка. В источниках 1С утверждается, что блокировка будет держаться до конца запроса, однако на практике это не так. S блокировка устанавливается только на время чтения строки. Как только строка прочитана, блокировка сразу же снимается, не дожидаясь окончания запроса, и не важно, вернет запрос эту строку или нет.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Такое поведение логично: незачем блокировать строку до конца запроса, ведь запрос может выполнятся несколько часов.
Уровень Read Committed защищает только от «грязного» чтения, при этом моментом чтения считается не время завершения запроса, а лишь момент чтения строки данных. Проверить это можно с помощью SQL Profiler и событий *Lock:Acquired* (установка блокировки) и *Lock:Released* (снятие блокировки). При этом нужно помнить, что X блокировка СУБД и любые блокировки 1С всегда держатся до конца транзакции. Если такой вопрос будет задан на экзамене 1С:Эксперт, то лучше отвечать, как написано в желтых книжках, что S блокировка для Read Committed держится до конца запроса, хотя это и не так. В принципе, большой роли это не играет, и процесс оптимизации никак не меняется. Поэтому часто по ходу курса в целях упрощения будет говориться, что для Read Committed блокировка при чтении держится до конца запроса

- В таблице строка *НаборЗаписей.Прочитать()* указана два раза, и это сделано намеренно. Метод *Прочитать()* для набора записей устанавливает S блокировку на уровне СУБД (если используется 8.2 или 8.3 в режиме совместимости с 8.2) и разделяемую управляемую блокировку 1С. Время жизни этих блокировок разное, поэтому данный метод записан отдельно для блокировки 1С и для блокировки СУБД
- В колонке блокировка 1С значение **Р** означает разделяемую управляемую блокировку, а значение **И** - это исключительная управляемая блокировка
- Любая управляемая блокировка всегда длится до конца транзакции, независимо от того, является она разделяемой или исключительной. X блокировка СУБД так же, независимо от режима блокировки и уровня изоляции, всегда будет держаться до конца транзакции.

Режим блокировки и гранулярность

Режим блокировки	Гранулярность и уровень изоляции	СУБД		
		Файловая	Блокировочники	Версионники
			MS SQL и DB2	PostgreSQL и Oracle
Автоматический	Минимальная гранулярность	**Таблица**	Строка	**Таблица**
	Уровень изоляции	Serializable	Repeatable read или Serializable	Serializable

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Управляемый	Минимальная гранулярность	**Таблица**	Строка	Строка
	Уровень изоляции	Serializable	8.2 Read committed	Read committed (поведение аналогично RCSI)
			8.3 Read committed snapshot	

Как видно из таблицы, при работе в файловом режиме будет блокироваться вся таблица в любом случае.

Если в автоматическом режиме будет использоваться СУБД «версионник», то также будет блокироваться вся таблица. Например, нельзя будет параллельно проводить документы одного вида, даже если данные в них разные.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Таблица режимов, уровней изоляции и проблем параллельной работы

Режим блокировки	В транзакции	1С	Ур. изол.	Действие	Блокировка	Потер. обн.	Грязн. чтение	Неп. чтение	Фант. чтение
Упр.	Нет	8.2	RU	Чтение		+	-	-	-
		8.3	RCSI	Чтение		+	+	-	-
	Да	8.2	RC	Чтение	S	+	+	-	-
				Запись	X				
		8.3	RCSI	Чтение		+	+	-	-
				Запись	X				
Авт.	Нет	8.2, 8.3	RU	Чтение		+	-	-	-
	Да		RR	Чтение	S	+	+	+	-
				Чтение «ДЛЯ ИЗМЕНЕНИЯ»	U				
				Запись	X				
			S	Чтение	RangeS, S	+	+	+	+
				Чтение «ДЛЯ ИЗМЕНЕНИЯ»	RangeU, U				
				Запись	RangeX, X				

Расшифровка сокращений уровней изоляции транзакции:

RU – Read uncommitted

RCSI – Read committed snapshot isolation

RC – Read committed

RR – Repeatable read

S – Serializable

Как видно из таблицы, если конфигурация работает в автоматическом режиме блокировок, то поведение 8.2 и 8.3 одинаково. Не следует забывать, что если в 8.3 включить режим совместимости с 8.2, то уровни изоляции будут использоваться те же, что и в 8.2.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 31

Основные причины избыточных блокировок

Чаще всего в системе встречаются следующие причины избыточных блокировок:

- Автоматический режим блокировок
- Неоптимальная работа запроса
- Методические ошибки при использовании объектов конфигурации
  - Константы
  - Последовательности
  - Регистры бухгалтерии
  - Регистры накопления
- Блокирующее чтение итогов в начале транзакции
- Выгрузка изменений по плану обмена
- Изменение большого количества данных в одной транзакции
- Эскалация блокировок.

Далее разберем каждую из проблем подробно.

Видеоурок. Автоматический режим блокировок

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример избыточной блокировки пустой таблицы.

Видеоурок. Неоптимальная работа запроса

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Схему блокировки запросом излишних данных
- В каких случаях запрос не накладывает блокировок.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Неоптимальная работа запроса. Пример

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример избыточной блокировки запросом.

Видеоурок. Методические ошибки. Константы

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Структуру хранения констант в разных версиях 1С
- Можно ли параллельно записывать разные константы.

Видеоурок. Методические ошибки. Последовательность

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Структуру хранения объекта метаданных *Последовательность*
- Возможные проблемы при работе с последовательностью.

Видеоурок. Методические ошибки. Регистр накопления. Запись задним числом

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что происходит при изменении остатка за предыдущий месяц, и как это влияет на производительность.

Видеоурок. Методические ошибки. Регистр накопления. Разделение итогов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- Как с помощью разделения итогов повысить параллельность работы с регистром накопления.

Видеоурок. Методические ошибки. Регистр бухгалтерии

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как повысить параллельность записи в регистр бухгалтерии.

Видеоурок. Механизм разделения итогов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Принцип работы разделения итогов
- Включение и выключение разделения итогов.

Видеоурок. Особенности использования разделения итогов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каком случае имеет смысл использовать разделение итогов.

Видеоурок. Разделение итогов в автоматическом режиме

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Работает ли разделение итогов в автоматическом режиме
- Особенности использования разделителя в автоматическом режиме.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Старая методика контроля остатков

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Почему нежелательно сначала контролировать остатки и только потом делать запись.

Видеоурок. Новая методика контроля остатков

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В чем преимущество новой методики
- Требуется ли устанавливать явную управляемую блокировку при использовании новой методики.

Видеоурок. БлокироватьДляИзменения и 8.2

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Зачем нужно свойство набора записей *БлокироватьДляИзменения*, и в каком случае его использовать
- Принцип работы *БлокироватьДляИзменения*
- К чему может привести отсутствие данного свойства у набора записей в 8.2.

Видеоурок. БлокироватьДляИзменения и 8.3

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Возможные ошибки в 8.3, если не указывать *БлокироватьДляИзменения*.

Видеоурок. БлокироватьДляИзменения дополнение

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Нюансы работы свойства *БлокироватьДляИзменения*.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Новая методика и партионный учет

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как избавиться от списания партий при проведении
- Можно ли при партионном учете использовать новую методику.

Видеоурок. Режим «Не удалять автоматически»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Влияет ли режим удаления движений на производительность.

Видеоурок. Режим «Удалять автоматически»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- О влиянии автоматического удаления движений на параллельность работы.

Видеоурок. Режим «Удалять автоматически при отмене проведения»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Почему данный режим используется по умолчанию, и как он влияет на параллельность.

Видеоурок. Оптимизация при перепроведении

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как платформа оптимизирует перепроведение документов.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Выгрузка изменений по плану обмена

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Почему выгрузка изменений может приводить к избыточным блокировкам
- Что блокируется при выгрузке изменений.

Видеоурок. Изменение большого числа данных в транзакции

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Почему нежелательно открывать длинные транзакции
- Почему нежелательно изменять в транзакции большой объем данных.

Видеоурок. Эскалация блокировок СУБД

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как происходит эскалация блокировок СУБД
- Как можно увидеть эскалацию.

Видеоурок. Событие Lock:Escalation

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как увидеть эскалацию на блокировках СУБД в SQL Profiler.

Видеоурок. Эскалация блокировок 1С

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Когда происходит эскалация на сервере приложений
- Как увидеть эскалацию на управляемых блокировках.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 32

Видеоурок. ЦУП. Сбор данных для анализа

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример сбора данных о блокировках с помощью ЦУП
- Рекомендуемое время сбора показателей.

Видеоурок. ЦУП. Анализ блокировок. Пример 1

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример анализа ожиданий на блокировках.

Видеоурок. ЦУП. Анализ блокировок. Пример 2

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример анализа ожиданий на блокировках и исправление найденных проблем.

Видеоурок. ЦУП. Анализ блокировок. Проверка оптимизации

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как проверить эффект от оптимизации блокировок.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. ЦУП. Анализ блокировок СУБД. Воспроизведение

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как воспроизвести ожидание на блокировках СУБД.

Видеоурок. ЦУП. Анализ блокировок СУБД. Расследование

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как блокировки СУБД отображаются в ЦУП, и как их анализировать.

Видеоурок. ЦУП. Анализ блокировок. Пример 3

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример анализа ожиданий на блокировках реальной системы.

Видеоурок. Сервис. Сбор и выгрузка данных

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример сбора и оперативной выгрузки данных в сервис.

Видеоурок. Сервис. Анализ управляемых блокировок

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как анализировать ожидания на управляемых блокировках.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Сервис. Анализ блокировок СУБД

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как анализировать ожидания на блокировках СУБД
- Как анализировать параметры запроса.

Видеоурок. Сервис. Анализ ожиданий. Пример 1

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример анализа ожиданий с одного из реальных проектов.

Видеоурок. Сервис. Анализ ожиданий. Пример 2

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример ошибки сервиса при анализе ожиданий.

Видеоурок. Кто кого заблокировал. Консоль кластера

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как с помощью консоли кластера определить, какой пользователь был заблокирован и кто виновник блокировки
- Особенности отображения при ожиданиях на блокировках 1С и СУБД.

Видеоурок. Кто кого заблокировал. Монитор активности

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как использовать данный инструмент MS SQL Server
- Как по номеру соединения понять, какие пользователи 1С участвуют в ожидании.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Кто кого заблокировал. sys.dm_tran_locks

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что использовать вместо sp_lock
- Как с помощью скрипта понять, какие объекты заблокированы.

Видеоурок. Кто кого заблокировал. Обработка для отображения текущих блокировок MS SQL

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как увидеть заблокированные записи в терминах 1С

Видеоурок. Кто кого заблокировал. Постфактум

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Обзор инструментов, с помощью которых можно увидеть ожидающих пользователей и на блокировках 1С, и на блокировках СУБД.

Ожидания на блокировках. Итоги

Блокировки являются крайне важной темой и, пожалуй, одной из самых сложных для понимания. Для того чтобы разобраться во всех тонкостях работы блокировок и транзакций, необходимо не только хорошо знать теорию, но и проделать самостоятельно большое число различных опытов.

В 8.3 проблем с ожиданиями будет меньше, чем в 8.2, т.к. там используется уровень изоляции RCSI, при котором чтение в транзакции выполняется без установки блокировки.

Если конфигурация работает в автоматическом режиме управления блокировкой данных и в ней наблюдаются ожидания на блокировках, то первое, что нужно сделать, – это перейти на управляемые блокировки. Возможно, одного этого будет достаточно для устранения проблем.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Если после перехода какие-то проблемы останутся, необходимо настроить инструменты и собрать данные, после чего провести анализ.

Следует помнить, что сами по себе инструменты бесполезны. Чтобы их использовать, нужно обладать определенными знаниями.

При использовании регистров нужно стараться максимально использовать возможности разделения итогов. Если по регистру всегда есть контроль остатков, то включать разделение итогов не имеет смысла. Если по регистру иногда контроль остатков нужен, а иногда нет, то необходимо использовать свойство набора записей *БлокироватьДляИзменения* там, где выполняется контроль остатков, иначе возможна взаимоблокировка (8.2) или отрицательный остаток (8.3).

Если пользователи пытаются изменить одни и те же данные и проблема блокировок технически не решается, тогда надо решить ее организационно, либо немного изменить логику работы системы. Например, можно двигать границу последовательности регламентным заданием, а обмены проводить в ночное или обеденное время.

Параллельность работы очень сильно зависит от структуры объекта. Ниже приведена таблица параллельности работы в управляемом режиме блокировок с разными объектами метаданных.

Объект	Особенности	Возможность параллельной записи
Константа	До версии 8.2.14	Нельзя менять константы параллельно
	Начиная с 8.2.14 и выше	Разные константы можно менять параллельно. Одну константу параллельно менять нельзя
Последовательность	Проблемы возможны только при сдвиге границы последовательности	Можно, если отличается хотя бы одно из измерений последовательности
Любые объектные данные (справочник, документ, задача и т.д.)	Ссылка объекта всегда уникальна	Можно записывать разные объекты одного типа
Регистр сведений	Подчинен регистратору	Можно, если регистраторы отличаются
	Независимый непериодический	Можно, если отличается хотя бы одно измерение
	Независимый периодический	Можно, если отличается период или хотя бы одно измерение

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Регистр накопления	Разделитель итогов включен и используется	Можно, даже если значения измерений совпадают
	Разделитель выключен	Можно, если отличается хотя бы одно измерение
	Разделитель выключен. Текущие итоги выключены. Итоги рассчитаны по текущий месяц	Можно, если отличается хотя бы одно измерение
	Разделитель выключен. Текущие итоги выключены. Итоги рассчитаны на дату, которая раньше всех записей таблицы регистра	Можно, если отличается хотя бы одно измерение
	Разделитель выключен. Итоги выключены	Можно, даже если значения измерений совпадают, т.к. запись идет только в таблицу движений, где разделителем является регистратор
Регистр бухгалтерии	Разделитель итогов включен и используется	Можно, даже если счет и значения измерений совпадают
	Разделитель выключен. Текущие итоги используются	Можно, если отличается счет или хотя бы одно измерение
	Разделитель выключен. Текущие итоги выключены. Итоги рассчитаны по текущий месяц	Можно, если отличается счет или хотя бы одно измерение
	Разделитель выключен. Текущие итоги выключены. Итоги рассчитаны на дату, которая раньше всех записей в таблице регистра	Можно, если отличается счет или хотя бы одно измерение
	Разделитель выключен. Итоги выключены	Можно, даже если счет и значения измерений совпадают, т.к. запись идет только в таблицу движений, где разделителем является регистратор

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Регистр расчета в данной таблице специально не указан, т.к. это будет частью домашнего задания.

Таблица приведена для управляемого режима блокировки данных, и возникающие ожидания будут именно на управляемых блокировках. В автоматическом режиме за счет блокировки соседних значений диапазона индекса (Range блокировки) параллельность при работе с необъектными данными будет ниже.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Взаимоблокировки

Занятие 33

Описание проблемы

Взаимоблокировка (deadlock) – это тупиковая ситуация, когда минимум две транзакции ждут друг друга.

Ожидания на блокировках могут быть необходимыми, но взаимоблокировка всегда является ошибкой, и ее всегда можно устранить.

Транзакции не могут «знать», что они находятся в состоянии взаимной блокировки, т.к. каждая из них «думает», что она в состоянии обычного ожидания. Для решения подобной ситуации нужен кто-то третий, наблюдающий со стороны. В роли такого внешнего наблюдателя выступает менеджер блокировок.

В MS SQL Server менеджер блокировок каждые 5 секунд проверяет систему на наличие в ней транзакций, ожидающих друг друга. Если такие транзакции найдены, то сервер завершает ту транзакцию, которую можно быстрее всего отменить. В результате одна транзакция все равно завершится успешно. Если взаимная блокировка была обнаружена, то интервал проверки уменьшается до 100 мс.

Взаимные блокировки также могут возникнуть и на управляемых блокировках. В этом случае менеджер блокировок сервера приложений сразу распознает такие ситуации и выдает сообщение об ошибке транзакции «жертве».

Очень часто после устранения одной взаимной блокировки возникает другая, которая всегда была в системе, но не проявлялась, т.к. при наличии первой взаимоблокировки до второй дело не доходило. По этой причине нужно быть готовым к тому, что трудозатраты на решение проблем взаимоблокировок могут быть несколько больше, чем ожидалось изначально.

На данный момент есть несколько инструментов, с помощью которых можно довольно легко и быстро диагностировать проблемы взаимных блокировок. Как правило, анализ собранных данных не занимает много времени, в большинстве случаев причины взаимоблокировок очевидны. Исключением являются взаимные блокировки на управляемых блокировках, в этом случае придется анализировать технологический журнал вручную.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Есть всего 2 типа взаимоблокировок, которые могут встречаться в различных вариациях:

- **Повышение режима блокировки ресурса**. Другое название – повышение уровня изоляции ресурса, хотя уровень изоляции – понятие, скорее применимое к транзакциям, а не к ресурсам. Данный тип практически никогда не встречается в управляемом режиме блокировок
- **Разный порядок захвата ресурсов.** Наиболее распространенный вид взаимоблокировок, при этом в роли ресурсов могут выступать как разные таблицы/индексы, так и разные строки одной таблицы. Если в системе есть взаимоблокировка на управляемых блокировках, то это, как правило, именно данный тип.

В данном разделе будут рассмотрены как основные, так и более редкие причины возникновения взаимных блокировок.

Видеоурок. Повышение режима блокировки ресурса. Пример 1

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Схему возникновения взаимоблокировок данного типа
- Особенности взаимоблокировок данного типа
- Пример воспроизведения.

Видеоурок. Повышение режима блокировки ресурса. Пример 2

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Приемы написания кода, которые могут привести к взаимоблокировкам
- Пример взаимоблокировки при объектном чтении.

Видеоурок. Повышение режима блокировки ресурса. Пример 3

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример взаимоблокировки данного типа на управляемых блокировках.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Повышение режима блокировки ресурса. Решение

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Приемы программирования, позволяющие избегать взаимоблокировок из-за повышения режима блокировки ресурса
- Пример решения взаимоблокировки данного типа.

Видеоурок. Захват ресурсов в разном порядке. Воспроизведение

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Описание и схему возникновения данного типа взаимоблокировок
- Примеры воспроизведения.

Видеоурок. Захват ресурсов в разном порядке. Решение

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Приемы программирования, позволяющие избегать взаимоблокировок данного типа.

Занятие 34

Видеоурок. БлокироватьДляИзменения

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Схему и описание взаимоблокировки из-за разделителя итогов.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Взаимоблокировка из-за запроса со сканированием

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как неоптимальный запрос может привести к возникновению взаимоблокировки
- Как решить данную проблему.

Видеоурок. Взаимоблокировка из-за распараллеливания

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- В каком случае взаимоблокировка может произойти по вине СУБД
- Как можно решить данную проблему.

Видеоурок. Взаимоблокировка из-за эскалации

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как эскалация может способствовать появлению взаимоблокировок.

Видеоурок. Распределенная взаимоблокировка. Схема

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что такое распределенная взаимоблокировка, и как она может появиться.

Видеоурок. Распределенная взаимоблокировка. Пример

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример воспроизведения распределенной взаимоблокировки.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Сервис анализа взаимоблокировок. Установка и настройка

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как установить и настроить сервис анализа взаимных блокировок
- На что обращать внимание при настройке.

Видеоурок. Сервис анализа взаимоблокировок. Воспроизведение и выгрузка

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как воспроизвести взаимоблокировку и оперативно выгрузить данные в сервис.

Видеоурок. Сервис анализа взаимоблокировок. Анализ

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример взаимоблокировки данного типа на управляемых блокировках.

Видеоурок. ЦУП. Воспроизведение и сбор данных о взаимоблокировках

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример сбора показателей для анализа взаимоблокировок в ЦУП.

Видеоурок. ЦУП. Анализ взаимоблокировок

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример анализа различных типов взаимоблокировок с помощью ЦУП.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Анализ взаимоблокировок на блокировках 1С

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- С помощью какого инструмента производиться анализ взаимоблокировок на управляемых блокировках
- Как проводить анализ таких взаимоблокировок.

Видеоурок. Сервис. Ошибки при анализе

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример ошибочного разбора взаимной блокировки.

Видеоурок. Пример взаимоблокировки из-за разделителя

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример разбора взаимоблокировки из-за включенного разделителя итогов с одного из проектов по оптимизации.

Видеоурок. Пример взаимоблокировки из-за сканирования

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример разбора взаимоблокировки из-за неоптимального запроса с одного из проектов по оптимизации.

Видеоурок. Пример взаимоблокировки из-за разного порядка захвата ресурсов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Наглядный пример взаимоблокировки из-за разного порядка захвата ресурсов.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Взаимоблокировки. Итоги

Взаимоблокировки могут возникнуть как на блокировках СУБД, так и на управляемых блокировках, но, независимо от места их появления, причины возникновения одинаковы. Существуют всего две основные причины взаимных блокировок, но проявляться они могут по-разному.

Для решения проблем с взаимоблокировками используется либо облачный сервис, либо ЦУП. Конечно, можно с помощью SQL Profiler собрать графы взаимоблокировки, но без контекстов вызовов из логов 1С от этого инструмента будет мало пользы.

Как правило, анализ и устранение взаимных блокировок не занимает много времени, но проблема может быть в том, что после устранения одной взаимоблокировки может появиться другая.

Возможна ситуация, когда база находится в автоматическом режиме управления блокировкой данных и там наблюдаются взаимные блокировки. В этом случае возникает вопрос, что делать сначала: переводить конфигурацию на управляемые блокировки или устранять взаимные блокировки?

При переводе на управляемые блокировки часть взаимных блокировок (из-за повышения режима блокировки) может уйти, т.к. запрос будет «отпускать» данные сразу после выполнения, не дожидаясь конца транзакции.

Если проблема была в разном порядке захвата ресурсов (например, разный порядок записи регистров), то взаимоблокировка останется, но будет уже не на сервере СУБД, а на сервере 1С, и расследовать ее будет немного сложнее.

В результате оптимальнее сначала устранить взаимные блокировки в автоматическом режиме и только потом переходить на управляемые блокировки. Много времени борьба с взаимоблокировками не займет, но вам точно не придется разбираться с анализом взаимоблокировок на стороне 1С.

Для того чтобы избежать взаимоблокировок, необходимо придерживаться нескольких простых правил:

Блокируйте данные всегда в одинаковом порядке. Лучше всего, чтобы порядок записи всегда определялся платформой. Если в логике конфигурации так заложено, что данные намеренно блокируются в разном прядке, тогда блокируйте изначально сразу все объекты.

Блокируйте данные сразу с необходимым режимом блокировки. Если вы читаете данные, которые потом будете записывать, то еще при чтении эти данные нужно заблокировать.

Если конфигурация в автоматическом режиме блокировок, тогда нужно в запросе использовать опцию ДЛЯ ИЗМЕНЕНИЯ чтобы поставить U блокировку.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Если используется управляемый режим, то взаимоблокировки не будет. Следует помнить, что при работе с объектными данными в управляемом режиме, если после чтения объект был изменен другой транзакцией, то в момент записи объекта сработает оптимистическая объектная блокировка. В результате появится ошибка, сообщающая о том, что объект уже был изменен.

Оптимизируйте запросы, выполняющиеся в транзакции. Неоптимальные запросы в транзакции могут приводить к взаимным блокировкам из-за захвата «чужих» строк. Эта рекомендация не относится к системам, которые работают в управляемом режиме с использованием СУБД «версионника» либо режима изоляции RCSI.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Приемы ускорения различных операций

Занятие 35

Иногда приходится сталкиваться с ситуациями, когда оптимизация «в лоб» не помогает достичь желаемого результата. В этом случае могут помочь различные нетривиальные варианты решения проблем. Как правило, в таких случаях надо менять сам подход к решению проблемы, попытаться либо избежать узкого места, либо организовать параллельную обработку данных.

Некоторые виды таких проблем и их решения будут описаны в данном разделе.

Многопоточная обработка данных

В платформе 1С уже давно существует возможность выполнять параллельные вычисления (с некоторыми ограничениями). Выполнять многопоточную обработку можно с помощью фоновых заданий, при этом скорость обработки данных может увеличиться в несколько раз по сравнению с однопоточным вариантом.

Пример из практики. Заказчику требовалось перепровести 100 тыс. документов за 8 часов. До начала оптимизации один документ перепроводился за 4,32 сек. и перепроведение 100 тыс. документов заняло бы 5 суток. При заявленных требованиях один документ должен был бы перепроводиться за 0,28 секунды, и при этом документы делали бы движения по регистрам накопления и бухгалтерии. В разумный срок добиться такой скорости крайне сложно. Для решения задачи было реализовано параллельное проведение документов фоновыми заданиями. Алгоритм определял наборы документов, которые не пересекаются по набору значений измерений, чтобы избежать блокировок, и проводил документы с разными значениями в разных потоках. В результате все документы перепровелись менее чем за 8 часов, и заказчик остался полностью доволен результатом.

Многопоточную обработку можно применять практически везде, где работа идет с независимыми блоками данных:

- При проведении документов по разным наборам значений измерений
- При восстановлении последовательности по различным измерениям
- При загрузке/выгрузке большого объема данных
- В любых других задачах, где данные не пересекаются.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

При реализации распараллеливания нужно учитывать мощности вашего оборудования, а также возможность появления эскалации. В случае, если один поток вызовет эскалацию, то остальные потоки будут ждать, даже если обрабатываемые данные не пересекаются.

Также нужно грамотно подбирать число потоков: излишнее число потоков может дать обратный результат и привести к замедлению. Обычно число потоков подбирается индивидуально для каждой системы, но, в большинстве случаев оптимальным является количество 8-10 потоков.

Видеоурок. Распараллеливание. Задача и реализация

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Примеры задач, которые можно выполнить в несколько потоков
- На что обращать внимание при разбиении данных на потоки
- Пример кода для многопоточного выполнения.

Видеоурок. Распараллеливание. Воспроизведение

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример ускорения с помощью многопоточной обработки
- Насколько скорость выполнения зависит от числа потоков.

Видеоурок. Запись в транзакции

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как ускорить запись большого объема данных
- Какие есть нюансы при открытии длинных транзакций.

Видеоурок. Ускорение записи в регистры

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как ускорить запись большого объема данных в регистры
- В каких случаях можно использовать данный способ оптимизации.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 36

Видеоурок. Динамическое считывание включено

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что такое динамическое считывание, и как оно работает.

Видеоурок. Динамическое считывание выключено

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как работает динамический список, если основная таблица указана, но динамическое считывание данных выключено.

Видеоурок. Динамическое считывание включено, и основная таблица не указана

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как работает список с произвольными запросами при отсутствии основной таблицы.

Видеоурок. Общие рекомендации по работе с динамическими списками

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Особенности работы с динамическими списками
- Каких правил придерживаться при разработке динамических списков.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Рекомендации по написанию кода

При написании кода рекомендуется придерживаться нескольких правил, благодаря которым можно избежать некоторых проблем с производительностью.

Минимизировать число клиент-серверных вызовов. Чем реже клиент обращается к серверу, тем меньше затраты на передачу данных. Это особенно критично при использовании веб-клиента. Необходимо по возможности сразу получать весь массив данных для обработки.

Для отслеживания числа клиент-серверных вызовов следует использовать окно показателей производительности, где видны текущие и накопленные серверные вызовы, их длительность и объем отправленных и принятых данных. Следует помнить, что программное изменение формы также приводит к повторным вызовам, т.к. надо заново передавать данные перерисованной формы.

Сократить объем передаваемых данных. Данный пункт не должен противоречить первому пункту: лучше прочитать 1000 строк, но один раз, чем 1000 раз обращаться к серверу и читать по 1 строке. Смысл этой рекомендации в том, что не нужно передавать лишние данные.

Например, не стоит использовать объектное чтение, если можно получить данные запросом. Запрос получит только запрашиваемые поля, а при объектном чтении будут получены все поля объекта, включая все табличные части, даже если идет обращение к реквизиту через точку. Если объект хранит реквизит с типом *ХранилищеЗначений* и там лежит объемный файл, тогда ситуация еще более усугубляется, т.к. этот файл будет постоянно передаваться по сети. Лучше всего в данном случае сделать отдельный регистр и хранить такие крупные объекты там.

Сюда же относятся внеконтекстные вызовы. Если вам не нужны данные формы, то не следует использовать вызовы с контекстом, т.к. это существенно увеличивает объем передаваемых данных.

Получение реквизитов через точку от полей составного типа. При объектном чтении реквизитов полей составного типа на уровне СУБД будет происходить соединение со всеми таблицами, входящими в этот составной тип. В запросе мы можем ограничить число таблиц, с которыми нужно соединиться, но при объектном чтении такой возможности нет, поэтому по возможности лучше всегда читать данные запросом.

Порядок проверки условий в коде. В запросе не важно, в каком порядке заданы поля, но при задании условий в коде это имеет значение. Когда условие задается в коде, то оно проверяется в том порядке, как оно написано (слева направо).

Допустим, есть следующий код:

```
…
Если (Активность=Истина) ИЛИ (ТяжелыйЗапрос.Выполнить().Пустой()) Тогда
…
```


Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

В этом случае, если *Активность=Истина*, то вторая часть условия вообще не будет проверяться и запрос не будет выполнен, т.к. это не повлияет на результат вычисления логического выражения. Если же *Активность=Ложь*, то только тогда будет проверятся вторая часть условия и запрос будет выполнен.

При написании таких условий в самое начало лучше ставить наиболее простые и селективные условия, которые сработают с наибольшей вероятностью.

Поиск большого числа строк из одной таблицы значений в другой таблице значений по сложным условиям. Если таблицы значений большие, то поиск, организованный с помощью кода, может занять много времени. Возможно, в данном случае будет более оптимально поместить таблицы значений во временные таблицы и выполнить поиск запросом с помощью соединения этих таблиц.

Использование *ЗаполнитьЗначенияСвойств().* Если есть 2 набора данных с одинаковыми именами полей, то, чтобы скопировать данные из строки одного набора в строку другого, следует использовать вышеописанную команду. Не нужно для этого запускать цикл или писать дополнительный код.

Получение данных в событиях, связанных с отрисовкой интерфейса. Допустим, логикой работы обработки заложено получение каких-то данных при возникновении событий *ПриАктивацииСтроки(), ОбновлениеОтображения(), ПриВыводеСтроки()* и т.д. В данном случае нужно либо избавиться от такого подхода, либо сделать получение данных максимально быстрым. Например, при листании списка нужно выводить статус документа, который рассчитывается по сложным алгоритмам. Если делать расчет каждый раз при выводе строки, то форма будет работать очень медленно. В данном случае пользователю, скорее всего, не нужны статусы всех документов, а только нескольких конкретных. Для этого можно отображать статус только для выбранного документа по нажатию на кнопку, либо сделать обработчик ожидания, срабатывающий каждые 0,5 секунд и получающий информацию по статусу. Также нужно избегать объектного чтения данных при выводе строки.

Особенности работы с веб-клиентом. В идеале, конечно, лучше использовать тонкий клиент с подключением через веб-сервер. Эта связка будет работать намного быстрее и стабильнее, чем веб-клиент, что позволит избежать целого ряда различных проблем. В большинстве случаев веб-клиент можно заменить тонким клиентом, но, к сожалению, не всегда.

Если в конфигурации планируется использовать веб-клиент, то отладку при разработке тоже лучше вести на веб-клиенте, в этом случае большая часть проблем будет обнаружена еще на этапе разработки. Работе интерфейса надо уделить особое внимание. При использовании веб-клиента абсолютно точное время выполнения действия можно получить лишь вручную с помощью секундомера, т.к. очень многое зависит не только от платформы, но и от самого браузера. Причем замер надо проводить именно на компьютере пользователя.

Наличие большого числа клиент-серверных вызовов особенно критично как раз для веб-клиента.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Использование сложных форм с большим числом элементов, условные форматирования и т.п. также сильно снижают производительность интерфейса. Поэтом надо стремится сделать форму максимально простой и при необходимости разбить одну сложную форму на несколько простых.

Веб-клиент имеет некоторые особенности при работе с ключевым словом *Знач* при объявлении параметров процедур и функций. Дело в том, что при клиент-серверном взаимодействии это ключевое слово означает совсем не то, что при работе внутри одного компьютера, клиентского или серверного. Когда мы используем *Знач* при объявлении параметра серверной процедуры и вызываем ее с клиента, это означает, что значение этого параметра не будет передано обратно на клиент. Если же мы не устанавливаем *Знач*, а обычно так и есть, то происходит следующее.

Допустим, мы вызываем серверную процедуру и передаем в нее массив. Предположим, что на клиенте мы даже не собираемся потом этим массивом пользоваться. Он просто был параметром и на самом деле нам больше не нужен. Но когда серверный вызов закончится, массив будет упакован в XML или JSON (на веб-клиенте), и будет передан обратно на клиент. Таким образом, получается, что передаются лишние данные. Поэтому, если значение является параметром и возвращать его не нужно, следует писать у таких параметров ключевое слово *Знач*.

Есть большие отличия при работе с длительными операциями между тонким клиентом и веб-клиентом. Тонкий клиент будет ждать, пока закончится вызов, каким бы длинным он не был. Время ожидания веб-клиента зависит от типа используемого браузера. Например, при использовании браузера Safari таймаут равен 8 секундам, через этот промежуток времени вызов будет прерван браузером.

Для устранения такой ситуации рекомендуется использовать механизм длительных операций БСП. Тот функционал, который мы хотим выполнить на сервере, вызывается внутри фонового задания. На клиенте подключается обработчик ожидания, который время от времени проверяет, не появился ли ответ сервера. В результате проблема решается, и пользователь может работать, не дожидаясь завершения вызова.

Индексация таблицы значений. Если есть большая таблица значений, по которой необходимо выполнять поиск, то для ускорения поиска можно создать индекс по искомой колонке. На больших объемах данных это может дать заметное преимущество в скорости.

Получение остатков на дату, а не на момент времени. Если требуется получить остаток не на начало месяца, то при прочих равных условиях лучше в качестве параметра для получения остатка передавать значение типа *Дата*, а не *Момент времени*. При использовании момента времени индекс по таблице движений не может быть полностью использован и идет скан. Если указывать дату, то индекс используется и запрос работает гораздо быстрее.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Тестирование

Занятие 37

При внедрении в организации с большим числом пользователей очень важно провести предварительное тестирование. Этапа тестирования все равно не избежать и тут возможны два варианта развития событий. Либо в качестве тестировщиков будут выступать реальные пользователи в первые дни работы в новой системе, и тогда всплывут все ошибки и недоработки, которые повлияют и на работу компании и на репутацию IT отдела. Либо тестирование будет проводиться виртуальными пользователями еще до внедрения системы с помощью специальных инструментов, и явные ошибки и узкие места можно будет сразу исправить.

Само тестирование занимает немного времени, но вот подготовка и организация этого процесса могут быть довольно трудозатратными. Тем не менее, тестирование оправдано для тех компаний, где бесперебойная работа системы имеет первостепенное значение. Например, планируется запуск системы, где будет одновременно работать 1000 пользователей, и при этом цена одного часа простоя может измеряться сотнями тысяч рублей. В данном случае для сокращения возможных издержек лучше потратить время и деньги на предварительное тестирование, чем считать убытки из-за программных ошибок.

Существуют различные виды тестирования, но основные – это нагрузочное и функциональное тестирование.

Нагрузочное тестирование позволяет оценить производительность, стабильность и масштабируемость информационной системы.

Функциональное тестирование позволяет выявить ошибки в логике работы системы и недостаток функционала.

В данном разделе в общих чертах будет рассмотрена методика подготовки и проведения нагрузочного тестирования.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Методика проведения нагрузочного тестирования

Общая схема

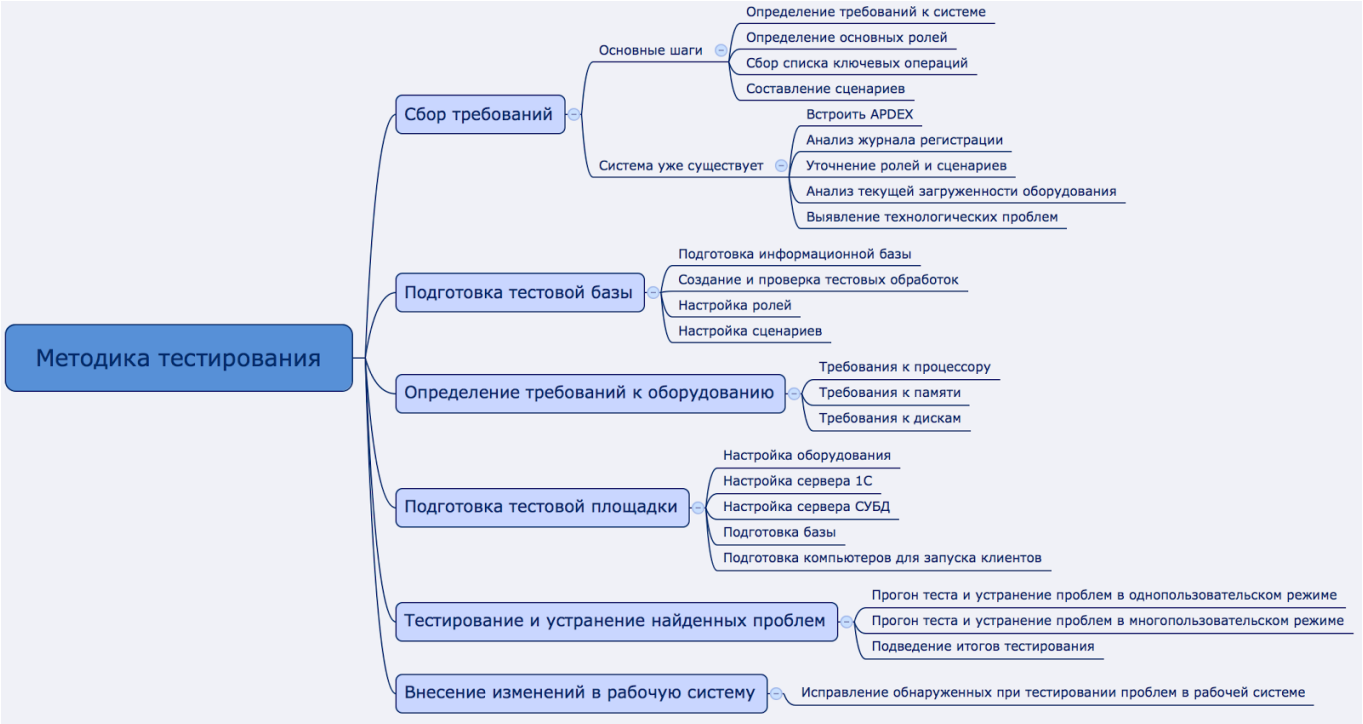
Нагрузочное тестирование необходимо для того, чтобы обеспечить технологическое качество работы системы.

Под технологическим качеством понимается:

- Доступность
- Стабильность
- Устойчивость
- Работоспособность
- Производительность
- Масштабируемость
- Отсутствие падения показателей при изменении системы.

Если система не имеет какой-то части функционала, которая требуется пользователям (например, нет нужного отчета), то это не является проблемой технологического качества.

Общая схема проведения тестирования выглядит следующим образом.



(изображение можно увеличить без потери качества)

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Разберем каждый из пунктов подробно.

Сбор требований

На данном этапе нужно сформулировать требования к системе. Например, требования к производительности (такие-то операции должны выполняться за такое-то время) и к стабильности (отсутствие падений и зависаний на протяжении такого-то времени).

В первую очередь необходимо собрать основную информацию о системе, а для этого нужно ответить на следующие вопросы:

- Какую версию платформы планируется использовать?
- Какая СУБД будет использоваться?
- На какой операционной системе (или системах) будет развернута информационная система?
- Каково максимальное число одновременно работающих пользователей?
- Будет ли меняться нагрузка на систему в разные периоды времени?
- Есть ли достаточное количество лицензий для проведения тестирования?
- Какие предъявляются требования к производительности и стабильности системы?
- Каковы сроки проведения тестирования?

Далее нужно определить основные роли. Под ролью понимается некая группа пользователей, для которой характерна определенная совокупность действий. Например, роль «Кладовщик» или роль «Менеджер». Каждая из ролей выполняет характерный только для нее список операций, однако некоторые операции могут выполняться разными ролями.

Затем необходимо определиться со списком ключевых операций в системе. Обычно невозможно учесть все операции, которые выполняются пользователями, да это и не нужно: достаточно учесть 20% операций, которые создают 80% нагрузки (правило Парето).

Далее необходимо создать сценарий, т.е. описать, сколько пользователей и под какой ролью будет работать, а также какие операции они будут выполнять. В результате, сценарий может выглядеть примерно следующим образом.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Роль	Количество пользователей	Ключевая операция	Количество операций на 1 пользователя в час
Кладовщик	2	Проведение документа «Приход»	1
		Проведение документа «Перемещение»	3
		Проведение документа «Отгрузка»	10
		Формирование отчета «Остатки на складах»	3
Оператор	4	Проведение документа «Продажа»	17
		Проведение документа «Заказ»	10
Менеджер	1	Проведение документа «Заказ»	21
		Создание элемента справочника «Контрагенты»	2
		Формирование отчета «Остатки на складах»	4

В идеале, число пользователей в сценарии должно совпадать с максимальным числом активно работающих пользователей в рабочей системе. Конечно, это не всегда осуществимо, тогда число пользователей в процессе тестирования должно быть максимально возможным.

Чем точнее будет сценарий, тем точнее тестирование будет моделировать нагрузку в реальной системе. Добиться 100% воспроизведения работы пользователей практически невозможно, поэтому здесь нужно постараться отделить главное от второстепенного. Иногда на составление сценария и согласование его с заказчиком может уйти довольно много времени. Сценарий обязательно должен содержать все ключевые операции.

Если информационная система уже существует, то для составления сценария можно использовать данные из этой системы. Например, можно проанализировать журнал регистрации и понять, какие действия и с какой частотой выполняют различные группы пользователей. Максимально точно определить частоту выполнения ключевых операций можно с помощью APDEX. В этом случае можно использовать данные из регистра замеров. Так же ничто не мешает проанализировать существующую систему с помощью сервисов, ЦУП или иных инструментов. Таким образом можно даже до тестирования определить некоторые неоптимальности и в дальнейшем с помощью собранной информации уточнить и подкорректировать сценарий.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Подготовка тестовой базы

Данный этап можно разбить на несколько составляющих:

- Подготовка тестовой базы
  - Создание копии рабочей базы или заполнение пустой базы тестовыми данными
  - Объединение тестовой базы с конфигурацией «Тест-Центр»
- Создание обработок, эмулирующих действия пользователя
- Создание ролей и сценариев.

Желательно за основу взять копию реальной базы. Если система только проектируется, то необходимо наполнить базу тестовыми данными. Лучше всего заполнить базу с запасом, т.е. ввести немного больше данных, чем есть на текущий момент. Также надо подготовить наборы данных, которыми будут оперировать роли. Если операторы проводят документы одного вида, то надо предусмотреть, чтобы данные в этих документах либо не пересекались, либо пересекались частично. Не должно быть ситуации, при которой все пользователи работают с одним набором данных, ведь в рабочей системе такого не бывает.

Сценариев лучше всего создать несколько:

- Отдельный сценарий для каждой роли, в каждой роли по одному пользователю
- В одном сценарии все роли, в каждой роли по одному пользователю
- В одном сценарии все роли, в каждой роли половина пользователей от требуемой нагрузки
- В одном сценарии все роли, в каждой роли все пользователи этой роли.

Такой подход позволит увеличивать нагрузку постепенно, и благодаря этому тест будет проще в отладке и настройке.

Объединение базы с «Тест-Центром», а также создание обработок, ролей и сценариев будет рассмотрено в видеоуроках.

Определение требований к оборудованию

Суть методики заключается в том, что тест проводится на эталонном оборудовании, замеряется загруженность оборудования и потом, исходя из замеров производительности и загруженности, вычисляются параметры целевого оборудования.

Выбор эталонного оборудования зависит от того, существует ли уже система, или она только проектируется. Если система существует, тогда в качестве эталонного можно взять текущее оборудование. Если система только проектируется, тогда можно ориентироваться на информацию об оборудовании с различных проектов ЦКТП http://v8.1c.ru/expert/cts/serv.html.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Эталонная система должна быть максимально близка к целевой, в ней должна использоваться та же СУБД, та же версия 1С и ОС и т.д.

На основании данных о загруженности оборудования во время теста можно путем экстраполяции вычислить, какие параметры должны быть у целевого оборудования.

Процессор

Для расчета физического количества ядер целевого сервера можно использовать следующую формулу:

Число ядер целевого сервера = Число ядер эталонного сервера * Средняя загруженность процессора эталонного сервера / 100 / Количество пользователей эталонной системы * Количество пользователей целевой системы

Например, планируется запуск системы, где в момент пиковой нагрузки будет 1 000 одновременно работающих пользователей. Тестирование было проведено на 100 пользователях, при этом на эталонном сервере 1С использовалось 4 ядра процессора, и процессор был загружен в среднем на 27 %. На эталонном сервере СУБД было 8 ядер, и процессор был загружен в среднем на 15 %. В результате для целевых серверов с нагрузкой в 1 000 одновременно работающих пользователей получим следующие данные:

Число ядер целевого сервера 1С = 4 * 27 / 100 / 100 * 1 000 = 10,8

Число ядер целевого сервера СУБД = 8 * 15 / 100 / 100 * 1 000 = 12

Значения могут получиться дробными, поэтому необходимо округлить их в большую сторону. Конечно, нет процессоров с числом ядер равным 11, в этом случае необходимо найти процессор, у которого ядер больше, чем 11, но ни в коем случае не меньше.

Частота процессора также очень важна, она напрямую влияет на скорость работы системы, особенно это критично для сервера приложений. Если используется тонкий клиент, то частота процессора особенно критична. Желательно использовать процессоры с частотой 3ГГц и выше. Процессоры с частотой 2,5 ГГц можно рассматривать только как минимальный порог.

Диск

С дисками все несколько сложнее, чем с процессорами, в связи с чем расчетные показатели получаются менее точными (более грубая оценка данных). За основу расчета берется процент активности диска, который можно получить, используя одноименный счетчик системного монитора. С помощью полученных данных рассчитывается, во сколько раз более производительная дисковая подсистема требуется для целевой системы.

Целевая производительность = % активности (загруженности) диска эталонной системы / 100 * Количество активных пользователей целевой системы / Количество активных пользователей эталонной подсистемы

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Например, планируется запуск системы на 1 000 пользователей, тестирование было проведено на 100 пользователях, при этом загруженность диска на сервере 1С составила 22 %. Подставив значения в формулу, получим:

Целевая производительность = 22 / 100 * 1 000 / 100 = 2,2

Получается, что для 1 000 пользователей на сервере 1С нужна в 2,2 раза более производительная дисковая подсистема.

Производительность дисковой подсистемы измеряется в операциях ввода / вывода за единицу времени. Для сравнения производительности дисков лучше не доверять информации от производителей, а использовать специальные утилиты, например, SQLIO или CrystalDiskMark.

Память

Чтобы подсчитать, сколько памяти потребляет сервер 1С, необходимо сложить объем памяти, потребляемый всеми процессами сервера приложений, а именно: *ragent*, *rphost* и *rmngr*.

Чтобы понять, сколько памяти занимает сервер СУБД, следует использовать счетчик системного монитора Total Server Memory (KB).

В результате, чтобы рассчитать требуемый объем памяти для целевой системы, можно использовать следующую формулу:

Требуемый объем памяти МБ = Объем занятой памяти / Количество пользователей эталонной системы * Количество пользователей целевой системы + Память для операционной ситемы

Например, сервер СУБД во время работы теста на 100 пользователей занял 11 200 МБ оперативной памяти. В таком случае для 1 000 пользователей потребуется:

Требуемый объем памяти МБ = 11 200 / 100 * 1 000 + 2 048 = 114 048 Мб = 111,3 ГБ

Показания загруженности оборудования необходимо снимать только за период выполнения теста, при отсутствии какой-либо посторонней нагрузки. При снятии параметров загруженности диска также необходимо следить за тем, чтобы в системе была свободная память, иначе загруженность диска будет вызвана просто нехваткой памяти.

Следует отметить, что все вышеперечисленные формулы носят лишь рекомендательный характер.

Также нужно учитывать, что нагрузка на оборудование в очень большой степени зависит от качества программного кода конкретной конфигурации. После оптимизации нагрузка на оборудование может как снизиться (например, если оптимизировали запросы), так и возрасти (например, если убрали ожидания на блокировках).

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Подготовка тестовой площадки

На данном этапе необходимо развернуть тестовый стенд и в это понятие входят следующие действия:

- Подготовка и настройка оборудования
  - Настройка счетчиков загруженности оборудования
  - Настройка виртуальных машин (при необходимости)
  - Настройка веб-серверов (при необходимости)
- Настройка СУБД
  - Установка параметров сервера СУБД
  - Выполнение регламентных операций
- Настройка сервера приложений 1С
- Установка и настройка подготовленной тестовой базы
- Настройка инструментов мониторинга производительности
  - Настройка сервисов или ЦУП
- Подготовка и настройка компьютеров, на которых будут запущены виртуальные пользователи.

Если используется виртуальная машина, то необходимо проследить за тем, чтобы аппаратные ресурсы, выделенные данной виртуальной машине, были строго зафиксированы.

Тестирование и устранение найденных проблем

После настройки необходимо выполнить само тестирование.

Не рекомендуется сразу запускать полноценный сценарий, лучше всего это делать постепенно увеличивая нагрузку. Это может значительно упростить анализ в случае возникновения проблем. Ранее рекомендовалось создать четыре сценария, вот их и надо поочередно запускать.

Сначала рекомендуется выполнить сценарии с одной ролью и одним пользователем, собрать данные о производительности, определить и исправить узкие места. Как только тест в однопользовательском режиме будет показывать приемлемые результаты, следует перейти к следующему сценарию и т.д.

При тестировании могут проявиться как проблемы производительности, так и проблемы стабильности. Как бороться с проблемами производительности уже было разобрано в предыдущих разделах. Вопросы стабильности будут разбираться далее по курсу.

На данном этапе необходимо выявить и исправить все найденные неоптимальности. В итоге тест должен быть полностью работоспособен и все ключевые операции должны иметь требуемое значение APDEX.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

В результате данного этапа необходимо зафиксировать, какие проблемы были обнаружены и как они были решены.

Ключевая операция	Проблема	Решение	APDEX		Среднее время		Трудозатраты
			до	после	до	после	

«До» и «после» – это значения до и после оптимизации.

Внесение изменений в рабочую систему

Если тестирование выполнялось для уже существующей системы, то необходимо внести исправления для найденных проблем в рабочую систему. Не обязательно вносить все исправления сразу, лучше это делать поэтапно. Даже если проблема проявляется только во время тестирования, исправление все равно рекомендуется внести.

Занятие 38

Тест-центр

Описание и схема работы

Тест-центр – это специальная конфигурация, предназначенная для автоматизации многопользовательских нагрузочных испытаний. С ее помощью можно моделировать работу системы без участия реальных пользователей.

С помощью «Тест-центра» можно:

- Описать многопользовательский сценарий любой сложности
- Автоматически запускать тест и контролировать его выполнение
- Собирать и анализировать результаты тестирования.

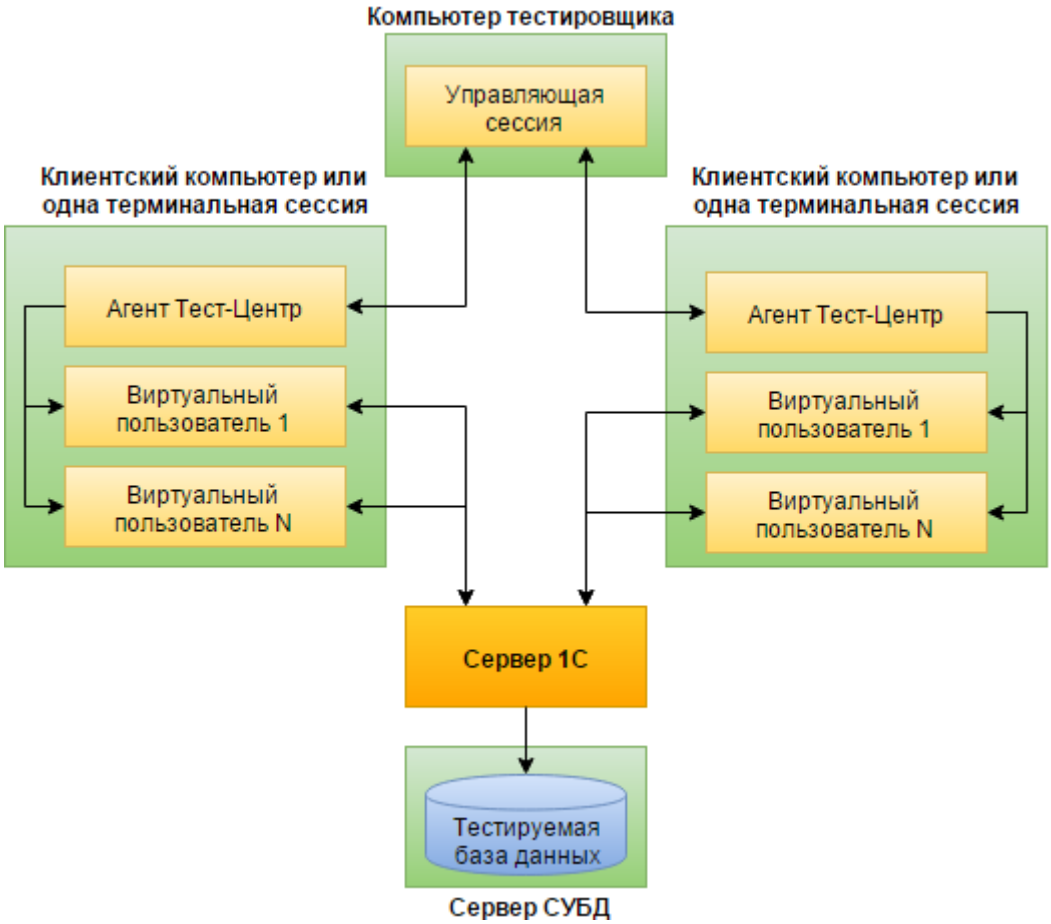
Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Для работы конфигурацию «Тест-центр» необходимо объединить с исследуемой конфигурацией.

Схема работы «Тест-Центра» выглядит следующим образом:



В работе «Тест-центра» участвуют следующие компоненты:

Управляющая сессия – запускает процесс тестирования и отображает результаты теста. Управляющая сессия взаимодействует только с агентами и может быть только одна. Ничто не мешает запустить эту сессию на одном из компьютеров, где запущены виртуальные пользователи. Управляющая сессия обязательно должна быть запущена в управляемом приложении.

Агент – это сессия, которая управляет виртуальными пользователями. Именно агент запускает и завершает работу виртуальных пользователей. Обычно на одном клиентском компьютере или в одной терминальной сессии только один агент. Агент может быть запущен как в управляемом, так и в обычном приложении.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Виртуальные пользователи – это сессии, которые эмулируют работу реальных пользователей. Виртуальные пользователи не взаимодействуют между собой, они взаимодействуют только с агентом и сервером приложений. Число виртуальных пользователей ограничено только количеством лицензий, возможностями оборудования и операционной системы. Если используется терминальный сервер, то желательно не делать в одной терминальной сессии более 100 виртуальных пользователей. Для каждой терминальной сессии нужно будет запустить своего агента. Виртуальные пользователи могут быть запущены как в управляемом, так и в обычном приложении.

Видеоурок. Объединение конфигурации с «Тест-Центром»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как встроить конфигурацию «Тест-центр» в исследуемую базу
- Какие дополнительные действия необходимо выполнить после объединения.

Видеоурок. Пример простого теста. Создание обработки

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как с помощью обработки запрограммировать простейший сценарий.

Видеоурок. Пример простого теста. Создание сценария

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Какие справочники необходимо заполнить для создания сценария
- Как создать роли и настроить обработку
- Как создать и настроить сценарий.

Видеоурок. Пример простого теста. Запуск сценария

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как запустить сценарий на выполнение, и что при этом нужно учитывать
- Надо ли делать контроль остатков при выполнении тестирования.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Встраивание замеров

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как встроить замеры времени в выполняемые операции.

Видеоурок. Отладка виртуальных пользователей

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как можно отладчиком подключиться к виртуальным пользователям
- Как настроить автоматическое подключение виртуальных пользователей к отладчику.

Занятие 39

Автоматизированное тестирование

Принцип работы автоматизированного тестирования

Начиная с версии 8.3 для управляемого приложения появилась поддержка автоматизированного тестирования.

Автоматизированное тестирование – это имитация интерактивных действий пользователя и проверка результатов этих действий.

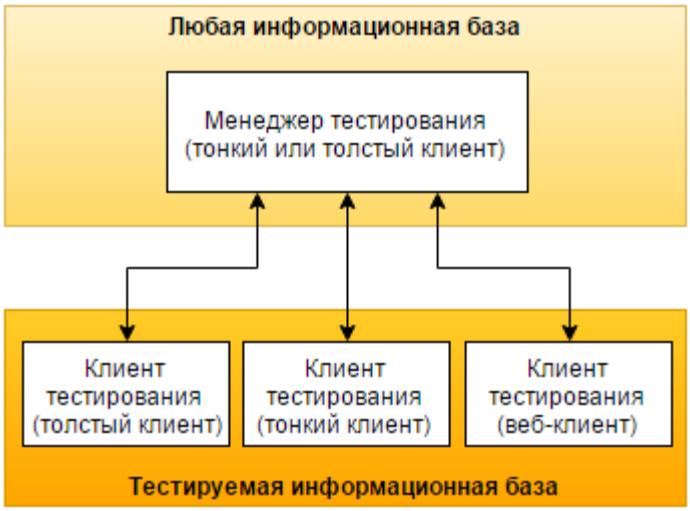
Данный механизм никак не связан с «Тест-центром», он присутствует в платформе по умолчанию.

Автоматизированное тестирование представляет собой взаимодействие двух клиентских приложений, одно из которых является менеджером тестирования, а другое – клиентом тестирования.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



Один менеджер тестирования может быть одновременно подключен к нескольким клиентам тестирования, но один клиент может быть подключен только к одному менеджеру.

Менеджер тестирования отвечает за:

- Установку соединения с клиентом тестирования
- Выполнение алгоритма теста и отправку команд клиентам для выполнения интерактивных действий
- Оценку результатов теста (при необходимости).

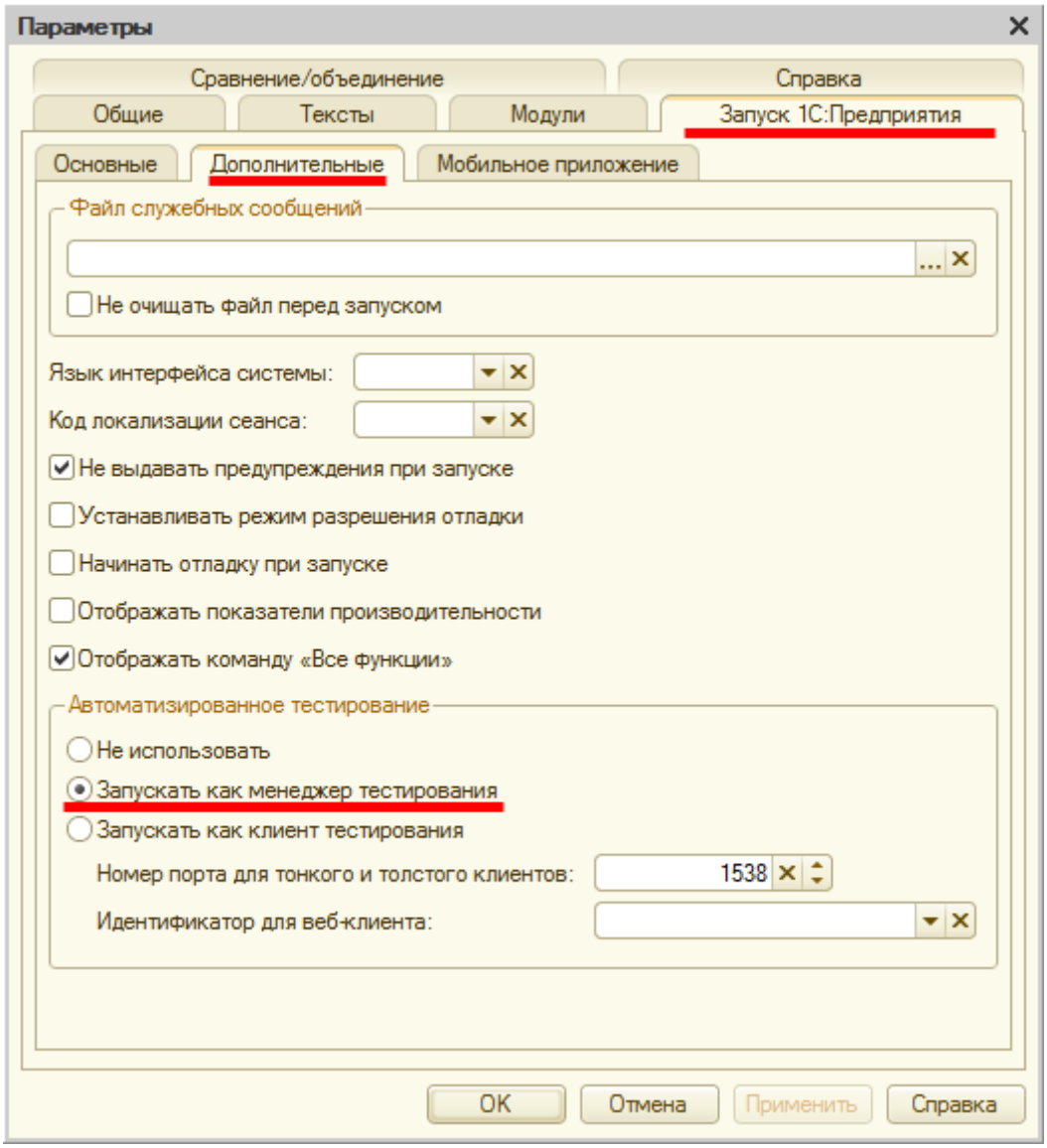
В качестве менеджера тестирования можно запустить как тонкий, так и толстый клиент. Есть два способа запуска менеджера тестирования: с помощью конфигуратора и с помощью ключа командной строки.

Для запуска из конфигуратора необходимо зайти в меню *Сервис – Параметры* и установить соответствующую настройку.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



После установки данного параметра приложение будет запускаться из конфигуратора как менеджер тестирования.

Второй способ – использовать ключ командной строки */TestManager*. Например, чтобы запустить тонкий клиент в базе «Test» как менеджер тестирования, следует выполнить следующую команду:

```
1cv8c ENTERPRISE /IBName "Test" /TestManager
```

Менеджер тестирования может быть запущен в любой информационной базе и на любом компьютере, это не обязательно должна быть тестируемая база и тот компьютер, на котором будут запущены клиенты.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Клиент тестирования отвечает за:

- Выполнение команд менеджера тестирования
- Передачу менеджеру данных, которые требуются для оценки результатов теста.

Клиент тестирования должен быть запущен в тестируемой базе как толстый, тонкий или как веб-клиент.

Для запуска в режиме клиента тестирования необходимо использовать ключ командной строки */TestClient -Tport 1843*. Параметр *Tport* не является обязательным, его следует указывать в том случае, если на одном компьютере используется несколько клиентов тестирования, при этом у каждого клиента должен быть свой уникальный, не занятый другим приложением, порт. По умолчанию используется порт 1538. Если указать порт, который уже занят, то сам клиент запустится, но при попытке менеджера подключиться к этому клиенту возникнет исключительная ситуация. По этой причине не следует использовать номера портов, которые используются другими приложениями. Также следует помнить, что сервер приложений по умолчанию использует порт 1540, а кластер по умолчанию использует порт 1541, поэтому на этих портах клиенты тестирования работать не будут.

При использовании порта по умолчанию (1538), была замечена следующая особенность: менеджер тестирования иногда зависает при попытке получить главное окно. Поэтому лучше использовать порт отличный от 1538, тогда менеджер всегда работает без проблем.

Чтобы запустить тонкий клиент как клиент тестирования на 1529 порту, нужно выполнить следующую команду:

```
1cv8c ENTERPRISE /IBName "Test" /TestClient -Tport 1529
```

Таким образом менеджер тестирования может однозначно идентифицировать клиента по имени компьютера и номеру порта.

Сценарий

Сценарий в автоматизированном тестировании – это код на языке 1С, который имитирует интерактивные действия пользователя. Сценарий можно либо написать полностью вручную, либо записать интерактивные действия пользователя и на их основе сгенерировать код сценария.

Для реализации сценария в платформе введена поддержка объекта *ТестируемоеПриложение*, а также поддержка новых типов данных. Далее на примерах будут рассмотрены возможности использования данного объекта и его методов.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Автоматизированное тестирование. Создание сценария вручную

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как создать сценарий для автоматизированного тестирования
- Как использовать навигационные ссылки в сценарии
- Как запустить сценарий на выполнение.

Видеоурок. Автоматизированное тестирование. Упрощение сценария

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Обязательны ли все команды, описанные в сценарии
- Как можно упростить сценарий.

Видеоурок. Запись действий пользователя

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как запустить приложение для записи действий пользователя
- Как записать все интерактивные действия пользователя.

Видеоурок. Создание сценария по записанным действиям

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как создать сценарий на основе записанных действий
- Настройки формы создания сценария
- Как встроить сценарий в обработку тестирования.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Выполнение записанного сценария

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Требует ли записанный сценарий отладки
- Особенности работы записанного сценария.

Видеоурок. Сценарий с двумя клиентами тестирования

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Может ли один менеджер тестирования управлять более чем одним клиентом тестирования
- Особенности выполнения кода при работе нескольких клиентов тестирования.

Тестирование. Итоги

Тестирование – процесс довольно трудоемкий и затратный, но приносит свои плоды. Даже минимальное тестирование помогает избежать многих проблем при внедрении системы или при обновлении. Тестирование помогает выявить не только узкие места в коде, но и определить требования к оборудованию.

Для проведения многопользовательских нагрузочных тестов обычно используется специальная конфигурация «Тест-центр», входящая в состав КИП. Для работы с «Тест-центром» его необходимо встроить в исследуемую базу и запрограммировать специальные обработки, которые будут создавать нагрузку и выполнять действия, соответствующие определенной роли. При выполнении теста очень важно собирать показатели производительности, обычно для этого используется подсистема APDEX, но время выполнения также можно замерить инструментами самого «Тест-центра».

Если необходимо замерить интерактивные действия, например, открытие формы или проведение документа из формы журнала документов, тогда окончание замера необходимо встраивать в код обработчика ожидания в процедуре *ТЦВыполнить()* после выполнения операции. Если ключевая операция начинается и завершается на сервере, то следует завершать замер процедурой *ОценкаПроизводительностиКлиентСервер.ЗакончитьЗамерВремени(КлючеваяОперация, ВремяНачала)*.

Тестирование также можно проводить с помощью автоматизированного тестирования, которое появилось в версии 8.3 и доступно только для управляемого приложения.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Данный функционал позволяет воспроизводить интерактивные действия пользователя практически со 100 % детализаций. При желании действия пользователя можно записать и потом воспроизвести с помощью сценария. Данный функционал доступен в платформе по умолчанию и не требует установки каких-либо дополнительных компонентов.

Ничто не мешает использовать оба этих инструмента одновременно, например, при желании можно запустить виртуальное рабочее место в качестве клиента или менеджера тестирования, указав соответствующий ключ в параметрах клиента.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Кластер серверов

Занятие 40

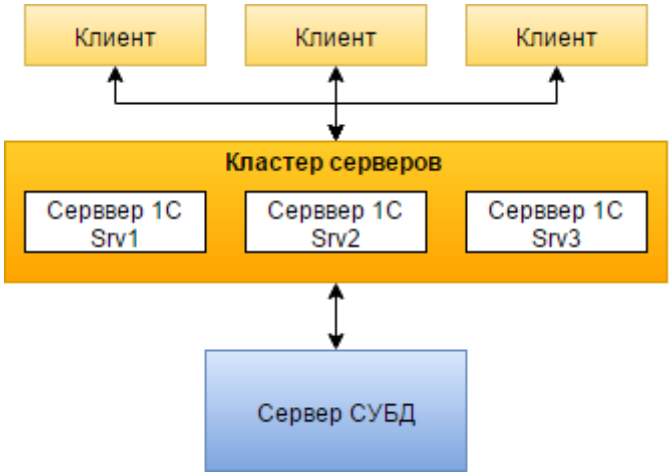
Кластер серверов – это основной компонент платформы, обеспечивающий взаимодействие между пользователями и СУБД. Также он обеспечивает возможность горизонтального масштабирования и отказоустойчивой работы платформы.

Кластер состоит из большого числа различных компонентов, каждый из которых, в свою очередь, имеет ряд настроек. В данном разделе будет подробно разобрано устройство и настройки кластера как для платформы 8.2, так и для 8.3.

Основные понятия

Общая схема работы кластера

Кластер называется именно так, потому что может состоять из нескольких компьютеров. Схему кластера можно представить следующим образом:



Как видно из схемы, клиентские приложения не взаимодействуют напрямую с сервером СУБД, вся работа идет через кластер серверов.

Возможное количество серверов в кластере не ограничено.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

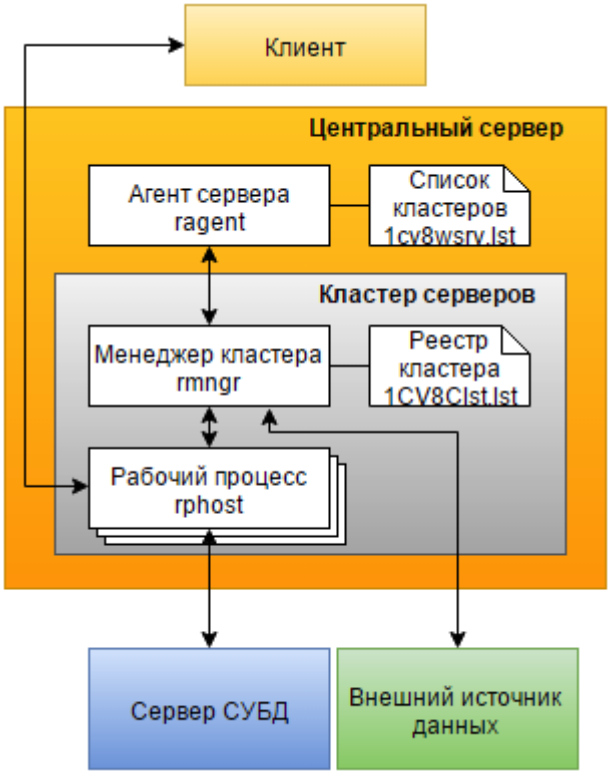
Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Серверы, входящие в состав кластера, называют рабочими серверами. В 8.2 один из рабочих серверов является центральным, в 8.3 центральными могут быть все рабочие серверы.

Чтобы клиент мог подключится к кластеру, он должен знать имя компьютера центрального сервера кластера (или его IP адрес) и номер порта кластера, который указывается через двоеточие. Заметим, что номер порта можно и не указывать, по умолчанию для кластера используется порт 1541.

Как правило, на одном компьютере запущен один рабочий сервер, но ничто не мешает запустить на одном компьютере сразу несколько рабочих серверов, при этом номера портов у них должны различаться.

На рисунке представлена подробная схема простейшего кластера платформы 8.3, состоящего из одного рабочего сервера и, т.к. сервер один, он же является центральным.



Работу кластера обеспечивают следующие процессы:

- **ragent.exe (агент сервера)** – это процесс, который необходимо запустить на компьютере, чтобы компьютер мог быть включен в состав кластера. Можно сказать, что рабочий сервер – это компьютер, на котором запущен процесс ragent. Данный процесс хранит список кластеров, для которых этот рабочий сервер является центральным. Список кластеров хранится в файле 1cv8wsrv.lst (в 8.2 файл называется srvribrg.lst).

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Как видно из схемы, фактически ragent не является частью кластера. Данный процесс также отвечает за работу с серверным ключом защиты. Фактически данный процесс и есть сервер 1С, он может быть запущен как служба или как отдельное приложение. Рекомендуется запускать его именно как службу, при этом пользователю, от имени которого запускается эта служба, должны быть назначены следующие локальные политики безопасности:

- Вход в качестве сервиса (Log on as a service)
- Вход в качестве пакетного задания (Log on as a batch job)

Также пользователь должен входить в группу «Пользователи журналов производительности (Performance Log Users)»

- **rmngr.exe (менеджер кластера)** – данный процесс отвечает за работу кластера. Если менеджер запущен на центральном сервере, то он называется главным менеджером кластера и ведет реестр кластера, который хранится в файле 1CV8Clst.lst (в 8.2 файл называется 1CV8Reg.lst). В реестре кластера хранится информация о зарегистрированных базах этого кластера и прочих служебных данных. В 8.3 центральных серверов может быть несколько, следовательно, и реестр кластера будет создан на каждом из них. В 8.2 центральный сервер может быть только один. В реестре кластера хранится следующая информация:
  - Список баз кластера
  - Список рабочих серверов и рабочих процессов кластера
  - Список сервисов и менеджеров кластера
  - Список администраторов кластера.

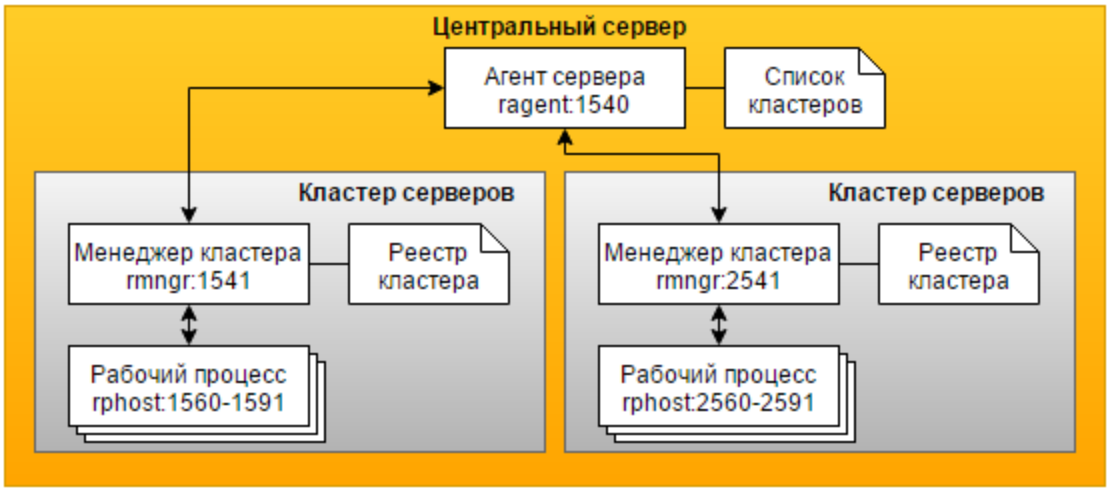
  Также менеджер кластера выполняет и другие функции, среди которых особо стоит выделить следующие:
  - Взаимодействует с другими серверами, входящими в кластер
  - Обеспечивает работу менеджера управляемых блокировок
  - Отвечает за автонумерацию
  - Выполняет запросы к внешним источникам данных.
- **rphost.exe (рабочий процесс)** – данный процесс обслуживает клиентские приложения, выполняет прикладной код, помеченный как исполняемый на сервере, и взаимодействует с сервером СУБД. Количество рабочих процессов может быть различным, это определяется настройками, которые зависят от версии платформы, а также аппаратной конфигурацией компьютера. Как правило, основная нагрузка ложится именно на рабочие процессы и именно они обычно потребляют больше всего ресурсов сервера.

На одном центральном сервере может располагаться несколько кластеров.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



Через двоеточие на рисунке указаны номера портов. По умолчанию для центрального сервера (агента) используется порт 1540, для менеджера (rmngr) порт 1541. В строке подключения к базе данных указывается именно порт менеджера кластера.

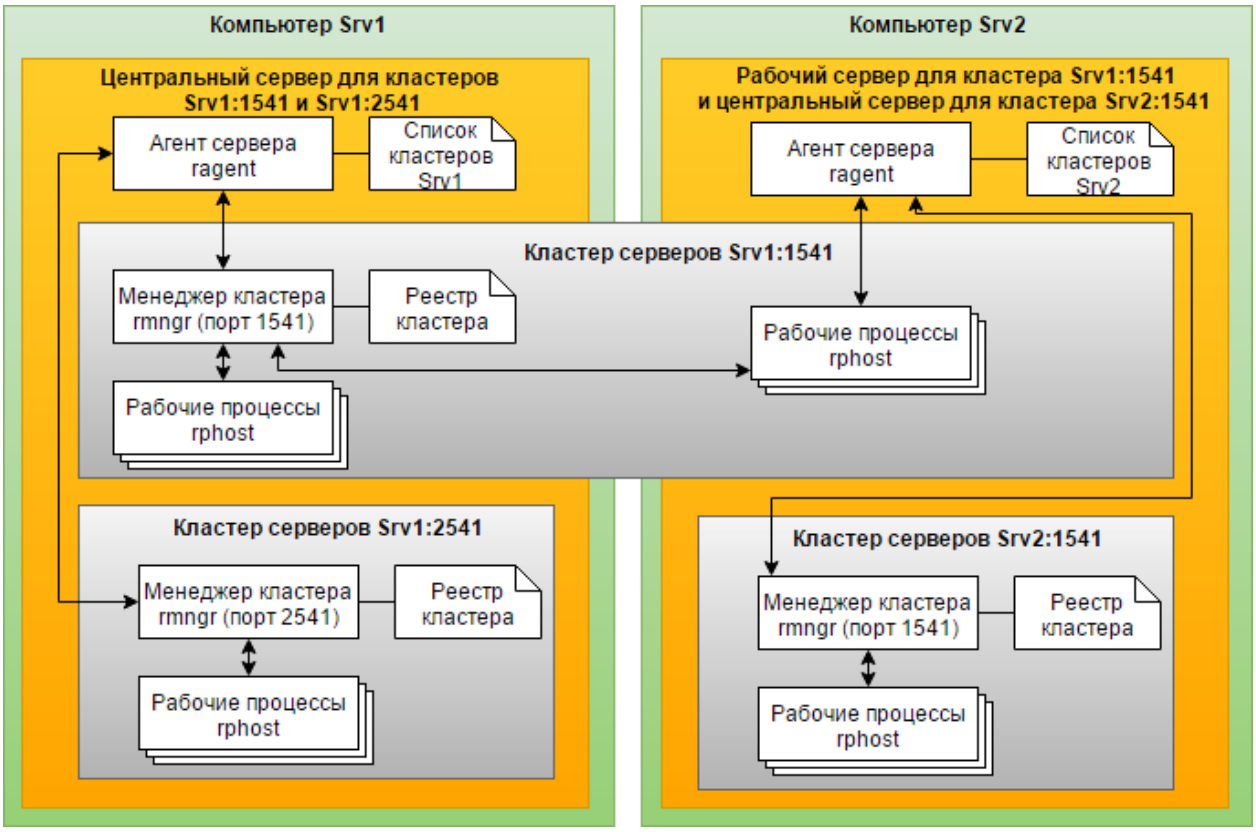
Для рабочих процессов используется диапазон портов, по умолчанию – от 1560 до 1591. При запуске рабочий процесс будет выбирать свободный порт из указанного диапазона. Номера портов можно задать при создании кластера и рабочего сервера. После создания кластера и сервера поменять их порты нельзя, но в любой момент можно изменить диапазон портов для рабочих процессов.

Один кластер может размещаться на нескольких серверах, при этом один сервер может быть центральным для одного кластера и в то же время рабочим сервером для другого кластера.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

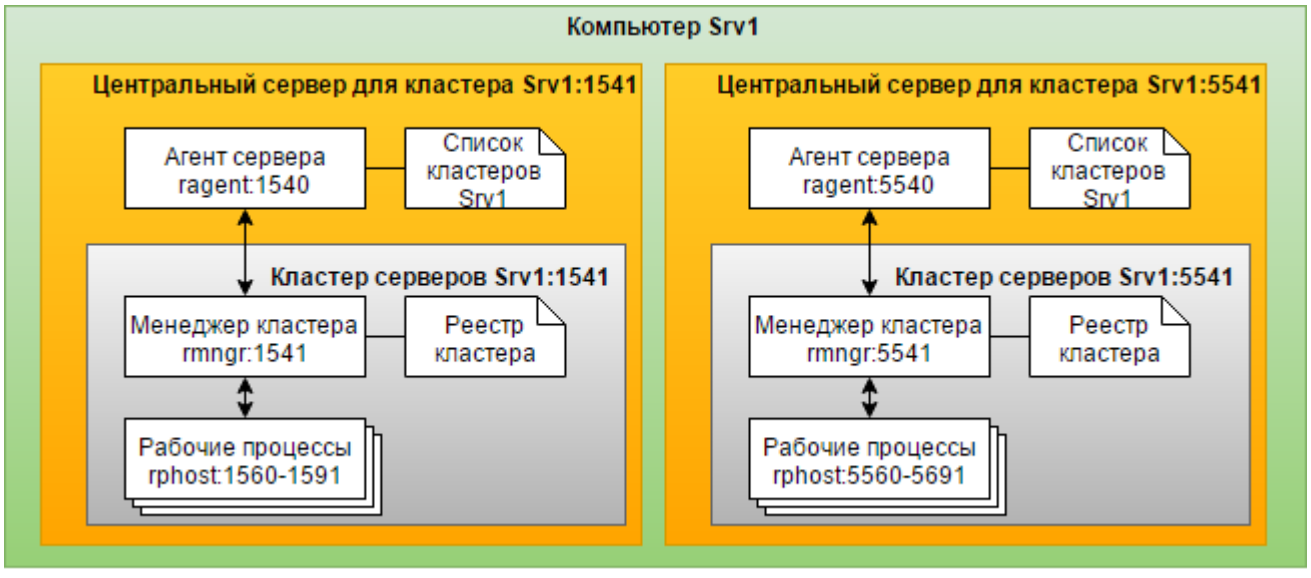
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



На приведенном выше рисунке сервер Srv1 является центральным для двух кластеров Srv1:1541 и Srv1:2541, а сервер Srv2 является рабочим для кластера Srv1:1541 и центральным для кластера Srv2:1541.

При необходимости на одном компьютере можно запустить несколько рабочих серверов, у каждого из которых будут свои кластеры.



Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

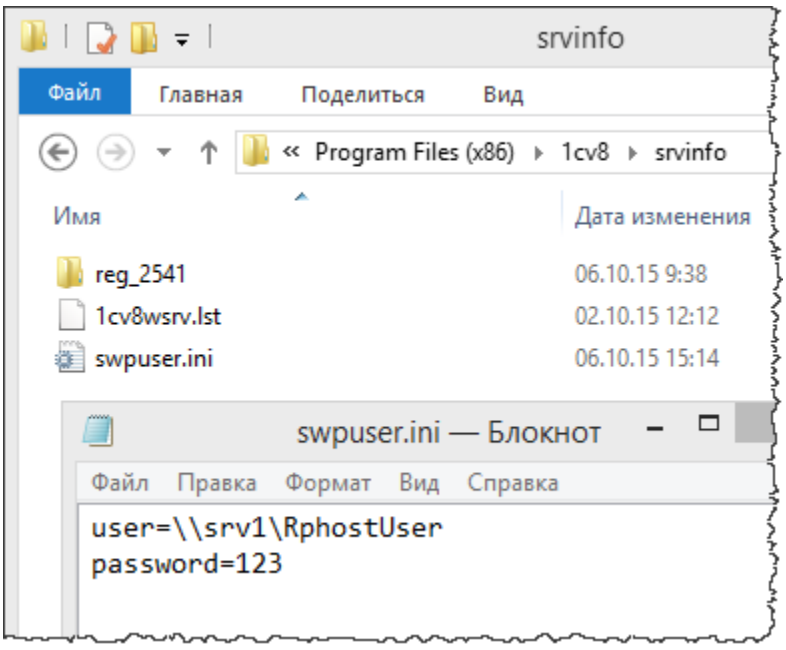
Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Например, один сервер может использоваться для отладки, а второй для рабочих баз. На рисунке показаны только центральные серверы, но на одном компьютере можно запускать параллельно как рабочие, так и центральные серверы – это не имеет значения, главное, чтобы номера портов у процессов в пределах одного компьютера не пересекались.

Запуск рабочего процесса под другим пользователем

По умолчанию рабочий процесс запускается под тем же пользователем, что и агент сервера.

При необходимости рабочий процесс (*rphost*) можно запустить под пользователем, отличным от пользователя, под которым запущен *ragent*, таким образом можно предотвратить программный доступ к служебным файлам кластера. Чтобы запускать процесс *rphost* под отдельным пользователем, необходимо в каталоге центрального сервера кластера создать файл swpuser.ini



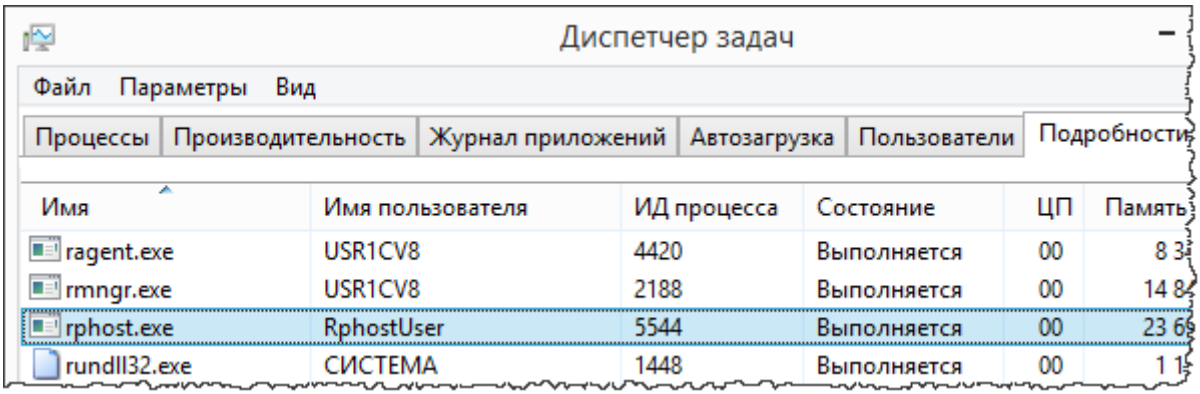
Пользователя, под которым планируется запускать *rphost*, необходимо включить в две локальные политики безопасности:

- Вход в качестве сервиса (Log on as a service)
- Вход в качестве пакетного задания (Log on as a batch job).

После перезапуска службы сервера 1С процесс *rphost* будет запущен под тем пользователем, который указан в файле.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



Если процесс *rphost* запущен под отдельным пользователем, и в свойствах информационной базы не указан логин для доступа к СУБД, тогда доступ к СУБД будет производится под тем пользователем, под которым запущен *rphost*. В данном случае необходимо завести этого пользователя на сервере СУБД и дать ему соответствующие права.

Соединения и сеансы

Сеанс

Сеанс определяет активного пользователя информационной базы и поток управления этого пользователя.

Можно сказать, что кластер серверов не видит пользователей, он видит только сеансы и сеансовые данные. Для него пользователи представлены сеансами.

В консоли кластера даже нет раздела *Пользователи*, под пользователями кластер понимает сеансы, о чем свидетельствует соответствующая иконка у пункта *Сеанс* в виде пользователей.

Следует уточнить, что под активным сеансом не обязательно понимается клиентское соединение, это может быть:

- Экземпляр клиентского приложения «1С:Предприятие»
- Экземпляр веб-приложения, в котором исполняется веб-клиент
- Экземпляр внешнего соединения (полученный из объекта *COMConnector*)
- Экземпляр фонового задания
- Обращение к Web-сервису.

Сеанс содержит в себе определенную информацию, такую как:

- Наименование информационной базы
- Номер сеанса
- Имя аутентифицированного пользователя информационной базы

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

- Язык интерфейса
- Значения параметров сеанса
- Временные хранилища
- Статистику работы сеанса
- Информацию форм управляемого приложения
- Некоторые внутренние данные платформы.

Эта информация называется **сеансовыми данными.** У каждого активного пользователя свои сеансовые данные, которые актуальны только на время работы этого пользователя. Как только пользователь вышел из базы (завершил сеанс), его сеансовые данные удаляются.

Данные сеансов хранятся на кластере серверов, за это отвечает менеджер кластера, именно для этого существует сервис сеансовых данных. Чтобы ускорить работу системы, данные сеансов кэшируются в рабочих процессах и в толстых клиентах.

Если активный сеанс не выполнил ни одного обращения к кластеру за 20 минут и сеанс не назначен соединению, то сеанс удаляется вместе с данными сеанса. Для поддержания сеанса тонкий клиент и веб-клиент обеспечивают обращение к кластеру не реже 1 раза в 10 минут.

В платформе 8.3 сеанс может быть **активным** или **спящим**.

Сеанс становится спящим в одном из следующих случаев:

- При нештатном разрыве соединения с сеансом, например, при отключении клиентского компьютера от сети
- Если сеанс не проявляет активности в течение времени, указанного в настройке «Время засыпания пассивного сеанса». Обычно, даже если сеанс бездействует, он один раз в 5-10 минут опрашивает сервер. Поэтому настройку «Время засыпания пассивного сеанса» не рекомендуется ставить меньше 10 минут.

Любая активность клиента приводит к тому, что сеанс становится активным.

Спящий сеанс автоматически завершается в следующих случаях:

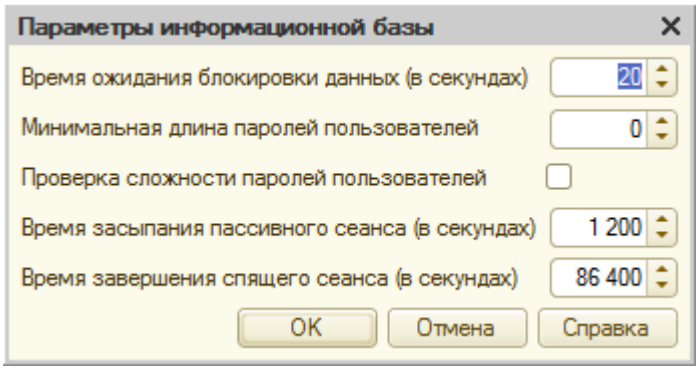
- По истечении времени, указанного в настройке «Время завершения спящего сеанса»
- При возникновении конфликтов блокировок между блокировками спящего и активного сеансов.

Настройки «Время засыпания пассивного сеанса» и «Время завершения спящего сеанса» указываются в конфигураторе, меню *Администрирование – Параметры информационной базы*.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



Соединение

Соединение является средством доступа сеансов к кластеру серверов, содержит ограниченное множество данных соединения, не отождествляется с активным пользователем. Также соединение используется для взаимодействия между процессами кластера.

Другими словами, сеанс получает доступ к кластеру с помощью соединения. Количество соединений ограничено, и как только соединение больше не нужно сеансу, оно возвращается в пул соединений. Если сеанс не обращается к кластеру (пользователь бездействует), то ему не назначается соединение, т.е. сеанс может существовать без соединения.

Кстати, сеансовые данные хранятся на сервере, поэтому разрыв соединения сеансу не страшен (если длится меньше 20 минут), ведь соединение – это только средство доступа.

Соединения также используются для взаимодействия процессов кластера, т.е. рабочие процессы (*rphost*) общаются с менеджером кластера (*rmngr*) с помощью соединений, а не с помощью сеансов.

Отличие соединений от сеансов

Чтобы описать основное отличие, проще всего прибегнуть к аналогии.

Сеанс – это пассажир, а соединение – это такси. Когда пассажиру нужно добраться домой (сеансу нужно подключится к серверу), он вызывает такси (сеансу назначается соединение из пула соединений). Если, добравшись домой, пассажир вдруг захочет поехать снова на работу, а такси уже уехало (после подключения случился разрыв соединения), то пассажир вызывает новое такси и едет по своим делам (сеансу назначается новое соединение).

Из этой аналогии хорошо видно, что сеанс и соединение - это не одно и тоже, и если случится разрыв соединения, то сеанс спокойно сможет его пережить.

Настройки центрального сервера

Центральный сервер – это сервер, на котором запущен главный менеджер кластера.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Настройка кластера осуществляется с помощью утилиты «Администрирование серверов 1С:Предприятия», обычно ее называют консолью кластера. Консоль кластера зависит от версии сервера: нельзя управлять кластером на релизе 8.3.6.2041, если запустить консоль для релиза 8.3.6.2237.

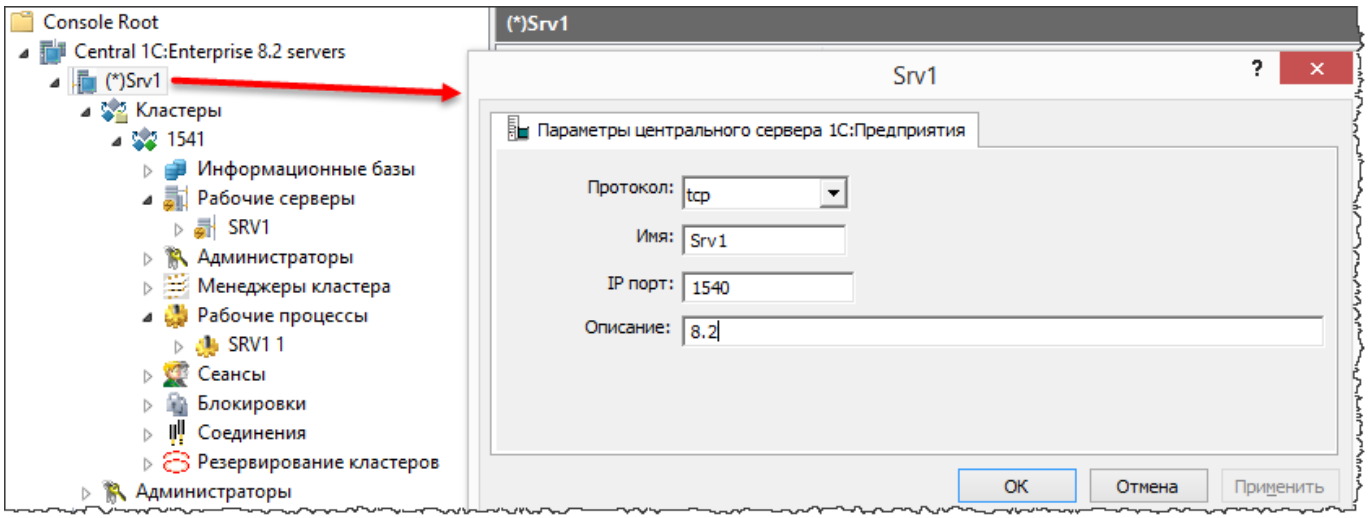
С помощью консоли можно не только выполнять настройку кластера, но и просматривать число сеансов и соединений, на каких серверах и на каких рабочих процессах работают соединения, видеть время вызова сервера СУБД и многое другое. Консоль кластера можно опрашивать программно и получать все необходимые данные. Например, ЦУП рисует графики оперативных показателей, программно опрашивая консоль кластера.

Отметим, что с помощью консоли нельзя создать центральный сервер, можно только подключиться к уже существующему центральному серверу. Для того чтобы создать центральный сервер, необходимо на компьютере запустить службу сервера 1С.

Консоль кластера не обязательно запускать на том компьютере, где работает сервер, она может быть запущена где угодно, главное, чтобы версия консоли совпадала с версией сервера.

Настройки центрального сервера в 8.2 и 8.3 одинаковы: это протокол, имя компьютера и номер порта центрального сервера. На самом деле данные настройки нужны только для подключения к уже существующему серверу. Указав настройки, мы просто выставляем параметры поиска центрального сервера.

У центрального сервера всего три настройки: протокол, имя сервера и IP порт. Протокол доступен только TCP, поэтому особого смысла эта настройка не имеет. Имя – это имя компьютера, на котором установлен центральный сервер. В качестве имени можно также указать IP адрес, а если консоль запущена на том же компьютере, где установлен центральный сервер, то можно указать значение localhost. IP порт – это номер порта, на котором работает центральный сервер на указанном компьютере. По умолчанию для центрального сервера используется порт 1540. Если углубиться в детали, то порт центрального сервера – это порт процесса *ragent*.



Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 41

Платформа 8.2

Видеоурок. Версия 8.2. Настройки кластера. Порт

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Можно ли изменить номер порта после создания кластера
- Нужно ли указывать служебный порт.

Видеоурок. Версия 8.2. Настройки кластера. Защищенное соединение

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как можно защитить трафик между клиентом и сервером
- Какие степени защиты существуют, и как они влияют на производительность.

Видеоурок. Версия 8.2. Настройки кластера. Интервал перезапуска

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение интервала перезапуска
- Пример автоматического перезапуска рабочего процесса.

Видеоурок. Версия 8.2. Настройки кластера. Ограничение по памяти

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- Как автоматически ограничить использование памяти рабочим процессом
- Пример автоматического перезапуска рабочего процесса при превышении допустимого объема памяти.

Видеоурок. Версия 8.2. Настройки кластера. Выключенные процессы

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Зачем нужна настройка «Выключенные процессы останавливать через»
- Пример использования настройки и рекомендуемые значения.

Видеоурок. Версия 8.2. Рабочий сервер - создание и настройка

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как создать рабочий сервер
- Какие настройки рабочего сервера указывать.

Видеоурок. Версия 8.2. Создание рабочего процесса

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как создать рабочий процесс
- Какие настройки рабочего процесса указывать.

Видеоурок. Версия 8.2. Распределение нагрузки по процессам

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение свойства рабочего процесса «Доступная производительность»
- По какому принципу пользователи распределяются по рабочим процессам.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Версия 8.2. Распределение нагрузки по серверам

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как нагрузка распределяется по серверам
- Сколько рабочих процессов создавать.

Видеоурок. Версия 8.2. Перенос сервисов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как распределить нагрузку путем переноса сервисов
- Используется ли такой метод на практике.

Занятие 42

Видеоурок. Версия 8.2. Отказоустойчивость *rphost*

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что происходит, если рабочий процесс падает.

Видеоурок. Версия 8.2. Отказоустойчивость *rmngr*

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- К чему приводит падение менеджера кластера.

Видеоурок. Версия 8.2. Отказоустойчивость *ragent*

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- К чему приводит падение агента сервера.

Видеоурок. Версия 8.2. Использовать как резервный

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- За что отвечает свойство рабочего процесса «Использовать как резервный»
- Применяется ли данное свойство на практике.

Видеоурок. Версия 8.2. Отказоустойчивость рабочего сервера. Схема

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что произойдет, если из строя выйдет рабочий сервер
- Будет ли кластер неработоспособен при выходе из строя рабочего сервера.

Видеоурок. Версия 8.2. Отказоустойчивость рабочего сервера. Пример

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример выхода из строя рабочего сервера.

Видеоурок. Версия 8.2. Отказоустойчивость центрального сервера. Схема

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что произойдет, если из строя выйдет центральный сервер кластера.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Версия 8.2. Отказоустойчивость центрального сервера. Пример

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример выхода из строя центрального сервера.

Видеоурок. Версия 8.2. Резервирование кластеров. Схема

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как работает механизм резервирования кластеров
- Что происходит в случае выхода из строя активного кластера.

Видеоурок. Версия 8.2. Резервирование кластеров. Пример

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как настроить группу резервирования кластеров
- Пример использования группы резервирования кластеров.

Видеоурок. Версия 8.2. Резервирование кластеров. Особенности

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Особенности использования группы резервирования кластеров
- Можно ли задать группу резервирования кластеров в строке подключения базы.

Видеоурок. Версия 8.2. Резервирование кластеров. Минусы

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- Недостатки использования группы резервирования кластеров
- Особенности использования лицензий при резервировании кластеров.

Видеоурок. Версия 8.2. Рабочий сервер в роли резервного. Схема

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Можно ли использовать рабочий сервер в качестве резервного
- Схему работы такого решения.

Видеоурок. Версия 8.2. Рабочий сервер в роли резервного. Пример

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример настройки резервного сервера в качестве рабочего.

Занятие 43

Платформа 8.3

В данном разделе будут описаны отличительные особенности работы кластера серверов 8.3 в сравнении с 8.2. Те настройки, которые полностью совпадают с версией 8.2, здесь рассматриваться не будут, т.к. они уже описаны выше.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

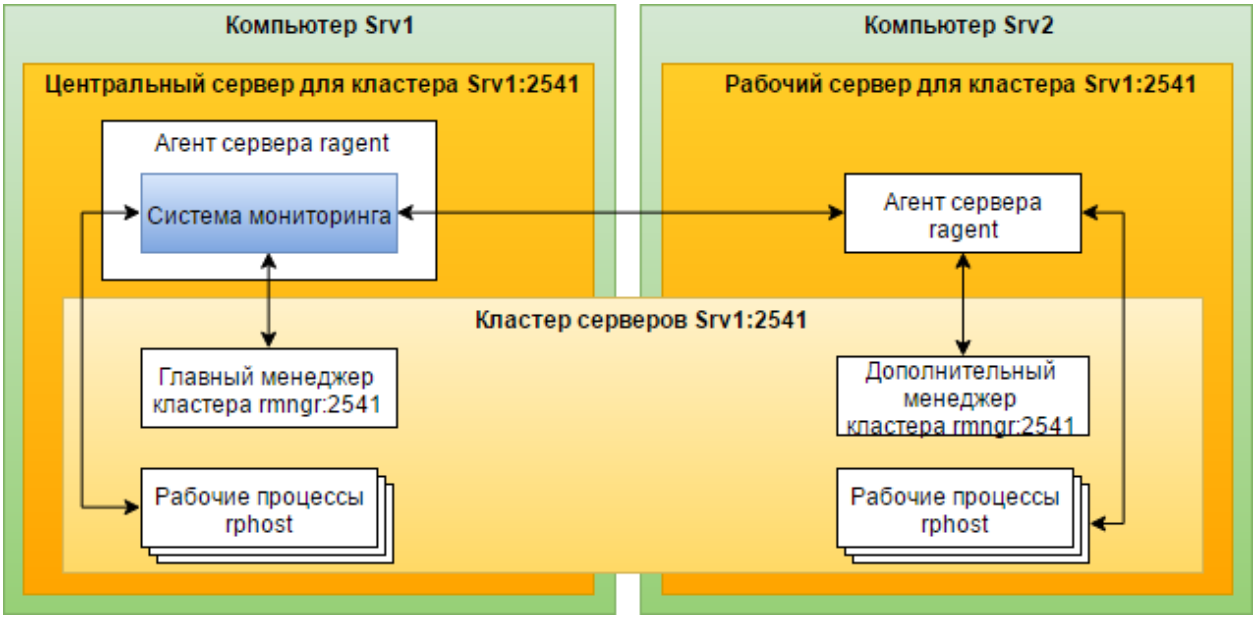
Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Система мониторинга

Начиная с версии 8.3.6.1977 в платформе появился новый механизм под названием **Система мониторинга**. Задача данной системы – обеспечение более стабильной работы кластера.

Данная система работает на центральном агенте сервера и следит за всеми процессами кластера (*rphost*, *rmngr* и *ragent*). Если в кластере больше одного рабочего сервера, то система мониторинга также отслеживает процессы всех остальных серверов. Для этого она связывается с агентом сервера на другом компьютере, и уже он опрашивает свои процессы. В отличие от версии 8.2, в версии 8.3 на других рабочих серверах дополнительный менеджер кластера создается по умолчанию.

Схема взаимодействия системы мониторинга с процессами кластера выглядит следующим образом:



Система мониторинга опрашивает все процессы один раз в 10 секунд, и для каждого из процессов выполняет проверку по следующим критериям:

- **Вычисление доступной производительности.** Выполняется микротест для определения загруженности каждого процесса. Это необходимо для балансировки нагрузки
- **Объем памяти, занимаемый процессом.** Анализируется только объем, занимаемый процессами rmngr и rphost. Если объем памяти rphost превышает порог, указанный в настройках кластера, срабатывает механизм ограничения по памяти.
  Принцип работы данного механизма уже был описан в видеоуроке «Версия 8.2. Настройки кластера. Ограничение по памяти»

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

- **Отслеживание и завершение рабочих процессов, удаленных из реестра кластера.** Если процесс должен был завершиться, но не завершился в течение 20 минут, то он признается проблемным
- **Соединение с процессом.** Система мониторинга каждые 10 секунд опрашивает процессы кластера, таймаут соединения составляет 20 секунд. Попытки установить соединение происходят последовательно. Следующей попытки не будет, пока не будет получен ответ от предыдущей попытки. Если не удалось установить соединение с процессом в течение 20 секунд, он признается проблемным.
  Что такое проблемный процесс и что с ним происходит, мы рассмотрим чуть позже.
- **Вычисление среднего количества ошибок.** Платформа вычисляет, сколько было в среднем ошибок (событий *EXCP*) на число обращений к серверу (событий *CALL*) за последние 5 минут. Это необходимо для настройки параметра кластера «Допустимое отклонение количества ошибок сервера». Параметр «Допустимое отклонение количества ошибок сервера» определяет, на сколько процентов должно быть превышено среднее количество ошибок, чтобы процесс стал считаться проблемным.
  Например, среднее число ошибок за 5 минут равно 10, при этом параметр кластера «Допустимое отклонение количества ошибок сервера» равен 50%. Это значит, что если какой-либо рабочий процесс вызовет в среднем за 5 минут более 15 ошибок, то он будет признан проблемным.

Если процесс признан проблемным, то он может быть завершен системой автоматически – это зависит от параметра кластера «Принудительно завершать проблемные процессы». Если в технологическом журнале включено создание дампов, то при завершении процесса образуется дамп. Работа с технологическим журналом будет подробно разобрана в отдельной главе.

Следует отметить, что механизм системы мониторинга только недавно появился в платформе, в связи с чем он не всегда работает так, как заявлено. Например, было замечено, что зачастую зависшие рабочие процессы не завершаются по истечении 20 минут, хотя система мониторинга должна была отследить эту ситуацию и завершить их.

Видеоурок. Версия 8.3. Настройки кластера

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Настройки кластера версии 8.3 и их отличия от настроек версии 8.2.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Версия 8.3. Максимальный объем памяти рабочих процессов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение настройки
- Что происходит, если данный параметр превышен.

Видеоурок. Версия 8.3. Безопасный расход памяти за один вызов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение настройки
- Что происходит, если данный параметр превышен.

Видеоурок. Версия 8.3. Объем памяти, до которого сервер считается производительным

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение настройки
- Что происходит, если данный параметр превышен.

Видеоурок. Версия 8.3. Регулировка числа рабочих процессов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как регулировать число рабочих процессов.

Видеоурок. Версия 8.3. Менеджер под каждый сервис

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- Назначение настройки
- Стоит ли устанавливать данный флаг.

Видеоурок. Версия 8.3. Центральный сервер

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как сделать несколько центральных серверов в кластере.

Видеоурок. Версия 8.3. Требования назначения функциональности

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Назначение данного объекта
- Как можно использовать требования назначения функциональности.

Видеоурок. Версия 8.3. Обслуживание базы на отдельном сервере

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как с помощью требования назначения функциональности перенести работу с определенной базы на отдельный сервер.

Видеоурок. Версия 8.3. Фоновые задания на отдельном сервере

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как с помощью требования назначения функциональности перенести работу всех фоновых заданий на всех базах на отдельный сервер.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Версия 8.3. Распределение клиентов по рабочим процессам

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Каким образом выбирается рабочий процесс для подключения клиента.

Занятие 44

Видеоурок. Версия 8.3. Центральные серверы

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как создать и настроить центральный сервер
- Пример подключения к другому центральному серверу.

Видеоурок. Версия 8.3. Центральные серверы и отказоустойчивость

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Какое влияние на отказоустойчивость оказывает количество центральных серверов.

Видеоурок. Версия 8.3. Центральные серверы и отказоустойчивость. Пример

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример отказоустойчивости кластера с двумя центральными серверами.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Версия 8.3. Уровень отказоустойчивости

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Описание и назначение параметра
- Возможный диапазон значений параметра.

Видеоурок. Версия 8.3. Отказоустойчивость. Ситуация 1

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Схему работы системы при выходе из строя единственного центрального сервера.

Видеоурок. Версия 8.3. Отказоустойчивость. Ситуация 1. Пример

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Реальное поведение системы при выходе из строя единственного центрального сервера.

Видеоурок. Версия 8.3. Отказоустойчивость. Ситуация 2

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Схему работы при выходе из строя одного рабочего сервера.

Видеоурок. Версия 8.3. Отказоустойчивость. Ситуация 2. Пример 1

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Реальное поведение системы при выходе из строя одного рабочего сервера, если уровень отказоустойчивости равен 1.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Версия 8.3. Отказоустойчивость. Ситуация 2. Пример 2

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Реальное поведение системы при выходе из строя одного рабочего сервера, если уровень отказоустойчивости равен нулю.

Видеоурок. Версия 8.3. Отказоустойчивость. Ситуация 3

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Поведение системы при выходе из строя двух рабочих серверов.

Видеоурок. Версия 8.3. Расчет уровня отказоустойчивости

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Особенности формулы расчета уровня отказоустойчивости.

Видеоурок. Версия 8.3. Влияние отказоустойчивости на производительность

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Каким образом отказоустойчивость влияет на производительность.

Видеоурок. Версия 8.3. Связь уровня отказоустойчивости и требований назначения функциональности

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- Особенности установки требований назначения функциональности при использовании уровня отказоустойчивости выше нуля
- Как требования назначения функциональности могут снизить устойчивость системы.

Видеоурок. Версия 8.3. Минусы механизма отказоустойчивости

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Нюансы и особенности при настройке и использовании механизма отказоустойчивости.

Видеоурок. Отличия в механизме лицензирования 8.2 и 8.3

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как перенести сервис лицензирования на другой компьютер.

Видеоурок. Рекомендации по общей архитектуре

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Общие рекомендации по архитектуре кластера.

Рекомендуемые настройки кластера серверов

Настройки кластера

Рассмотрим, какие настройки кластера желательно использовать.

Интервал перезапуска – 86 400 секунд. Рекомендуется один раз в сутки перезапускать рабочие процессы.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Допустимый объем памяти – здесь необходимо провести определенные расчеты. При указании данного параметра надо учитывать, что при перезапуске рабочего процесса сначала стартует новый рабочий процесс, при этом «старый» процесс может еще работать то время, которое указанное в параметре «Выключенные процессы останавливать через». Получается, что в течение времени «Выключенные процессы останавливать через» будут работать оба процесса и, естественно, они оба будут занимать память. Это необходимо учитывать при расчете допустимого объема памяти. Поэтому формула расчета допустимого объема памяти выглядит следующим образом:

Допустимый объем памяти = Общий объем памяти – Объем памяти для ОС (3 ГБ) – Объем одного rphost * Количество rphost

Допустим, на сервере установлено 50 ГБ оперативной памяти, при этом у вас работает 5 рабочих процессов, каждый из которых занимает примерно 7 ГБ памяти.

Допустимый объем памяти = 50 – 3 – 5 * 7 = 12

Таким образом, в приведенном примере необходимо установить допустимый объем памяти равный 12 ГБ. Как только этот предел будет достигнут, запустится другой рабочий процесс и все соединения перейдут на него, а через время, указанное в параметре «Выключенные процессы останавливать через», первый процесс будет завершен.

Интервал превышения допустимого объема памяти – рекомендуется установить значение 60 секунд. Слишком большое значение параметра может привести к тому, что на сервере закончится свободная оперативная память.

Выключенные процессы останавливать через – рекомендуется поставить значение 60 секунд – за это время, как правило, все соединения успевают завершить свою работу. Большое значение данного параметра может снизить производительность сервера, т.к. все это время процесс будет отнимать его ресурсы.

Допустимое отклонение количества ошибок сервера – на данный момент нет официальных рекомендаций по данному параметру, но, исходя из описания его работы, можно совершенно точно считать проблемными те процессы, у которых ошибок на 50% больше среднего значения. Можно поставить данный параметр в значение 50%.

Принудительно завершать проблемные процессы – флаг необходимо установить, так как в этом случае, если процесс будет признан проблемным, он автоматически будет завершен и стартует новый процесс.

Уровень отказоустойчивости – если в составе кластера более одного сервера, то не рекомендуется ставить значение этого параметра выше 1, иначе затраты на синхронизацию между серверами возрастут.

Режим распределения нагрузки – рекомендуется оставить значение по умолчанию – «Приоритет по производительности». В данном случае нагрузка будет распределяться исходя из свойства рабочих процессов «Доступная производительность».

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Данный параметр имеет смысл менять только в том случае, если параметр «Доступная производительность» у процессов считается некорректно, что приводит к перекосу в нагрузке в сторону одного рабочего сервера.

Настройки рабочего сервера

Максимальный объем памяти рабочих процессов – рекомендуется оставить значение по умолчанию - 0. В этом случае максимальный объем памяти рабочих процессов будет равен 80% от объема памяти, установленной на данном компьютере.

Безопасный расход памяти за один вызов – рекомендуется оставить значение по умолчанию – 0. В этом случае безопасный расход памяти будет равен 5% от значения показателя «Максимальный объем памяти рабочих процессов».

Объем памяти рабочих процессов, до которого сервер считается производительным – рекомендуется оставить значение 0, в этом случае балансировка нагрузки будет происходить исходя из других параметров. Имеет смысл изменять, только если вы явно хотите ограничить нагрузку на данный сервер.

Количество ИБ на процесс – рекомендуется оставить значение по умолчанию – 8. Данный параметр имеет смысл менять, только если базы мало нагружены, тогда можно увеличить данный параметр, чтобы они все обслуживались одним процессом. Если же базы сильно нагружены, то можно, наоборот, понизить значение данного параметра. При изменении параметра следует отслеживать, как изменилось поведение системы, не появились ли ошибки и не снизилась ли скорость работы системы.

Количество соединений на процесс – если на сервере используется один процессор, рекомендуется оставить значение по умолчанию - 128. Данный параметр имеет смысл менять, только если соединений с базой очень много и при этом каждое из них дает небольшую нагрузку. В этом случае порог можно поднять, чтобы не создавать слишком большое число рабочих процессов. Если на сервере используется больше одного процессора (не ядра, а именно процессора), то следует использовать следующую формулу:

ЧислоСоединенийНаПроцесс = РасчетноеМаксКоличествоСоединений / КоличествоУзловNUMA

РасчетноеМаксКоличествоСоединений – это максимальное число соединений к данному серверу 1С.

КоличествоУзловNUMA – это количество NUMA узлов, которые используются на данном компьютере. Можно уточнить эту информацию у производителя.

Данный расчет необходим для того, чтобы избежать ситуации, когда часть ядер загружена на 100%, а другие ядра простаивают.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

При этом все равно возможна ситуация, при которой будет сильный перекос в загрузке ядер. Это можно увидеть либо в диспетчере задач, либо в системном мониторе добавить счетчик загруженности процессора с разбивкой по ядрам.

В таком случае лимит соединений на процесс следует уменьшать из расчета

ЧислоСоединенийНаПроцесс = РасчетноеМаксКоличествоСоединений / (КоличествоNUMA * Коэффициент)

Коэффициент изначально равен 2, но его можно увеличивать на единицу до тех пор, пока нагрузка по ядрам не выровняется.

Менеджер под каждый сервис – данный параметр лучше не устанавливать, т.к. он пока является экспериментальным. Если параметр установить, на сервере будет запущен отдельный процесс rmngr для каждого сервиса.

Центральный сервер – если в кластере используется более одного сервера, тогда имеет смысл у еще одного рабочего сервера поставить данный флаг: в результате у вас будет минимум 2 центральных сервера. Желательно, чтобы центральные серверы территориально находились в разных местах. Если один из них выйдет из строя, пользователи все равно смогут подключаться к базе по старому адресу, но работать будут со вторым центральным сервером. Это повышает устойчивость кластера, но снижает производительность, поэтому делать в кластере более 2 центральных серверов стоит только в том случае, если цена простоя в работе очень высока и стабильность предпочтительнее производительности. Рабочие серверы можно добавлять в кластер, но не обязательно все их делать центральными.

Прочие настройки

Количество рабочих процессов для 8.2 необходимо определять вручную. Желательно создавать минимум 2 рабочих процесса в любом случае, независимо от разрядности системы.

Если используется 8.3, то можно настроить требования назначения функциональности, например, использовать выделенный сервер лицензирования, тогда не будет ограничений по апгрейду серверов. Однако следует помнить, что при выходе из строя того сервера, на котором расположено какое-либо требование назначения функциональности, это требование не сможет больше выполнить ни один из оставшихся серверов. Например, если сервер лицензирования выйдет из строя, лицензии станут недоступны.

Возможность переноса сервисов между кластерами в 8.2 лучше не использовать, т.к. польза от этого малозаметна, но возможны негативные побочные эффекты в случае отказа сервера. Например, если перенести сервис фоновых заданий на отдельный компьютер, то это не значит, что сами задания будут выполняться именно там, это значит, что управление этими заданиями будет выполняться на втором сервере. Если сервис фоновых заданий перенесен на второй сервер и при этом он вышел из строя, тогда фоновые задания не смогут запускаться на первом сервере, даже если он будет доступен.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

В 8.3 эта ситуация исправлена: если сервер, на котором работал сервис фоновых заданий, завершит работу, то этот сервис запустится на другом сервере и фоновое задание выполнится. Однако, если установить требование назначения функциональности и перенести фоновые задания на отдельный сервер, то, если он выйдет из строя, фоновые задания не будут выполняться.

Кластер серверов. Итоги

Кластер серверов предоставляет широкие возможности для горизонтального масштабирования и повышения отказоустойчивости. Для настройки кластера необходимо знать и понимать его устройство и принципы работы. При этом следует помнить, что чем проще устроен и настроен кластер, тем быстрее и стабильнее он работает.

Отметим, что при большом количестве серверов нагрузка между ними будет разделена, но накладные расходы на синхронизацию тоже возрастут.

Механизмы отказоустойчивости в версиях 8.2 и 8.3 реализованы различным образом, однако на практике механизмы платформы 8.3 часто работают не очень стабильно.

В платформе 8.3 реализован механизм *Требований назначения функциональности*, с его помощью можно более тонко настраивать серверы под определенные задачи. Благодаря данному механизму можно сделать, например, сервер лицензий, который будет только выдавать лицензии, что особенно полезно при использовании виртуальных машин. При настройке требований надо учитывать настройки отказоустойчивости.

Кластер серверов может располагаться на компьютерах с разными версиями операционных систем. Например, в составе одного кластера могут одновременно находиться как 32, так и 64-разрядные серверы, а также серверы под управлением разных операционных систем, Windows и Linux.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Технологический журнал

Занятие 45

Описание и назначение

Технологический журнал (ТЖ) – это инструмент для логирования работы платформы на низком уровне. С помощью ТЖ можно записывать практически любые действия платформы: выполнение запросов и вызовов сервера, ошибки и т.д. Кроме того, ТЖ позволяет настроить создание аварийных дампов в случае падения процессов платформы.

С помощью ТЖ можно выяснить, какие запросы работают медленно и откуда они вызываются, при выполнении какого кода «падают» рабочие процессы сервера, куда «утекает» память и многое, многое другое.

Все инструменты анализа производительности платформы используют ТЖ для получения информации. При необходимости с помощью ТЖ можно написать свой собственный инструмент анализа производительности и стабильности.

ТЖ может собирать данные как для процессов сервера 1С, так и для клиентских приложений. Соответственно, и набор событий, которые можно фиксировать в ТЖ, будет отличаться. Логи и дампы клиентских приложений крайне редко вызывают интерес, поэтому в этом разделе будут рассматриваться только логи сервера. Тем не менее, вся рассмотренная в дальнейшем информация может быть использована для анализа как серверных, так и клиентских логов.

С помощью ТЖ можно собирать логи и настраивать формирование дампов в случае аварийного завершения процессов платформы.

Логи – это файлы с расширением *log*, где информация хранится в текстовом виде. Логи собираются для каждого сервера отдельно.

Дампы – это файлы с расширением *mdmp*, которые хранят в себе содержимое оперативной памяти процесса на момент «падения». Дамп бывает крайне необходим для расследования проблем стабильности платформы. Самостоятельно невозможно анализировать дампы, т.к. для этого нужен исходный код платформы, но можно отправить их в техническую поддержку или на партнерский форум и получить решение проблемы. Подробнее о том, как можно использовать ТЖ и дампы для решения проблем стабильности, будет рассказано в отдельной главе.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Размещение файла logcfg.xml

Настройки технологического журнала задаются с помощью файла logcfg.xml.

По умолчанию технологический журнал включен и работает (даже если файл logcfg.xml отсутствует), но собирает очень ограниченный объем данных.

Под минимальным объемом данных подразумеваются две вещи:

- Создание дампов минимального размера в случае аварийного завершения работы процессов кластера 1С (*ragent*, *rmngr* или *rphost*).

  По умолчанию дамп создается в каталоге:

  %USERPROFILE%\Local Settings\Application Data\1C\1Cv82\dumps

  Если используется Windows Vista и выше, то дамп будет создан в каталоге:

  %LOCALAPPDATA%\1C\1Cv8\dumps

  Здесь и далее для 8.2 вместо каталога *1Cv8* используется *1Cv82*.

- Для 8.3 в минимальный ТЖ входит формирование логов с одним событием SYSTEM с уровнем *Error*.

  Логи сохраняются в каталоге:

  %USERPROFILE%\Local Settings\Application Data\1C\1Cv8\logs

  Для Windows Vista и старше используется каталог:

  %LOCALAPPDATA%\1C\1Cv8\logs

Данные логи по умолчанию будут храниться 24 часа, более старые логи платформа удаляет автоматически.

Чаще всего информации из ТЖ по умолчанию недостаточно, поэтому необходимо его настраивать вручную. Чтобы произвести тонкую настройку ТЖ, необходимо создать файл logcfg.xml с определенной структурой в определенном месте. Структура этого файла будет рассмотрена далее.

Данный файл необходимо разместить в каталоге:

C:\Program Files\1Cv8\conf

В этом случае настройки ТЖ будут действовать для всех версий платформы 1С, установленных на данном компьютере, и для всех пользователей. Этот вариант используется чаще всего, и именно его рекомендуется использовать в абсолютном большинстве случаев.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

При настройке ЦУПа, облачных сервисов и прочих инструментов контроля производительности, в которых нужно указывать путь к файлу logcfg.xml, также лучше использовать именно этот каталог, иначе при обновлении платформы или при изменении имени пользователя, под которым запущена служба сервера 1С, описанные инструменты перестанут работать и их придется перенастраивать.

Тем не менее, есть и другие варианты, хотя и используются они гораздо реже. Рассмотрим лишь те, которые могут понадобиться при анализе производительности с наибольшей вероятностью.

Чтобы настроить ТЖ только для одной версии платформы, размещаем logcfg.xml в каталоге:

C:\Program Files\1Cv82\8.2.19.106\bin\conf

Где 8.2.19.106 – это номер нужной версии платформы.

Крайне редко, но все же может возникнуть необходимость настроить ТЖ отдельно для каждого пользователя, под которым запущена служба сервера 1С.

Тогда необходимо разместить файл logcfg.xml в каталоге:

%USERPROFILE%\Local Settings\Application Data\1C\1Cv8\Conf

Для ОС Windows Vista и старше:

%LOCALAPPDATA%\1C\1Cv8\Conf

Это может потребоваться, если, например, одна служба сервера 1С используется как рабочая, а вторая – для отладки. При необходимости можно запустить службы под разными пользователями и собирать ТЖ только для одной из них, чтобы не загружать второй сервер и не собирать в логах лишние данные, либо сделать для каждой из служб свои настройки ТЖ.

Настройки из logcfg.xml считываются не моментально, а каждые 60 секунд, причем каждый из процессов кластера считывает файл настроек независимо от других процессов. Например, сначала могут появиться логи процесса *rmngr*, и только через 45 секунд – логи *rphost*.

Если удалить или переименовать файл logcfg.xml, то будет собираться ТЖ по умолчанию.

Для полного выключения ТЖ можно создать logcfg.xml с пустым содержимым.

Если в кластере более одного сервера, то необходимо включать ТЖ на каждом из серверов, ведь если логи включены только на одном сервере, то собранной информации может быть недостаточно для расследования.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Структура файла logcfg.xml

Как уже было сказано выше, для тонкой настройки ТЖ используется файл logcfg.xml. Давайте подробно разберем структуру этого файла.

Например, мы разместили в каталоге *C:\Program Files\1Cv82\conf* файл logcfg.xml со следующим содержанием:

```
<config xmlns="http://v8.1c.ru/v8/tech-log">
<dump location="C:\1C_Info\Dumps" create="1" type="3" prntscrn="false"
externaldump="1"/>
<log location="C:\1C_Info\Logs" history="1">
<event>
<ne property="name" value=""/>
</event>
<property name="all"/>
</log>
</config>
```

Подробно рассмотрим каждую строку.

1. **<config xmlns="http://v8.1c.ru/v8/tech-log">**

Определяет начало настроек ТЖ и указывает на пространство имен xml, эта строка всегда идет первой и остается неизменной по содержанию.

2. **<dump location="C:\1C_Info\Dumps" create="1" type="3" prntscrn="false" externaldump="1"/>**

Данная строка служит для создания и настройки аварийного дампа в случае аварийного завершения одного из процессов сервера 1С.

Параметр *location* указывает место создания дампа. В нашем примере это *C:\1C_Info\Dumps*.

Если параметр *create=«0»* или *create=«false»*, то дамп не будет создан.

Параметр *type* определяет, насколько полный дамп нужно создавать. Если этот параметр равен 3, то в дамп будет записано содержимое всей памяти процесса. Рекомендуется использовать именно это значение параметра, т.к. в этом случае в дамп будет записан максимальный объем информации, что значительно облегчит возможное расследование.

Параметр *type* равный 0 означает, что будет собран минимальный дамп, информации в котором, может не хватить для расследования.

Для параметра *type* возможен довольно большой перечень значений, подробнее о них можно прочитать в руководстве администратора, но на практике другие значения параметра почти никогда не используются.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Если раздела *<dump>* нет, то в случае аварийного завершения процесса будут созданы минимальные дампы, которые будут сохранены в каталог для дампов по умолчанию: *%USERPROFILE%\Local Settings\Application Data\1C\1Cv8\dumps*

Параметр *prntscrn* может принимать значения 0 (*false*) или 1 (*true*) и отвечает за создание скриншота экрана во время падения процесса. Как правило, эта информация никакой ценности не несет, поэтому параметр обычно принимает значение *false* или 0.

Параметр *externaldump* может принимать значения 0 (*false*) или 1 (*true*). Если параметр выключен, то дамп будет создаваться тем процессом, который завершается аварийно. Если параметр включен, то дамп будет создан с использованием внешней утилиты (*dumper.exe*), которая входит в комплект поставки платформы. Данный способ считается более надежным, т.к. в этом случае исключается возможность зависания процесса во время создания дампа. Параметр можно использовать, только если сервер 1С установлен на ОС Windows.

Последние два параметра не являются обязательными, и их можно вообще не указывать.

3. **<log location="C:\1C_Info\Logs" history="1">**

Открывает раздел с настройками логов. Здесь настраивается каталог для хранения логов и время хранения логов в часах. В данном случае логи будут храниться только за последний час (*history=1*). Файлы логов старше указанного времени платформа удалит самостоятельно.

Параметр *location* определяет каталог для записи логов. Необходимо помнить, что выбранный каталог должен быть пустым и кроме файлов логов в нем не должно быть никаких посторонних файлов. В противном случае платформа не сможет записывать логи в указанный каталог.

Разделов *<log>* может быть несколько, в каждом разделе могут быть указаны свои фиксируемые события, время хранения и каталог для хранения файлов.

4. **<event>**

Открывает раздел для фильтрации и настройки тех событий, которые мы будем собирать в логах ТЖ. Разделов *<event>* также может быть несколько – по одному разделу на одно событие.

5. **<ne property="name" value=""/>**

Определяет, какое именно событие мы будем фиксировать и в каком случае.

ne – это условие неравенства (not equal), дословно строка читается так: если свойство события «Имя» не равно значению «», тогда записываем это событие в ТЖ, а так как у любого события есть имя, то данное условие заведомо всегда будет выполняться, и мы будем фиксировать абсолютно все события технологического журнала.

Обычно такая настройка не используется, т.к. сбор полного технологического журнала будет замедлять работу системы, логи быстро займут все свободное место на диске, да и разобраться в гигабайтах текстовой информации потом будет непросто. Поэтому обычно используют фильтрацию по событиям, но об этом поговорим чуть позже.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

6. **</event>**

Закрывает раздел <event>. После этого, если нужно фиксировать несколько событий, можно начинать новый раздел *</event>*.

7. **<property name="all">**

Здесь мы определяем, какие свойства событий необходимо фиксировать. Обычно это значение остается по умолчанию *«all»*, т.е. будут записываться все свойства событий, которые определены в разделе <*event*>.

8. **</log>**

Закрывает раздел <*log*>, после этого можно начинать новый раздел <*log*>.

9. **</config>**

Определяет конец настроек ТЖ.

На первый взгляд, такой вариант настройки может показаться сложным, но это только на первый взгляд. На самом деле, благодаря такому формату записи можно создавать очень гибкие настройки, что в дальнейшем позволяет сильно облегчить расследование различных проблемных ситуаций.

События технологического журнала

Писать в ТЖ все события платформы не имеет большого смысла: такое подробное логирование сильно загрузит сервер 1С, будет занимать очень много места на диске, и, к тому же, будет крайне проблематично найти нужную информацию в гигабайтах текста.

Для того чтобы записывать только нужную информацию, существуют события ТЖ и фильтрация этих событий.

Например, когда возникает какая-либо исключительная ситуация, то в ТЖ записывается событие EXCP. Если выполняется запрос к базе MS SQL Server, возникает событие DBMSSQL и т.д.

Все события технологического журнала для платформы 8.3 описаны в таблице.

Событие	Описание
ADMIN	Действия администратора кластера серверов 1С:Предприятия. Рекомендуется завести администраторов кластеров с логином, который соответствует фамилии администраторов, тогда с помощью данного события будет понятно, кто и что менял в настройках кластера.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

ATTN	Мониторинг состояния кластера. Данное событие пишется системой мониторинга кластера, если один из процессов не прошел по критериям проверки. Критерии проверки описаны в разделе «Система мониторинга».
CALL	Входящий удаленный вызов (удаленный вызов на стороне приемника вызова). Например, если вы из клиента вызываете функцию на сервере, то в ТЖ на сервере будет записано событие CALL.
CONN	Установка или разрыв TCP-соединения между процессами системы «1С:Предприятие».
CLSTR	Выполнение операций, изменяющих работу кластера серверов.
EDS	События, связанные с внешними источниками данных.
DB2	Исполнение операторов SQL СУБД IBM DB2.
DBMSSQL	Исполнение операторов SQL СУБД Microsoft SQL Server.
DBPOSTGRS	Исполнение операторов SQL СУБД PostgreSQL.
DBORACLE	Исполнение операторов SQL СУБД Oracle Database.
DBV8DBEng	Исполнение операторов SQL файловой СУБД.
EXCP	Исключительная ситуация приложения системы «1С:Предприятие», которое штатно не обрабатывается и может послужить причиной аварийного завершения серверного процесса или подсоединенного к нему клиентского процесса. Данное событие не всегда означает ошибку, большая часть таких записей – это информационный мусор.
EXCPCNTX	События, которые начались, но не закончились в момент возникновения нештатной ситуации. Обычно это событие отдельно не включается, т.к. оно пишется даже в том случае, если включено событие EXCP.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

FTEXTCheck	Возникает при проверке файлов индекса полнотекстового поиска.
FTEXTUpd	Возникает во время обновления файлов индекса полнотекстового поиска.
HASP	Обращение к аппаратному ключу защиты (HASP).
LEAKS	Событие, связанное с утечкой памяти, которая может быть вызвана ошибками в коде конфигурации.
MAILPARSEERR	Событие, формируемое в том случае, если во время разбора почтового сообщения возникла ошибка.
MEM	События, связанные с увеличением объема памяти, занятой процессам и сервера (ragent, rmngr, rphost).
PROC	Событие, относящееся к процессу целиком и влияющие на дальнейшую работоспособность процесса. Например: старт, завершение, аварийное завершение и т. п.
QERR	Событие, связанное с обнаружением ошибок компиляции запроса или ограничением на уровне записей и полей базы данных
SCALL	Исходящий удаленный вызов (исходящий вызов на стороне источника вызова). Например, если вы из клиента вызываете функцию на сервере, то в ТЖ на клиенте будет записано событие SCALL.
SCOM	Событие создания или удаления серверного контекста, обычно связанного с информационной базой.
SDBL	Исполнение запросов к модели базы данных 1С:Предприятия 8. Например, с помощью данного события можно определить длительность транзакции.
SESN	Действия, относящиеся к сеансу работы. Например: начало сеанса, окончание сеанса и т. д.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

SRVC	События, связанные с запуском, остановкой и оповещениями сервисов кластера серверов.
TLOCK	Установка управляемой блокировки. Используется различными инструментами для анализа ожиданий на управляемых блокировках. Длительность данного события – это время ожидания предоставления блокировки.
TDEADLOCK	Обнаружена взаимоблокировка на управляемых блокировках. С помощью данного события и события Tlock можно расследовать взаимоблокировки на управляемых блокировках.
TTIMEOUT	Ошибка по таймауту на управляемых блокировках. Возникает, если управляемая блокировка не была установлена в течение максимального периода предоставления блокировки (по умолчанию 20 секунд).
VRSCACHE	Работа кэша серверных вызовов.
VRSREQUEST	Запрос к серверу для получения какого-либо ресурса.
VRSRESPONSE	Ответ сервера.
SYSTEM	Системные события механизмов платформы, предназначенные для анализа сотрудниками фирмы «1С».

Особо отметим, что события пишутся в технологический журнал после завершения, а не в момент их начала. Это связано с тем, что в логах отображается длительность события. Для того чтобы получить момент начала события, нужно из момента его завершения вычесть длительность.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 46

Видеоурок. Включение ТЖ

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как включить технологический журнал
- Каким образом записываются логи технологического журнала
- Как понять, какой лог к какому процессу относится.

Видеоурок. Ошибки при настройке ТЖ

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Наиболее популярные ошибки при настройке ТЖ.

Видеоурок. Структура логов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как устроены логи технологического журнала
- Как узнать длительность события и точное время события.

Видеоурок. Свойства событий

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как записываются свойства события
- Основные свойства события DBMSSQL.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Фильтрация по событиям

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как сделать фильтр по событиям
- Как записывать одновременно несколько событий.

Видеоурок. Фильтрация по свойствам

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как сделать фильтр по свойствам события.

Видеоурок. Условия фильтрации

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Какие виды сравнения можно использовать в ТЖ
- Особенности использования условия по *like*.

Видеоурок. Комбинация фильтров

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как можно комбинировать различные фильтры на события и на свойства.

Видеоурок. Запись в разные каталоги

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как записывать разные события в разные каталоги
- Особенности взаимного расположения каталогов.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Получение плана запроса

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как с помощью ТЖ получить план запроса
- Особенности получения планов запросов.

Видеоурок. Отличия ТЖ 8.2 и 8.3

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Отличия в формате логов 8.2 и 8.3
- Особенности фильтрации по длительности для разных версий платформы.

Видеоурок. Обработка для настройки ТЖ

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как можно настроить ТЖ с помощью специальной обработки
- Для каких случаев обработка не подходит.

Видеоурок. Влияние на производительность

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как ТЖ влияет на производительность сервера 1С
- Что необходимо сделать, чтобы снизить это влияние.

Видеоурок. Сбор ТЖ на клиенте

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Зачем может потребоваться собирать логи на клиенте
- Какая настройка скорее всего потребуется.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Анализ ожиданий на управляемых блокировках

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как с помощью ТЖ провести анализ ожиданий на управляемых блокировках.

Видеоурок. Анализ взаимоблокировки на блокировках 1С

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как с помощью ТЖ провести анализ взаимных блокировок на управляемых блокировках.

Рекомендуемая настройка на каждый день

Особо отметим, что в рабочей системе технологический журнал должен быть включен и настроен в обязательном порядке: в этом случае при возникновении любой ситуации, требующей дополнительного расследования, на руках уже будет материал для анализа.

Компания «1С» рекомендует использовать следующую настройку для ежедневного использования.

```
<?xml version="1.0" encoding="UTF-8"?>
<config xmlns="http://v8.1c.ru/v8/tech-log">
<dump create="true" location="C:\DUMPS" type="3" prntscrn="false" externaldump="1"/>
  <log location="C:\LOGS\All" history="28">
    <event>
        <eq property="Name" value="EXCP"/>
    </event>
    <event>
        <eq property="Name" value="PROC"/>
    </event>
    <event>
        <eq property="Name" value="ADMIN"/>
    </event>
    <event>
        <eq property="Name" value="CONN"/>
    </event>
    <event>
        <eq property="Name" value="SESN"/>
    </event>
    <event>
        <eq property="Name" value="CLSTR"/>
```


Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

```
    </event>
    <event>
        <eq property="Name" value="SRVC"/>
    </event>
    <property name="all"/>
  </log>
  <log location="C:\LOGS\CallScall" history="28">
    <event>
        <eq property="Name" value="CALL"/>
    </event>
    <event>
        <eq property="Name" value="SCALL"/>
    </event>
        <property name="Context">
    <event>
        <eq property="name" value=""/>
    </event>
    </property>
    <property name="all"/>
  </log>
</config>
```

Это официально рекомендованная настройка, но на практике в подавляющем большинстве случаев такая настройка будет избыточной. Логи с такой настройкой будут содержать очень много событий CALL и SCALL, что сильно увеличит их размер.

Для большинства случаев подойдет более простая настройка:

```
<config xmlns="http://v8.1c.ru/v8/tech-log">
      <dump create="true" location="C:\DUMPS" type="3" prntscrn="0"
externaldump="1"/>
      <log location="C:\LOGS\All" history="28">
      <event>
            <eq property="Name" value="EXCP"/>
      </event>
      <event>
            <eq property="Name" value="PROC"/>
      </event>
      <event>
            <eq property="Name" value="ADMIN"/>
      </event>
      <event>
            <eq property="Name" value="CLSTR"/>
      </event>
      <event>
            <eq property="Name" value="SRVC"/>
      </event>
      <property name="all"/>
      </log>
</config>
```


Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Технологический журнал. Итоги

Технологический журнал – это основной источник данных для всех инструментов анализа производительности платформы. С помощью технологического журнала можно зафиксировать не только длительные запросы с планами и контекстами, но и практически любые другие события.

С помощью логов можно расследовать различные специфичные ошибки, например, медленную работу системы только под определенным пользователем, или найти длительные транзакции.

Особую ценность ТЖ представляет при расследовании проблем стабильности. С его помощью можно настраивать создание дампов при «падении» процессов и расследовать причину этих «падений».

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Расследование проблем стабильности

Занятие 47

Описание проблем стабильности

К проблемам стабильности относятся «падения» или «зависания» процессов кластера, а также различные системные ошибки по вине платформы.

«Падение» процесса означает, что процесс самопроизвольно выгружается из памяти. Например, при использовании платформы 8.2 бесконечная рекурсия будет приводить к падению рабочего процесса. В 8.3 данная ситуация предусмотрена, и падения не будет, но у пользователя появится ошибка «Переполнение стека встроенного языка на сервере» с контекстом кода, который привел к переполнению.

«Зависание» процесса означает, что процесс остается в памяти, но не принимает новые соединения и не обслуживает уже существующие.

Чаще всего проблемы стабильности наблюдаются у процесса *rphost*, т.к. именно на него приходится большая часть нагрузки.

Для расследования проблем стабильности нужно знать и понимать, что происходило в этот момент, что и где делал пользователь. Чтобы расследовать большую часть проблем стабильности, достаточно собирать ТЖ, рекомендуемый для ежедневного использования.

Видеоурок. Дампы

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что такое дамп, и зачем он нужен
- Как включить сбор дампов.

Видеоурок. Расследование падений

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- Какой ТЖ необходимо настраивать для расследования «падений»
- Как найти причину «падения».

Видеоурок. Расследование падений с помощью специалистов фирмы 1С

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что делать, если не удалось выяснить причину «падений» самостоятельно.

Видеоурок. ЦКТП

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что такое проект ЦКТП
- В каком случае может помочь проект ЦКТП.

Видеоурок. Расследование «зависаний»

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как можно расследовать «зависания».

Видеоурок. Утечки памяти

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что такое утечки памяти, и как они проявляются.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Выявление утечек памяти

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как понять, что в системе есть утечки памяти
- Какие инструменты для этого можно использовать.

Видеоурок. Определение базы с утечками памяти

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как можно определить базу, в которой наблюдаются утечки памяти.

Видеоурок. Расследование утечек памяти с помощью ТЖ

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как с помощью ТЖ определить проблемный контекст
- Как посмотреть, сколько памяти было занято за вызов сервера.

Видеоурок. Расследование утечек памяти. Дампы

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что делать, если самостоятельно расследовать утечки не удалось.

Видеоурок. Рабочий процесс занял много памяти. Версия 8.2

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Что делать, если рабочий процесс занял много памяти
- Как исправить эту ситуацию
- Как избежать такой проблемы в будущем.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 48

Различные проблемы стабильности и их решение

Ошибка «Недостаточно памяти для выполнения запроса»

Ошибка часто встречается, если используется 32-разрядный сервер 1С, и может быть вызвана несколькими причинами:

- **Фрагментация адресного пространства.** Рекомендуется периодически перезагружать сервер приложений или рабочие процессы.
- **Не хватает памяти рабочему процессу при высокой нагрузке.** Рекомендуется добавить новые рабочие процессы.
- **Запрос действительно возвращает слишком много данных.** Рекомендуется исправить запрос.

Если используется 64-разрядный сервер приложений, данная ошибка возникает крайне редко.

Ошибка «Не обнаружен ключ защиты программы»

Основная причина данной ошибки – неверно настроенный файл nethasp.ini на компьютерах пользователей

Необходимо переписать файл следующим образом:

```
[NH_COMMON]
NH_TCPIP = Enabled      ; // вкл. использование TCP/IP
[NH_TCPIP]
NH_SERVER_ADDR = 192.0.0.1    ; // IP-адрес сервера с ключом
NH_USE_BROADCAST = Disabled ; // выкл. широковещательный запрос
```

Таким образом клиентское приложение будет «знать», на каком компьютере расположен ключ, и сразу обращаться именно туда.

Очень часто долгий запуск клиентского приложения как раз связан с излишне продолжительным поиском ключа.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Есть и более редкие причины данной ошибки:

- Еще одна возможная причина ошибки – это использование протокола IPv6. Следует помнить, что HASP License Manager работает только с протоколом IPv4
- Установка HASP License Manager на терминальный сервер при неполных правах администратора
- Ключ неисправен. Это можно проверить, переставив ключ на другой компьютер, либо включить в ТЖ событие HASP: если код возврата в логах будет FFFFFFFF, значит, ключ неисправен.

Не обнаружена программная лицензия

Типичные причины данной проблемы:

- **Изменилась аппаратная конфигурация сервера.** Следует помнить, что лицензия позволяет добавлять новое оборудование, но компоненты оборудования нельзя заменять. Также нельзя делать аппаратные мощности меньше, чем было на момент активации. Например, если на момент активации на сервере было установлено 128 ГБ памяти, то память можно добавить, но делать меньше 128 ГБ нельзя.
- **На виртуальной машине включено динамическое распределение ресурсов.** Если оно включено, то размер памяти и количество ядер на виртуальной машине могут измениться, что приведет к описанной ошибке. Для того чтобы избежать данной проблемы, следует жестко зафиксировать количество ресурсов для виртуальной машины.
- **Есть копии файла лицензии, которые доступны серверу приложений.** Не следует держать копию файла лицензии в каталоге, который доступен серверу приложений, иначе сервер деактивирует лицензию и выдаст ошибку.

Если лицензия была деактивирована, не важно по какой причине, то активировать ее повторно не получится. Обычно с лицензией выдается 3 запасных пин-кода. Для повторной активации придется использовать один из этих кодов. Если нужно будет активировать лицензию в четвертый раз, то придется обращаться в фирму «1С» с соответствующим запросом.

Очистка каталога сеансовых данных

Иногда в системе наблюдаются различного рода необычные ошибки и странное поведение процессов кластера:

- Не стартуют рабочие процессы
- Самопроизвольно запускается очень большое количество рабочих процессов
- Выдается сообщение «Ошибка формата потока»
- Замедление работы сервера приложений
- Прочие непонятные ошибки.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Если при возникновении этих ошибок не помогает даже перезапуск сервера 1С, то рекомендуется выполнить следующие действия:

- Остановить службу агента сервера
- Очистить каталог: *C:\Program Files\1cv82\srvinfo\reg_<НомерПортаКластера>\snccntx*
- Запустить агент сервера.

Очень часто этот способ помогает решать различные нестандартные проблемы.

Расследование проблем стабильности. Итоги

Проблемы стабильности гораздо сложнее в расследовании, чем проблемы производительности. Тем не менее, иногда их можно успешно решать самостоятельно, грамотно используя технологический журнал. Следует помнить, что сбор данных для решения проблем стабильности может вызвать замедление работы пользователей, особенно это касается расследования утечек памяти.

Следует отметить, что чем проще система, тем стабильнее она работает. Например, стабильнее всего работает приложение на обычных формах, при этом управляемое приложение выглядит, конечно, красивее, но работает медленнее и менее стабильно. Веб-клиент работает наименее быстро и стабильно.

Фирма «1С» постоянно делает множество оптимизаций и улучшений в работе платформы, но на данный момент лучше все же по возможности отказаться от использования веб-клиента: это позволит избежать целого ряда возможных проблем как со скоростью, так и со стабильностью. В качестве альтернативы веб-клиенту можно использовать тонкий клиент с подключением через веб-сервер, либо использовать терминальный сервер. Любой из этих двух вариантов всегда будет работать гораздо быстрее и стабильнее, чем веб-клиент.

Иногда для решения проблем стабильности полезно знать, какая программа захватила тот или иной файл, в этом случае можно использовать приложение Process Explorer.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Резервное копирование

Занятие 49

Основные понятия

В данной главе будут рассмотрены вопросы выполнения резервного копирования средствами MS SQL Server.

Зачастую многие специалисты делают резервное копирование с помощью выгрузки информационной базы в dt файл, что является в корне неверным. Выгрузка предназначена для переноса информационных баз между различными средами, например, при смене СУБД, но никак не для резервных копий.

Еще одно неоптимальное решение для резервного копирования – это копирование файлов базы данных MS SQL Server (mdf и ldf), что также является неправильным.

В обоих описанных случаях нельзя сделать копирование, не прерывая работы пользователей. При этом получившийся файл копии, даже с учетом сжатия будет довольно большим, что увеличивает стоимость хранения данных, к тому же нет возможности восстановить состояние системы на произвольный момент времени.

В данном разделе будет рассмотрен способ, который позволяет выполнять резервное копирование, не прерывая работу пользователей, получать файлы резервных копий меньшего размера, тратить меньше времени, а также автоматизировать процесс копирования.

Тема резервного копирования довольно обширна, в ней есть масса нюансов и тонкостей, подробное рассмотрение которых легко может вместиться в отдельную книгу. В данной главе будет рассмотрен необходимый минимум информации, достаточный для того, чтобы настроить резервное копирование, подходящее для подавляющего большинства информационных систем. Будут разобраны способы, позволяющие организовать восстановление системы практически на любой момент времени, в том числе и на момент сбоя.

Типы резервных копий

MS SQL Server поддерживает 3 основных типа резервных копий: полная, разностная и журнал транзакций.

Рассмотрим поочередно каждый из этих типов.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

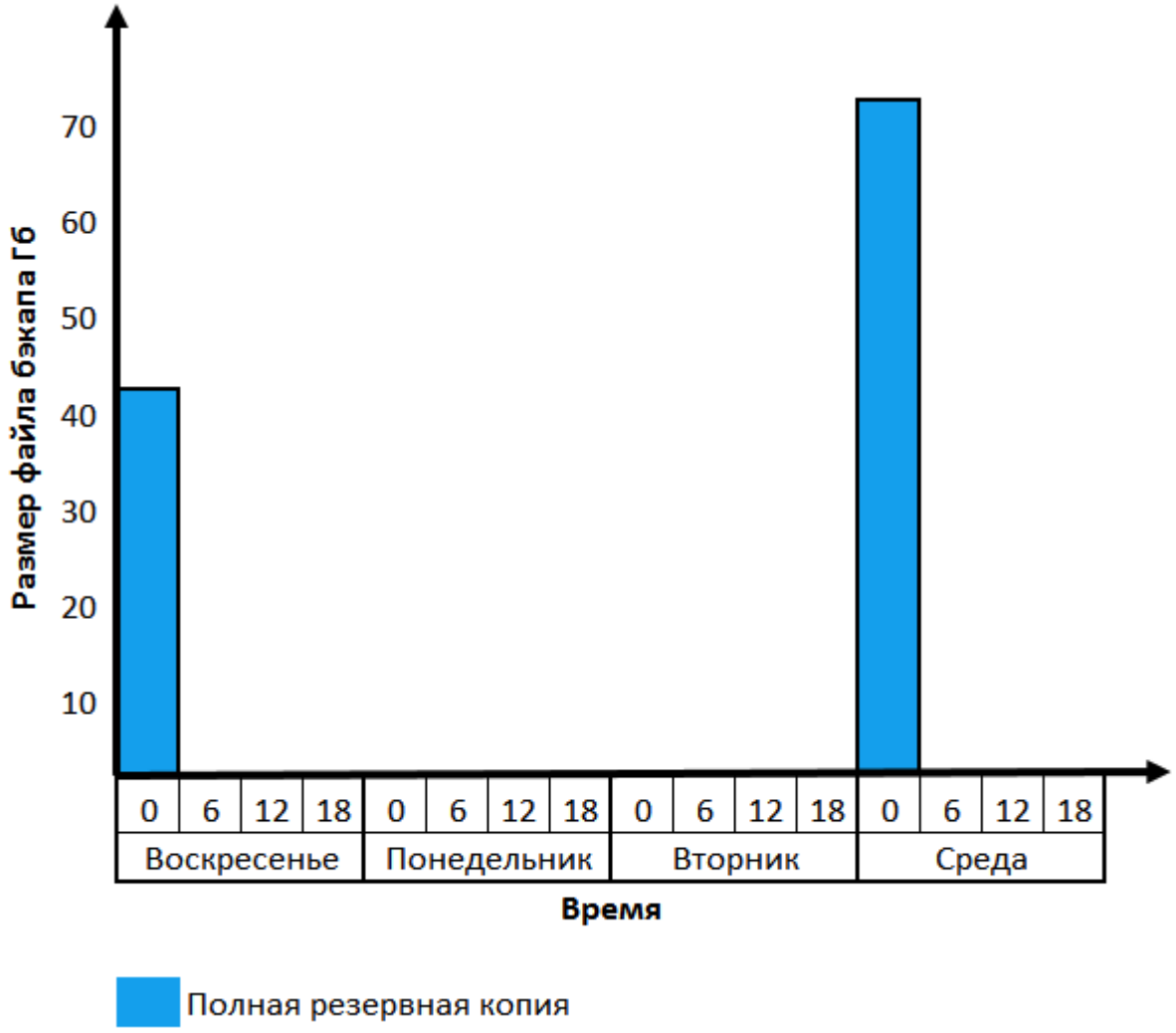
Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Полная резервная копия (Full backup)

В данном случае будет создана полная копия базы данных, которая включает в себя и файл данных, и файл журнала транзакций. Это самый простой и понятный тип бэкапа. Обычно полный бэкап выполняется 1 раз в неделю, и чем больше размер базы, тем дольше он выполняется и тем больше размер файла бэкапа.

Допустим, есть база размером 40 ГБ, и каждый день, включая выходные дни, база прирастает на 10 ГБ. Для простоты будем считать, что старые данные в базе не меняются, а только записываются новые. При этом два раза в неделю, в воскресенье и в среду в полночь, выполняется полный бэкап без сжатия.

Эту ситуацию можно отразить на графике, где на одной оси показан размер файла бэкапа, а на второй – время.



Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

На оси времени, одна клетка означает 6 часов, цифрой обозначается первый час шестичасового интервала.

Следует учитывать, что в бэкап попадают данные, которые были в базе данных на момент начала создания бэкапа. Если во время создания бэкапа в базе данных было что-то изменено, то это изменение в данный бэкап не попадет.

Если произойдет сбой, то для восстановления базы данных потребуется только загрузить один из файлов полного бэкапа и восстановить состояние базы либо на начало воскресенья, либо на начало среды.

Если сбой произойдет в промежутке между бэкапами, то данные с 00:00:00 воскресенья до момента сбоя будут утеряны. Получается, что интервал потери данных в этом случае равен нескольким дням, что очень много.

Конечно, можно было бы делать полный бэкап каждый день или даже чаще, но это вызовет целый ряд других проблем:

- Размер полного бэкапа сопоставим с размером базы (если не включено сжатие), что требует много дискового пространства для его хранения
- Создание полного бэкапа отнимает достаточно много времени: чем больше база, тем дольше создается бэкап
- Создание полного бэкапа сильнее нагружает оборудование, чем другие виды резервного копирования, т.к. обрабатывается большой объем данных.

Разностная (дифференциальная) резервная копия (Differential backup)

Для того чтобы избежать вышеописанных проблем, существует разностная резервная копия, или дифференциальный бэкап. В этом случае в файл бэкапа будут помещены только те данные, которые были изменены с момента последнего полного бэкапа. Если сделать несколько разностных бэкапов, то более поздние бэкапы будут включать в себя данные предыдущих.

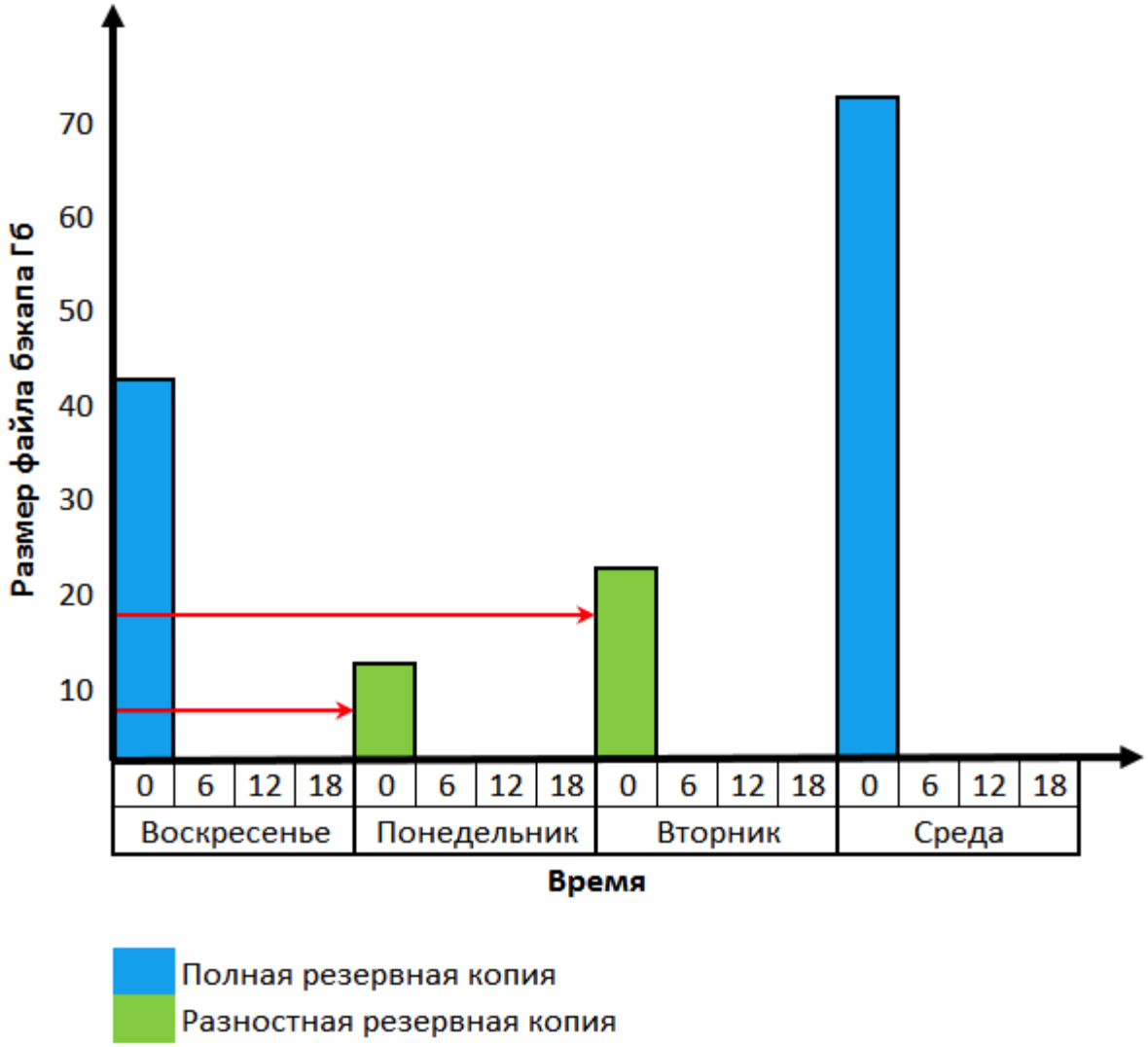
Допустим, что помимо создания полных бэкапов два раза в неделю, было также настроено создание разностных бэкапов каждый день в полночь, за исключением тех дней, когда выполняется полный бэкап.

С помощью графика это можно представить следующим образом.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



Красная линия показывает, за какой интервал времени хранит данные бэкап, расположенный справа от стрелки.

Например, бэкап вторника хранит в себе данные начиная с 0.00 воскресенья (время начала создания полного бэкапа) до 0.00 вторника (момент начала создания разностного бэкапа).

На графике видно, что разностная копия с каждым днем увеличивается, т.к. включает в себя все данные, которые есть в предыдущих разностных копиях. В понедельник разностный бэкап составлял 10 ГБ, а во вторник – уже 20 ГБ, из которых 10 ГБ – данные, добавленные за воскресенье, и 10 ГБ – новые данные за понедельник. В четверг разностный бэкап будет составлять 10 ГБ, т.к. в среду был сделан полный бэкап.

За счет того, что разностные бэкапы хранят только измененные данные, они занимают гораздо меньше места и создаются намного быстрее полных бэкапов.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Теперь в рассматриваемом примере можно восстановить данные на начало каждого дня. Например, если потребуется восстановить данные на начало вторника, тогда необходимо будет сначала восстановить полный бэкап, созданный в воскресенье, а потом применить разностный бэкап, созданный во вторник.

Если нет полного бэкапа, то разностный бэкап не имеет смысла, т.к. разностный бэкап хранит только дельту данных. Чтобы применить дельту, нужна опорная точка, в роли которой и выступает полный бэкап.

При использовании разностного бэкапа интервал потери данных сократился до одного дня. Этот результат, конечно, намного лучше, чем несколько дней из предыдущего примера, но все равно недостаточен с точки зрения надежности. В случае сбоя в вечернее время будут утеряны данные за целый рабочий день.

Резервная копия журнала транзакций (Log backup)

Перед тем как ознакомиться с данным разделом, рекомендуется повторить раздел «Общие сведения о файле данных и журнале транзакций» и освежить в памяти понятия **файл данных** и **файл журнала транзакции**.

Хотя разностная копия и занимает меньше места, чем полный бэкап, но выполнять такое копирование каждый час для сервера будет довольно ресурсоемкой работой, к тому же с каждым разом разностный бэкап будет создаваться все медленнее и медленнее и занимать все больше места, пока не будет сделан очередной полный бэкап.

Для того чтобы сократить интервал возможной потери данных до нескольких часов или даже нескольких минут, существует возможность создавать бэкап журнала транзакций (бэкап лога). В этом случае создается не копия самих данных, а копия журнала транзакций. Такой бэкап будет создаваться гораздо быстрее и иметь гораздо меньший размер, чем рассмотренные ранее бэкапы.

За счет малого объема и нетребовательности к ресурсам бэкап лога можно создавать каждые несколько минут. Обычно бэкап лога выполняют 1 раз в 15–30 минут.

Следует помнить, что файл журнала транзакций не хранит сами данные, он лишь содержит информацию о том, какие изменения были произведены в базе, что было и что стало в результате работы каждой транзакции. Используя эту информацию, можно восстановить состояние базы на определенный момент времени. Журнал не хранит информацию, которая не нужна для восстановления данных, например, там нет информации о том, какой запрос привел к изменению или какой план был у этого запроса.

Бэкап лога доступен только в том случае, если у базы установлен **полный режим восстановления.** О режимах восстановления и о том, как они связаны с бэкапами, будет рассказано в отдельном разделе курса. Сейчас достаточно знать, что у всех баз 1С режим восстановления по умолчанию как раз полный, так что бэкап лога для них доступен.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

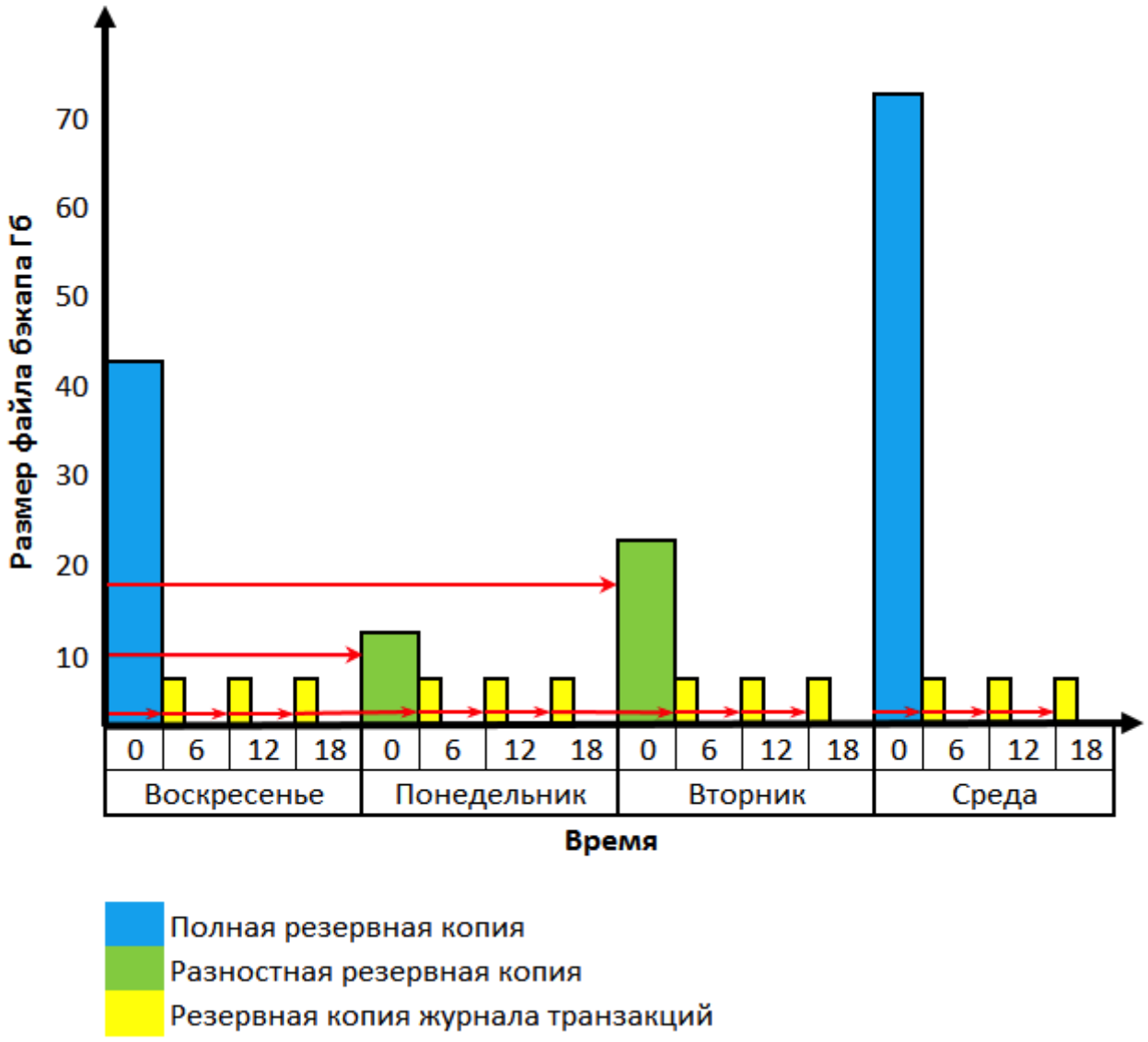
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

При использовании полной модели восстановления размер журнала транзакций увеличивается по мере того, как в системе происходят изменения данных. При выполнении резервного копирования журнала (именно журнала, а не данных), он усекается, т.е. уменьшается в размере. Именно за счет этого размер бэкапа лога получается таким маленьким и создается так быстро.

Бэкап журнала транзакций содержит в себе данные, которые были изменены начиная с момента предыдущего бэкапа журнала транзакций, а не с момента полного или разностного бэкапа. Например, если делать бэкап журнала каждый час, то его размер будет примерно одинаков, т.к. бэкап журнала за второй час не будет включать в себя данные за первый час, а это значит, что размер бэкапа будет минимальным.

Допустим, было решено каждые 6 часов выполнять бэкап журнала транзакций, за исключением тех случаев, когда выполняется полный или разностный бэкап. Представим, что за 6 часов логи растут на 5 ГБ, тогда график примет следующий вид:



Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Из графика видно, что бэкап лога имеет примерно одинаковый размер, т.к. не хранит в себе данные предыдущих бэкапов. Например, бэкап журнала, созданный в 18.00 в понедельник, хранит в себе только измененные данные журнала начиная с 12.00 до 18.00 понедельника, т.е. с момента начала создания предыдущего бэкапа лога до момента создания бэкапа в 18.00.

Таким образом, имея бэкапы логов и полный бэкап базы, можно восстановить состояние системы на любой момент времени.

Например, если во вторник в 18.30 произошел сбой, то для того чтобы восстановить базу на ближайший к сбою момент времени, можно воспользоваться одним из двух вариантов.

Вариант 1

Необходимо сначала накатить полный бэкап, созданный в воскресенье, потом разностный бэкап, созданный во вторник, и потом последовательно накатить бэкапы журналов: сначала 6-часовой, потом 12-часовой и в конце 18-часовой.

Вариант 2

Допустим, что бэкапы разностных копий были повреждены или удалены. В этом случае все равно можно восстановить базу данных на требуемый момент времени, используя бэкапы логов.

Для этого необходимо сначала накатить полный бэкап, созданный в воскресенье, а потом последовательно накатить все бэкапы журналов, начиная с 6-часового бэкапа воскресенья и заканчивая 18-часовым бэкапом вторника. Этот вариант будет работать дольше, чем первый, но его также можно использовать.

Этот вариант возможен, т.к. журнал хранит список всех изменений, происходящих с базой: что было и что стало с данными, а это значит, что, последовательно применив все изменения из журнала, мы получим базу в том состоянии, в котором она была на момент создания бэкапа лога.

В любом из двух вариантов данные будут потеряны максимум за последние 30 минут.

Последовательность наката бэкапов журнала очень важна. Например, если имеется 6-часовой бэкап, но 12-часовой бэкап лога будет поврежден или утерян, то мы не сможем накатить 18-часовой бэкап лога. В результате получится восстановить состояние базы только на 6.00 утра.

Когда речь идет о бэкапах лога, возникает понятие цепочки журналов транзакций.

Цепочка журналов транзакций – это непрерывная последовательность резервных копий журнала транзакций. Резервные копии журналов должны строго следовать друг за другом. Если в цепочке не хватит хотя бы одного журнала, то цепочка прервется, а это значит, что восстановить данные из бэкапа лога можно будет только до момента обрыва цепочки. В приведенном примере цепочка прерывается на 12-часовом бэкапе лога, потому что он поврежден или утерян, а это значит, что нельзя использовать бэкап на 18 часов, т.к. в нем хранится только дельта данных по отношению к 12-часовому бэкапу. Из этого следует, что можно использовать только бэкапы журнала, созданные до 12 часов вторника.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

При наличии непрерывной цепочки журналов транзакций можно восстановить состояние базы не только на момент создания последнего бэкапа лога, но и на любой момент времени в пределах цепочки журналов. Например, получен полный бэкап воскресенья на 0.00 и есть цепочка журналов транзакций начиная с 6.00 воскресенья до 18.00 вторника (включительно), как это показано на рисунке выше. В таком случае возможно восстановить базу на любой момент времени в промежутке между 0.00 воскресенья и 18.00 вторника.

Как восстановить базу на определенный момент времени, будет показано в соответствующем видеоуроке.

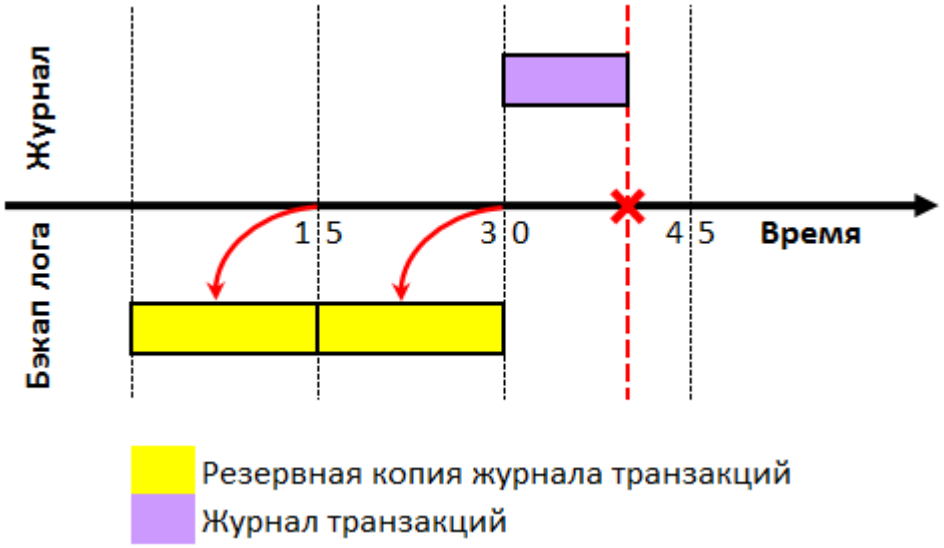
Следует понимать, что сам по себе бэкап лога бесполезен. Имея только бэкапы журнала, нельзя восстановить базу, для этого нужен хотя бы один полный бэкап, при этом цепочка журналов транзакций от полного бэкапа до последнего бэкапа журнала должна быть непрерывной.

Как видно из вышеописанного примера, бэкапы лога крайне важны и их обязательно нужно делать регулярно в течение дня. Если бэкапа лога нет, то систему можно будет восстановить только на момент создания последней разностной копии (при наличии предварительно сделанной полной резервной копии).

Резервная копия заключительного фрагмента журнала

Даже если выполнять бэкап лога каждые 15 минут, то данные за несколько минут перед сбоем все равно могут быть утеряны.

Допустим, бэкап лога выполняется каждые 15 минут, при этом сбой случился на 40 минуте. Эту ситуацию можно представить следующим образом.



Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Черной пунктирной линией показаны 15-минутные отрезки, а красной пунктирной линией показан момент времени, когда произошел сбой. Видно, что в 15 минут был создан бэкап журнала, который включает в себя информацию с начала часа по 15-ю минуту. В 30 минут был создан еще один бэкап, который включает в себя журнал с 15-й по 30-ю минуту. После последнего бэкапа лога журнал продолжал фиксировать изменения, и в 40 минут произошел сбой.

Если использовать только полный бэкап и бэкапы логов, то можно восстановить базу данных на момент 30-й минуты указанного часа. Данные, введенные с 30-й по 40-ю минуту, будут утеряны. Чтобы не потерять данные даже за эти 10 минут, следует использовать резервную копию заключительного фрагмента журнала.

Резервная копия заключительного (конечного) фрагмента журнала – это бэкап лога базы уже после того, как произошел какой-то сбой, например, диск с файлами данных базы вышел из строя. В бэкап конечного фрагмента входят данные журнала, которые не попали в предыдущий бэкап лога.

Если в результате сбоя сам журнал не пострадал (он был на другом диске), то можно восстановить данные на момент сбоя, т.е. на 40-ю минуту. Это значит, что никакие данные не будут утеряны, кроме транзакций, которые были открыты на момент сбоя – по ним будет сделан откат.

Если в результате сбоя файл журнала был поврежден, тогда восстановить данные на момент сбоя уже не получится. В этом случае придется взять бэкапы логов и восстановить базу на 30-ю минуту. Именно поэтому, а также по ряду других причин, рекомендуется располагать файлы журналов на надежном носителе с дублированием, например, использовать RAID10.

Чтобы восстановить базу на время, максимально близкое к моменту сбоя, необходимо выполнить следующую последовательность действий:

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

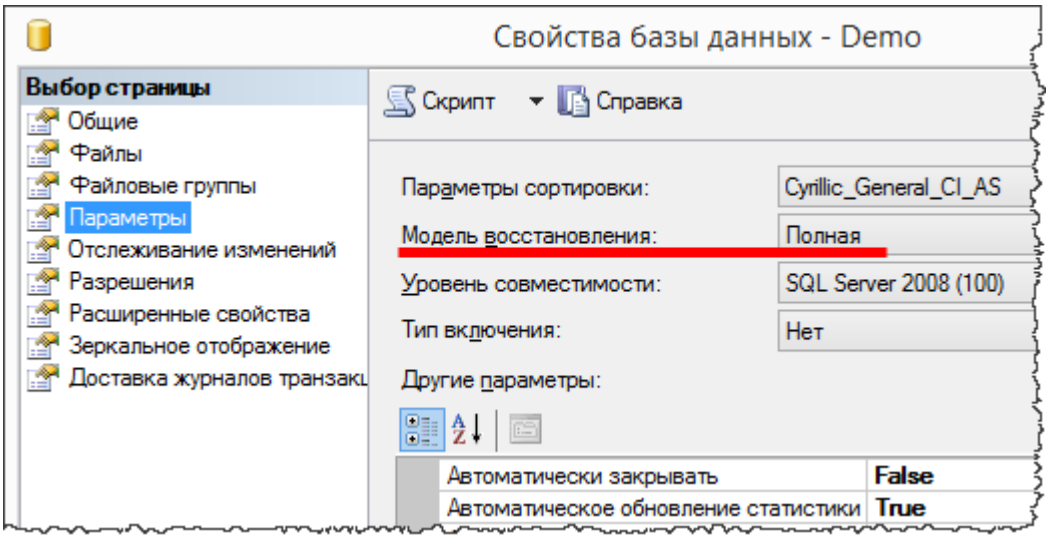
- Сделать резервную копию заключительного фрагмента журнала
- Накатить полный бэкап, ближайший по времени к моменту сбоя
- Накатить разностный бэкап, ближайший по времени к моменту сбоя
- Накатить поочередно бэкапы логов с момента разностного бэкапа до момента сбоя
- Накатить бэкап заключительного фрагмента журнала.

В этой цепочке разностных бэкапов может и не быть, главное, чтобы были полный бэкап и непрерывная цепочка журналов с момента полного бэкапа.

Отметим, что выполнять всю последовательность действий вручную совершенно не обязательно: MS SQL Server самостоятельно сортирует и восстанавливает бэкапы в нужной последовательности.

Модели восстановления

У всех баз данных MS SQL Server есть свойство «Модель восстановления».



Модель восстановления – это свойство базы данных, которое управляет процессом регистрации транзакций, определяет, требуется ли для журнала транзакций резервное копирование и какие типы операций восстановления доступны.

Если говорить проще, то данное свойство определяет поведение журнала транзакций, а именно – за какой период будут храниться данные в журнале и какие события в нем будут фиксироваться. Модель восстановления играет важную роль в процессе восстановления базы из бэкапа.

Модель восстановления может принимать одно из трех значений: полная, простая и с неполным протоколированием. Модель восстановления можно изменить в свойствах базы в любой момент времени с любого на любое другое значение.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Рассмотрим каждую из моделей восстановления подробнее.

Полная (Full)

При полной модели восстановления журнал транзакций хранит максимальную информацию обо всех транзакциях. При этом хранятся не только текущие активные транзакции, но и все изменения всех предыдущих транзакций. Если, например, бэкап журнала не выполнялся в течение месяца, тогда файл журнала транзакций содержит информацию обо всех изменениях в базе, которые произошли за этот месяц. Следовательно, при полной модели восстановления файл журнала постоянно увеличивается до тех пор, пока не будет сделан бэкап лога. Если не делать бэкап лога длительное время, файл журнала может расти неограниченно и со временем занять все свободное место на диске.

В момент создания копии журнала он усекается и в нем остаются только данные об активных на текущий момент транзакциях. Усечение журнала не означает уменьшение размера файла журнала на диске, просто место внутри этого файла будет освобождено. При выполнении полного или разностного бэкапа усечение журнала не происходит.

Чтобы сократить объем, занимаемый файлом журнала на диске, необходимо выполнить сжатие журнала, т.е. выполнить команду DBCC SHRINKFILE. Следует понимать, что усечение журнала – это освобождение места внутри физического файла, а сжатие файла журнала – это уменьшение размера файла журнала на диске.

Особо отметим, что не следует путать полную модель восстановления и полный бэкап – это разные понятия.

Простая (Simple)

В данном случае в журнале транзакций будет храниться минимальный набор данных. Когда файл журнала транзакций будет заполнен на 70%, система начнет писать данные в начало файла, затирая самые ранние данные. При такой модели восстановления файл журнала транзакций будет занимать минимальный объем на диске, но не будет хранить никакой истории транзакций. Так как история транзакций не хранится, то нет смысла делать бэкапы логов.

Таким образом, если для базы данных выбрана простая модель восстановления, то бэкап журнала транзакций для нее становится недоступным.

С неполным протоколированием (Bulk logged)

Дополнение к полной модели восстановления и отличается от нее лишь тем, что в журнал транзакций пишется минимальная информация при массовых операциях по изменению данных.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

В данной модели восстановления при массовом изменении данных журнал транзакций не будет разрастаться так сильно, как при полной модели. Для баз данных, работающих на платформе 1С:Предприятие, данная модель не имеет большого практического смысла, поэтому далее будут рассматриваться только первые две модели восстановления.

Влияние модели восстановления на резервное копирование

Модель восстановления напрямую влияет на процесс восстановления данных из бэкапов.

Полная модель восстановления

Если база данных использует полную модель восстановления, то с этой базы можно делать все 3 вида резервных копий: полную, разностную и журнала транзакций. Само название «Полная модель восстановления» говорит о том, что можно использовать максимальные возможности резервного копирования. При наличии полного бэкапа и цепочки журналов транзакций можно восстановить состояние базы данных на любой момент времени с момента полного бэкапа до момента создания последнего бэкапа лога в цепочке.

Допустим, есть следующие бэкапы:

- Полный бэкап за воскресенье 0.00
- Бэкап лога за воскресенье 6.00
- Бэкап лога за воскресенье 12.00
- Бэкап лога за воскресенье 18.00
- Разностный бэкап за понедельник 0.00
- Бэкап лога за понедельник 6.00
- Бэкап лога за понедельник 12.00
- Бэкап лога за понедельник 18.00
- Разностный бэкап за вторник 0.00

В этом случае можно восстановить состояние системы на любой момент времени начиная с воскресенья 0.00 и заканчивая понедельником до 18.00, а также на момент времени 0.00 вторника. Чтобы восстановить данные на 14.00 понедельника, нужно будет:

- Сначала развернуть полный бэкап за воскресенье
- Сделать накат разностного бэкапа за понедельник
- Сделать накат трех бэкапов лога за 6, 12 и 18 часов.

Таким образом, если бэкапы логов выполняются достаточно часто, то риск потери данных будет минимальным.

Рассмотрим еще один пример.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Допустим, есть тот же набор бэкапов, но два из них повреждены:

- Полный бэкап за воскресенье 0.00
- Бэкап лога за воскресенье 6.00
- Бэкап лога за воскресенье 12.00
- Бэкап лога за воскресенье 18.00 (файл бэкапа поврежден)
- Разностный бэкап за понедельник 0.00
- Бэкап лога за понедельник 6.00
- Бэкап лога за понедельник 12.00 (файл бэкапа поврежден)
- Бэкап лога за понедельник 18.00
- Разностный бэкап за вторник 0.00

В таком случае можно восстановить состояние системы на следующих интервалах времени:

- С 0.00 воскресенья до 12.00 воскресенья на любой момент времени
  - Накатить полный бэкап, потом 2 бэкапа лога
- С 0.00 понедельника до 6.00 понедельника на любой момент времени
  - Накатить полный бэкап, потом разностный бэкап и в конце бэкап лога.

Из-за того, что цепочка журналов прервалась, восстановить данные в промежутке межу поврежденным бэкапом и следующим за ним полным или разностным бэкапом невозможно. В данном примере нельзя восстановить систему на момент в интервале с 12.00 воскресенья до 0.00 понедельника и с 6.00 понедельника до 0.00 вторника.

Простая модель восстановления

Если база данных использует простую модель восстановления, то бэкап журнала для нее недоступен, т.к. в этом случае журнал не хранит всю историю транзакций. В данном случае доступны только полные и разностные резервные копии. Это значит, что данные можно восстановить не на любой момент времени, а только на момент создания полного или разностного бэкапа.

Допустим, есть следующие бэкапы:

- Полный бэкап за воскресенье 0.00
- Разностный бэкап за понедельник 0.00
- Разностный бэкап за вторник 0.00

В данном случае нет возможности восстановить базу данных на любой момент времени. При наличии таких бэкапов базу можно восстановить только на начало любого из трех дней.

Это значит, что есть риск потерять данные за один рабочий день, т.к. разностный бэкап в течение дня обычно не выполняется, чтобы не создавать излишнюю нагрузку на оборудование.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Какую модель восстановления выбрать?

Для рабочих баз следует использовать полную модель восстановления, чтобы иметь возможность выполнять бэкапы логов и при необходимости восстановить базу данных на момент сбоя. Минус данной модели только в том, что размер файла журнала растет, пока не будет сделан бэкап лога.

Для баз разработчиков, различных технических и тестовых баз, например, баз сервисов и ЦУП, можно использовать простую модель восстановления, т.к. там обычно не требуется восстанавливать данные на какой-то определенный момент времени. Файл журнала для этих баз расти практически не будет.

Модель	Преимущества	Недостатки	Применение
Полная	Восстановление на любой момент времени	Растет журнал лога, необходимо периодически делать его бэкап	Для рабочих баз
Простая	Журнал практически не растет	Нельзя восстановить на любой момент времени	Для тестовых и технических баз

Стратегия резервного копирования

У системного администратора обязательно должна быть стратегия резервного копирования: какие бэкапы когда создавать, сколько будут хранится копии бэкапов, где они будут хранится и т.д.

Не рекомендуется хранить бэкапы на том же физическом диске, на котором располагается сама база данных. В большинстве случаев подходит следующая стратегия резервного копирования:

Создание бэкапов

Полный бэкап – 1 раз в неделю на выходных. Конечно, можно делать полный бэкап каждый час, но это сильно загрузит оборудование и бэкапы будут занимать слишком много места на диске.

Разностный бэкап – 1 раз в день в ночное время.

Бэкап лога – каждые 15–30 минут. Бэкап лога можно делать и чаще, но обычно это не имеет большого смысла, т.к. в случае сбоя можно будет сделать бэкап конечного фрагмента журнала. Даже если бэкап конечного фрагмента сделать не удастся (что бывает крайне редко), то потеря данных за 15-30 минут обычно не является большой проблемой: данные за потерянные 15-30 минут можно быстро внести вручную.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Местоположение бэкапов

Бэкапы должны храниться на физическом диске, отличном от диска, на котором расположена база данных.

Рекомендуется хранить файлы бэкапов минимум в двух разных местах.

Обычно одна копия хранится территориально там же, где расположена база данных, но на другом диске. Это нужно для того, чтобы в случае сбоя можно было быстро произвести восстановление и не тратить время на копирование файлов бэкапа из другого места.

Вторая копия должна храниться территориально в другом месте, например, на удаленном сервере или в облачном хранилище. Вторая копия необходима на случай повреждения или утери первой копии.

При разработке стратегии резервного копирования нужно исходить из того факта, что весь мир настроен против вас. Уборщицы, американские шпионы, метеориты и дикие животные – все они только и ждут за углом, чтобы уничтожить ваши данные. Не давайте им шанса.

При выборе носителя для бэкапов следует помнить о том, что чем выше у него скорость передачи данных, тем быстрее восстановится база из бэкапа. Хранить полный бэкап базы в 1Тб на диске с интерфейсом USB 2.0 – не очень хорошая идея.

Чтобы снизить стоимость хранения, можно для локальной копии бэкапа использовать быстрый диск, но хранить там бэкап только за последнюю неделю, а на удаленном сервере использовать более медленный диск и хранить там бэкапы за последние несколько недель. В подавляющем большинстве случаев, даже если случится сбой, то для восстановления данных будет достаточно локальной копии бэкапа.

Период хранения бэкапов

Если полный бэкап создается каждую неделю, то обычно нет смысла хранить бэкапы старше 2-3 недель. В некоторых случаях может потребоваться хранить полные бэкапы за несколько лет, например, по требованию бухгалтерии.

Инструкция по восстановлению

Обязательно должна быть четкая и понятная инструкция, описывающая порядок действий в случае сбоя. Даже если систему обслуживает опытный специалист, который все знает и все умеет, в стрессовой ситуации он тоже может что-то забыть. Это может привести к ошибками или просто увеличить время простоя. Наличие инструкции имеет еще один неочевидный плюс: если специалист, обслуживающий систему, уходит в отпуск, он должен передать план действий в случае сбоя своему заместителю. В первую очередь это в интересах самого специалиста, т.к. вряд ли кому-то понравится вызов на работу ранним утром во время отпуска.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Инструкция должна быть максимально простой, но подробной. Возможно, стоит записать видеоинструкцию: это может занять меньше времени, чем написание ее бумажного варианта.

Запомните главное: даже самая плохая инструкция из 10 строк гораздо лучше, чем вообще никакой.

Регулярные проверки

Как минимум раз в месяц необходимо выполнять плановую проверку восстановления из бэкапов. Например, можно восстанавливать бэкап рабочей базы в копию. Проверки полезны по следующим причинам:

- **Оценка времени на восстановление.** Вы должны точно знать сколько времени займет восстановление базы. Например, если полгода назад база восстанавливалась за 10 минут, то это вовсе не значит, что сейчас она будет восстанавливаться за то же время. Необходимо проверить разные сценарии восстановления, например, сколько времени займет восстановление из локальной копии бэкапа, а сколько – из копии на удаленном сервере. Скорость соединения с удаленным сервером здесь будет играть важную роль.
- **Отработка нештатных ситуаций.** Что делать, если полный бэкап оказался поврежден? Как взять бэкап с удаленного сервера, если нет соединения с интернетом или соединение очень медленное? Что делать, если есть полный бэкап и бэкапы логов, но разностный бэкап оказался поврежден? Ответы на все эти вопросы нужно обязательно получить, проверив все на практике. Все возможные нештатные ситуации, которые удалось придумать, и пути их решения нужно обязательно отразить в инструкции. Потратив один раз время на создание инструкции, в будущем вы сэкономите гораздо больше и времени, и нервных клеток, а также гарантируете себе спокойный отпуск без неожиданностей.
- **Регулярная практика.** Если случится сбой, думать будет уже некогда, в этом случае нужно на автомате выполнить уже отработанные действия, сверяясь с инструкцией при необходимости. Регулярные проверки как раз дают возможность эти действия отработать. Как говорится: кто готов, тому не надо готовиться.

Для того чтобы не забывать регулярно выполнять плановую проверку, лучше всего настроить автоматическое напоминание. Например, можно делать проверку бэкапов 1 раз в каждый первый понедельник месяца.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 50

Создание и восстановление резервных копий

Видеоурок. Основные настройки резервного копирования

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как выбрать тип резервной копии
- Как указать каталог по умолчанию для создания резервных копий
- Как сжать резервную копию.

Видеоурок. Создание полного бэкапа интерактивно

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как создать полный бэкап средствами Management Studio
- Какие настройки указывать при создании полного бэкапа.

Видеоурок. Восстановление из полного бэкапа

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как восстановить базу из полного бэкапа
- Какие настройки указывать при восстановлении полного бэкапа.

Видеоурок. Сжатие бэкапов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как можно сжать бэкап
- Насколько сильно может уменьшиться размер бэкапа.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Создание бэкапа с помощью скрипта

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как создать бэкап с помощью команд MS SQL Server
- Как автоматически сгенерировать такой скрипт.

Видеоурок. Создание разностного бэкапа

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как создать разностный бэкап
- Какие настройки указывать при создании разностного бэкапа.

Видеоурок. Восстановление из разностного бэкапа

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как восстановить базу из разностного бэкапа.

Видеоурок. Создание второго разностного бэкапа

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Чем отличается создание второго разностного бэкапа от первого
- Как меняется размер разностного бэкапа в зависимости от его «удаленности» от полного бэкапа
- Восстановление на момент создания разностного бэкапа.

Видеоурок. Создание бэкапа лога

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучив этот урок, Вы узнаете:

- Как создать бэкап лога транзакций
- Можно ли сжимать бэкап лога.

Видеоурок. Восстановление бэкапа лога

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как восстановить базу из бэкапа лога
- Как восстановить базу на определенный момент времени с использованием бэкапа лога.

Видеоурок. Создание цепочки журналов транзакций

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Пример создания цепочки журналов транзакций
- Пример восстановления системы на произвольный момент времени в пределах цепочки журналов транзакций.

Видеоурок. Бэкап конечного фрагмента журнала транзакций

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как сделать бэкап конечного фрагмента журнала транзакций
- Как восстановить состояние базы на момент сбоя.

Видеоурок. Стратегия резервного копирования

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как часто и какие бэкапы делать
- Стратегия резервного копирования, подходящая для большинства баз.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Автоматизация создания бэкапов

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как создать план обслуживания, создающий бэкапы
- Как настроить расписание для каждого типа резервной копии.

Резервное копирование. Итоги

Резервное копирование – это важнейшая операция, которая должна выполняться в обязательном порядке. Самым оптимальным способом является создание резервных копий средствами СУБД. Таким образом можно делать как полные, так и дифференциальные бэкапы.

Использование дифференицальных бэкапов существенно снижает нагрузку на оборудование и используемое дисковое пространство. Полный бэкап хранит полную копию базы данных на момент создания бэкапа. Разностный бэкап хранит только изменения, которые произошли в базе с момента создания полного бэкапа.

Чтобы восстановить состояние базы на любой момент времени, следует использовать бэкапы лога (журнала транзакций), но это возможно только в том случае, если база использует полную модель восстановления данных. В простой модели восстановления бэкап лога недоступен.

Системный администратор в обязательном порядке должен иметь стратегию резервного копирования: какие бэкапы создавать и с какой периодичностью, где и сколько они будут храниться и т.д. Все это должно быть четко и понятно описано в инструкции.

Рекомендуется написать подробную инструкцию по восстановлению базы. В инструкции должны быть подробно описаны все шаги, а также возможные проблемы и пути их решения.

Отдельно следует позаботиться о резервном копировании файлов журнала регистрации платформы, т.к. файлы журнала регистрации не хранятся в СУБД.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Подготовка к аттестации 1С:Эксперт

Данный раздел является обязательным только для тех слушателей курса, которые готовятся к экзамену «1С:Эксперт по технологическим вопросам». Если вы не планируете сдавать этот экзамен, то этот раздел курса можно пропустить. Однако, при наличии свободного времени и желания расширить свои знания в вопросах оптимизации, рекомендуется всем слушателям курсов ознакомиться с данной информацией.

Занятие 51

Регламент экзамена

Сертификация длится 4 дня и состоит из 2 этапов: первый этап (1 день) – это сам экзамен, второй этап (3 дня) – это тренинг с практическими заданиями.

Отсюда следует, что если вы не сдали экзамен в первый день, то это еще не значит, что вы провалили сертификацию – у вас будет целых 3 дня, чтобы исправить ситуацию.

Для успешной сертификации необходимо набрать минимум 3 плюса за все 4 дня экзамена, но даже если вы получили 3 плюса, это еще не гарантия того, что вы получите сертификат. В подавляющем большинстве случаев экзаменуемые получают 1 или 2 плюса в первый день и дополнительные плюсы в течение следующих 3 дней. Окончательное решение о выдаче сертификата принимает преподаватель, при этом он ориентируется не только на плюсы, но и на общее впечатление, которое вы произвели.

За неправильные ответы экзаменаторы ставят минусы, но они не отменяют ваших плюсов.

Чтобы набрать плюсы, нужно:

- Правильно ответить на теоретическом экзамене
- Быстро и правильно сделать практические задания
- Быть активным на тренинге.

Разберем эти пункты подробнее.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

1. **Правильно ответить на теоретическом экзамене.**

Первый день полностью посвящен экзамену, время для подготовки – 1 час, но в действительности вы можете готовиться столько, сколько посчитаете нужным. Когда вы будете готовы, выходите и пишите свое имя на доске под предыдущим именем. После ответа, перед выходом из аудитории нужно будет стереть или зачеркнуть свое имя. При этом во время подготовки вы можете пользоваться дополнительными материалами: книгами, смартфоном и пр. Экзаменационный билет – это лишь повод поговорить. Преподаватели отлично знают, что билеты есть в сети, поэтому большее внимание они будут уделять именно дополнительным вопросам.

На вопросы лучше отвечать своими словами, а не заученными фразами. Вопросы могут быть самые разные, от: «Как происходит подсчет лицензий в облачных сервисах?» – до: «Что такое закон Амдала, и в чем его практический смысл?». Примеры вопросов для самопроверки, в том числе и дополнительных вопросов, приведены конце данного раздела.

В экзаменационном билете раньше было 3 вопроса, сейчас – 10. Возможно, когда вы будете сдавать экзамен, число вопросов снова изменится. Если ответите правильно на все вопросы билета и на все дополнительные вопросы, то получите 3 плюса сразу в первый день. Но не все так просто, как кажется. Преподаватель будет задавать очень много дополнительных вопросов на произвольные темы. И самое сложное как раз заключается в ответах на эти дополнительные вопросы. При этом по отвечающему сразу видно, понимает ли он свой ответ или просто произносит заученную фразу, поэтому лучше отвечать на вопросы своими словами.

Не отвечайте на вопрос фразами из книг или документации, постарайтесь передать ту же мысль, но по-своему. Можно сначала ответить так, как написано в книге, а потом рассказать, как вы понимаете процитированный текст.

Необходимо не только дать ответ на вопрос, но и объяснить, почему именно этот ответ правильный. При этом надо быть готовым к различным уточняющим вопросам, вплоть до того, какой уровень изоляции будет в том или ином случае и какой план запроса получится, если выполнить определенный запрос, и т.д.

Домашние задания во многом похожи на экзаменационные билеты, и если вы их сделаете самостоятельно, то на экзамене у вас не должно возникнуть сложностей.

Очень часто на экзамене просят пошагово описать свои действия. Зная одну лишь теорию, будет сложно ответить на такой вопрос. Например, могут попросить описать шаг за шагом ваши действия, если потребуется собрать данные о загруженности оборудования на Linux или описать свои действия при обнаружении взаимных блокировок на блокировках 1С и т.д.

Чтобы ответить на все эти вопросы, нужно все проверять на практике, за плечами должна быть не только голая теория. Чем больше будете экспериментировать, тем лучше.

Если вы чувствуете, что отвечаете по одному из вопросов слабо, хотя до этого вы все ответили верно, то попросите преподавателя задать вам дополнительные вопросы. В крайнем случае, можно сказать, что ответ вы сейчас не знаете, но если бы вы столкнулись с таким вопросом на практике, то сделали бы такие-то и такие-то действия, чтобы узнать правильный ответ.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Встречаются и каверзные вопросы, например: «Как с помощью одного лишь SQL Profiler узнать, что есть избыточные ожидания на блокировках?». Ответ должен быть примерно следующим: «SQL Profiler показывает, что было ожидание на блокировках СУБД с помощью события Lock:Acquired. Но было ли это ожидание необходимым или излишним, точно сказать нельзя, для этого нужно ставить ЦУП и собирать аналитический показатель Анализ ожиданий на блокировках».

Первый день хоть и не является определяющим, но у преподавателя складывается о вас определенное впечатление, поэтому желательно постараться заработать максимальное количество плюсов в первый день. Большая часть экзаменующихся в первый день 3 плюса не получает, а зарабатывает недостающие плюсы уже на тренинге.

В данном курсе достаточно информации для подготовки и сдачи экзамена в первый день. Но чтобы эта информация лучше усвоилась, нужно все проверить на практике.

Чем больше опытов вы проведете, тем выше ваши шансы.

Часто перед ответом на вопрос по билетам экзаменаторы дают вопросы из комплекта «1С:Профессионал по технологическим вопросам» без вариантов ответа. Вам необходимо будет написать свои ответы.

Проверьте себя, пройдя бесплатное тестирование на сайте: http://dist.edu.1c.ru/1CEduWeb/

2. Быстро и правильно сделать практические задания

На тренинге помимо лекций будут и практические задания. Многие задания из курса (особенно необязательные) похожи на практические задания тренинга. Как правило, большая часть учащихся само задание выполняет быстро. Но сложности возникают, когда преподаватель просит пояснить, почему вы выполнили задание именно так. Нужно обязательно объяснить, почему вы сделали именно так, а не иначе, доказать, что именно это решение правильное.

После выполнения задачи желающий (обычно это первый, кто сделал правильно) может выйти «к доске» и за компьютером преподавателя показать, как он выполнил задание. В аудитории есть проектор, так что все экзаменуемые смогут увидеть правильное решение.

За выход к доске вы зарабатываете дополнительный плюс, но это не является гарантией того, что вы прошли сертификацию. Т.е. выходить к доске желательно, но не обязательно.

Для выполнения заданий нужно уметь пользоваться такими инструментами как ЦУП, Тест-Центр, SQL Profiler и технологический журнал. Поэтому желательно заранее установить их себе на компьютер и выполнить практические задания данного курса с использованием этих инструментов. О сервисах Гилева на экзамене лучше вообще не упоминать.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

3. **Быть активным на тренинге**

На тренинге необходимо проявлять активность, задавать вопросы и участвовать в обсуждениях. Не нужно задавать простые или глупые вопросы, так вы только рискуете заработать минус. Если задаете вопрос, то он должен быть действительно интересным. Очень хорошо, если вы сможете поделиться своим опытом из практики оптимизации, рассказать, как помог вам тот или иной прием.

Активно участвуйте в обсуждениях, вас должны запомнить как грамотного специалиста, который уже имеет за плечами определенный опыт.

Помните, что все должно быть в меру, не стоит весь тренинг сидеть на «галерке» молча, но и чрезмерно усердствовать с активностью тоже нежелательно.

Вас должны запомнить, но не как человека, надоедающего вопросами и не дающего вставить слово преподавателю, а как грамотного специалиста, которому есть что рассказать коллегам.

Советы по сдаче экзамена

Не торопитесь с ответом

У вас есть время на подготовку, вас никто не торопит, поэтому лучше подумать минуту и ответить правильно, чем ответить сразу, но неверно. Помните, что вопрос может быть с «подвохом».

Например:

Вопрос: Какие события нужно включить в ТЖ, чтобы расследовать взаимоблокировки СУБД, используя только технологический журнал?

Ответ: Для расследования нужны трассировки блокировок СУБД, без них взаимоблокировку СУБД расследовать нельзя. С помощью одного лишь ТЖ можно расследовать только взаимоблокировки управляемых блокировок.

Так что торопиться с ответом не стоит.

Старайтесь сделать задачу первым, чтобы выйти и показать решение, в этом случае вы заработаете дополнительный плюс.

Отвечайте на вопросы «своими словами», **не надо произносить заученные фразы**. Главное – донести до преподавателя, что вы действительно понимаете тему, а не просто заучили наизусть правильный ответ. Отвечайте как можно более полно и развернуто.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Не волнуйтесь. Очень многие не сдали экзамен не потому, что не знали правильного ответа, а просто потому, что не смогли справиться с волнением и не ответили на простой вопрос, хотя знали ответ, но «все вылетело из головы».

Не расстраивайтесь, если вы не получили 3 плюса в первый день. Сертификация идет 4 дня, а не 1. В большинстве случаев экзаменуемые добирают плюсы уже по ходу тренинга. Известен случай, когда очень грамотный специалист получил 1 плюс в первый день (из-за волнения не смог ответить на вопрос) и сильно расстроился, что помешало ему заработать недостающие баллы на тренинге. При этом его коллега (слабее по подготовке) с тем же результатом не сдался и в конце тренинга получил заветный сертификат. Так что не сдавайтесь раньше времени, боритесь до последней минуты.

Если у вас есть знакомые, которые уже получили сертификат эксперта, пообщайтесь с ними. Наверняка они дадут вам ценный совет и расскажут, что все не так сложно и страшно, как кажется.

Обязательно сделайте все домашние задания этого курса, особенно необязательные – таким образом вы получите необходимые практические навыки, которые пригодятся вам на тренинге.

Для лучшего понимания материала **самостоятельно сделайте свою тестовую базу** и воспроизведите на ней различные проблемы производительности. Например, воспроизведите неоптимальные запросы и посмотрите получившийся план, воспроизведите взаимоблокировки, поставьте Linux и пособирайте данные о загруженности оборудования.

Специально создайте избыточные блокировки и взаимоблокировки и расследуйте их с помощью ЦУП.

Поставьте платформу 8.3, посмотрите настройки кластера, сравните с 8.2. «Прощупайте» все своими руками: когда все действия выполняешь самостоятельно, все воспринимается и запоминается гораздо лучше.

Поставьте PostgreSQL и поэкспериментируйте на нем. Поставьте точки останова при проведении документов. Посмотрите, в каком режиме блокировок документы проводятся параллельно, а в каком нет. Что будет, если указать одинаковые данные в документах, что будет, если указать разные данные. Пробуйте разные варианты. Сейчас на экзамене стали задавать различные вопросы по поведению системы при работе с «версионниками», в частности, с PostgreSQL, так что будьте к этому готовы.

Плотно поработайте с технологическим журналом. На экзамене очень любят задавать вопросы по ТЖ. Обязательно прочитайте соответствующий раздел документации, поработайте с основными событиями. Разберитесь, что пишется в свойства событий *TLOCK* и *TDEADLOCK*, поработайте с фильтрами. Будет просто отлично, если вы заучите наизусть события, их основные свойства (имя базы, пользователь, контекст, приложение, все свойства учить не нужно), как они называются и что они означают.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Изучите события SQL Profiler. В частности, уделите внимание событиям раздела *Lock*, вы должны точно знать, зачем они нужны и что означают. Все события подробно описаны здесь: https://msdn.microsoft.com/ru-ru/library/ms175481(v=sql.120).aspx

Получите граф взаимоблокировки с помощью SQL Profiler, попробуйте сами его прочитать и т.д. В жизни с событиями *Lock* вам вряд ли предется работать, для этого есть сервисы и ЦУП, но для экзамена это нужно знать.

Не нужно упоминать на экзамене о том, что проходили этот курс и об инструментах Гилева.

Данный курс не является официальным источником информации, это прямой конкурент курсам учебного центра. Лучше не говорить на экзамене, что вы проходили этот курс, иначе число дополнительных вопросов может резко возрасти. Об облачных сервисах также лучше не говорить, т.к. это неофициальный инструмент и прямой конкурент ЦУП.

Не спорьте с преподавателем. Это как русская рулетка: если повезет, и вы его убедите, то он поставит вам плюс, но это будет сложно сделать. Лучше отвечать так, как написано в документации, хотя практика зачастую отличается от теории – такие моменты описаны в курсе. Если вы все же решите оспорить его мнение, то сначала проведите опыты самостоятельно, не нужно слепо верить документации и этому курсу, проверьте все на практике. Если есть четкое воспроизведение ситуации, тогда, вооружившись ноутбуком, можно попробовать оспорить мнение преподавателя.

Решение задач, аналогичных экзаменационным

На аттестации обычно решается 4 задачи, хотя возможно и меньше, в зависимости от свободного времени. Первая задача посвящена поиску неоптимальных запросов, вторая – написанию простейшего теста, который будет создавать и проводить копию документа. Как это сделать, уже было рассмотрено в курсе.

Еще две задачи посвящены воспроизведению и анализу ожиданий на блокировках и взаимным блокировкам.

Видеоурок. Неоптимальные запросы при проведении

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как найти неоптимальный запрос при проведении
- Как оптимизировать этот запрос.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Ожидания на блокировках. Сценарий

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как написать сценарий для проверки излишних ожиданий на блокировках.

Видеоурок. Ожидания на блокировках. Воспроизведение

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как проверить, что ожидания на блокировках действительно есть.

Видеоурок. Ожидания на блокировках. Анализ

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как провести анализ ожиданий на блокировках и выполнить оптимизацию.

Видеоурок. Ожидания на блокировках. Проверка оптимизации

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как проверить, что ожидания на блокировках больше не воспроизводятся.

Видеоурок. Взаимоблокировки. Сценарий

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как написать сценарий проверки взаимоблокировок.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Видеоурок. Взаимоблокировки. Воспроизведение

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как убедиться, что взаимоблокировки действительно присутствуют.

Видеоурок. Взаимоблокировки. Анализ и оптимизация

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как оптимизировать и устранить взаимные блокировки.

Видеоурок. Взаимоблокировки. Проверка оптимизации

Данный урок записан в видеоформате. Обратитесь к материалам на стартовой странице курса

Изучив этот урок, Вы узнаете:

- Как убедиться, что взаимоблокировки были устранены.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Занятие 52

Литература для подготовки

При подготовке к экзамену желательно использовать все возможные источники для получения максимально полной и подробной информации. Большая часть требуемой информации есть в курсе, и, тем не менее, будет очень полезно прочитать дополнительный материал, в котором о тех же вещах рассказано другим языком, а также есть дополнительная полезная информация.

Рекомендуется прочитать следующие книги:

- Настольная книга 1С:Эксперта по технологическим вопросам. Автор: Е.В. Филиппов. Обязательно второе издание, в первом много ошибок и неточностей. Во втором ошибки тоже есть, но их гораздо меньше
- 1С:Корпоративный инструментальный пакет 8. Версия 2.0. Руководство по использованию. 3-е издание (в двух частях)
- Microsoft SQL Server 2012 Создание запросов. Учебный курс Microsoft экзамен 70-461. Авторы: Ицик Бен-Ган, Деян Сарка, Рон Талмейдж. Книга довольно большая, поэтому рекомендую прочитать только три главы:
    - Глава 14. Использование инструментов анализа производительности запросов
    - Глава 15. Реализация индексов и статистика
    - Глава 17. Основные сведения о дальнейших аспектах оптимизации
- Любые книги, связанные с работой и администрированием MS SQL Server.

Разделы с диска ИТС:

- Разработка и администрирование – Технологические вопросы крупных внедрений (все статьи раздела обязательны к прочтению)
- Разработка и администрирование – Документация – 1С:Предприятие 8.3 – Клиент-серверный вариант. Руководство администратора. Рекомендуется прочитать следующие разделы:
    - Глава 2. Клиент-серверный вариант работы
    - Глава 5. Администрирование
- Разработка и администрирование – Документация – 1С Предприятие 8.3 – Руководство администратора. Рекомендуется прочитать следующие разделы:
    - Глава 9. Защита от несанкционированного использования: особенности и настройка (особенно раздел по программным лицензиям)
    - Приложение 3. Описание и расположение служебных файлов
    - Приложение 5. Обработка ошибок

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

- Разработка и администрирование – платформа 1С:Предприятие 8 – Администрирование (желательно все разделы)
- Заметки из зазеркалья – это раздел на сайте 1С, где периодически анонсируются новые механизмы платформы. В данном разделе есть интересные статьи, посвященные в том числе и оптимизации.

Обязательно используйте самый последний диск ИТС, т.к. информация может быть обновлена или добавлена новая.

Дополнительные вопросы для самопроверки

Данные вопросы предназначены для проверки вашей готовности к экзамену. Это список наиболее часто задаваемых вопросов. Естественно, вопросы на экзамене постоянно меняются, поэтому здесь приведены только наиболее распространенные из них. Ответы на большую часть вопросов можно найти в материалах курса. При необходимости следует провести опыты в своей тестовой конфигурации.

Попробуйте ответить на все вопросы самостоятельно. Если не знаете ответ на какой-либо вопрос, повторите соответствующий раздел курса.

Ответы на наиболее сложные вопросы и на вопросы, которые не описаны в курсе, приведены на отдельной странице.

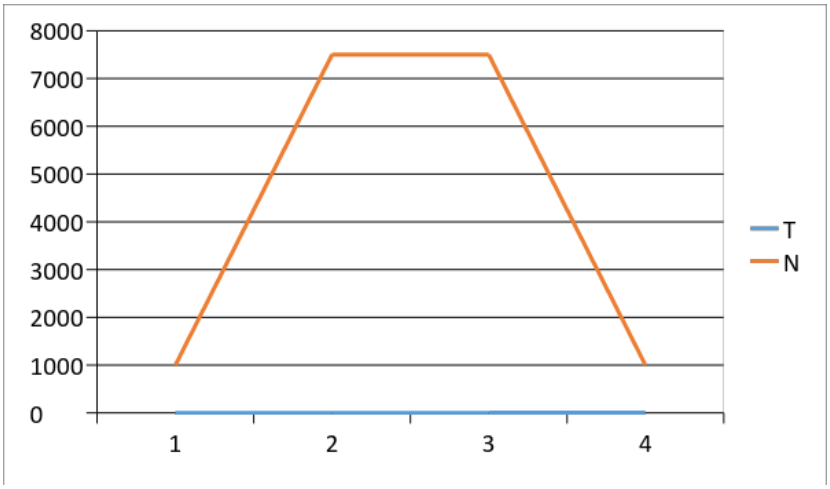
1. Каковы причины появления ошибки «Недостаточно памяти для выполнения запроса»?
2. Документ проводится быстро в однопользовательском режиме и медленно при многопользовательской работе. В чем может быть причина? Предложите план по расследованию этой проблемы
3. Для работы системы используется сетевой ключ на 50 пользователей, который установлен на сервере предприятия. Во время работы системы у различных пользователей периодически возникает ошибка «Не обнаружен ключ защиты программы». Укажите возможные причины возникновения ошибки и предложите способы их устранения
4. Чем СУБД «версионник» отличается от «блокировочника»?
5. Какие кластерные индексы создаются платформой автоматически для документов и справочников?
6. Устанавливает ли код *Запрос.Выполнить()* управляемую блокировку, если используется режим управляемых блокировок?
7. Есть ли отличия в индексах, создаваемых платформой автоматически для регистров, в версиях 8.2 и 8.3?
8. Как проблема медленных запросов пересекается с проблемой ожиданий на блокировках?
9. Мы сформировали таблицу по ключевым операциям согласно APDEX. А что делать дальше?

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

10. Устанавливает ли код *Ссылка.ПолучитьОбъект()* управляемую блокировку, если используется режим управляемых блокировок? Если да, то в какой момент эта блокировка снимается?
11. Устанавливает ли код *НаборЗаписей.Прочитать()* управляемую блокировку, если используется режим управляемых блокировок? Если да, то в какой момент эта блокировка снимается?
12. Операция выполняется медленно. Как узнать, кто «виноват»: сервер 1С или сервер СУБД? Предложите план расследования
13. Эксперт провел нагрузочное тестирование по ключевой операции работающей системы. Для ключевой операции было определено целевое время T = 1 секунда. При этом ключевая операция выполнялась 17 343 раза. В итоге был получен следующий график распределения:



Каков будет APDEX?

14. Каким образом можно замерить время выполнения операции?
15. Может ли ТЖ фиксировать блокировки? Если да, то какие именно: блокировки СУБД или 1С?
16. С чего начать, если пользователи жалуются на производительность?
17. Как определить, что мы избавились от избыточных блокировок?
18. Что такое блокировка, и для чего она нужна?
19. Кто и когда устанавливает блокировку?
20. Кто и когда снимает блокировку?
21. Что такое ожидание на блокировке, и в каком случае оно происходит?
22. Что такое транзакция? Для чего она нужна?
23. Поддерживаются ли вложенные транзакции в 1С?
24. Когда открывается транзакция?
25. Что такое уровень изоляции транзакции?
26. Какие уровни изоляции транзакции поддерживаются 1С:Предприятием 8?
27. Чем уровни изоляции отличаются (с практической точки зрения)?

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

28. Как блокировка связана с транзакцией?
29. Как блокировка связана с запросом?
30. В результате анализа проблемы взаимоблокировки была обнаружена ее причина: запрос в транзакции без опции «ДЛЯ ИЗМЕНЕНИЯ» и затем запись прочитанных данных. Почему в этом случае возникает взаимоблокировка? Как и почему изменится ситуация, если использовать опцию «ДЛЯ ИЗМЕНЕНИЯ»?
31. Что такое взаимоблокировка?
32. Как взаимоблокировка связана с неоптимальными запросами?
33. Каковы основные причины взаимоблокировок?
34. В какой момент начинает действовать свойство набора записей БлокироватьДляИзменения при установленном значении Истина?
35. Нарисуйте на бумаге схемы двух основных типов взаимоблокировок
36. Что такое план запроса, и зачем он нужен?
37. Кто и когда формирует план запроса?
38. Пользователю системы назначили новую роль, после чего у него возникли проблемы производительности. Укажите наиболее вероятную причину, объясните, почему это могло произойти.
39. Как можно повлиять на выбор плана запроса?
40. Как отобрать ТОП 5 медленных запросов при выполнении какой-либо обработки или проведении документа?
41. В каком случае сканирование таблицы – это хорошо?
42. В каком случае использование *Nested Loops* – это хорошо?
43. Какие типичные ошибки в коде конфигурации и структуре метаданных могут привести к неоптимальной работе запроса? Дайте рекомендацию по устранению ошибок.
44. Что такое «Сеанс» и «Соединение»? Чем они отличаются?
45. При проведении документа происходит следующая проверка: если в некотором справочнике есть элемент с определенным названием, то проведение успешно завершается, если элемента с таким именем нет, его нужно создать. Какие проблемы возможны при большом количестве проводимых документов? Как можно реализовать данный функционал и избежать проблем?
46. Что такое закон Амдала? В чем его практический смысл?
47. Нужны ли дополнительные действия при загрузке базы с 8.2 в 8.3 для включения READ COMMITTED SNAPSHOT, например, нужно ли выгрузить базу в dt-файл и затем загрузить из него или достаточно просто обновить платформу?
48. Каким образом при использовании разделителей происходит эскалация блокировок? При использовании только независимых или независимых и совместных? Блокируется вся таблица без учета разделителя или с его учетом в каждом случае?
49. Как влияет конфигурация RAID на методику определения узких мест?
50. Связь между объектным чтением и управляемыми блокировками. После объектного чтения каких объектов они накладываются?
51. *Запрос.Выполнить().Выбрать()* – результат запроса помещается в память сразу или нет? А в случае *Запрос.Выполнить().Выгрузить()*?
52. Как «уронить» сервер 1С 8.2 в тестовых целях?

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

53. Как понять, хватает ли SQL серверу оперативной памяти?
54. В конфигурации определен регистр «Взаиморасчеты», который имеет следующий набор измерений:
    - Договор (тип *Справочник.Договора*)
    - Контрагент (тип *Справочник.Контрагенты*)

    Данные измерения не проиндексированы.

    В транзакции выполняется следующий запрос:

```
ВЫБРАТЬ
      Взаиморасчеты.Контрагент КАК Контрагент,
      Взаиморасчеты.Договор КАК Договор,
      Взаиморасчеты.СуммаОстаток КАК Сумма
      ИЗ
            РегистрНакопления.Взаиморасчеты.Остатки( , Контрагент=&Контрагент) КАК
Взаиморасчеты
```

    Ответьте на следующие вопросы:

- Какие записи в регистре будут заблокированы? Почему именно эти записи?
- Как уменьшить количество заблокированных записей, не меняя текст запроса?
- Какие записи регистра будут заблокированы, если задать условие только по договору, без условия по контрагенту? Почему?
- Какие записи регистра будут заблокированы, если задать условие и по договору, и по контрагенту? Объясните причину блокировки именно этих записей?
- Будут ли отличаться ответы на вышеприведенные вопросы в зависимости от платформы 8.2 и 8.3?
- Что изменится, если выполнять запрос на платформе 8.3 с режимом совместимости 8.2?
- Что изменится, если выполнять запрос вне транзакции?
- Какой уровень изоляции транзакции будет использоваться, если выполнять этот запрос в транзакции в управляемом режиме управления блокировкой данных на платформе 8.2? А если использовать платформу 8.3?
- Можно ли оптимизировать данный запрос с помощью конструкции «ВЫРАЗИТЬ»?
- Изменится ли скорость работы запроса, если получать остаток на середину месяца? Обоснуйте свой ответ.

Ответы на некоторые наиболее сложные вопросы

1. Документ проводится быстро в однопользовательском режиме и медленно при многопользовательской работе. В чем может быть причина? Предложите план по расследованию этой проблемы.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Ответ: Причина, скорее всего, в избыточных блокировках, т.к. проблема имеет место только при многопользовательской работе. Для расследования проблемы необходимо настроить ЦУП и сравнить значения показателей «Суммарное время выполнения запросов» и «Суммарное время ожиданий на блокировках». Если время ожидания на блокировках отнимает большую часть времени запросов (30-50% или более), то, скорее всего, в системе есть избыточные ожидания на блокировках.

2. Есть ли отличия в индексах, создаваемых платформой автоматически для регистров, в версиях 8.2 и 8.3?

   Ответ: Да некоторые отличия есть.

Объект	**Описание индекса**	**8.2**	**8.3**
Регистр сведений, периодический, независимый	Кластерный	Период+Изм1+ …+ИзмN	Изм1+…+ИзмN+ Период
Регистр сведений, периодический, подчиненный регистратору	Кластерный	Период+Изм1+ …+ИзмN	Изм1+…+ИзмN+ Период+Активность

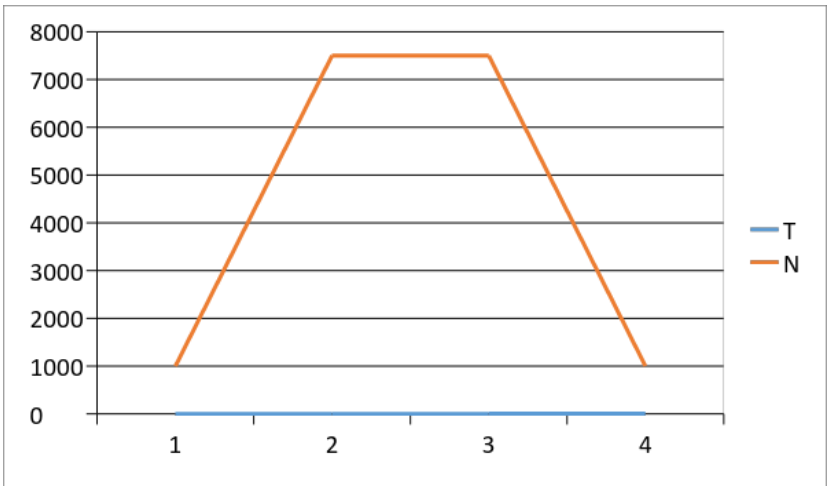
3. Операция выполняется медленно. Как узнать, кто виноват: сервер 1С или сервер СУБД? Предложите план расследования

   Ответ: необходимо сделать трассировку SQL Profiler при выполнении этой операции, выгрузить трассировку в таблицу и запросом получить суммарное время выполнения запросов. Если время выполнения запросов будет занимать большую часть времени выполнения операции, то проблема на стороне СУБД, иначе – на стороне 1С (или сетевого интерфейса).

4. Эксперт провел нагрузочное тестирование по ключевой операции работающей системы. Для ключевой операции было определено целевое время T = 1 секунда. При этом ключевая операция выполнялась 17 343 раза. В итоге он получил следующий график распределения:

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»



Каков будет APDEX?

Ответ: подставляем данные из задачи в формулу расчета APDEX = (NS + NT / 2) / N
N – общее число выполнений данной операции. N = 17 343
NS – число выполнений со временем от 0 до T. NS = 0 (судя по графику)
NT – число выполнений со временем от T до 4T. NT = 17 343 (судя по графику)
APDEX = (0 + 17343 / 2) / 17 343 = 0,5
APDEX равен 0,5, что соответствует оценке «Очень плохо»

5. Может ли ТЖ фиксировать блокировки? Если да, то какие именно: блокировки СУБД или 1С?

   Ответ: ТЖ может фиксировать только блокировки 1С, для этого нужно включить фиксацию события *TLOCK*. Событие пишется, после того как завершилась установка управляемой блокировки, и неважно, явная эта блокировка или нет. Событие записывается именно после установки блокировки, а не после того, как она будет снята.

6. С чего начать, если пользователи жалуются на производительность?

   Ответ: начинаем с определения списка ключевых операций, определения целевого времени и т.д. Действуем согласно методике APDEX, т.е. сначала собираем информацию о текущем состоянии системы и только потом производим анализ и оптимизацию.

7. Как определить, что мы избавились от избыточных блокировок?

   Ответ: Настраиваем ЦУП и смотрим соотношение показателей «Суммарное время выполнения запросов» и «Суммарное время ожиданий на блокировках». Если время ожиданий значительно меньше времени выполнения запросов или ожиданий нет вообще, то можно сказать, что от этой проблемы мы избавились.

8. Пользователю системы назначили новую роль, после чего у него возникли проблемы производительности. Укажите наиболее вероятную причину, объясните, почему это могло произойти.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/
Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Ответ: Причина, скорее всего, в RLS. Необходимо проанализировать роли пользователя, возможно, стоит объединить несколько разных ролей в одну и назначить ему. Также если проблемы производительности возникают только на определенных операциях, то, возможно, стоит перенести эти операции в привилегированный модуль.

9. Как отобрать ТОП 5 медленных запросов при выполнении какой-либо обработки или проведении документа?

   Ответ: собрать трассировку с помощью SQL Profiler, выгрузить ее в таблицу и выполнить запрос SELECT TOP 5 с сортировкой по полю *Duration*

10. Что такое «Сеанс» и «Соединение»? Чем они отличаются?

    Ответ: сеанс определяет текущего пользователя, причем пользователь – это необязательно именно клиентское соединение, это также может быть фоновое задание или внешнее соединение через COM-объект и т.д.
    Соединение – это средство доступа сеанса к кластеру серверов. Другими словами, сеанс получает доступ к кластеру с помощью соединения. Количество соединений ограничено, и как только соединение больше не нужно сеансу, оно возвращается в пул соединений. Если сеанс не обращается к кластеру (пользователь бездействует), то ему не назначается соединение, т.е. сеанс может существовать и без соединения.

11. При проведении документа происходит следующая проверка: если в некотором справочнике есть элемент с определенным названием, то проведение успешно завершается, если элемента с таким именем нет, его нужно создать. Какие проблемы возможны при большом количестве проводимых документов? Как можно реализовать данный функционал и избежать проблем?

    Ответ: при одновременном проведении двух документов может получиться так, что оба документа создадут элемент справочника с одинаковым именем, если до этого элемента с таким именем там не было. Решить данную проблему для разных версий платформы можно по-разному, также решение зависит от режима управления блокировкой данных.

    Опишем возможные решения описанной выше ситуации.

Режим управления блокировкой данных	**8.2**	**8.3**
Автоматический	Необходимо выполнять код внутри конструкций Попытка Исключение. Сначала записываем элемент, а после записи выполняем запрос на поиск элемента с таким же именем, но с условием, что его код меньше, чем код того элемента, который сейчас записывается. Если запрос вернет хотя бы одну строку, тогда необходимо отменить запись элемента, потому что элемент с таким именем ранее уже был записан.	

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Управляемый	В данном случае, если нет элемента с нужным наименованием, то придется блокировать все элементы этого справочника исключительной управляемой блокировкой и только после этого создавать новый элемент. Подумайте, почему здесь не подойдет никакой другой способ.	Добавляем поле «Наименование» в поля блокировки данных и далее накладываем явную управляемую исключительную блокировку по полю «Наименование».

12. Что такое закон Амдала? В чем его практический смысл?

    Ответ: закон Амдала гласит, что если мы будем выполнять программу на компьютере, где установлено в 2 раза больше процессоров, то ускорение хоть и будет, но оно будет менее, чем в 2 раза. В любой программе есть последовательная часть, которую невозможно выполнить параллельно. Более подробное описание закона Амдала есть в дополнительных материалах курса.

13. Нужны ли дополнительные действия при загрузке базы с 8.2 в 8.3 для включения READ COMMITTED SNAPSHOT, например, нужно ли выгрузить базу в .dt файл и затем загрузить из него или достаточно просто обновить платформу?

    Ответ: Если в 8.3 будет включен режим совместимости с 8.2, то уровень изоляции READ COMMITTED SNAPSHOT использоваться не будет. Для того чтобы этот уровень изоляции использовался, необходимо отключить режим совместимости с 8.2, а также использовать режим управления блокировкой данных «Управляемый» и MS SQL Server 2005 или выше.

14. При использовании разделителей каким образом происходит эскалация блокировок? При использовании независимых, независимых и совместных? Блокируется вся таблица без учета разделителя или с его учетом в каждом случае?

    Ответ: эскалация происходит в пределах разделителя, но это касается только управляемых блокировок. Если эскалация произойдет на сервере СУБД, то она будет на всю таблицу.

15. Как влияет конфигурация RAID на методику определения узких мест?

    Ответ: нужно посмотреть, сколько дисков в данном массиве работает параллельно для чтения\записи данных. Допускается очередь к дискам, равная 2, на каждый диск, работающий параллельно.

16. *Запрос.Выполнить().Выбрать()* – сразу в память помещается все или нет? А в случае *Запрос.Выполнить().Выгрузить()*?

    Ответ: *Выгрузить()* – сразу в память, *Выбрать()* – данные выгружаются в память рабочего процесса порционно.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

17. Как «уронить» сервер 1С 8.2 в тестовых целях?

    Ответ: выполнить бесконечную рекурсию на сервере.

18. Как понять, хватает ли SQL серверу оперативной памяти?

    Ответ: необходимо включить в Performance Monitor счетчик *SQLServer\Buffer manager\Buffer cache hit ratio*. Данный счетчик показывает процент найденных в буферном пуле (в оперативной памяти) страниц, что исключает необходимость чтения данных с диска. Чем ближе показатель к 100 %, тем лучше. Если показатель ниже 97 % на протяжении длительного времени, то это может говорить о том, что серверу СУБД не хватает памяти, т.к. он слишком часто обращается к диску для чтения данных.

19. В конфигурации определен регистр «Взаиморасчеты», который имеет следующий набор измерений:

    - Договор (тип *Справочник.Договора*)
    - Контрагент (тип *Справочник.Контрагенты*).

    Данные измерения не проиндексированы.

    В транзакции выполняется следующий запрос:

```
ВЫБРАТЬ
      Взаиморасчеты.Контрагент КАК Контрагент,
      Взаиморасчеты.Договор КАК Договор,
      Взаиморасчеты.СуммаОстаток КАК Сумма
      ИЗ
            РегистрНакопления.Взаиморасчеты.Остатки( , Контрагент=&Контрагент) КАК
Взаиморасчеты
```

Вопросы:

- Какие записи в регистре будут заблокированы? Почему именно эти записи?
  Ответ: будут заблокированы все записи в таблице итогов, т.к. для таблицы итогов регистра накопления создается составной индекс по набору измерений, в данном случае Договор+Контрагент. Условия по договору у нас нет, следовательно пропущено условие в индексе и он не может использоваться, значит, единственный способ поиска – это сканирование.

- Как уменьшить количество заблокированных записей, не меняя текст запроса?
  Ответ: необходимо проиндексировать поле «Контрагент», в этом случае будет использоваться индекс по этому полю и будут заблокированы записи только с указанным контрагентом.

- Какие записи регистра будут заблокированы, если задать условие только по договору, без условия по контрагенту? Почему?

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Ответ: будут заблокированы все записи по данному договору независимо от контрагента. Поле «Договор» находится на первом месте в индексе, следовательно оно может использоваться для поиска и индекс будет задействован.

- Какие записи регистра будут заблокированы, если задать условие и по договору, и по контрагенту? Объясните причину блокировки именно этих записей?
  Ответ: в этом случае будут заблокированы только записи с указанным договором по выбранному контрагенту. Индекс будет использоваться, т.к. условие полностью подходит под составной индекс Договор+Контрагент. Данный запрос будет выполнятся наиболее оптимально.

- Будут ли отличаться ответы на вышеприведенные вопросы в зависимости от платформы 8.2 и 8.3?
  Ответ: нет, все ответы, описанные выше, справедливы как для 8.2, так и для 8.3.

- Что изменится, если выполнять запрос на платформе 8.3 с режимом совместимости 8.2?
  Ответ: ничего не изменится, даже уровень изоляции транзакции останется таким же, как в 8.2, т.к. используется режим совместимости.

- Что изменится, если выполнять запрос вне транзакции?
  Ответ: запрос не будет накладывать блокировок, в остальном ситуация не изменится.

- Какой уровень изоляции транзакции будет использоваться, если выполнять этот запрос в транзакции в управляемом режиме управления блокировкой данных на платформе 8.2? А если использовать платформу 8.3?
  Ответ: для 8.2 будет использоваться режим READ COMMITTED, для 8.3 – READ COMMITTED SNAPSHOT.

- Можно ли оптимизировать данный запрос с помощью конструкции «ВЫРАЗИТЬ»?
  Ответ: нет, т.к. поля запроса содержат по одному типу данных. Использование «ВЫРАЗИТЬ» в данной ситуации бессмысленно.

- Изменится ли скорость работы запроса, если получать остаток на середину месяца? Обоснуйте свой ответ.
  Ответ: если поставить условие по дате на середину месяца, то, для того чтобы получить остаток, необходимо будет обращаться не только к таблице остатков, но и к таблице движений выбранного регистра, что несколько замедлит запрос. Если дату в запросе не указывать или указывать начало первого дня месяца, то данные будут выбраны только из таблицы итогов: такой подход обеспечит максимальное быстродействие.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Подготовка к аттестации 1С:Эксперт. Итоги

Аттестация на Эксперта по праву считается одной из самых сложных, в первую очередь потому, что здесь нужно больше знаний по механизмам СУБД и по администрированию, чем по программированию на 1С. В отличие от многих других экзаменов, здесь правильный ответ на вопрос не имеет решающего значения, здесь важно доказать, почему этот ответ правильный. Нужно быть готовым к тому, что на каждый ваш вопрос, вы услышите пять вопросов «почему» и десять «зачем». Допрос будет очень подробным.

Даже если в первый день экзамена вы себя не проявили, это абсолютно ничего не значит, скорее, это норма. Основные баллы набираются по ходу тренинга. Можете заранее заготовить интересные и оригинальные вопросы, только не нужно спрашивать банальные вещи, которые и так всем известны.

Для успешной подготовки к экзамену вам обязательно нужно выполнить все задания курса, особенно необязательные. Также необходимо повторить все действия из курса на тестовой базе, провести свои эксперименты и прочитать соответствующую литературу.

Очень желательно перед экзаменом применить свои знания на практике, например, на своей рабочей системе. Тогда вы сможете поделиться на экзамене своим опытом с коллегами.

На аттестацию всегда приезжает много хороших специалистов и просто интересных людей со всей России и не только, так что не упускайте возможности пообщаться с ними, обменятся опытом и завести полезные знакомства.

Следует помнить, что 1С:Эксперт – это прежде всего человек, обладающий определенными знаниями, а не соответствующей бумажкой. Например, известны случаи, когда человека снимали с проекта из-за некомпетентности, хотя сертификат у него был, или когда человек оптимизировал систему на 1 000 одновременно работающих пользователей, но сертификат ему так и не дали. Конечно, сертификат вещь важная и полезная, но знания и практический опыт во много раз важнее.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Полезная информация по оптимизации

Занятие 53

Скрипты и динамические представления

В данном разделе представлены различные запросы, скрипты и динамические представления MS SQL Server, которые могут быть полезны при расследовании различных ситуаций.

Эта информация похожа на «швейцарский нож»: есть всего понемногу, при этом никогда не знаешь, когда это потребуется. Поэтому здесь не дается конкретных рекомендаций, когда использовать эти инструменты: ситуаций может быть бесконечное множество, и в какой момент к ним обратиться, вы должны определить самостоятельно.

MS SQL Server сохраняет различную статистику о работе системы с момента последнего перезапуска службы сервера СУБД. Чем дольше работает сервер, тем точнее будет эта статистика. Все запросы к динамическим представлениям можно выполнять непосредственно в Management Studio.

Следует помнить, что статистика, которая накоплена с помощью нижеперечисленных системных представлений, собирается только с момента последнего запуска службы MS SQL Server. Чем дольше работает MS SQL Server без перезапуска, тем точнее будет данная статистика.

Данная информация может и вовсе не понадобиться для расследования проблем производительности (часто так и происходит), однако иногда она может быть полезной.

Статистика ожиданий MS SQL Server

Сервер СУБД фиксирует информацию о том, что кто-то кого-то ждет, и сохраняет ее. Система обычно фиксирует огромное количество ожиданий, и все они означают ожидание доступа к различным ресурсам. Для примера, ожидание PAGEIOLATCH_EX означает, что поток ожидает чтения страницы данных с диска в буферный пул. Ожидание LCK_M_X означает, что поток ожидает возможности наложить эксклюзивную блокировку на какие-то данные. Есть много различных видов ожиданий, но не все они означают проблему.

Можно получить накопленную статистику ожидания, используя динамическое представление sys.dm_os_wait_stats.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

С помощью нижеописанного скрипта можно выяснить, какие типы ожиданий встречаются наиболее часто. В данном скрипте уже исключены те ожидания, которые точно не являются проблемными.

```
WITH [Waits] AS
    (SELECT
        [wait_type],
        [wait_time_ms] / 1000.0 AS [WaitS],
        ([wait_time_ms] - [signal_wait_time_ms]) / 1000.0 AS [ResourceS],
        [signal_wait_time_ms] / 1000.0 AS [SignalS],
        [waiting_tasks_count] AS [WaitCount],
        100.0 * [wait_time_ms] / SUM ([wait_time_ms]) OVER() AS [Percentage],
        ROW_NUMBER() OVER(ORDER BY [wait_time_ms] DESC) AS [RowNum]
    FROM sys.dm_os_wait_stats
    WHERE [wait_type] NOT IN (
        N'BROKER_EVENTHANDLER',         N'BROKER_RECEIVE_WAITFOR',
        N'BROKER_TASK_STOP',            N'BROKER_TO_FLUSH',
        N'BROKER_TRANSMITTER',          N'CHECKPOINT_QUEUE',
        N'CHKPT',                       N'CLR_AUTO_EVENT',
        N'CLR_MANUAL_EVENT',            N'CLR_SEMAPHORE',
        N'DBMIRROR_DBM_EVENT',          N'DBMIRROR_EVENTS_QUEUE',
        N'DBMIRROR_WORKER_QUEUE',       N'DBMIRRORING_CMD',
        N'DIRTY_PAGE_POLL',             N'DISPATCHER_QUEUE_SEMAPHORE',
        N'EXECSYNC',                    N'FSAGENT',
        N'FT_IFTS_SCHEDULER_IDLE_WAIT', N'FT_IFTSHC_MUTEX',
        N'HADR_CLUSAPI_CALL',           N'HADR_FILESTREAM_IOMGR_IOCOMPLETION',
        N'HADR_LOGCAPTURE_WAIT',        N'HADR_NOTIFICATION_DEQUEUE',
        N'HADR_TIMER_TASK',             N'HADR_WORK_QUEUE',
        N'KSOURCE_WAKEUP',              N'LAZYWRITER_SLEEP',
        N'LOGMGR_QUEUE',                N'ONDEMAND_TASK_QUEUE',
        N'PWAIT_ALL_COMPONENTS_INITIALIZED',
        N'QDS_PERSIST_TASK_MAIN_LOOP_SLEEP',
        N'QDS_CLEANUP_STALE_QUERIES_TASK_MAIN_LOOP_SLEEP',
        N'REQUEST_FOR_DEADLOCK_SEARCH', N'RESOURCE_QUEUE',
        N'SERVER_IDLE_CHECK',           N'SLEEP_BPOOL_FLUSH',
        N'SLEEP_DBSTARTUP',             N'SLEEP_DCOMSTARTUP',
        N'SLEEP_MASTERDBREADY',         N'SLEEP_MASTERMDREADY',
        N'SLEEP_MASTERUPGRADED',        N'SLEEP_MSDBSTARTUP',
        N'SLEEP_SYSTEMTASK',            N'SLEEP_TASK',
        N'SLEEP_TEMPDBSTARTUP',         N'SNI_HTTP_ACCEPT',
        N'SP_SERVER_DIAGNOSTICS_SLEEP', N'SQLTRACE_BUFFER_FLUSH',
        N'SQLTRACE_INCREMENTAL_FLUSH_SLEEP',
        N'SQLTRACE_WAIT_ENTRIES',       N'WAIT_FOR_RESULTS',
        N'WAITFOR',                     N'WAITFOR_TASKSHUTDOWN',
        N'WAIT_XTP_HOST_WAIT',          N'WAIT_XTP_OFFLINE_CKPT_NEW_LOG',
        N'WAIT_XTP_CKPT_CLOSE',         N'XE_DISPATCHER_JOIN',
        N'XE_DISPATCHER_WAIT',          N'XE_TIMER_EVENT')
    )
SELECT
    [W1].[wait_type] AS [WaitType],
    CAST ([W1].[WaitS] AS DECIMAL (16, 2)) AS [Wait_S],
    CAST ([W1].[ResourceS] AS DECIMAL (16, 2)) AS [Resource_S],
    CAST ([W1].[SignalS] AS DECIMAL (16, 2)) AS [Signal_S],
```


Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

```
    [W1].[WaitCount] AS [WaitCount],
    CAST ([W1].[Percentage] AS DECIMAL (5, 2)) AS [Percentage],
    CAST (([W1].[WaitS] / [W1].[WaitCount]) AS DECIMAL (16, 4)) AS [AvgWait_S],
    CAST (([W1].[ResourceS] / [W1].[WaitCount]) AS DECIMAL (16, 4)) AS [AvgRes_S],
    CAST (([W1].[SignalS] / [W1].[WaitCount]) AS DECIMAL (16, 4)) AS [AvgSig_S]
FROM [Waits] AS [W1]
INNER JOIN [Waits] AS [W2]
    ON [W2].[RowNum] <= [W1].[RowNum]
GROUP BY [W1].[RowNum], [W1].[wait_type], [W1].[WaitS],
    [W1].[ResourceS], [W1].[SignalS], [W1].[WaitCount], [W1].[Percentage]
HAVING SUM ([W2].[Percentage]) - [W1].[Percentage] < 95; -- percentage threshold
GO
```

Запросы и базы, создающие нагрузку

С помощью данного прямого запроса можно определить те запросы, которые выполняют максимальное количество физических (с диска) и логических (из памяти) чтений.

```
SELECT
SUM(qs.total_physical_reads) as physical_reads,
SUM(qs.total_logical_reads) as logical_reads
into T1 FROM (
select top 100000
*
from
sys.dm_exec_query_stats qs
where qs.last_execution_time > (CURRENT_TIMESTAMP - '01:00:00.000')
order by qs.total_physical_reads desc
) as qs
;
select top 100
(qs.total_physical_reads) as physical_reads,
(qs.total_logical_reads) as logical_reads,
qp.query_plan,
st.text,
dtb.name,
qs.*,
st.dbid
INTO T2
FROM
sys.dm_exec_query_stats qs
CROSS APPLY sys.dm_exec_query_plan(qs.plan_handle) qp
CROSS APPLY sys.dm_exec_sql_text(qs.sql_handle) st
left outer join sys.databases as dtb on st.dbid = dtb.database_id
where qs.last_execution_time > (CURRENT_TIMESTAMP - '01:00:00.000')
order by qs.total_physical_reads desc;
select
(T2.physical_reads*100/T1.physical_reads) as percent_physical_reads,
(T2.logical_reads*100/T1.logical_reads) as percent_logical_reads,
T2.*
```


Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

```
from
T2 as T2
INNER JOIN T1 as T1
ON 1=1
order by T2.total_physical_reads desc;
drop table T2;
drop table T1;
```

Запросы с высокими издержками на ввод-вывод

```
SELECT TOP 100
[Average IO] = (total_logical_reads + total_logical_writes) / qs.execution_count
,[Total IO] = (total_logical_reads + total_logical_writes)
,[Execution count] = qs.execution_count
,[Individual Query] = SUBSTRING (qt.text,qs.statement_start_offset/2,
(CASE WHEN qs.statement_end_offset = -1
THEN LEN(CONVERT(NVARCHAR(MAX), qt.text)) * 2
ELSE qs.statement_end_offset END - qs.statement_start_offset)/2)
,[Parent Query] = qt.text
,DatabaseName = DB_NAME(qt.dbid)
FROM sys.dm_exec_query_stats qs
CROSS APPLY sys.dm_exec_sql_text(qs.sql_handle) as qt
ORDER BY [Average IO] DESC;
```

Самые часто выполняемые запросы

```
SELECT TOP 100
[Execution count] = execution_count
,[Individual Query] = SUBSTRING (qt.text,qs.statement_start_offset/2,
(CASE WHEN qs.statement_end_offset = -1
THEN LEN(CONVERT(NVARCHAR(MAX), qt.text)) * 2
ELSE qs.statement_end_offset END - qs.statement_start_offset)/2)
,[Parent Query] = qt.text
,DatabaseName = DB_NAME(qt.dbid)
FROM sys.dm_exec_query_stats qs
CROSS APPLY sys.dm_exec_sql_text(qs.sql_handle) as qt
ORDER BY [Execution count] DESC;
```

Определение контекста проблемных запросов

Допустим, что с помощью скрипта или SQL Profiler были пролучены тексты SQL запросов, которые выполняются медленно, либо сильно нагружают базу. Сами по себе эти тексты запросов бесполезны, необходимо найти место в конфигурации, откуда они вызываются, и это можно сделать только с помощью ТЖ.

Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

```
<config xmlns=" v8.1c.ru/v8/tech-log »»">
<log location="C:\Query" history="2">
<event>
<eq property="Name" value="DBMSSQL"/>
<like property="Sql" value="%AccumRg105%"/>
</event>
<property name="all"/>
</log>
</config>
```

Здесь очень важно указать как можно более точные фильтры, для того чтобы не собирать лишней информации.

Базы, нагружающие диск

```
WITH DB_Disk_Reads_Stats
AS
(SELECT DatabaseID, DB_Name(DatabaseID) AS [DatabaseName], SUM(total_physical_reads)
AS [physical_reads]
 FROM sys.dm_exec_query_stats AS qs
 CROSS APPLY (SELECT CONVERT(int, value) AS [DatabaseID]
              FROM sys.dm_exec_plan_attributes(qs.plan_handle)
              WHERE attribute = N'dbid') AS F_DB
 GROUP BY DatabaseID)
SELECT ROW_NUMBER() OVER(ORDER BY [physical_reads] DESC) AS [row_num],
      DatabaseName, [physical_reads],
      CAST([physical_reads] * 1.0 / SUM([physical_reads]) OVER() * 100.0 AS
DECIMAL(5, 2)) AS [Physical_Reads_Percent]
FROM DB_Disk_Reads_Stats
WHERE DatabaseID > 4 -- system databases
AND DatabaseID <> 32767 -- ResourceDB
ORDER BY row_num OPTION (RECOMPILE);
```

Длительные транзакции

```
DECLARE @curr_date as DATETIME
SET @curr_date = GETDATE()
select --SESSION_TRAN.*,
SESSION_TRAN.session_id AS connectID, -- "Соединение с СУБД" в консоли кластера 1С
--TRAN_INFO.*,
TRAN_INFO.transaction_begin_time,
DateDiff(MINUTE, TRAN_INFO.transaction_begin_time, @curr_date) AS Duration, --
Длительность в минутах
TRAN_INFO.transaction_type, -- 1 = транзакция чтения-записи; 2 = транзакция только
для чтения; 3 = системная транзакция; 4 = распределенная транзакция
TRAN_INFO.transaction_state,
```


Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

```
-- 0 = Транзакция еще не была полностью инициализирована
-- 1 = Транзакция была инициализирована, но еще не началась
-- 2 = Транзакция активна
-- 3 = Транзакция закончилась. Используется для транзакций «только для чтения»
-- 4 = Фиксирующий процесс был инициализирован на распределенной транзакции.
Предназначено только для распределенных транзакций. Распределенная транзакция все
еще активна, но дальнейшая обработка не может иметь место
-- 5 = Транзакция находится в готовом состоянии и ожидает разрешения
-- 6 = Транзакция зафиксирована
-- 7 = Производится откат транзакции
-- 8 = откат транзакции был выполнен.
--CONN_INFO.*,
CONN_INFO.connect_time,
CONN_INFO.num_reads,
CONN_INFO.num_writes,
CONN_INFO.last_read,
CONN_INFO.last_write,
CONN_INFO.client_net_address,
CONN_INFO.most_recent_sql_handle,
--SQL_TEXT.*,
SQL_TEXT.dbid,
db_name(SQL_TEXT.dbid) AS IB_NAME,
SQL_TEXT.text,
--QUERIES_INFO.*,
QUERIES_INFO.start_time,
QUERIES_INFO.status,
QUERIES_INFO.command,
QUERIES_INFO.wait_type,
QUERIES_INFO.wait_time,
PLAN_INFO.query_plan

FROM sys.dm_tran_session_transactions AS SESSION_TRAN
JOIN sys.dm_tran_active_transactions AS TRAN_INFO
ON SESSION_TRAN.transaction_id = TRAN_INFO.transaction_id
LEFT JOIN sys.dm_exec_connections AS CONN_INFO
ON SESSION_TRAN.session_id = CONN_INFO.session_id
CROSS APPLY sys.dm_exec_sql_text(CONN_INFO.most_recent_sql_handle) AS SQL_TEXT
LEFT JOIN sys.dm_exec_requests AS QUERIES_INFO
ON SESSION_TRAN.session_id = QUERIES_INFO.session_id
LEFT JOIN (
SELECT VL_SESSION_TRAN.session_id AS session_id,
VL_PLAN_INFO.query_plan AS query_plan
FROM sys.dm_tran_session_transactions AS VL_SESSION_TRAN
INNER JOIN sys.dm_exec_requests AS VL_QUERIES_INFO
ON VL_SESSION_TRAN.session_id = VL_QUERIES_INFO.session_id
CROSS APPLY sys.dm_exec_text_query_plan(VL_QUERIES_INFO.plan_handle,
VL_QUERIES_INFO.statement_start_offset, VL_QUERIES_INFO.statement_end_offset) AS
VL_PLAN_INFO) AS PLAN_INFO
ON SESSION_TRAN.session_id = PLAN_INFO.session_id
ORDER BY transaction_begin_time ASC
```


Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Список длительных транзакций

```
SELECT transaction_id, *
FROM sys.dm_tran_active_snapshot_database_transactions
ORDER BY elapsed_time_seconds DESC;
```

Использование кэша сервера СУБД

```
SELECT TOP(100) [type], SUM(single_pages_kb) AS [SPA Mem, Kb]
FROM sys.dm_os_memory_clerks
GROUP BY [type]
ORDER BY SUM(single_pages_kb) DESC;
```

Использование кэша базами

```
SELECT DB_NAME(database_id) AS DB,COUNT(row_count)*8.00/1024.00 AS MB,
COUNT(row_count)*8.00/1024.00/1024.00 AS GB
FROM sys.dm_os_buffer_descriptors
GROUP BY database_id
ORDER BY MB DESC
```

Определение свободного места в базе Tempdb

```
SELECT SUM(unallocated_extent_page_count) AS [free pages],
(SUM(unallocated_extent_page_count)*1.0/128) AS [free space in MB]
FROM sys.dm_db_file_space_usage;
```

Запросы, нагружающие процессор за последний час

```
SELECT
SUM(qs.max_elapsed_time) as elapsed_time,
SUM(qs.total_worker_time) as worker_time
into T1 FROM (
      select top 100000
      *
      from
      sys.dm_exec_query_stats qs
      where qs.last_execution_time > (CURRENT_TIMESTAMP - '01:00:00.000')
      order by qs.total_worker_time desc
      --order by qs.max_elapsed_time desc
) as qs;
```


Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

```
select top 100
(qs.max_elapsed_time) as elapsed_time,
(qs.total_worker_time) as worker_time,
qp.query_plan,
st.text,
dtb.name,
qs.*,
st.dbid
INTO T2
FROM
sys.dm_exec_query_stats qs
CROSS APPLY sys.dm_exec_query_plan(qs.plan_handle) qp
CROSS APPLY sys.dm_exec_sql_text(qs.sql_handle) st
left outer join sys.databases as dtb on st.dbid = dtb.database_id
where qs.last_execution_time > (CURRENT_TIMESTAMP - '01:00:00.000')
order by qs.total_worker_time desc
--order by qs.max_elapsed_time desc;
select
(T2.elapsed_time*100/T1.elapsed_time) as percent_elapsed_time,
(T2.total_worker_time*100/T1.worker_time) as percent_worker_time,
T2.*
from
T2 as T2
INNER JOIN T1 as T1
ON 1=1
order by T2.total_worker_time desc
--order by T2.max_elapsed_time desc;
drop table T2;
drop table T1;
```

Нагрузка на процессор в разрезе баз

```
WITH DB_CPU_Stats
AS
(SELECT DatabaseID, DB_Name(DatabaseID) AS [DatabaseName], SUM(total_worker_time) AS
[CPU_Time_Ms]
FROM sys.dm_exec_query_stats AS qs
CROSS APPLY (SELECT CONVERT(int, value) AS [DatabaseID]
             FROM sys.dm_exec_plan_attributes(qs.plan_handle)
             WHERE attribute = N'dbid') AS F_DB
GROUP BY DatabaseID)
SELECT ROW_NUMBER() OVER(ORDER BY [CPU_Time_Ms] DESC) AS [row_num],
      DatabaseName, [CPU_Time_Ms],
      CAST([CPU_Time_Ms] * 1.0 / SUM([CPU_Time_Ms]) OVER() * 100.0 AS DECIMAL(5, 2))
AS [CPUPercent]
FROM DB_CPU_Stats
WHERE DatabaseID > 4 -- system databases
AND DatabaseID <> 32767 -- ResourceDB
ORDER BY row_num OPTION (RECOMPILE);
```


Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

Индексы с высокими затратами на содержание

```
SELECT TOP 10
[Average CPU used] = total_worker_time / qs.execution_count
,[Total CPU used] = total_worker_time
,[Execution count] = qs.execution_count
,[Individual Query] = SUBSTRING (qt.text,qs.statement_start_offset/2,
(CASE WHEN qs.statement_end_offset = -1
THEN LEN(CONVERT(NVARCHAR(MAX), qt.text)) * 2
ELSE qs.statement_end_offset END -
qs.statement_start_offset)/2)
,[Parent Query] = qt.text
,DatabaseName = DB_NAME(qt.dbid)
FROM sys.dm_exec_query_stats qs
CROSS APPLY sys.dm_exec_sql_text(qs.sql_handle) as qt
ORDER BY [Average CPU used] DESC;

Индексы с высокими
-- Create required table structure only.
-- Note: this SQL must be the same as in the Database loop given in the following
step.
SELECT TOP 1
      [Maintenance cost]  = (user_updates + system_updates)
      ,[Retrieval usage] = (user_seeks + user_scans + user_lookups)
      ,DatabaseName = DB_NAME()
      ,TableName = OBJECT_NAME(s.[object_id])
      ,IndexName = i.name
INTO #TempMaintenanceCost
FROM   sys.dm_db_index_usage_stats s
INNER JOIN sys.indexes i ON  s.[object_id] = i.[object_id]
   AND s.index_id = i.index_id
WHERE s.database_id = DB_ID()
   AND OBJECTPROPERTY(s.[object_id], 'IsMsShipped') = 0
   AND (user_updates + system_updates) > 0 -- Only report on active rows.
   AND s.[object_id] = -999  -- Dummy value to get table structure.
;

-- Loop around all the databases on the server.
EXEC sp_MSForEachDB     'USE [?];
-- Table already exists.
INSERT INTO #TempMaintenanceCost
SELECT TOP 10
       [Maintenance cost]  = (user_updates + system_updates)
       ,[Retrieval usage] = (user_seeks + user_scans + user_lookups)
      ,DatabaseName = DB_NAME()
       ,TableName = OBJECT_NAME(s.[object_id])
       ,IndexName = i.name
FROM   sys.dm_db_index_usage_stats s
INNER JOIN sys.indexes i ON  s.[object_id] = i.[object_id]
   AND s.index_id = i.index_id
WHERE s.database_id = DB_ID()
   AND i.name IS NOT NULL     -- Ignore HEAP indexes.
   AND OBJECTPROPERTY(s.[object_id], ''IsMsShipped'') = 0
```


Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»

```
   AND (user_updates + system_updates) > 0 -- Only report on active rows.
ORDER BY [Maintenance cost]  DESC
;
'
-- Select records.
SELECT TOP 10 * FROM #TempMaintenanceCost
ORDER BY [Maintenance cost]  DESC
-- Tidy up.
DROP TABLE #TempMaintenanceCost
```


Бесплатные материалы проекта:
http://курсы-по-1с.рф/free/

Электронная почта: support@Kursy-po-1C.ru
Телефоны: **8-800 1000-613**, +7(495) 7777-096

Copyright © 2012-2015, Насипов Фарит и Евгений Гилев, ООО «ОдинКурс»