

Наиболее интересные возможности SQL Server 2005 с точки зрения администраторов баз данных

Брайан Найт (Brian Knight)

Вероятно, вы уже довольно много читали об усовершенствованиях в SQL Server 2005, касающихся программируемости и создания интеллектуальных ресурсов предприятия, но что это дает администратору баз данных?

В этой статье я продемонстрирую вам некоторые из любимых мною функциональных возможностей SQL Server 2005.

Можно представить, как уже сейчас преподаватели курсов облизаваются, предвкушая обучение все новых групп пользователей SQL Server, поскольку здесь есть чему поучиться. В SQL Server 2005 существует множество новых возможностей.

Новая среда управления

Можете забыть все, что вы знали об инструментах управления в SQL Server 2000, за исключением, пожалуй, Profiler (в который в SQL Server 2005 также были внесены некоторые усовершенствования). Одним из са-

мых незначительных дополнений инструментария является средство управления SQL Configuration Manager (рис. 1). Оно заменяет многое из того, что ранее выполнял апплет обслуживания Services на контрольной панели Control Panel, и предоставляет еще кое-какие дополнительные возможности. Например, вы можете запустить и остановить службу и изменить пароль, применяемый для их запуска, — очень похоже на то, как это делал апплет Services. Вы сможете также изменить порт, который прослушивает SQL Server, или создать псевдонимы, как это делали программы клиентской и серверной конфигурации сети в SQL Server 2000. По сути, он комбинирует все старые службы и приложения управления сетью в одном центральном интерфейсе MMC.

Теперь центральным инструментом управления является Microsoft SQL Server Management Studio (рис. 2). Этот инструмент лучше, чем Enterprise Manager. Вместо применения консоли MMC, которая используется в Enterprise Manager, специалисты Microsoft решили пойти новым путем и задействовали облегченную версию Visual Studio 2005. Окружение строится для управления сотнями баз предприятия на

Рис. 1. SQL Configuration Manager

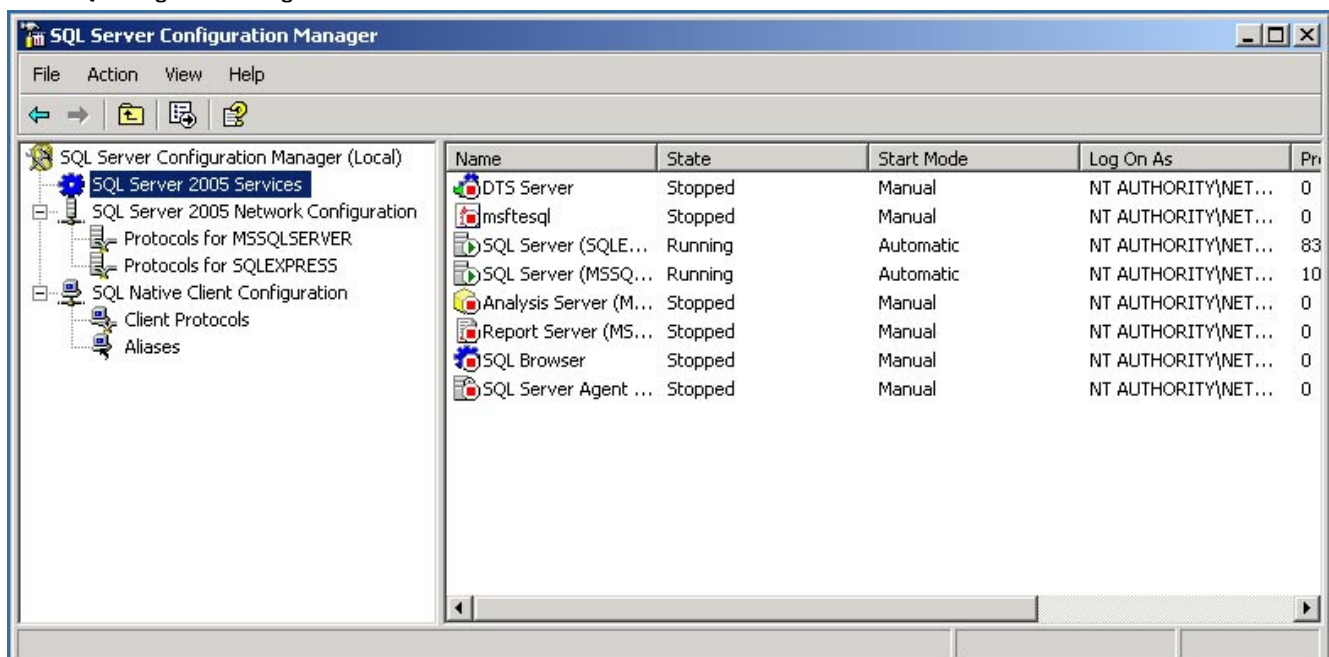
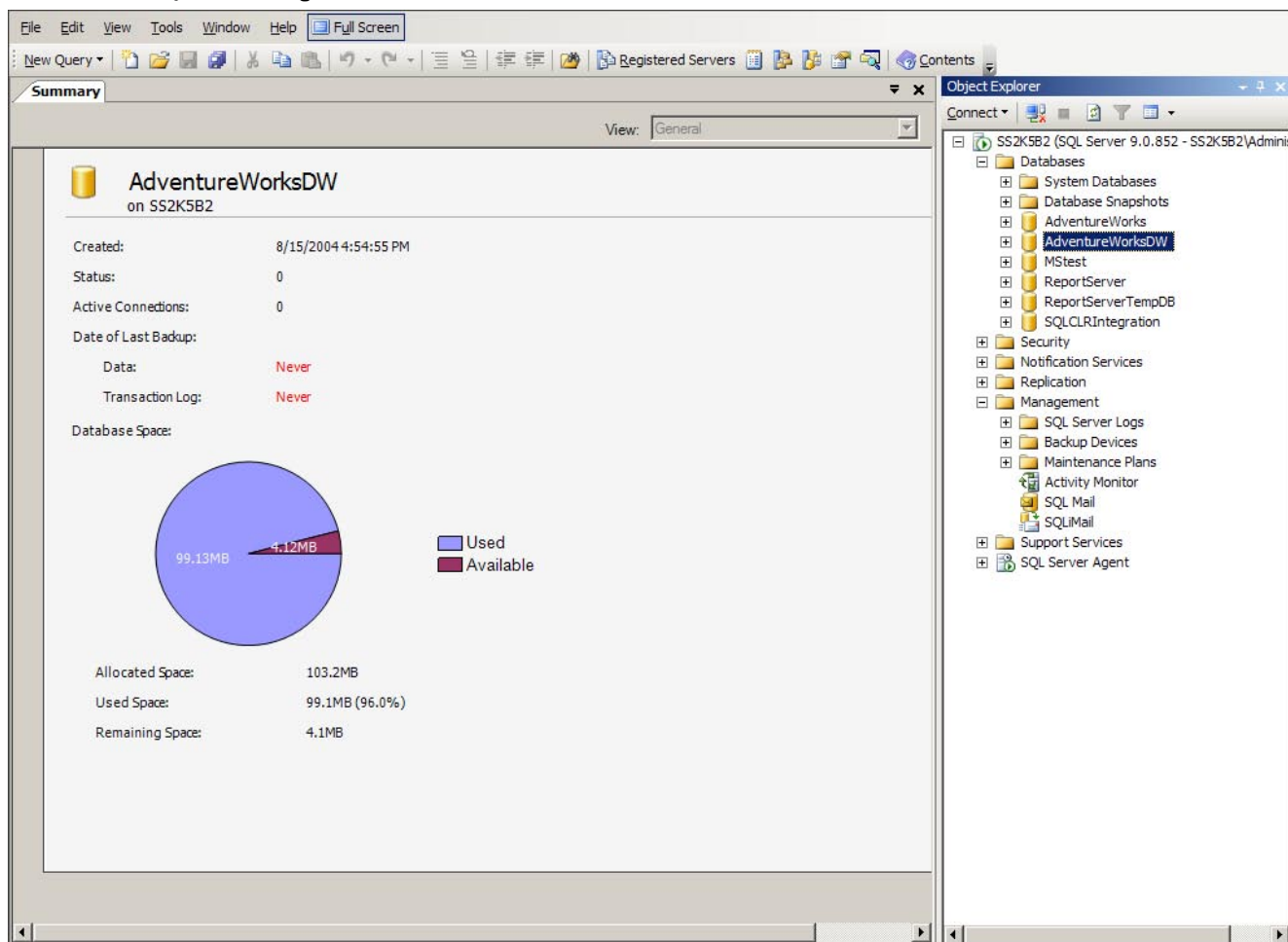


Рис. 2. Microsoft SQL Server Management Studio



одном сервере. К примеру, когда вы раскрываете дерево баз данных Databases, инструментальные средства начинают опрос с целью формирования списка баз данных, и, пока этот процесс продолжается, вы можете переключиться на другую задачу. Обычно в SQL Server 2000 вам приходилось ждать окончания формирования списка баз данных или таблиц. В больших вычислительных средах это может занять 5-10 минут.

Все оперативно

Одним из важнейших качеств, востребованных в активном окружении баз данных, является поддержание оперативности. Поскольку версии SQL Server 2000 еще предстояло пройти большой путь, вы не могли выполнять в оперативном режиме множество действий. В SQL Server 2005 внесены потрясающие исправления: теперь можно восстанавливать базы данных, даже когда пользователи все еще подсоединены к базам, выполняя свои запросы. Такая возможность доступна для восстановления файлов и групп файлов в изданиях

SQL Server масштаба Enterprise, Developer и Evaluation. При этом недоступны только те данные, которые восстанавливаются в данный момент. Остальная часть базы остается по-прежнему доступной. Такие же возможности предусмотрены для операций повторного построения индексов и команд DBCC. Вы сможете построить заново кластеризованный индекс во время внесения активных изменений в данные.

Секционирование данных

Общей проблемой администраторов баз данных была обработка больших таблиц. Большие таблицы вызывают трудности при внесении в них значительных объемов данных и удалении порций информации. Рассмотрим случай из жизни, с которым большинству из нас пришлось столкнуться в среде, сопутствующей базе данных AdventureWorks.

Допустим, что вы администратор баз данных и в вашем распоряжении имеется большая таблица транзакций TransactionHistory. Раз в месяц вы переписываете из нее транзакции старше 2 лет в архивную

таблицу TransactionHistoryArchive. Поскольку в таблице TransactionHistory более 40 миллионов записей, их переписывание занимает часы, даже несмотря на наличие кластеризованного индекса по столбцу с датой выполнения транзакции TransactionDate. Такой сценарий является прекрасным случаем для секционирования данных.

Нам придется провести секционирование обеих таблиц, TransactionHistory и TransactionHistoryArchive по столбцу TransactionDate по месяцам. Прежде чем перемещать данные посредством секционирования, необходимо проверить два момента: схемы обеих таблиц должны соответствовать друг другу, и в таблице-адресате должна существовать пустая секция, которая сможет принять данные. Как только вы будете готовы, запускайте ряд команд ALTER TABLE для миграции данных между этими таблицами. Этот процесс протекает мгновенно, потому что не происходит физического перемещения страниц или строк данных. Вместо этого вы просто указываете в метаданных записей этого диапазона времени, что они принадлежат другой таблице.

Моментальные снимки и зеркальные копии баз данных

Создание зеркальных баз данных — одна из самых предсказуемых возможностей SQL Server 2005 с высоким коэффициентом готовности. Это во многом похоже на отправку журнала, когда журнал транзакций восстанавливается на вторичном сайте автоматически. В создании зеркальных баз данных вторичный сайт (называемый зеркалом) получает резервную копию журнала транзакций с первичного сайта (называемого главой). Имеется также третий сервер, называемый свидетелем, который отслеживает доступность каждого сервера в этом взаимодействии. По желанию можно объединить свидетель с первичным или вторичным сервером, но делать это не рекомендуется, так как создается единая точка отказа.

Имеется ряд ключевых различий между зеркальным копированием баз данных и пересылкой журнала. Если при создании зеркальной копии первичный сайт по каким-либо причинам откажет, то зеркальный сайт немедленно (в течение одной-двух секунд) подхватит владение данными и выйдет из нескончаемого режима восстановления. Благодаря новой ревизии MDAC клиенты будут автоматически перенаправлены на зеркальный сайт. Традиционная отгрузка журнала предполагает, что эти функции будут встроены в приложение. Обработка отказа при этом производится вручную.

Существует несколько искусственных приемов преодоления с использованием мгновенных снимков. Большое достоинство зеркальных копий баз данных в том, что взаимодействовать могут географически удаленные сайты. К примеру, мой домашний офис в Джексонвилле может переслать данные на наш чикагский резервный сайт для чрезвычайных ситуаций, и все приложения автоматически перенесутся следом, как только я демонтирую базу данных для ее технического обслуживания. Недостатком и зеркальных баз данных, и отгрузки журналов является то, что требования к пространству возрастают вдвое, поскольку зеркальная копия базы данных переносится на другой сервер.

Моментальные снимки

Моментальный снимок базы данных, как следует из названия, является снимком состояния базы на заданный момент времени. Его можно только читать, и он статичен. Как только моментальный снимок будет сделан, начинают отслеживаться все изменения страниц данных, которые произошли в базе с этого момента. По существу, журнал изменений содержит страницу данных, которая была изменена, а первоначальный вид страницы хранится в моментальном снимке. Сначала моментальный снимок очень мал, он содержит мало данных. Но со временем это может вызвать проблемы с пространством на сервере по мере того, как все больше и больше данных обновляется в базе.

Для создания моментального снимка с помощью T-SQL можно воспользоваться синтаксисом CREATE DATABASE:

```
CREATE DATABASE AdventureWorks_DecReport ON
( NAME = AdventureWorks_Data, FILENAME =
'C:\Program Files\Microsoft SQL
Server\MSSQL.1\MSSQL\Data\
AdventureWorks_DecReport.ss' )
AS SNAPSHOT OF AdventureWorks;
GO
```

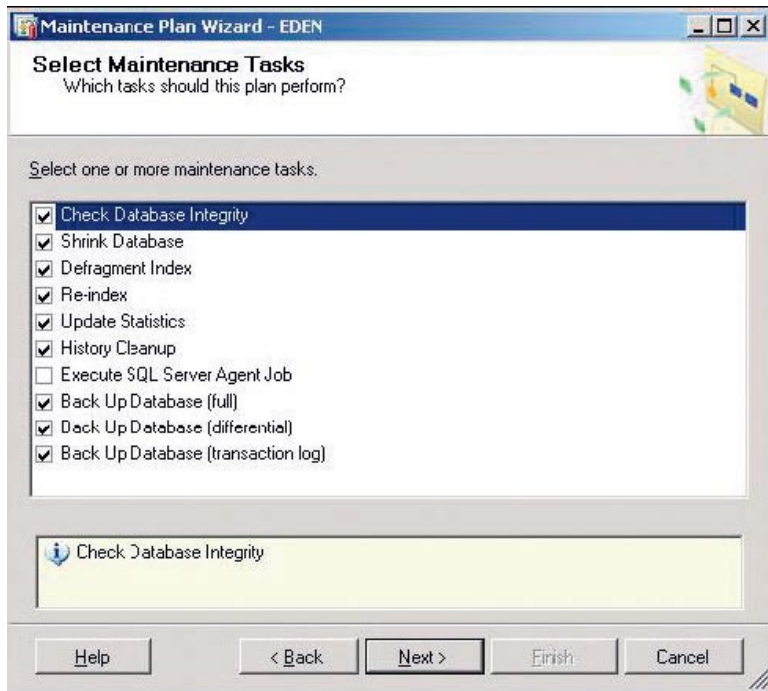
К моментальному снимку можно подсоединиться и обращаться с запросами, как к физической базе данных, за исключением того, что нельзя применять операторы UPDATE, INSERT и DELETE. Любой пользователь с полномочием RESTORE может воспользоваться моментальным снимком как источником данных для восстановления:

```
RESTORE DATABASE database FROM DATABASE_SNAPSHOT
= AdventureWorks_DecReport
```

Чтобы удалить моментальный снимок, надо просто выполнить стандартную команду DROP DATABASE и указать его имя:

```
DROP DATABASE AdventureWorks_DecReport
```

Рис. 3. Конфигурация плана сопровождения

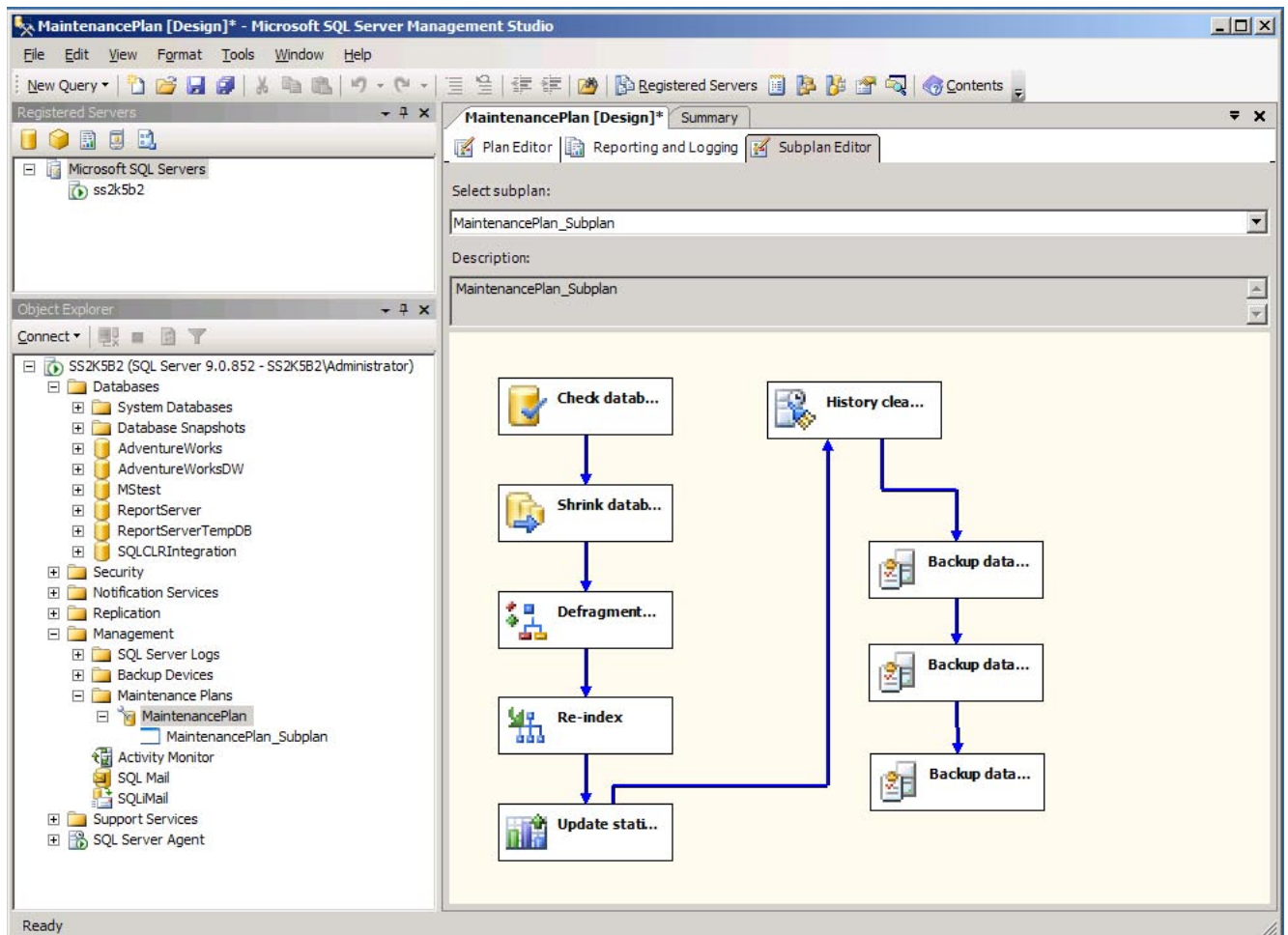


Планы сопровождения

Планы сопровождения в версиях SQL Server 7.0 и 2000 были блаженством для тех администраторов баз данных, кто хотел, чтобы базовое сопровождение сервера выполнялось на регулярной основе. Однако им не хватало функциональных возможностей, чтобы стать инструментом моей мечты. Например, нельзя было настроить план сопровождения, который отвечал бы вашим собственным стандартам. Планы сопровождения в SQL Server 2005 ушли далеко вперед. При запуске плана вас спрашивают, какие задачи вам хотелось бы осуществлять в плане сопровождения (рис. 3).

После того как вы ответите на ряд вопросов, план сопровождения будет создан в виде пакета DTS (рис. 4). Преимущество создания плана сопровождения в виде пакета DTS в том, что он допускает настройку. К примеру, если мне необходимо получить

Рис. 4. План сопровождения в виде пакета DTS



извещения о неудачном завершении резервного копирования, но меня не слишком беспокоит отсутствие статистики обновления, я могу настроить пакет DTS на работу именно таким образом. Можно также заставить пакет по завершении работы инициировать другое задание SQL Server или исполнить код T-SQL, а также переупорядочить последовательность выполняемых действий по сопровождению.

Усовершенствование безопасности

Эта широкая категория представляет большую работу, которая была проделана в SQL Server 2005 для администраторов. Сейчас мир стал несколько иным, чем пять лет назад. Мы должны намертво закрывать дверные замки баз данных, чтобы дать отпор хакерам и соответствовать новым правилам вроде тех, что требует Sarbanes-Oxley.

Важным изменением, которое поможет SQL Server соответствовать новым правительственным постановлениям, является предоставление администратору возможности заставить пользователя менять пароль с периодичностью, обусловленной серверной политикой. Можно заблокировать вход в SQL Server для пользователя, который не меняет свой пароль через заданное количество дней.

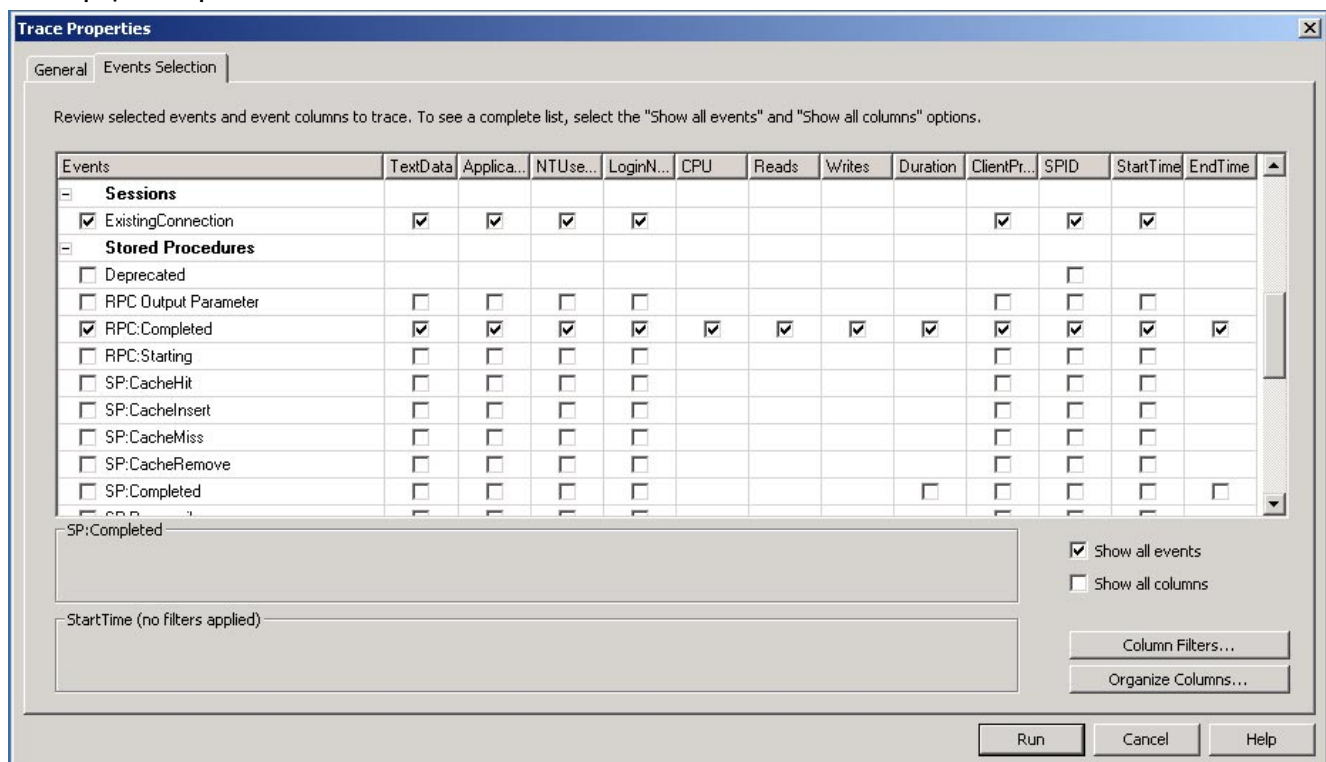
Постоянное ноющее чувство неудовлетворенности у администратора вызвано тем, как SQL Server 2000 привлекает пользователей к схемам. В SQL Server 2000, если пользователь bknight создает объект History, таблица может быть названа bknight.History. Но в этом случае нельзя удалить пользователя bknight до тех пор, пока не будет изменен владелец объекта. Однако тем самым вы переназначаете полномочия на объект, что и вызывает головную боль у администратора.

В SQL Server 2005 схема может не быть привязана к пользователю. Если вы создаете схему, например Accounting, а затем таблицу, принадлежащую этой схеме, то таблица может быть названа Accounting.History. Плюс в том, что схема может принадлежать роли базы данных, так что вы не привязаны к заданному пользователю. Наконец, теперь вы узнаете, что системные объекты являются членами схемы sys schema.

SQL Profiler и консультант по настройке Database Tuning Advisor

Мастер настройки индексов Index Tuning Wizard получил новое имя и новые функциональные возможности. Теперь он называется консультантом по настройке баз данных Database Tuning Advisor (DTA) и способен

Рис. 5. Процесс выбора события в новом Profiler



формулировать рекомендации по секционированию таблиц и индексов. Он также обладает легко перестраиваемой конфигурацией, что дает администратору больше возможностей управления. Выполнение рекомендаций не будет пустой тратой времени. Когда вы первый раз откроете Profiler, вам может показаться, что ничего не изменилось. Но скоро заметите улучшение. При создании новой трассы вам предложат матрицу из событий и столбцов, которые вы будете выбирать. SQL Server 2000 позволял вам выбрать столбец, не соответствовавший событию. К примеру, событию Audit Logout не соответствует столбец TextData. В SQL Server 2005 вам не удастся сделать такую ошибку (**рис. 5**).

Теперь в Profiler мне больше всего нравится возможность привязывать его к результатам системного мониторинга. Другими словами, вы можете выбрать строку в Profiler и увидеть, что этот запрос сделал с процессором и оперативной памятью сервера, — причем на том же самом экране. Теперь вы можете отслеживать события в Analysis Services и выдавать результаты трассировки в файл XML, что улучшает их переносимость. Наконец, вы можете агрегировать трассу по заданным ключам, а не переносить данные в SQL Server для выполнения этой функции.

Заключение

Если вы еще не сделали этого, непременно обновите версию SQL Server Beta 2 до последней проверенной сообществом версии для предварительного просмотра Community Technology Preview (CTP). На момент публикации этой статьи последней была апрельская версия 2005 года. В каждом выпуске версии для предварительного просмотра вычищаются несколько важных функциональных возможностей. К середине лета 2005 года они будут зафиксированы. Последний выпуск работает намного быстрее и содержит гораздо меньше ошибок, чем предыдущие бета-версии SQL Server 2005.

Брайан Найт (Brian Knight) обладает сертификатами SQL Server MVP и MCDBA. Он является одним из основателей SQLServerCentral.com и работает в Allstate менеджером баз данных. Брайан — автор нескольких книг.

