

## ГЛАВА 6

# Администрирование служб Integration Services SQL Server и настройка производительности

**В** продолжение темы о том, как SQL Server 2005 изменяет роль администратора БД, мы рассмотрим новые инструментальные средства, которые помогают администратору в обслуживании различных компонентов SQL Server. Пакет бизнес-логики SQL Server 2008 включает службы интеграции (Integration Services, SSIS), службы анализа (Analysis Services, SSAS) и службы отчетности (Reporting Services, SSRS). Мы остановимся на службах интеграции и службах анализа, так как прежде всего именно эти компоненты требуют глубокого понимания для осуществления успешного управления. В этой главе мы рассмотрим различные возможности администрирования, необходимые для управления службами Integration Services. Сначала проведем общий обзор служб Integration Services, чтобы у читателя была возможность познакомиться с ними поближе и получить более полное представление о настройках, требующих внимания со стороны администратора. После ознакомления с архитектурой служб Integration Services сфокусируем внимание читателя на их администрировании, включая конфигурацию, протоколирование событий и отображение активности. Затем перейдем к рассмотрению различных административных задач, которые необходимы для работы с пакетами служб интеграции, финансовым компонентом SSIS, включая создание, управление, выполнение и развертывание. В конце этой главы читатель познакомится с тем, как обеспечить безопасность всех служб Integration Services.

---

**Примечание.** Более детальную информацию о службах Integration Services можно получить в книге Брайана Найта и др. *Professional SQL Server 2008 Integration Services* (издательство Wrox, 2008).

---

## Обзор служб Integration Services

Конечно, самым существенным новшеством бизнес-логики при переходе с SQL Server 2000 на SQL Server 2008 можно считать компоненты, отвечающие за передачу и обработку данных. На платформе 2000 для этих целей использовалась служба преобразования данных (Data Transformation Services, DTS), в то время как на платформах 2005 и 2008 для этих целей предусмотрены службы Integration Services. Различие в возможностях этих двух служб более значительно, чем можно было бы судить по простой перемене имени, так как службы Integration Services представляют собой полностью “переделанные” службы DTS. В действительности ни одного исходного кода из DTS в службах Integration Services не осталось. Подробнее о переходе со службы SQL Server 2000 DTS на работу со службами Integration Services можно будет узнать в главе 3, “Рекомендуемые методы обновления SQL Server 2008”.

Службы Integration Services представляют собой решение, обеспечивающее интеграцию и обработку данных на уровне предприятия и имеющее своей целью выборку, преобразование и загрузку (extraction, transformation, and loading, ETL) данных из разных источников разными адресатами. В состав служб Integration Services включен широкий диапазон инструментальных средств и мастеров, которые помогают в создании управляющей логики и потоков данных.

## Применение служб Integration Services

Перед тем как перейти к обсуждению компонентов служб Integration Services, следует глубже изучить самые популярные бизнес-решения, влекущие за собой создание SSIS-решений.

Одним из самых популярных сценариев является сценарий, который предполагает совместное использование данных, полученных из различных источников, хранящихся в различных системах хранения. В этом сценарии службы SSIS будут отвечать за подключение к каждому источнику данных, выборку данных и слияние их в один общий набор данных. В топологии современных информационных систем этот сценарий приобретает все большую популярность, так как деловая архивная информация не нужна для постоянной работы, но неоценима при анализе тенденций в бизнесе или для оценки соответствия требованиям. Этот сценарий имеет место, когда различные направления одного бизнеса используют в своей работе различные технологии или разные схемы предоставления одних и тех же данных. В этих случаях службы SSIS используются для обеспечения однородности информации. Службы SSIS безупречно обрабатывают различные данные, полученные из различных источников, преобразуют типы данных, разбивают или сливают столбцы, отыскивают описательную информацию, которая в таких ситуациях является существенным подспорьем.

Еще одним популярным общим сценарием является заполнение и ведение хранилищ и магазинов данных. В этих бизнес-приложениях объемы данных имеют тенденцию к увеличению, а временное окно, в котором должна выполняться выборка, преобразование и загрузка данных, имеет тенденцию к уменьшению. Службы SSIS включают возможность проводить массовую загрузку данных непосредственно из простых файлов в SQL Server и также имеют компонент, отвечающий за выполнение массовой загрузки в SQL Server. Ключевой возможностью для данных большого объема и ситуаций взаимного обогащения и преобразования является повторяемость (возможность повторного запуска). Службы SSIS включают контрольные точки для обеспечения перезапуска пакета из задания или контейнера, что позволяет “элегантно” обрабатывать различные типы ошибок, возникающие во время выполнения сценариев загрузки комплексных данных.

Еще одним важным фактором при загрузках в хранилища данных является возможность направлять данные в конкретное место из различных таблиц или файлов.

В теории баз данных такое действие называется *денормализацией*. Пакеты SSIS позволяют объединять данные в один набор данных и проводить загрузку данных в таблицы в едином процессе, не разбивая процесс на отдельные этапы, и без промежуточного запоминания данных на каждом шаге процесса.

Наконец, часто требуется управление хронологией или разбиение по хронологии в хранилищах данных для получения возможности анализа активности по состоянию на определенный момент времени. Такое управление историей позволяет создавать различные сценарии обновления, и службы SSIS обрабатывают их с помощью мастера Slowly Changing Dimension. Этот мастер динамически создает и настраивает набор задач преобразования данных, которые применяются для вставки и обновления записей, обновления связанных записей и добавления новых столбцов в таблицы для поддержки управления историей.

Часто бывает так, что бизнес-приложения получают данные извне. Потом их необходимо стандартизировать и обработать перед загрузкой в систему. При этом обычно в различных областях бизнеса применяются различные стандарты и форматы представления информации или при покупке данных, как это бывает в случае адресных данных. Иногда форматы данных различаются из-за того, что платформы, с которых они получены, отличаются от платформ, для которых они предназначены. Для этого случая службы SSIS имеют в своем арсенале широкий набор задач преобразования данных для осуществления широкого диапазона функций обработки данных, преобразования их и дополнения. Службы SSIS позволяют заменить значения или получить описания по кодовым значениям с использованием точного или эвристического поиска. Идентификация записей, которые могут быть дубликатами, с применением группирующих преобразований служб SSIS позволяет успешно удалять их перед загрузкой адресатом.

Возможность динамической настройки преобразований данных представляет собой одно из основных требований современного бизнеса, так как часто в зависимости от самих данных или от итогового результата и количества значений данные должны обрабатываться по-разному. Службы SSIS имеют в своем арсенале богатый набор преобразований, которые могут использоваться для разбиения или объединения данных на основании значений данных, включая различные объединения или вычисления на основании анализа различных частей набора данных и загрузки различных частей данных для различных адресатов. В частности, контейнеры служб SSIS поддерживают вычисление выражений, перечислений в информационных наборах и выполнение задач управления на основе результатов, полученных с использованием данных.

Наконец, обязательно должны быть оперативные административные функции, требующие автоматизации. Службы SSIS включают полный набор задач, предназначенных для выполнения этих административных функций. Можно воспользоваться задачами, специально разработанными для копирования объектов SQL Server или для оказания помощи при загрузке массовых данных. Кроме того, для выполнения прохода по всем имеющимся серверам и выполнения административных операций на каждом сервере вашей среды в службах SSIS можно воспользоваться средством SQL Management Objects (SMO). Еще существует возможность запланировать график работы всех пакетов и решений служб SSIS с помощью заданий SQL Server Agent (SQL Server Agent).

## Четыре основных компонента служб Integration Services

Службы Integration Services состоят из четырех компонентов, показанных на рис. 6.1:

- собственно служба SSIS;
- рабочий механизм и рабочие компоненты SSIS;

- ❑ объектная модель SSIS;
- ❑ механизм данных и компоненты данных SSIS.

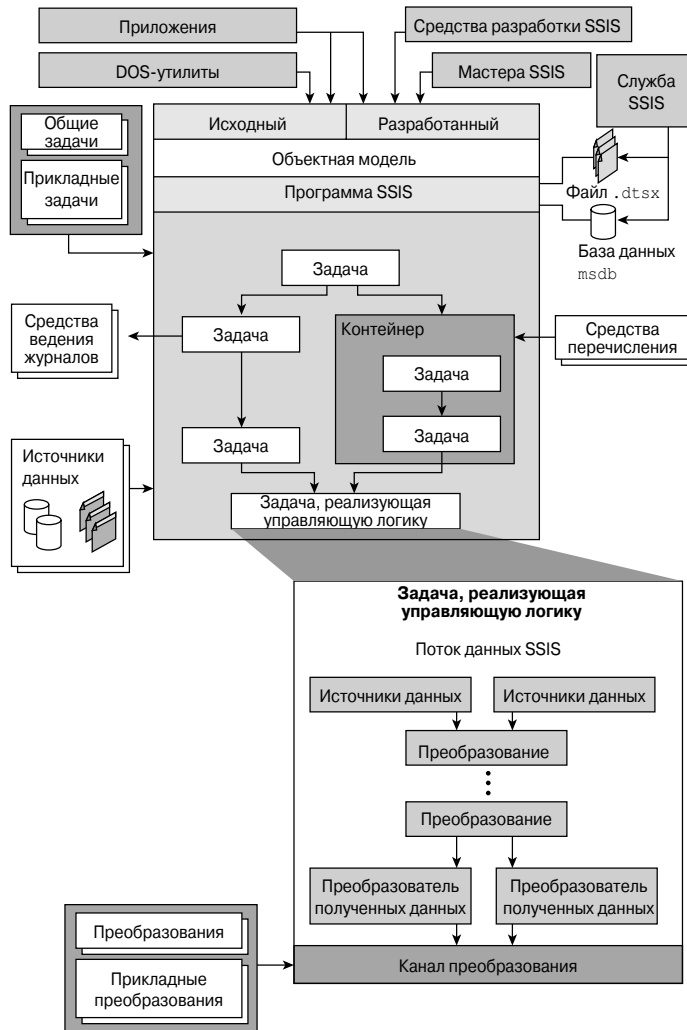


Рис. 6.1. Четыре составляющие служб Integration Services

## Службы Integration Services

Компонент архитектуры служб Integration Services, ответственный за управление выполняющимися пакетами и управление хранением пакетов, — это все службы SSIS.

## Рабочий механизм и рабочие компоненты служб Integration Services

Рабочий механизм служб SSIS отвечает за сохранение компоновки и дизайна пакетов, запуск пакетов и обеспечение такой дополнительной функциональности пакетов, как выполнение транзакций, обработка контрольных точек, настройка соединений,

обработка событий и регистрация. Специфические программы, которые входят в состав этого механизма, включают пакеты, контейнеры, задачи и обработчики событий. У служб SSIS имеется три основных ограничения: успех, завершение и отказ.

### **Объектная модель служб Integration Services**

Для доступа к средствам служб SSIS, утилитам командной строки и разработанным приложениям используется управляемый программный интерфейс (API-интерфейс), являющийся объектной моделью SSIS. Пока мы еще не погрузились в дебри информации об этой объектной модели, очень важно сказать, что она является основным компонентом служб Integration Services.

### **Механизм и компоненты потока данных служб Integration Services**

В рамках управляющей логики пакета служб SSIS задача потока данных создает экземпляры механизма потока данных. Этот механизм отвечает за обеспечение переноса данных в памяти от источника к адресату. Кроме того, этот механизм выполняет заказанные преобразования данных для определяемых вами целей. Три основных компонента, составляющих механизм потока данных, представляют собой источники, преобразования и адресаты. *Источники* обеспечивают соединение и точные данные из широкого диапазона источников, таких как таблицы или представления баз данных, файлы, электронные таблицы и даже XML-файлы. *Адресаты* позволяют выполнять вставку, обновление и удаление информации на подобном широком диапазоне адресатов. Наконец, *преобразования* позволяют проводить преобразование исходных данных перед их загрузкой адресатом с использованием таких возможностей, как поиск, слияние, манипулирование, разбиение, преобразование и получение информации.

### **Управление проектами и изменение управления**

Однозначно одной из областей, в которой потребуется полный пересмотр мировоззрения по сравнению с предыдущей версией SQL Server, является взаимодействие администратора БД с разработчиками. Общий подход администраторов и разработчиков к разработке реализуется посредством среды Business Intelligence Developer Studio (BIDS). С точки зрения служб Integration Services среда BIDS является своеобразным хранилищем принимаемых решений и разрабатываемых проектов. В целом, настройка решений и проектов в среде BIDS возлагается на разработчиков, однако при этом для консультаций при настройке различных аспектов этих решений вызываются администраторы. Администрирование и управление службами Integration Services в основном выполняется из среды SQL Server Management Studio. Часто перенос решений по службам Integration Services из одной среды в другую будет означать изменение динамической информации внутри пакета, а также установку информационных ссылок на пакеты. Примерами таких элементов являются настройки пакетов, XML-файлы или конфигурационные файлы и источники данных принятых решений, на которые даются ссылки.

Заметим, что когда разработчик щелкает на кнопке Save (Сохранить) или выполняет пакет в среде BIDS, старая версия пакета немедленно отвергается. Для того чтобы выйти из сложившейся ситуации, необходимо интегрировать среду разработки службы SSIS в исходную систему управления, например Visual SourceSafe. После такой интеграции, когда сохраняются старые пакеты, всегда можно сделать откат на более раннюю версию пакета. Можно использовать любой тип исходной системы управления, интегрируемой в Visual Studio 2008.

## Администрирование служб Integration Services

А сейчас, после того как мы поняли предназначение составляющих частей служб Integration Services, рассмотрим различные административные аспекты использования служб Integration Services и обсудим детали, необходимые для обретения уверенности в работе с компонентами службы. Начнем с обзора служб Integration Services, рассмотрим различные элементы настройки этих служб. Затем наступит очередь вопросов настройки свойств служб SSIS с использованием возможностей традиционной оснастки Службы Windows или диспетчера настройки SQL Server Configuration. Затем попробуем понять, что можно изменить в брандмауэре Windows, а потом рассмотрим управление и настройку журнала регистрации событий. И наконец, узнаем, как осуществляется мониторинг рабочих характеристик системы.

### Обзор

Службы Integration Services — это службы Windows, предназначенные для управления пакетами SSIS. Доступ к ним осуществляется из среды SQL Server Management Studio. Вот перечень возможностей, предоставляемых этими службами:

- запуск и остановка локальных и удаленных пакетов;
- управление локальными и удаленными пакетами;
- импорт-экспорт пакетов из различных источников;
- управление хранилищем пакетов;
- настройка папок хранилищ;
- остановка работающих пакетов в случае остановки службы;
- просмотр журнала регистрации событий Windows;
- подключение к многочисленным экземплярам служб SSIS.

Чтобы быть абсолютно точным, эта служба для разработки и выполнения пакетов не нужна. Самым главным предназначением этой службы является управление пакетами в среде Management Studio. Одно из преимуществ, которое можно получить от работы этой службы, заключается в том, что при работе со средством разработки SSIS (Designer SSIS) можно использовать эту службу в среде Business Intelligence Developer Studio для кэширования объектов, используемых в среде разработки, дополняя тем самым возможности среды разработки.

### Настройка

Настройка служб Integration Services включает просмотр и, возможно, модификацию XML-файла, отвечающего за рабочие настройки службы, установку свойств службы с использованием возможностей служб Windows или диспетчера настройки SQL Server (SQL Server Configuration Manager) и, вероятно, настройку брандмауэра Windows для того, чтобы открыть доступ к службам Integration Services.

### Конфигурационный XML-файл

Отвечающий за настройку служб Integration Services файл `MsDtsSrvr.ini.xml` по умолчанию размещен в папке `<устройство_SQL_Server>:\Program Files\Microsoft SQL Server\100\DTS\Binn`. Размещение конфигурационного файла можно изменить. Эта информация попадает в среду Management Studio из ключа

реестра Windows<sup>1</sup>: HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Microsoft SQL Server\100\SSIS\ServiceConfigFile. Этот файл включает установки для остановки пакетов в случае остановки службы, перечень корневых папок для отображения их в браузере Object Explorer среды Management Studio и установки для определения папок файловой системы, которые управляются службой.

Имя конфигурационного файла и его размещение могут быть изменены. Эта информация попадает в среду Management Studio из ключа реестра Windows<sup>2</sup>: HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\MSDTS\ServiceConfigFile. Как и при всех остальных манипуляциях с реестром, перед тем как внести изменения, вам следует сделать резервную копию реестра и перезапустить службу после внесения изменений, для того чтобы эти изменения вступили в силу.

Вот один пример изменения настройки, который необходимо сделать при подключении к именованному экземпляру SQL Server. Следующий пример отражает изменения, необходимые для работы с именованным экземпляром (MyServerName\MyInstanceName). Так происходит потому, что по умолчанию настройка для службы SSIS всегда указывает на "." и должна настраиваться на кластеризованный или именованный экземпляр.

```
<?xml version="1.0" encoding="utf-8"?>
<DtsServiceConfiguration xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <StopExecutingPackagesOnShutdown>true</StopExecutingPackagesOnShutdown>
  <TopLevelFolders>
    <Folder xsi:type="SqlServerFolder">
      <Name>MSDB</Name>
      <ServerName>MyServerName\MyInstanceName</ServerName>
    </Folder>
    <Folder xsi:type="FileSystemFolder">
      <Name>File System</Name>
      <StorePath>..\Packages</StorePath>
    </Folder>
  </TopLevelFolders>
</DtsServiceConfiguration>
```

В этом файле особенно нечего настраивать, но у него есть определенные интересные варианты применения. Первая строка сообщает пакетам, каким образом можно реагировать в том случае, если служба была остановлена. По умолчанию пакеты, с которыми работает служба, остановятся вместе с остановкой службы или завершат свою работу со сбоем. Вам также следует настроить пакеты для продолжения работы до тех пор, пока не завершат работу после завершения работы службы, изменив значение свойства StopExecutingPackagesOnShutdown на False.

```
<StopExecutingPackagesOnShutdown>false</StopExecutingPackagesOnShutdown>
```

Следующим традиционным шагом при работе с конфигурационными файлами является внесение поправок в сценарии, включая добавление дополнительных путей размещения пакетов, отличных от стандартного пути сохранения пакетов служб SSIS C:\Program Files\SQL Server\100\Packages, и создание централизованной структуры папок для нескольких серверов и сохранение в централизованной общедоступной папке конфигурационного файла.

<sup>1</sup> Уникальный идентификатор, приписываемый каждой порции информации в системной регистрационной базе данных. — *Примеч. пер.*

<sup>2</sup> Уникальный идентификатор, приписываемый каждой порции информации в системной регистрационной базе данных. — *Примеч. пер.*

## Создание центрального сервера SSIS

Многие большие компании используют в своей работе так много пакетов, что они пришли к решению отделить эту службу от SQL Server, разместив ее на отдельном сервере. Когда вы это сделаете, потребуется этот новый сервер лицензировать, как в случае, когда этот сервер является полноценным SQL Server. Преимущество такого решения заключается в том, что ваши пакеты SSIS не будут занимать память SQL Server в моменты большой загрузки, и в вашем распоряжении появляется центральная точка управления. Недостатки этого подхода заключаются в том, что необходимо лицензировать этот сервер отдельно, и во время отладки пакетов добавляется еще один уровень сложности. При создании выделенного сервера вы создаете фантастический способ простого масштабирования пакетов в результате добавления дополнительной памяти к центральному серверу, но также создаете дополнительную нагрузку, так как удаленные данные должны копироваться по сети для ввода их в буфер передачи данных.

Для того чтобы создать централизованный узел службы SSIS, необходимо только внести изменения в файл `MsDtsSrvr.ini.xml` и перезапустить службу. Эта служба может получать путь наподобие `\\ServerName\Share`, который будет указывать сразу на несколько удаленных серверов. При этом не рекомендуется прибегать к использованию общих устройств, так как учетная запись, запускающая службу SSIS, должна знать это устройство и иметь возможность создания необязательной зависимости с этой учетной записью. В следующем примере служба будет насчитывать пакеты, поступающие от трех серверов, один из которых является локальным, а другие — именованными экземплярами. После перезапуска службы в среде Management Studio можно будет увидеть шесть новых папок. Позднее в этой главе мы обязательно обсудим аспекты работы в среде Management Studio со службой SSIS.

```
<?xml version="1.0" encoding="utf-8" ?>
<DtsServiceConfiguration xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <StopExecutingPackagesOnShutdown>true</StopExecutingPackagesOnShutdown>
  <TopLevelFolders>
<Folder xsi:type="SqlServerFolder">
  <Name>Server A MSDB</Name>
  <ServerName>localhost</ServerName>
</Folder>
  <Name>Server B MSDB</Name>
  <ServerName>SQLServerB</ServerName>
</Folder>
<Folder xsi:type="File systemFolder">
  <Name>Server A File System</Name>
  <StorePath>P:\Packages</StorePath>
</Folder>
<Folder xsi:type="File systemFolder">
  <Name>Server B File System</Name>
  <StorePath>\\SQLServerB\Packages</StorePath>
</Folder>
</TopLevelFolders>
</DtsServiceConfiguration>
```

Следующим вопросом при работе с центральным узлом службы SSIS является планирование пакетов. Выполнение пакетов можно запланировать с помощью SQL Server Agent (SQL Server Agent) или с помощью такой системы планирования, как Планировщик заданий (Task Scheduler), имеющейся в арсенале ОС Windows. Но раз уж вы оплачиваете лицензию еще одного SQL Server, практичнее будет там устано-



вить SQL Server и воспользоваться его агентом SQL Server, так как он предоставит вам гораздо больше гибкости (позднее вы увидите почему). Кроме того, также можно сохранить конфигурационные и регистрационные таблицы на этом SQL Server для централизации обработки. Оба механизма проектирования описываются в этой главе.

После каждого внесения изменений в конфигурационный файл необходимо останавливать и запускать снова службу SSIS, как это описывается в следующих разделах. А теперь, после того, как мы узнали, каким образом можно настроить файл `MsDtsSrvr.ini.xml`, который отвечает за настройку служб Integration Services, необходимо узнать, как задавать свойства службы.

## Установка свойств служб с помощью оснастки Services

Как и любые другие службы Windows, службы Integration Services имеют свойства, позволяющие задавать условия их запуска. Это можно сделать, воспользовавшись обычной Windows-оснасткой Services (Службы):

- настройка типа запуска службы — Manual (Вручную), Automatic (Авто) или Disabled (Отключено);
- запрос о состоянии службы, т.е. она started (запущена), stopped (остановлена) или restarted (перезапущена);
- определение реакции компьютера на сбой службы;
- просмотр или модификация зависящих служб (ни одна из них не задается по умолчанию).

Для просмотра и изменения свойств служб SSIS с использованием Windows-оснастки Services следует придерживаться следующей последовательности действий.

1. Находясь в Control Panel (Панель управления), выбрать Administrative Tools (Администрирование), а там открыть оснастку Services (или из окна Category (Категории) из Performance and Maintenance ⇒ Administrative Tools (Производительность и обслуживание ⇒ Администрирование)).
2. Найти в списке служб SQL Server Integration Service и щелкнуть на этой службе правой кнопкой мыши.
3. Щелкнув правой кнопкой мыши на этой службе, выберите Properties (Свойства) для просмотра текущих установок службы.
4. На вкладке General (Общие) можно просматривать или вносить изменения в столбец Startup type (Тип запуска) (Automatic (Авто), Manual (Вручную) или Disabled (Отключено)). Затем, выбрав Manual или Automatic, изменить состояние службы на Start (Работает), Stop (Остановлена) или Resume (Возобновлена) (рис. 6.2).
5. На вкладке Log On (Вход в систему) можно просматривать значение учетной записи, используемой для запуска и остановки этой службы, и вносить в нее изменения. По умолчанию она будет работать под управлением учетной записи `NT AUTHORITY\NetworkService`.
6. На вкладке Recovery (Восстановление) можно настроить реакцию сервера на сбой в работе службы. Для этого задаются значения в строках First (Первый сбой), Second (Второй сбой) и Subsequent (Последующие сбои), равные Take No Action (Не выполнять никаких действий) (умолчание), Restart the Service (Перезапуск службы), Run a Program (Запуск программы) или Restart the Computer (Перезагрузка компьютера) (рис. 6.3). Кроме того, можно задать сброс счетчика сбоев службы по прошествии определенного срока в днях.

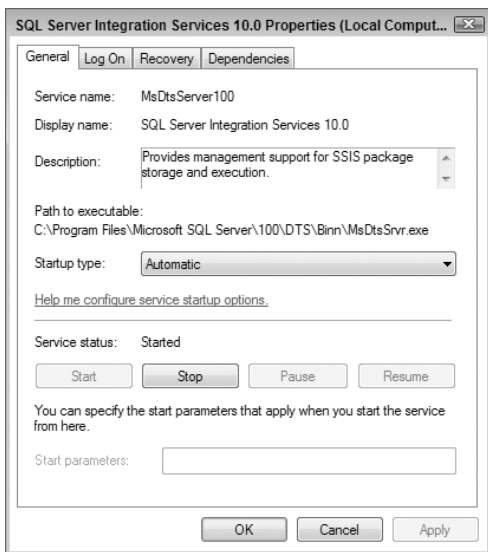


Рис. 6.2. Windows-оснастка Services

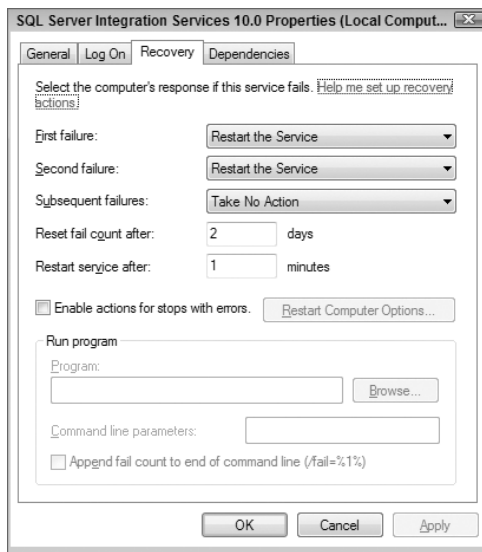


Рис. 6.3. Вкладка Восстановление

7. Наконец, на вкладке Dependencies (Зависимости) можно изменить список служб, от которых зависят службы SSIS (по умолчанию таких зависимостей нет), и просмотреть список служб, находящихся в зависимости от служб SSIS (по умолчанию таких зависимостей нет).

### Установка свойств службы с использованием диспетчера настройки SQL Server

Как и Windows-оснастка Services (Службы), диспетчер настройки SQL Server Configuration Manager позволяет настроить ограниченный набор свойств служб Integration Services. В частности, он позволяет настроить регистрационную информацию, используемую службой, и задавать режимы запуска службы.

Вот как можно просматривать и модифицировать свойства служб SSIS с помощью диспетчера настройки SQL Server (SQL Server Configuration Manager).

1. Откройте диспетчер настройки SQL Server из меню Start⇒All Programs⇒Microsoft SQL Server 2008⇒Configuration Tools (Пуск⇒Все программы⇒Microsoft SQL Server 2008⇒Средства настройки).
2. В списке служб, который располагается справа, щелкните правой кнопкой мыши на SQL Server Integration Services и выберите в появившемся меню Properties (Свойства).
3. На вкладке Log On (Вход в систему) можно просмотреть или изменить учетную запись, которая используется для запуска и работы службы. По умолчанию служба работает под управлением учетной записи NT AUTHORITY\NetworkService.
4. На вкладке Service (Служба) можно просматривать или вносить изменения в столбце Startup type (Тип запуска), задав значение Automatic (Авто), Manual (Вручную) или Disabled (Отключено).

Теперь, когда мы научились устанавливать свойства службы для служб Integration Services с помощью Window-оснастки Services или из диспетчера настройки SQL Server, посмотрим, что нам может понадобиться при внесении изменений в установки брандмауэра Windows для разрешения доступа к службам Integration Services.

### **Настройка брандмауэра Windows для осуществления доступа**

Вероятно, что вы уже обнаружили, что для того, чтобы обеспечить устойчивый доступ к службам Integration Services, требуется внесение изменений в работу брандмауэра Windows. Брандмауэр Windows контролирует доступ к определенным ресурсам компьютера прежде всего за счет ограничения доступа к портам. У нас нет возможности поменять порт, с которым работают службы Integration Services, так как они могут работать только с портом 135.

В дополнение к использованию пользовательского интерфейса, о котором шла речь в этом разделе, этот процесс можно запрограммировать с помощью сценария из командной строки, запустив следующую команду (заметим, что здесь строка разбита на несколько строк, но все-таки это одна команда).

```
netsh firewall add portopening protocol=TCP port=135 name="RPC (TCP/135)"
mode=ENABLE scope=SUBNET
netsh firewall add allowedprogram program="%ProgramFiles%\Microsoft SQL
Server\100\DTS\Binn\MsDtsSrvr.exe" name="SSIS Service" scope=SUBNET
```

Вот последовательность действий с брандмауэром Windows, позволяющая открыть доступ к службам Integration Services.

1. Находясь в Control Panel (Панель управления), откройте Windows Firewall (Брандмауэр Windows).
2. Выберите вкладку Exceptions (Исключения) и щелкните на кнопке Add program (Добавить программу).
3. В диалоговом окне Add program (Добавить программу) щелкните на кнопке Browse (Обзор) и выберите C:\Program Files\Microsoft SQL Server\100\DTS\Binn\MsDtsSrvr.exe. Конечно, для того, чтобы указать непосредственным образом компьютеры, имеющие доступ, задав их с помощью списка IP-адресов, подсетей или и того и другого, можно воспользоваться опцией Change Scope (Изменить область). В результате получим список исключений, указанный в диалоговом окне Windows Firewall (Брандмауэр Windows) (рис. 6.4).
4. Щелкните на кнопке Add Port (Добавить порт).
5. В диалоговом окне Add Port (Добавить порт) введите какое-то осмысленное описание порта наподобие RPC(TCP/135) Integration Services, в окошке ввода номера порта введите значение 135 и выберите протокол TCP. Конечно, для того, чтобы детализировать компьютеры, имеющие доступ к порту, задав список IP-адресов, подсетей или и того и другого, необходимо воспользоваться параметром Change Scope (Изменить область). Полученное исключение можно увидеть в диалоговом окне Windows Firewall на рис. 6.4.

К данному моменту мы много узнали о настройках, необходимых для служб Integration Services. Следующим шагом в понимании работы служб Integration Services является получение представления о протоколировании событий.

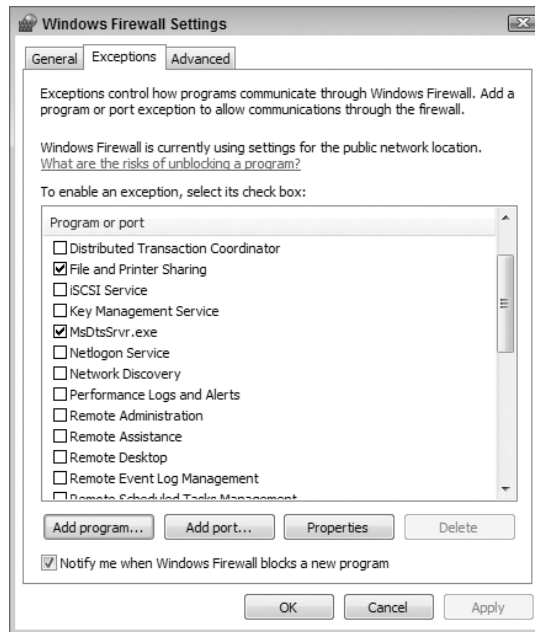


Рис. 6.4. Диалоговое окно Windows Firewall

## Журналы регистрации событий

Службы Integration Services записывают в журналы события, происходящие во время выполнения пакетов. Журналы регистрации событий SSIS могут вестись в формате текстового файла, журнала регистрации событий SQL Server, журнала регистрации событий Windows или в формате XML-файлов. Для того чтобы осуществлялось протоколирование, пакеты и задачи SSIS должны иметь активизированную возможность протоколирования. Протоколирование может проводиться на уровне пакетов, контейнеров и задач, при этом для пакетов, контейнеров и задач могут быть предусмотрены отдельные журналы регистрации.

Кроме того, для протоколирования событий должен быть определен и добавлен в пакет провайдер протокола. Эти журналы могут создаваться только на пакетном уровне, а на уровне задач или контейнеров должен использоваться один из журналов, созданных для пакета. После создания журналов внутри пакета их можно просмотреть с помощью средства просмотра событий Windows или из среды SQL Server Management Studio.

Вот каким образом можно просматривать журналы регистрации ошибок SSIS с использованием средства просмотра событий Windows.

1. Из Control Panel ⇒ Administrative Tools (Панели управления ⇒ Администрирование) (из окна Category (Категории) из Performance and Maintenance ⇒ Administrative Tools (Производительность и обслуживание ⇒ Администрирование)) откройте средство просмотра событий Windows.
2. Находясь в окне Event Viewer (Просмотр событий), щелкните на элементе Application (Приложение).
3. После появления оснастки Application (Приложение) найдите запись, имеющую в столбце Source (Источник) значение `SQLISService100` или `SQLISPackage100`.

Исходный журнал SQLISPackage100 будет сгенерирован из журнала пакета, а SQLISService100 будет содержать простые сообщения службы SSIS.

- Щелкните правой кнопкой мыши на этой записи и для просмотра описательной информации о записи выберите Event Properties (Свойства событий) (рис. 6.5).

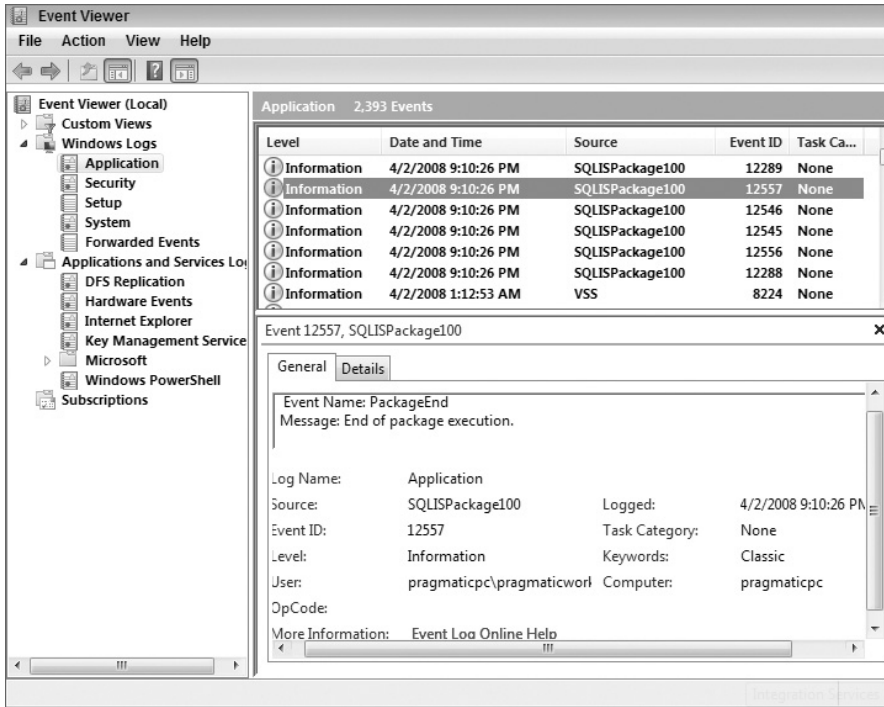


Рис. 6.5. Окно Event Viewer

А теперь посмотрим, каким образом можно просматривать эти события из среды SQL Server Management Studio.

- Откройте среду Management Studio и подключитесь к целевому серверу Integration Services.
- В браузере Object Explorer щелкните правой кнопкой мыши на узле Integration Services (расположен вверху), затем щелкните на появившемся элементе View Logs (Просмотр журналов).
- В секции Select Logs (Выбор журналов) выберите элемент SQL Server Integration Services.
- Детализированная информация о событии появится в нижней панели после щелчка на событии из списка, расположенного в верхней панели.

## Мониторинг активности

Контроль за работой служб Integration Services включает настройку протоколирования счетчиков производительности. Эти счетчики позволяют просматривать и анализировать потребление ресурсов, задействованных во время выполнения пакетов SSIS. А детальнее, протоколирование позволяет наблюдать за потреблением ресурсов во время выполнения пакетами задач по обслуживанию потоков данных.

Сначала обратим внимание на самые полезные счетчики, включая счетчик прочитанных записей (Rows read), использованных буферов (Buffers in use) и спулинга буферов (Buffers spooled). Счетчик прочитанных записей обеспечивает нас количеством прочитанных из источников данных записей во время выполнения пакета. Счетчик используемых буферов хранит количество конвейерных буферов (пула памяти) в конвейере пакетов. Наконец, счетчик спулинга буферов определяет количество буферов, использованных для обработки процессов передачи данных. Этот счетчик имеет особенно большое значение, так как он является хорошим индикатором того, что машине для выполнения всех задач не хватает физической памяти или не хватает виртуальной памяти во время обработки потока данных. Важность использования буферов, а не дисковой памяти трудно переоценить, так как службы SSIS настраиваются для осуществления быстрой и эффективной передачи данных прежде всего посредством эффективного использования памяти, а не дискового пространства. Когда службам SSIS приходится переходить на использование дисковой памяти, а не буферной, производительность выполнения операций с данными существенно снижается, о чем сразу же засвидетельствует счетчик буферов, сохраненных в области спулинга. Преимущество использования буферов вместо организации спулинга на диск особенно проявляется в разнице скорости выполнения пакета, которая может занять 20 минут по сравнению с 20 часами в некоторых случаях. При каждой операции спулинга на диск записывается буфер объемом 10 Мбайт.

Одним из примеров использования этих счетчиков производительности является проверка достаточности памяти для работы пакетов служб SSIS. Узким местом в любом процессе преобразования являются операции ввода-вывода, во время которых обрабатываемые данные сохраняются на диске. Службы Integration Services были разработаны для оптимизации использования ресурсов во время преобразования данных при передаче от источника адресату, включая попытки выполнения этих преобразований в оперативной памяти, без сохранения их при этом на диске со всеми накладными расходами, которые влекут за собой дисковые операции ввода-вывода. Следует добиваться, чтобы значение счетчика спулинга буфера оставалось на значении нуль (0), так как в этом случае во время процессов преобразования, выполняемых пакетами служб SSIS, будет использоваться только память. Если значение счетчика спулинга буферов превышает нулевое значение, это верный показатель того, что серверу, занятому обработкой пакетов служб SSIS, требуется добавить память.

Для проведения анализа операций с данными и планов выполнения запросов, сгенерированных для различных операций потока данных, потребуется SQL Server Profiler. Этой информацией можно воспользоваться при отладке индексов или применении других методов оптимизации источников данных, которые используются в вашем решении по использованию служб SSIS. Познакомиться с SQL Server Profiler можно в главе 13, “Мониторинг SQL Server”.

## Администрирование пакетов служб Integration Services

После ознакомления с самыми различными аспектами администрирования служб Integration Services рассмотрим подробнее основные аспекты, связанные с администрированием пакетов SSIS. Начнем с обзора элементов пакетов служб SSIS, а затем рассмотрим различные способы создания пакетов и методы управления пакетами служб SSIS. После выяснения, каким образом администрируются и управляются пакеты, приступим к вопросам размещения, выполнения и планирования пакетов и решений служб SSIS.

## Обзор

Пакет является основной организационной и рабочей единицей служб Integration Services. Пакеты включают в себя набор задач управляющей логики, связанных между собой ограничениями предшествования для управления последовательностью выполнения задач, и задач передачи данных, призванных управлять перемещением данных от источника к адресату, со множеством задач преобразования данных между ними и программами обработки событий, отвечающими за обмен различной информацией о состоянии пакета и его задач.

Для того чтобы понять различные административные функции пакетов, начнем с момента создания пакета. Проектируются пакеты с помощью мастера импорта и экспорта или из среды Business Intelligence Developer Studio. Затем созданный пакет сохраняется в файловой системе или базе данных SQL Server, для того чтобы можно было его использовать в дальнейшем. Часто приходится выполнять задачи перемещения пакетов из одного места хранения в другое с помощью таких инструментов, как утилита DTUtil, работающая из командной строки. Для запуска пакетов потребуется прибегнуть к помощи утилит DTEXEC или DTEXECUI. Запланировать выполнение пакетов можно с помощью Планировщика заданий SQL Agent. Наконец, управлять рабочими характеристиками и состоянием пакетов можно из среды Management Studio.

## Создание пакетов

Создать пакеты служб Integration Services можно с помощью мастера импорта и экспорта или создав решение Integration Services в среде Business Intelligence Developer Studio. В этой книге мы акцентируем внимание на работе с мастером импорта и экспорта, так как разработка пакетов в среде BIDS чаще всего является прерогативой разработчиков. Более того, существует множество источников, которые помогут вам самостоятельно разобраться в процессе создания и разработки пакетов в среде BIDS. Как администратору, вам необходимо понимать, каким образом вы можете настроить шаблоны пакетов для использования их членами коллектива разработчиков.

### **Использование мастера импорта и экспорта для создания пакетов**

При ежедневном управлении данными часто приходится перемещать или копировать их, пользуясь самыми различными источниками данных. Зачастую при выполнении таких сложных задач не требуется выполнения сложных преобразований или сложной логики. Для решения задач управления данными этого типа очень хорошо подходит мастер импорта и экспорта (Import and Export Wizard).

Основная предпосылка использования мастера импорта и экспорта заключается в возможностях преобразования, необходимого перед загрузкой данных адресату, которыми обладает этот мастер. Только этот мастер позволяет выполнять такие преобразования, как установка имен, типов данных и свойств типов данных столбцов, которые имеются у адресата. При использовании мастера импорта и экспорта преобразования на уровне столбца не поддерживаются.

Преобразование типов данных при переносе их от источника к адресату проводится с помощью файлов соответствия, по умолчанию размещенных в C:\Program Files\Microsoft SQL Server\100\DTS\MappingFiles. Примером такого файла может служить файл OracleClientToMSSql.xml, который используется для приведения типов данных СУБД Oracle в соответствие типам данных SQL Server. Вот пример XML-файла, который используется для преобразования типа данных DATE СУБД Oracle в тип данных datetime СУБД SQL Server.

```

<!-- DATE -->
<dtm:DataTypeMapping>
  <dtm:SourceDataType>
    <dtm:DataTypeName>DATE</dtm:DataTypeName>
  </dtm:SourceDataType>
  <dtm:DestinationDataType>
    <dtm:SimpleType>
      <dtm:DataTypeName>datetime</dtm:DataTypeName>
    </dtm:SimpleType>
  </dtm:DestinationDataType>
</dtm:DataTypeMapping>

```

В эту папку можно добавить свои файлы соответствия для обеспечения новых комбинаций соответствия, которые не были предусмотрены изначально. Кроме того, можно внести изменения в существующие файлы соответствия, если они как таковые не соответствуют вашим задачам. Следует заметить, что после добавления новых или изменения существующих файлов соответствия необходимо будет перезапустить мастер импорта и экспорта или среду Business Intelligence Developer Studio, для того, чтобы эти изменения вступили в силу.

Практиковаться в работе с мастером импорта и экспорта можно начать из среды Business Intelligence Development Studio, из среды SQL Server Management Studio или из командной строки, работая с утилитой DTSWizard (C:\Program Files\Microsoft SQL Server\100\DTS\Binn). Единственная особенность при запуске этого мастера из среды BIDS заключается в том, что этот мастер не может запускать полученный пакет на самом последнем шаге, вместо этого полученный пакет сохраняется как часть решения, в котором этот мастер был запущен.

Запустить мастер импорта и экспорта можно следующим способом:

- ❑ в среде Management Studio подключите сервер базы данных и внутри узла базы данных щелкните правой кнопкой мыши на базе данных, выберите Tasks (Задачи), а затем выберите Import Data (Импортировать данные) или Export Data (Экспортировать данные);
- ❑ в среде BIDS откройте решение SSIS, щелкните правой кнопкой мыши на папке SSIS Packages (Пакеты SSIS) и выберите элемент SSIS Import and Export Wizard (Мастер импорта и экспорта SSIS);
- ❑ также находясь в среде BIDS, щелкните на Project⇒SSIS Import and Export Wizard (Проект⇒ Мастер импорта и экспорта SSIS);
- ❑ из командной строки запустите утилиту DTSWizard.exe (C:\Program Files\Microsoft SQL Server\100\DTS\Binn).

Вот каким образом можно запустить мастер импорта и экспорта для перемещения данных.

1. Запустите мастер импорта и экспорта из среды Management Studio, среды BIDS или из командной строки.
2. На странице Choose a Data Source (Выбор источника данных) выберите место, откуда будут копироваться данные, задав источник данных (например, SQL Native Client, Excel или Flat File) и всю необходимую информацию об этом источнике данных (как, например, Server Name, Authentication и Database для SQL Native Client). В случае выбора Export Data (Экспорт данных) (в среде Management Studio) исходную информацию можно считать уже предварительно настроенной для включения сервера и базы данных, из которых мастер будет запущен.



3. На странице Choose a Destination (Выбор адресата) выберите место, куда будут копироваться данные, задайте источник данных и всю необходимую информацию об источнике данных. В случае выбора Import Data (Импорт данных) (в среде Management Studio) информацию об адресате можно считать уже предварительно настроенной для включения сервера и базы данных, из которых мастер будет запущен.
4. На странице Specify Table Copy or Query (Задать копию таблицы или запрос) можно задать копирование всего источника данных, выбрав параметр Copy data from one or more tables (Копировать данные из одной и более таблиц) или указав, что копируется только часть данных из источника данных, задав для определения передаваемых данных параметр Write a query (Запись запроса). На странице Provide a Query Source (Задать источник запроса) можно создать свой SQL-оператор или поискать уже готовый SQL-файл для последующей его загрузки и использования для определения исходных данных.
5. На странице Select Source Tables and Views (Выбор исходных таблицы и видов) укажите все таблицы и представления, которые будут включены в том случае, если предварительно не был выбран параметр Write a query (Запись запроса) для определения переносимых данных, так как в таком случае можно выбрать только одну таблицу или один запрос.
6. Дополнительно на странице Select Source Tables and Views можно настроить такие возможности, как оптимизация таблиц или запуск транзакции.
7. Кроме того, на странице Select Source Tables and Views при копировании таблиц или представлений, которые уже существуют в том месте, куда осуществляется копирование, можно выбрать параметр Edit Mappings (Редактирование размещений). Это позволит выполнять настройку таких возможностей, как удаление или добавление данных в существующую таблицу или активизацию идентичных вставок (рис. 6.6).

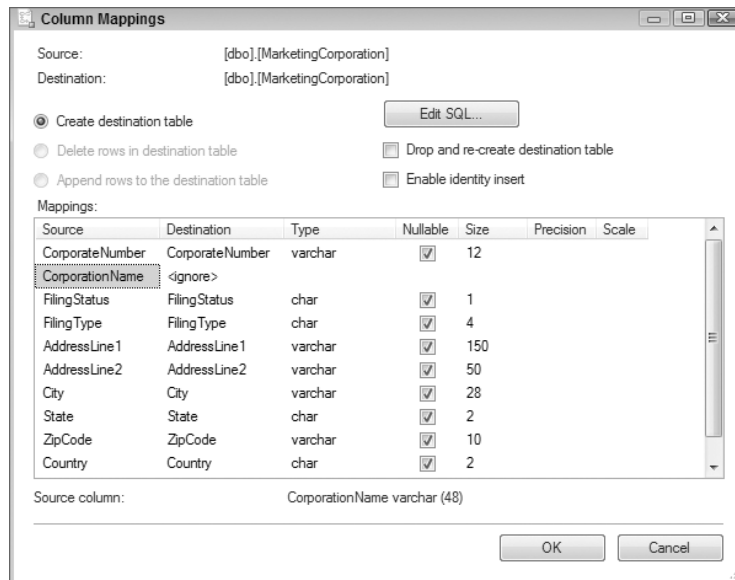


Рис. 6.6. Диалоговое окно мастера импорта и экспорта. Соответствие столбцов

8. Кроме того, на странице **Select Source Tables and Views** при создании новых таблиц или представлений, которые не существуют там, куда выполняется копирование, выберите **Edit Mappings** для настройки алгоритма копирования таким образом, чтобы таблицы адресата создавались или сначала удалялись, а потом создавались вновь. Кроме того, есть возможность внесения изменений в информацию соответствия адресата для всех столбцов, включая имя, тип данных, возможности хранения пустых значений, размера, точности и масштаба.
9. Наконец, при запуске из среды **Management Studio** на странице **Save and Execute Package (Сохранить и выполнить пакеты)** можно задать немедленное выполнение и дополнительно указать сохранение пакета SSIS. При сохранении на **SQL Server** или в файловую систему можно настроить уровни защиты сохраняемого пакета.

### Создание и использование шаблонов пакетов

Во время запуска нового решения **Integration Services** можно увидеть, что пакет уже включен в решение и готов к использованию. Несмотря на то, что заменить шаблон, принятый по умолчанию, который используется службами SSIS для создания стартового пакета (и всех последующих пакетов, созданных командой **New SSIS Package (Создать пакет SSIS)**), нельзя, создайте новый стандартный пакет или пакеты, которые будут использоваться в качестве шаблонов.

Основные преимущества создания таких пакетов:

- инкапсуляция общих элементов создания пакетов в каждый пакет (как протоколирование или обработка ошибочных ситуаций);
- улучшение рабочих характеристик на основании использования в пакетах общих переменных или конфигураций;
- применение таких стандартов проектирования пакетов, как аннотации;
- возможность отключения стандартного уровня защиты пакетов SSIS `EncryptSensitiveWithUserKey`, который может вызывать затруднения в работе целого коллектива разработчиков или при использовании сценария развертывания.

Чаще всего необходимость применения такого стандартного шаблона пакета возникает тогда, когда на уровне группы разработчиков принимается решение, что уровнем защиты пакетов по умолчанию будет `EncryptSensitiveWithPassword`. Так делается для того, чтобы распределить задачи по разработке пакетов между несколькими администраторами БД или разработчиками. Стандартизация и использование стандартного шаблона пакета вместо стандартного пакета SSIS позволяет автоматически задавать необходимый уровень защиты пакета. Кроме того, необходимость в таком стандартном шаблоне пакета может возникнуть при введении стандарта протоколирования и обработке ошибок в ваших пакетах. Пакет службы SSSS, принятый по умолчанию, не предусматривает такой функциональности, однако такую функциональность можно задать в стандартном шаблоне пакетов, а потом при работе пакетов каждый пакет будет изначально иметь такую встроенную функциональность.

Сделать пакет SSIS с помощью шаблона можно следующим образом.

1. Создать и сохранить пакет SSIS.
2. Скопировать файл пакета (файл с расширением `.dtsx`), который будет использоваться в качестве шаблона, в папку `DataTransformationItems`. Эта папка обычно размещена по пути `C:\Program Files\Microsoft Visual Studio 9.0\Common7\IDE\PrivateAssemblies\ProjectItems\DataTransformationProject\DataTransformationItems`.

После создания пакета SSIS в качестве шаблона его можно использовать в среде Business Intelligence Developer Studio. Вот каким образом можно использовать эти шаблоны в процессе разработки пакетов служб SSIS.

1. Откройте проект Integration Services в среде Business Intelligence Developer Studio.
2. В браузере Solution Explorer (Обозреватель решений) щелкните правой кнопкой мыши на решении или проекте (узел, расположенный выше всех) и выберите элемент меню Add (Добавить).
3. Из параметров, отображенных на подменю Add, выберите New Item (Новый элемент).
4. В диалоговом окне Add New Item – Project Name (Добавить новый элемент — Имя проекта) выберите пакет, который собираетесь использовать в качестве шаблона, и щелкните на элементе меню Add.

## Управление

Приступив к созданию собственно пакетов и решений, содержащих пакеты, необходимо отдельно остановиться на том, как управлять этими пакетами. Посмотрим, каким образом службы Integration Services помогают в управлении пакетами. Затем посмотрим, каким образом можно настроить среду Management Studio, чтобы она удовлетворяла задачам управления пакетами. Затем ознакомимся с утилитой управления пакетами DTUtil, затронув при этом некоторые детали, которые могут оказаться полезными при импорте и экспорте пакетов. А закончим этот экскурс просмотром возможностей пакетов, которые могут использоваться при проектировании пакетов и на стадиях разработки для помощи в управлении пакетами и решениями на этапе эксплуатации.

### Использование среды Management Studio для управления пакетами

Как уже рассказывалось ранее в разделе “Службы Integration Services”, пакетами можно управлять непосредственно из среды Management Studio и через связь со службами Integration Services. При подключении к службам появятся две основные папки: Running Packages (Работающие пакеты) и Stored Packages (Сохраненные пакеты). Пакеты, которые там можно увидеть, хранятся в таблице `sysssispackages` базы данных `msdb` или в папках файловой системы, которые определяются конфигурационным файлом служб Integration Services.

Основное назначение среды Management Studio заключается в управлении работающими пакетами и управлении пакетами, хранящимися в среде Integration Services. А теперь приступим к деталям.

Во-первых, в папке Running Packages вы сможете просмотреть информацию о работающих в данный момент пакетах. Информация об этих пакетах отображается на странице Summary (Итоги), в то время как информацию о конкретном выполняющемся пакете можно получить, щелкнув на папке Running Packages. Остановить выполнение пакета, указанного в этой папке, можно, щелкнув правой кнопкой мыши на пакете и выбрав в появившемся меню элемент Stop (Остановить).

Во-вторых, в систему хранения пакетов можно внести изменения, добавив новые папки и скопировав пакеты из хранилищ одного типа в хранилища другого типа с помощью утилит импорта и экспорта. Существует возможность настроить так называемые логические папки, отображаемые в папке MSDB среды Management

Studio. Для этого необходимо внести изменения в таблицу `sysssispackagefolders` в базе данных `msdb`. Корневыми папками в этой таблице являются те, в столбцах `parentfolderid` которых содержатся пустые значения. Для добавления логических папок в эту таблицу можно добавить значения, памятуя при этом, что столбцы `folderid` и `parentfolderid` содержат ключевые значения, которые используются для определения иерархии папок. Кроме того, в файловой системе, которую отображает среда Management Studio, можно задать папки по умолчанию. Об этом речь идет в этой главе в разделе “Конфигурационный XML-файл”. Об импорте и экспорте пакетов речь пойдет в другом разделе этой главы.

Основные задачи управления, которые можно решить в среде Management Studio:

- создание новых папок в браузере Object Explorer для отображения пакетов, сохраненных в файловой системе или в структуре SQL Server (таблицы `sysssispackages` базы данных `msdb`);
- импортирование пакетов;
- экспортирование пакетов;
- выполнение пакетов;
- удаление пакетов;
- переименование пакетов.

### **Работа с утилитой управления пакетами DTUtil**

Кроме возможностей среды Management Studio для управления пакетами, можно воспользоваться утилитой `DTUtil`. Основное, что необходимо о ней знать, это то, что эта утилита при работе с пакетами позволяет использовать планировщик или пакетные файлы операционной системы Windows. Как и при работе в среде Management Studio, эта утилита позволяет выполнять копирование, удаление, перемещение и даже верификацию наличия на сервере соответствующих пакетов.

Для этой утилиты можно задавать параметры `/SQL`, `/FILE` или `/DTS` для определения расположения пакетов, которыми вы хотите управлять. Параметрами также можно воспользоваться для определения режима работы утилиты при ее использовании. Параметры предваряются символом косой черты `/` или знаком минус `-` и могут задаваться в командной строке в любой последовательности.

Кроме того, эта утилита выдает код завершения, который позволяет понять, что в синтаксисе были обнаружены ошибки или просто было задано ошибочное сочетание параметров. Когда все проходит без ошибок, утилита `DTUtil` возвращает код 0 и выводит сообщение “The operation completed successfully”. Список возвращаемых кодов содержит следующие значения:

- 1 — сбой;
- 4 — невозможно найти пакет;
- 5 — невозможно загрузить пакет;
- 6 — невозможно выполнить команду.

При создании этих команд следует придерживаться следующих правил:

- значения параметров должны задаваться строками и выделяться двойными кавычками или не содержать пробелов;
- символ одинарной кавычки должен заключаться в двойные кавычки;
- для всех значений, кроме паролей, регистр не имеет значения.

Одним из способов применения утилиты DTUtil является регенерация идентификаторов пакетов для пакетов, скопированных из других пакетов. Следует помнить, что после копирования существующего пакета имя и идентификатор нового пакета соответствует имени и идентификатору исходного пакета. Приведем пример использования утилиты DTUtil с ключом /I [D Regenerate] для регенерации идентификаторов пакета. Конечно, для того, чтобы обновить несколько пакетов с помощью однократного выполнения утилиты DTUtil, можно создать пакетный файл, который может делать итерации в заданной папке для всех файлов с расширением .dtsx (пакет) и использует утилиту DTUtil для регенерации идентификаторов пакета. Для этого по подсказке введите следующую команду.

```
for %f in (<FilePath>\*.dtsx) do dtutil.exe /i /File %f
```

При необходимости выполнения этой команды из сценария можно воспользоваться следующим синтаксисом.

```
for %%f in (<FilePath>\*.dtsx) do dtutil.exe /i /File %%f
```

Понимание принципов работы утилиты DTExec добавляет в ваш арсенал управления пакетами очень мощное оружие. А сейчас мы посмотрим, как можно осуществить перемещение пакетов с использованием возможностей экспорта и импорта.

## Экспорт и импорт пакетов

Другая обязанность, которую должен в полной мере освоить администратор, подразумевает способы, которыми вы сможете перемещать пакеты между различными хранилищами и форматами. Функциональность, связанная с импортом и экспортом, позволяет добавлять или копировать пакеты из одного места хранения и формата в другое место хранения или формат. Таким образом, пакеты можно не только добавлять или копировать, можно менять при этом и формат хранения (например, из папок файловой системы в базу данных msdb SQL Server).

Вот как выполняется операция импортирования пакета с помощью служб Integration Services из среды Management Studio.

1. Откройте среду Management Studio и подключитесь к серверу Integration Services.
2. В браузере Object Explorer (Обозреватель объектов) перейдите в папку Stored Packages и выберите вложенную папку, в которую будете импортировать пакет.
3. Щелкните правой кнопкой мыши на нужной папке и выберите в появившемся меню элемент Import Package (Импортировать пакет).
4. В диалоговом окне Import Package (Импортировать пакет) выберите место расположения пакета из следующих вариантов SQL Server, File System или SSIS Package Store.
5. Если в диалоговом окне Import Package выбрано размещение пакета SQL Server, необходимо указать сервер, тип аутентификации, имя пользователя и пароль. Задайте сервер в диалоговом окне Import Package, когда пакет размещен в SSIS Package Store (Хранилище пакетов SSIS).
6. Все еще находясь в диалоговом окне Import Package, щелкните на кнопке Browse (Обзор), расположенной возле текстового поля Package path (Путь к пакету), и выберите импортируемый пакет. В том же диалоговом окне Import Package можно задать новое имя пакета, с которым он будет записан на новом месте, и указать уровень защиты пакета (рис. 6.7).

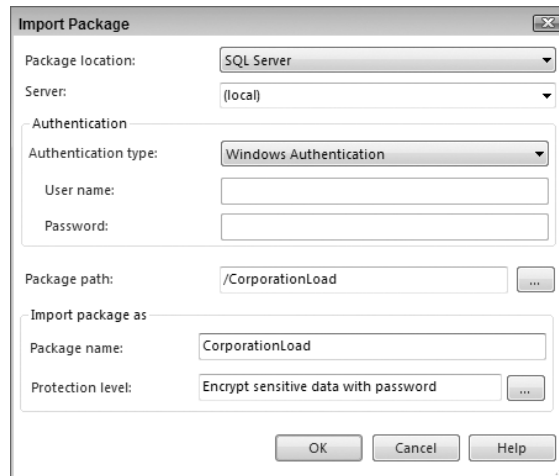


Рис. 6.7. Диалоговое окно Import Package

Операция по экспорту пакетов проводится аналогично импортированию пакетов. Одним существенным различием является то, что для экспорта пакета необходимо щелкнуть правой кнопкой на экспортируемом пакете и выбрать элемент меню Export (Экспорт). Кроме того, следует напомнить, что подобные операции по экспорту-импорту можно выполнить с помощью утилиты командной строки DTUtil.

## Развертывание

После того как пакеты и решения служб Integration Services были разработаны на локальных компьютерах или на специальных серверах, используемых для разработки, их необходимо развернуть для тестирования на промышленном сервере. Процесс развертывания можно начинать, когда пакеты отлажены и доказали свою работоспособность в среде Business Intelligence Development Studio.

Для развертывания пакетов или решений необходимо:

- создание утилиты развертывания пакетов и использование мастера инсталляции пакетов (Package Installer Wizard);
- использование утилит импорта или экспорта пакетов в среде Management Studio;
- сохранение или перемещение копий пакетов в файловой системе;
- работа с утилитой управления пакетами DTUtil.

Часто применение какого-то определенного метода и инструментов развертывания зависит от изменений, которые претерпело решение Integration Service. Например, при модификации только одного пакета из решения, содержащего 30 пакетов, использование утилиты импорта пакетов в среде Management Studio для сохранения и копирования пакетов в файловой системе может быть проще, чем развертывание всего решения с помощью утилиты развертывания пакета и мастера инсталляции пакетов.

Эти четыре метода развертывания еще можно оценить с точки зрения автоматического развертывания или развертывания вручную. Использование утилиты развертывания пакетов в совокупности с мастером инсталляции пакетов можно отне-

сти к категории методов автоматического развертывания, в то время как остальные методы представляют методы развертывания вручную. Посмотрим на детали каждого из этих методов развертывания, начиная с автоматического метода.

### Автоматическое развертывание пакета

Самый популярный метод развертывания пакетов предполагает использование утилиты развертывания пакетов. Эта утилита строит пакеты служб SSIS, производит настройку пакетов и всех файлов поддержки в специальной папке развертывания, вложенной в папку `bin` проекта Integration Services. Кроме того, эта утилита создает специальный выполняемый файл `DTSDeploymentManifest.xml` и размещает его внутри этой папки развертывания. После создания утилиты развертывания для установки пакетов выполняется специальный манифест-файл.

Метод развертывания состоит из двух отдельных шагов. Во-первых, сначала создается утилита развертывания, содержащая все развертываемые файлы. Во-вторых, с помощью мастера инсталляции пакетов выполняется развертывание этих файлов на целевом сервере.

#### Применение утилиты развертывания пакета

Для развертывания нашего решения Integration Services необходимо выполнить следующую последовательность операций с утилитой развертывания пакетов.

1. Откроем среду Business Intelligence Development Studio и решение Integration Services.
2. В браузере Solution Explorer (Обозреватель решений) щелкнем правой кнопкой мыши на решении или проекте (самый верхний узел) и в появившемся меню выберем элемент Properties (Свойства).
3. В диалоговом окне [Solution/Project Name] Property Pages ([Имя решения/проекта] страницы свойств) перейдем в секцию Deployment Utility (Утилита развертывания).
4. Находясь в секции Deployment Utility (Утилита развертывания) диалогового окна Property Pages (Страницы свойств), для параметра CreateDeploymentUtility (СоздатьУтилитуРазвертывания) зададим значение True (рис. 6.8).
5. Дополнительно можно настроить размещение таким образом, чтобы были возможны изменения настройки. Для этого устанавливается значение параметра AllowConfigurationChanges (РазрешитьИзмененияВНастройках) в True. Этот параметр обеспечивает возможность обновления настройки ключевых машинно-зависимых элементов ваших пакетов или элементов, зависящих от среды, каковыми являются имена баз данных или параметр, задающий Initial Catalog (Стартовый каталог)<sup>3</sup>, которые задаются в свойствах диспетчеров подключения базы данных.
6. После этого проект строится как обычно. Создается файл `DTSDeploymentManifest.xml`, и пакеты копируются в папку `bin/Deployment` или в папку, которая была задана параметром `DeploymentOutputPath` (ВыходнойПутьРазвертывания) на странице Project Property Page (Страница свойств проекта) в разделе Deployment Utility (Утилита развертывания).

Вследствие того что эта утилита копирует все файлы решения, вместе с проектом можно развернуть такие дополнительные файлы, как файл `Readme`. Для этого достаточно просто поместить эти файлы в папку `Miscellaneous` проекта Integration Services.

<sup>3</sup> Здесь речь идет о строке подключения к БД. — Примеч. ред.

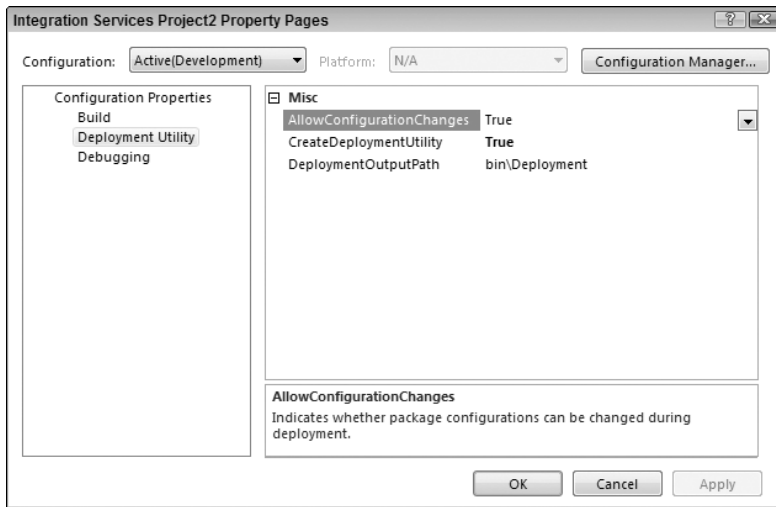


Рис. 6.8. Диалоговое окно Project Property Pages

### Работа с мастером инсталляции пакетов

После создания с помощью утилиты развертывания пакетов файла `SSISDeploymentManifest` наступает черед процедуры инсталляции пакетов. Для этого задействуется мастер инсталляции пакетов. Этот мастер работает как программа `DTSInstall.exe` и копирует пакеты и любую конфигурацию в указанное место.

Использование мастера инсталляции пакетов позволяет получить действительно полезную функциональность, которой трудно добиться при применении ручных методов развертывания. Одним из примеров такой функциональности служит возможность развертывания, как на файловом уровне, так и на SQL-уровне. Следует заметить, что в файловой системе всегда присутствуют файловые зависимости. Другой настолько важной, насколько и полезной возможностью такого развертывания является возможность вносить изменения в настройки, которые будут актуализироваться на целевом сервере развертывания. Это дает возможность, работая с мастером, вносить изменения в такие настройки, как имя сервера.

Вот те шаги, которые необходимо предпринять для успешного развертывания пакетов с помощью утилиты развертывания пакетов.

1. С помощью браузера Windows Explorer найдите место, где был создан файл `SSISDeploymentManifest` (обычно это папка решения или проекта `/bin/Deployment`).
2. После создания файлов в папке `Deployment` скопируйте в папку `Deployment` все файлы, находящиеся в ней, на целевой сервер развертывания.
3. Для запуска мастера инсталляции пакетов (`DTSInstall.exe`) откройте папку `Deployment` и дважды щелкните на файле `SSISDeploymentManifest` на целевом сервере развертывания.
4. На странице `Deploy SSIS Packages` (Развернуть пакеты SSIS) выберите место установки пакетов: в файловую систему или в SQL Server (рис. 6.9). Дополнительно можно задать проверку пакетов после инсталляции.



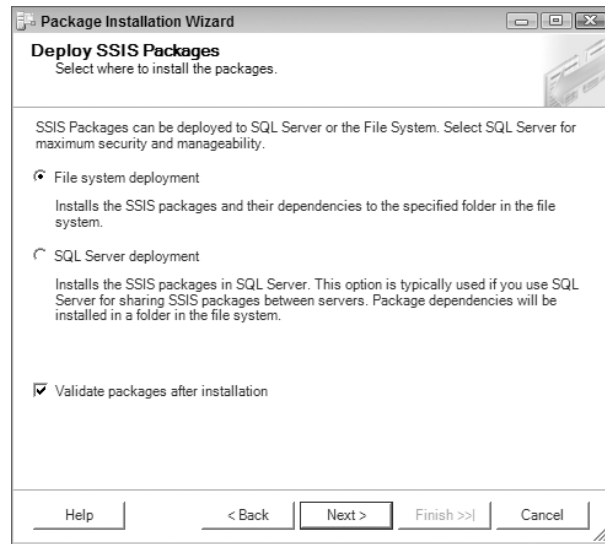


Рис. 6.9. Страница Deploy SSIS Packages

5. На странице Select Installation Folder (Выбор инсталляционной папки) задайте путь к папке развертывания в файловой системе или имя сервера и соответствующие привилегии сервера для развертывания на SQL Server.
6. При развертывании в SQL Server на странице Select Installation Folder задайте для зависимостей, необходимых для хранения в файловой системе пакета, путь к соответствующей папке.
7. Кроме того, если пакет включает настройки и при создании манифеста размещения было задано значение параметра AllowConfigurationChanges, равное True, появится страница Configure Packages (Настройка пакетов), позволяющая обновлять значения конфигурации.
8. Дополнительно в случае задания проверки пакетов на экран будет выведена страница Packages Validation (Проверка пакетов), позволяющая просмотреть результаты проверки.

## Развертывание пакетов вручную

Ранее в качестве методов развертывания вручную упоминались утилиты импорта и экспорта пакетов, сохранение или перемещение пакетов в файловой системе и работа с утилитой управления пакетами DTUtil. Основное различие между этими методами развертывания и работой с утилитой развертывания пакетов и мастером инсталляции пакетов заключается в том, что работа с ручными методами требует глубокого понимания того, какие файлы требуется переносить и где они должны размещаться на целевой машине. Кроме того, они не обладают автоматическими методами перенастройки динамических элементов в пакетах, которые зачастую требуют внесения изменений перед развертыванием на другой машине (такие, как значения Server и Initial String строки подключения).

Несмотря на недостаток “изящества” в этих методах, есть ситуации, когда они очень помогают, даже лучше автоматизированных методов. В качестве примера такой ситуации, когда применение ручных методов просто уместнее, можно упо-

мянуть ситуацию, когда в решении, содержащем 20 или больше пакетов, модифицируется один пакет. При автоматическом методе развертывания потребуется развернуть вновь все решение, в то время когда вручную можно импортировать, экспортировать, сохранить или перемещать только один нужный пакет. Такой подход дает определенные преимущества на некоторых уровнях тестирования, так как это позволяет изолировать влияние изменений одного пакета и снимает необходимость в ненужной проверке пакетов, которые не подвергались изменениям, только потому, что они были размещены повторно поверх старых пакетов.

### Развертывание пакетов посредством импортирования и экспортирования

Ранее в этой главе уже затрагивалась тема использования возможностей импортирования и экспортирования, что позволяет добавлять пакеты или копировать их из одного хранилища и формата хранения в другое хранилище или формат. Одним из самых очевидных применений этих возможностей является развертывание пакетов после завершения их разработки и тестирования.

Одним из интересных результатов этого подхода является использование процедур импортирования/экспортирования для изменения форматов хранения (например, из папок файловой системы в базу данных msdb SQL Server). Такое изменение форматов хранения может использоваться для восстановления данных после разрушения, как резервное средство сохранения решений Integration Services, и сохранения их в различных форматах и хранилищах.

### Развертывание пакетов посредством сохранения и переноса пакетов

Вероятно, самым простейшим способом развертывания пакетов является копирование их из папки bin и размещение на целевых серверах. Этот метод не имеет каких-либо других полезных применений, но может пригодиться для небольших решений Integration Services. Одной существенной возможностью, которая отсутствует у этого метода размещения, является возможность размещения пакетов в SQL Server.

### Развертывание пакетов с помощью утилиты DTUtil

Как и работа со средой Management Studio, работа с утилитой DTUtil позволяет копировать или перемещать пакеты. Как ранее отмечалось в этой главе, преимущество использования утилиты DTUtil заключается в возможности планирования запуска работы команд этой утилиты на будущее. Поэтому, пользуясь этими возможностями, можно запланировать размещение ваших пакетов на другом сервере, просто используя команды копирования или перемещения DTUtil для переноса модифицированных пакетов на целевой сервер.

В следующем примере показано, как команда копирования утилиты DTUtil может использоваться при развертывании.

```
dtutil /DTS srcPackage.dtsx /COPY SQL;destPackage
```

## Выполнение и планирование

Итак, мы рассмотрели методы создания, управления и развертывания решений Integration Services. Теперь уделим внимание методам выполнения и планирования выполнения этих решений. Как вы уже видели, задачи администрирования пакетов и решений могут решаться с применением самых разнообразных средств. В частности, выполнять пакеты можно из:

- ❑ средства Business Intelligence Development Studio;
- ❑ мастера импорта и экспорта SQL Server (при работе в среде Management Studio);

- ❑ утилиты выполнения пакетов `DTEXEC`, работающей из командной строки;
- ❑ утилиты выполнения пакетов `DTEXECUI`, работающей из командной строки;
- ❑ планировщика заданий SQL Server Agent.

Выбор используемого инструмента часто зависит от таких факторов, как цикл жизни пакета, с которым проводится работа в данный момент. Например, средство SSIS Designer в среде BIDS будет логичным выбором для выполнения пакета на этапе его разработки благодаря возможностям, разработанным для оказания помощи при разработке (таким, как визуализация выполнения пакета посредством изменения фонового цвета задач).

А теперь, зная какие возможности у нас имеются для выполнения или планирования выполнения пакетов, посмотрим, каким образом можно использовать каждую из них.

### **Запуск пакетов из среды Business Intelligence Development Studio**

Вероятнее всего, что первые пробы запуска пакетов происходят в среде BIDS, так как эта среда разработки чаще всего используется при создании решений Integration Services. В среде BIDS для этого достаточно щелкнуть правой кнопкой мыши на пакете, а затем просто выбрать из появившегося меню элемент `Execute Package` (Выполнить пакет) или просто нажать функциональную клавишу `<F5>` (или кнопку `Start` (Пуск) в строке меню).

### **Запуск пакетов с помощью мастера импорта и экспорта SQL Server**

При работе с мастером импорта и экспорта в среде Management Studio у вас имеется возможность немедленного выполнения пакета. Это дает возможность одновременного перемещения и выполнения пакетов.

### **Запуск пакетов с помощью утилиты DTEXEC**

Главным назначением утилиты `DTEXEC` является предоставление возможности запуска пакетов из командной строки, из сценария или с использованием утилиты планирования. Из командной строки можно задавать любую конфигурируемую возможность. Кроме того, пакеты можно загружать и запускать из SQL Server, служб SSIS Service и файловой системы.

При написании этих команд следует придерживаться следующих дополнительных синтаксических правил:

- ❑ все параметры обязательно предваряются косой чертой `/` или символом минуса `-`;
- ❑ если аргумент имеет пробел, этот аргумент заключается в кавычки;
- ❑ значения, содержащие одинарные кавычки внутри строк, заключенных в кавычки, берутся в двойные кавычки.

Общее синтаксическое правило строки `DTEXEC` можно записать следующим образом.

```
Dtexec /option value
```

Вот пример запуска пакета `CaptureDataLineage.dtsx`.

```
DTEXEC /FILE "C:\Program Files\Microsoft SQL Server\100\Samples\Integration
Services\Package Samples\CaptureDataLineage
Sample\CaptureDataLineage\CaptureDataLineage.dtsx" /CONNECTION
"(local).AdventureWorks2008";"\Data Source=(local);Initial
Catalog=AdventureWorks2008;Provider=SQLNCLI.1;Integrated Security=SSPI;Auto
Translate=False;" /" /MAXCONCURRENT " -1 " /CHECKPOINTING OFF /REPORTING
EWCDI
```

При выполнении пакета с помощью утилиты DTEXEC она возвращает код завершения. Эти значения могут быть такими:

- 0 — успешное завершение;
- 1 — сбой;
- 3 — выполнение отменено пользователем;
- 4 — пакет не найден;
- 5 — невозможно запустить пакет;
- 6 — синтаксическая ошибка.

Предусмотрено множество параметров, позволяющих изменять ход выполнения пакета. Некоторые примеры содержат параметр /Decrypt, который устанавливает пароль пакета, используемого для обеспечения безопасности пакета, или параметр /Set, который используется для присвоения значений переменным SSIS во время выполнения. Эти параметры обрабатываются в порядке их следования в командной строке. При использовании команд /Set и /ConfigFile значения также обрабатываются в порядке их следования. Кроме того, заметим, что регистр представления параметров и аргументов (за исключением паролей) значения не имеют.

### Запуск пакетов с помощью утилиты DTEXECUI

Настройку различных параметров, необходимых для работы пакетов, можно выполнять с помощью графического эквивалента утилиты DTEXEC: утилиты DTEXECUI. Пользуясь мастером, с помощью которого эта утилита собирает данные о выполнении пакета, вы сможете лучше понять назначение различных параметров и познакомиться с синтаксисом, необходимым для запуска пакета.

Утилита DTEXECUI запускается двойным щелчком на файле, имеющем расширение .dtsx. Сначала выбираются типы параметров, необходимые для запуска пакета, находящиеся с левой стороны страницы утилиты, а затем эти параметры настраиваются в основной части страницы. После того как все установки сделаны, вы попадете на последнюю страницу, на которую выводится командная строка, необходимая для выполнения пакета с выбранными нами параметрами (рис. 6.10).

После завершения работы со всем страницами и проверки запускаемой командной строки щелкните на кнопке Execute (Выполнить). Таким образом, с помощью утилиты DTEXECUI механизму Integration Services будет передана командная строка. Необходимо быть предельно осторожным при работе с этой утилитой на 64-битовой платформе, так как эта утилита предназначена для работы на платформе Win32, а не на Win64. Таким образом, для 64-битовых платформ необходимо пользоваться 64-битовой версией утилиты DTEXEC или работать только с компонентом агентом SQL Server.

Главной причиной нашего знакомства с обеими утилитами DTEXEC и DTEXECUI является то, что они хорошо зарекомендовали себя при тестировании пакетов и проверке правильности командной строки, работа которой запланирована с помощью SQL Server Agent. Теперь посмотрим, каким образом SQL Server Agent может провести запланированное выполнение наших пакетов Integration Services.

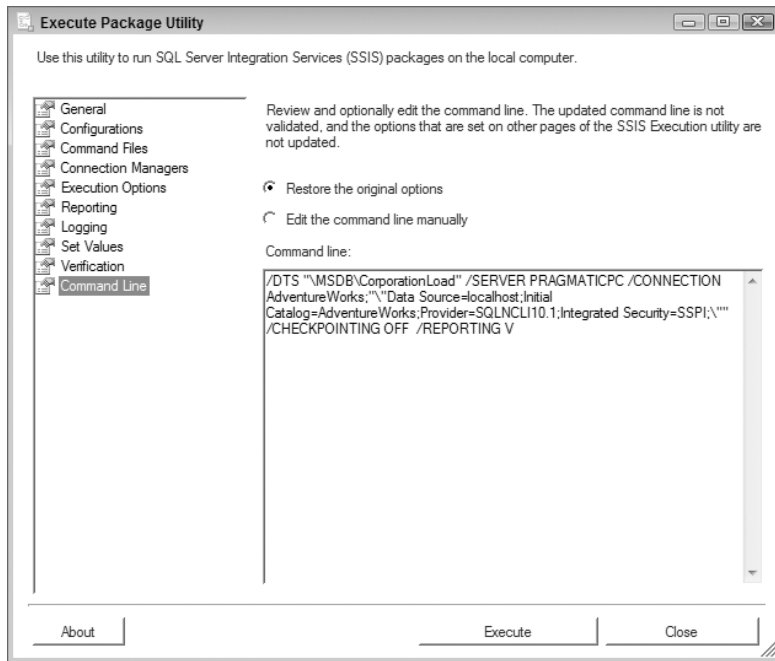


Рис. 6.10. Страница Execute Package Utility

## Планирование запуска заданий с помощью SQL Server Agent

Нет никаких сомнений, что в дальнейшем может потребоваться автоматизировать выполнение ваших пакетов Integration Services. В то время как для этих целей можно пользоваться любым известным средством, имеющимся в Windows, рассмотрим, чем в этом вопросе может помочь компонент SQL Server Agent.

Попробовать это можно, создав задание, включив в него затем по крайней мере один шаг в виде пакета SQL Server Integration Services. Можно настроить и другие параметры задания. Одним из таких параметров является возможность рассылки уведомлений по электронной почте, когда работа задания завершается, завершается успешно или завершается со сбоем. Другим параметром задания является возможность настройки рассылки уведомлений по критическим событиям SQL Server, критическим событиям по рабочим характеристикам системы, критическим событиям по WMI-событиям.

Вот каким образом можно активизировать SQL Server Agent на выполнение пакета.

1. Находясь в среде Management Studio, подключитесь к SQL Server.
2. В браузере Object Explorer (Обозреватель объектов) щелкните на элементе SQL Server Agent (Агент SQL Server).
3. В разделе SQL Server Agent браузера Object Explorer щелкните правой кнопкой мыши на папке Jobs (Задания) и в появившемся меню выберите элемент New Job (Создать новые задания).
4. На странице General (Общее) диалогового окна New Job (Новое задание) укажите имя, владельца, категорию и описание задания.

5. На странице Steps (Шаги) диалогового окна New Job (Новое задание) щелкните на кнопке New (Создать), расположенной в нижней части страницы.
6. В диалоговом окне New Job Step (Создать Новое задание) укажите имя шага и выберите тип SQL Server Integration Services Packages (Пакеты SQL Server Integration Services). Дополнительно, в других разделах, посвященных настройкам служб SSIS, размещенных как вкладки, настройте информацию, необходимую для запуска вашего пакета. Этот раздел SSIS по своей функциональности почти идентичен параметрам, которые указываются при использовании утилиты DTExecUI (рис. 6.11). У вас задан источник пакетов: SQL Server, file system (Файловая система) или SSIS Package Store (Хранилище пакетов SSIS). Затем следует указать пакет, работу которого требуется запланировать. На вкладке Command Line (Командная строка) можно просмотреть детализированную командную строку, которую сформирует SQL Server Agent для выполнения пакета. Полученную таким образом командную строку можно сравнить со значениями командной строки, сгенерированной утилитой DTExecUI во время тестирования выполнения пакета.

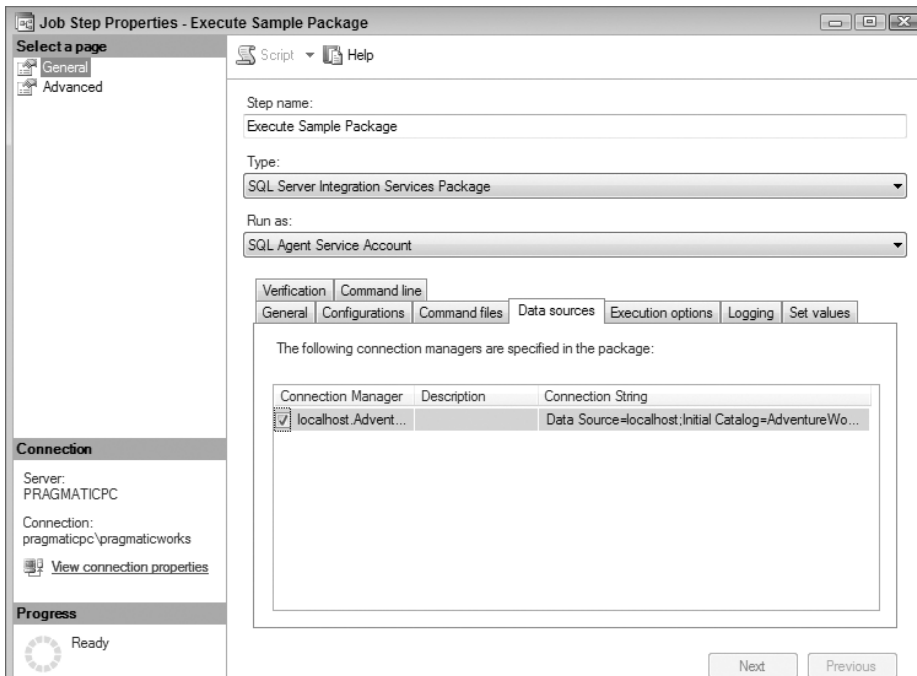


Рис. 6.11. Страница New Job Step

7. На странице Advanced (Дополнительно) диалогового окна New Job Step (Создать шаг задания) можно задать действия, которые будут выполняться, если выполнение шага завершится успешно, количество попыток повторного выполнения, интервал повторений и выполняемые шаги в случае сбоя в работе шага. После завершения настройки шага щелкните на кнопке ОК, появится страница Step (Шаг) диалогового окна New Job (Создать задание), показывающая новый шаг. После добавления нескольких шагов порядок выполнения этих шагов можно изменить на этой странице.

8. Графики выполнения, критические события, уведомления и целевые серверы можно поменять даже после подтверждения настройки шага в диалоговом окне New Job.

## Вопросы безопасности служб Integration Services

Мы рассмотрели наиболее важные вопросы администрирования пакетов, включая создание, управление, развертывание и выполнение решений Integration Services. Кроме того, нами были рассмотрены основные задачи администрирования служб Integration Services. А теперь перейдем к детальному обсуждению вопросов обеспечения безопасности служб Integration Services.

### Обзор

Для обеспечения целостности проекта пакетов, а также администрирования и выполнения пакетов служб Integration Services, как и все остальные службы, SQL Server использует уровни безопасности, опирающиеся на разнообразные механизмы. Безопасность служб SSIS обеспечивается как на стороне клиента, так и на стороне сервера, обеспечивая следующие возможности:

- уровни защиты пакетов для шифрования или удаления важных данных из пакета;
- уровни защиты пакетов с паролями для защиты всей информации или только важной информации;
- ограничение доступа к пакетам с помощью ролей;
- блокировка размещения файлов там, где могут храниться пакеты;
- присвоение пакетам сертификатов.

В общем случае важной информацией в пакетах считается такая информация, как пароли и строки соединения. Вы не сможете определить, какую информацию считать важной для служб SSIS, а какая таковой не является, пока это не потребуются сделать в прикладной задаче.

Службы Integration Services определяют как важную следующую информацию:

- пароль, указываемый в строке подключения (важная) или вся строка соединения (целиком);
- узлы XML, генерированные задачей, помеченные как важные для SSIS;
- переменные, помеченные как важные для SSIS.

### Обеспечение безопасности пакетов

Два основных метода обеспечения безопасности пакетов служб Integration Services включают установку уровней защиты пакетов и настройку соответствующих ролей служб SSIS. Рассмотрим оба этих метода.

#### Уровни защиты пакетов

В своей работе многие организации используют секретную информацию, кроме того, необходимо контролировать места ее размещения внутри самой организации. Пакеты могут снабжаться паролем, который бы смог предотвратить его выполнение лицом, не имеющим права на это.

Эти проблемы безопасности решаются службами Integration Services в результате применения уровней защиты пакетов. Во-первых, необходимо обеспечить, чтобы такая важная информация о том, где хранится такая секретная информация, как строки подключения, управлялась с помощью уровней защиты пакетов `EncryptSensitive`. Во-вторых, можно осуществлять управление теми, кто открывает или выполняет пакет с использованием паролей пакетов `EncryptAll`.

В нашем распоряжении в службах Integration Services имеются следующие уровни защиты пакетов:

- не сохранять секретную информацию;
- зашифровать (все/секретную) с помощью пользовательского ключа;
- зашифровать (все/секретную) с помощью пароля;
- возложить шифрование на сервер (только хранилище SQL).

Уровни защиты пакетов присваиваются в среде Business Intelligence Development Studio. Эти уровни защиты пакетов можно обновить после развертывания пакетов или во время импорта или экспорта пакетов с помощью среды Management Studio. Кроме того, изменить уровни защиты пакетов можно во время копирования пакетов из средства BIDS в любые другие места, в которых хранятся пакеты. Это требует выработки компромиссных решений между группой разработки и группой администрирования, так как разработчики могут настроить эти уровни для удовлетворения своих требований, учитывающих скорость разработки, а администраторы должны отслеживать эти уровни, чтобы поддерживать производственные стандарты безопасности.

## Роли базы данных служб Integration Services

При развертывании пакетов на SQL Server (они попадают в базу данных `msdb`) необходимо обеспечить защиту этих пакетов уже в самой базе данных. Как и в обычных базах данных, эта безопасность обеспечивается посредством ролей базы данных. Существует три фиксированные роли базы данных, которые можно применить и к базе данных `msdb`, для того чтобы управлять доступом к пакетам. Это роли `db_dtsadmin`, `db_dtsltduser` и `db_dtsoperator`.

Эти роли присваиваются пакетам в среде Management Studio, причем эти присвоения сохраняются в базе данных `msdb` в столбцах `readerrole`, `writerrole` и `ownersid` таблицы `sysssispackages`. Как видно из имен столбцов, вы можете просмотреть роли, имеющие право доступа к определенному пакету, просмотрев значение столбца `readerrole`, роли, имеющие право записи в определенный пакет, просмотрев значение в столбце `writerrole`, и роль, создавшую пакет, — в столбце `ownersid`.

Вот как присваиваются роли чтения и записи пакетам.

1. Откройте среду Management Studio и подключитесь к серверу Integration Services.
2. В браузере Object Explorer щелкните на папке Stored Packages (Хранящиеся пакеты) и щелкните на вложенной в нее папке для того, чтобы иметь возможность присвоения ролей.
3. Для присвоения ролей щелкните правой кнопкой мыши на вложенной папке.
4. В диалоговом окне Packages Roles (Роли пакетов) выберите роль читателя в списке Reader Role (Роль читателя) и роль писателя в списке Writer Role (Роль писателя).

Можно также создать свои роли, если операции чтения и записи существующих ролей не соответствуют вашим задачам обеспечения безопасности. Для определе-



ния таких ролей следует подключиться к экземпляру SQL Server и открыть узел `roles` (роли) в базе данных `msdb`. В узле `roles` щелкните правой кнопкой мыши на элементе `database roles` (роли БД) и выберите `New Database Role` (Создать роль БД). После добавления новой роли в базу данных `msdb`, перед тем как можно будет использовать эту роль, необходимо перезапустить службу `MSSQLSERVER`.

Эти роли позволяют настраивать в таблице `sysssispackages` базы данных `msdb` параметры безопасности пакетов с точки зрения операций записи и чтения для определенных пакетов. Применяв этот уровень безопасности, вы обеспечиваете безопасность на уровне сервера, базы данных и таблицы. Снова следует напомнить, что вопросы безопасности, которые мы здесь рассматриваем, касаются только хранения пакетов в SQL Server (конкретно в базе данных `msdb`). Далее мы рассмотрим возможности обеспечения безопасности для пакетов, хранимых вне SQL Server.

## Сохранение пакетов

Созданные пакеты хранятся в файловой системе в виде XML-файлов с расширением `.dtsx`. В разделе, посвященном размещению, было показано перемещение этих файлов на нужный сервер Integration Services с помощью утилит экспорта и импорта, утилиты `DTUtil` или мастера развертывания пакетов (`Package Installer Wizard`).

Если не предусмотрено сохранение пакетов и связанных с ними файлов (например, конфигурационных файлов) в SQL Server (база данных `msdb`), для обеспечения безопасности этих пакетов необходимо применить систему управления доступом к файловой системе NTFS.

А теперь посмотрим, как можно ограничить доступ к компьютерам, на которых работает служба SQL Server, так как они могут хранить пакеты как локально, так и удаленно.

## Запуск пакетов

После отработки вопросов защиты пакетов необходимо ограничить доступ к работающим в настоящее время на вашей системе пакетам. В частности, необходимо управлять правами тех, кто может просматривать работающие пакеты, и тех, кто может остановить работу выполняемых пакетов.

Служба SQL Server используется в среде Management Studio для определения списка выполняющихся в настоящее время пакетов. Только члены группы администраторов Windows имеют право на выполнение этих действий. Эта служба также может использоваться для перечисления папок, включая также и локальные папки. Эти папки представляют собой место, куда сохраняются ваши пакеты, поэтому очень важно, чтобы вы надлежащим образом контролировали доступ к компьютерам, на которых эта служба работает. Важно также и то, чтобы для хранения пакетов использовались папки формата NTFS, позволяющего управлять доступом.

## Ресурсы пакетов

После обсуждения проблем обеспечения безопасности пакетов, сохраненных в SQL Server или в файловой системе, и управления доступа к работающим пакетам, обратим внимание на различные ресурсы, используемые пакетами.

Ресурсы, используемые пакетами, включают:

- конфигурационные файлы;
- файлы с контрольными точками;
- файлы журналов.

Безопасность этих ресурсов необходимо обеспечивать, так как они могут содержать важную информацию. Например, конфигурационный файл часто содержит учетные данные и информацию о паролях источников данных, которые используются пакетами для получения данных. Даже если они не содержат важной информации, все равно необходимо позаботиться о том, чтобы во время своей работы пакеты смогли найти их и использовать в своей работе. Другим примером являются файлы контрольных точек, используемые для обеспечения возможности повторного старта пакетов. Эти файлы контрольных точек хранят в себе текущую информацию о состоянии при выполнении пакетов, включая текущее значение любой переменной, определенной в пакетах. Часто мы используем переменные для сохранения информации, которая может показаться важной и, как результат, потребуется гарантировать, чтобы доступ к этой информации имел только строго определенный персонал. В любом случае можно хранить эти данные в SQL Server и для обеспечения их безопасности пользоваться ролями базы данных служб Integration Services. В противном случае для обеспечения безопасности информации вам потребуется привлечь механизм управления правами доступа, предусмотренный в файловой системе NTFS.

## Цифровые подписи

В качестве последней линии обороны в деле обеспечения успешного использования пакетов во время их выполнения можно вспомнить цифровые подписи этих пакетов. После подписания пакета его можно выполнить с помощью утилиты или программы, но при этом во время выполнения программы производится проверка подписи, и в случае несоответствия подписи должно выводиться предупреждение или выполнение пакета будет просто блокироваться. Для того чтобы такая проверка производилась, необходимо свойству `CheckSignatureOnLoad` (ПроверитьПодписьПриЗагрузке) задать значение `True`.

## Резюме

В этой главе мы рассмотрели множество различных административных функций, имеющих отношение к службам Integration Services. После общего введения в Integration Services были рассмотрены различные задачи администрирования служб Integration Services, такие как управление службами Integration Services, включая настройки, журналы регистрации событий и мониторинг активности. Затем были рассмотрены различные административные задачи, направленные на работу с пакетами Integration Services, включая создание, управление, выполнение и развертывание. Наконец, мы посмотрели, каким образом можно обеспечить безопасность различных компонентов служб Integration Services. В главе 7, “Администрирование и настройка служб Analysis Services” будут рассмотрены аналогичные административные задачи, относящиеся к службе Analysis Services.