





**WebSphere Adapter for Oracle E-Business Suite - Руководство
пользователя
Версия 6, выпуск 2**

Примечание

Перед использованием данной информации и описанного в ней продукта обязательно ознакомьтесь с разделом “Примечания” на стр. 223.

Декабрь 2008

Это издание относится к продукту IBM WebSphere Adapter for Oracle E-Business Suite, версии 6, выпуска 2, варианта 0, а также более поздних, если не оговорено противное.

Комментарии об этом документе можно отправить по адресу <mailto://doc-comments@us.ibm.com>. Будем рады получить ваши отзывы.

IBM может использовать или распространять переданную вами информацию любым способом на свое усмотрение, без возникновения каких-либо обязательств перед вами.

© Copyright International Business Machines Corporation 2006, 2008. Все права защищены. US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

© Copyright International Business Machines Corporation 2006, 2008.

Содержание

Глава 1. Обзор WebSphere Adapter for Oracle E-Business Suite 1

Требования к аппаратному и программному обеспечению	1
Технический обзор WebSphere Adapter for Oracle E-Business Suite	2
Обработка исходящих запросов	4
Обработка входящих событий	17
Бизнес-объекты	22
Хранимые процедуры - Обзор	33
Бизнес-объект хранимой процедуры - Обзор	34
Хранимые процедуры, используемые вместо операций или в дополнение к ним	35
Хранимые функции - Обзор	40
Бизнес-объект запроса - Обзор	40
ESW (мастер внешних служб)	41

Глава 2. Планирование реализации адаптера. 43

Перед тем, как начать	43
Поддержка защиты конфиденциальных данных в файлах протоколов и трассировки	43
Защита	44
Идентификация пользователей	44
Варианты развертывания	45
WebSphere Adapters в кластерных средах	48
Поддержка кэширования подготовленных операторов	49

Глава 3. Примеры и учебники 51

Глава 4. Настройка модуля для развертывания 53

Обзор процедуры настройки модуля	53
Создание хранилища событий	55
Создание псевдонимов идентификации	56
Создание проекта	57
Добавление внешних зависимостей программного обеспечения	58
Установка свойств соединения для ESW (мастер внешних служб)	59
Настройка модуля для обработки исходящих запросов	60
Поиск объектов базы данных	60
Выбор и настройка бизнес-объектов	64
Установка глобальных свойств для операций	76
Установка свойств развертывания и генерация службы	78
Завершение настройки	84
Настройка модуля для обработки входящих событий	86
Поиск объектов базы данных	86
Выбор и настройка бизнес-объектов	89
Установка глобальных свойств для операций	95
Установка свойств развертывания и генерация службы	97

Завершение настройки	106
--------------------------------	-----

Глава 5. Изменение свойств спецификации взаимодействия с помощью редактора сборки 109

Глава 6. Развертывание модуля 111

Среды развертывания	111
Развертывание модуля для тестирования	111
Добавление внешних зависимостей	111
Генерация и связывание целевого компонента для тестирования входящей обработки	112
Подготовка к тестированию исходящих операций	114
Добавление модуля на сервер	115
Тестирование модуля для исходящей обработки с помощью клиента тестирования	116
Развертывание модуля в рабочей среде	116
Добавление зависимостей внешнего программного обеспечения на сервер	116
Установка файла RAR (только для модулей, использующих автономные адаптеры)	117
Экспорт модуля в виде файла EAR	118
Установка файла EAR	119

Глава 7. Администрирование модуля адаптера 121

Изменение свойств конфигурации для встроенных адаптеров	121
Настройка свойств адаптера ресурсов для встроенных адаптеров	121
Настройка свойств фабрики управляемых соединений (J2C) для встроенных адаптеров	123
Настройка свойств спецификации активации для встроенных адаптеров	125
Изменение свойств конфигурации для автономных адаптеров	127
Настройка свойств адаптера ресурсов для автономных адаптеров	127
Настройка свойств фабрики управляемых соединений (J2C) для автономных адаптеров	128
Настройка свойств спецификации активации для автономных адаптеров	130
Запуск приложения, использующего адаптер	131
Остановка приложения, использующего адаптер	131
Отслеживание производительности с помощью системы сбора статистики	132
Настройка системы сбора статистики	132
Просмотр статистики производительности	134
Включение трассировки в инфраструктуре событий общего формата (CEI)	136

Глава 8. Устранение неполадок и поддержка 137

Поддержка Анализатор протоколов и данных трассировки	137
Настройка протоколов и трассировки	138
Настройка свойств протокола	138
Изменение имен файлов протокола и трассировки	140
Поддержка FFDC	141
Сбой бизнеса	141
XAResourceNotAvailableException	142
Ресурсы для самостоятельного устранения неполадок	143
Устранение обычных неполадок	144

Глава 9. Справочная информация 151

Информация о бизнес-объекте	151
Атрибуты бизнес-объектов	151
Информация уровня приложения для атрибутов	153
Информация уровня приложения для бизнес-объектов	161
Соглашение об именах	164
Свойства конфигурации исходящего соединения	165
Свойства соединения для мастера	167

Свойства адаптера ресурсов.	172
Свойства фабрики управляемых соединений	176
Свойства спецификации взаимодействия	186
Свойства конфигурации входящего соединения	187
Свойства соединения для мастера	188
Свойства адаптера ресурсов.	193
Свойства спецификации активации	198
Глобализация	216
Глобализация и преобразование двунаправленного текста.	217
Свойства, доступные для преобразования двунаправленных данных	219
Сообщения адаптера	221
Связанная информация	221

Примечания. 223

Информация о программном интерфейсе	225
Товарные и сервисные знаки	225

Индекс 227

Глава 1. Обзор WebSphere Adapter for Oracle E-Business Suite

WebSphere Adapter for Oracle E-Business Suite позволяет создавать интегрированные приложения, обеспечивающие обмен информацией с базой данных. Применение адаптера позволяет приложению отправлять запросы к базе данных, а также получать из нее события, причем во многих случаях код SQL для этого не требуется.

Адаптер позволяет установить двустороннюю связь между приложением, работающим в WebSphere Process Server или WebSphere Enterprise Service Bus, и базой данных. Применение адаптера позволяет приложению отправлять запросы на чтение, создание, изменение или удаление данных в базе данных, причем во многих случаях для этого не требуется писать код SQL. Для обработки запросов, поступающих от приложения, адаптер обновляет таблицы базы данных с помощью запросов SQL или хранимых процедур. Приложение также может принимать события от базы данных, например, уведомления об обновлении конкретных таблиц. События, поступающие в результате изменений в базе данных, адаптер доставляет в приложение для обработки. С помощью уведомления о событии обновления в базе данных могут автоматически распространяться в другие приложения. Объединение обработки событий, выполняемой WebSphere Adapter for Oracle E-Business Suite, с другим адаптером позволяет автоматически распространять обновления в такие приложения, как Siebel, PeopleSoft и Oracle.

Адаптер предоставляет стандартный интерфейс, который интегрируется с различными версиями программного обеспечения базы данных Oracle; он может работать с сервером базы данных Oracle, поскольку этот сервер использует драйвер Java Database Connectivity (JDBC), поддерживающий спецификацию JDBC 2.0 и выше. Для обмена данными между приложением и базой данных адаптер использует бизнес-объекты, поэтому приложению не требуется использовать API JDBC. *Бизнес-объекты* - это контейнеры для данных приложения, которые представляют бизнес-функции или элементы, такие как таблица базы данных или результат запроса SQL. Адаптер распознает формат данных, предоставляемый приложением, и может обработать эти данные, выполнить операцию и отправить обратно результаты в этом формате.

Требования к аппаратному и программному обеспечению

Требования к аппаратному и программному обеспечению WebSphere Adapters приведены на Web-сайте службы поддержки IBM.

Требования WebSphere Adapters к аппаратному и программному обеспечению описаны в разделе <http://www.ibm.com/support/docview.wss?uid=swg27006249>.

Дополнительная информация

По следующим ссылкам предоставлена дополнительная информация по настройке и развертыванию адаптера:

- Матрица совместимости для WebSphere Business Integration Adapters и WebSphere Adapters определяет поддерживаемые версии требуемого программного обеспечения для адаптера. Для просмотра этого документа перейдите на страницу

поддержки WebSphere Adapters и выберите **Матрица совместимости** под заголовком **Связанная информация** в разделе **Дополнительная поддержка**: <http://www.ibm.com/software/integration/wbiadapters/support/>.

- Технические комментарии для WebSphere Adapters содержат описание способов обхода неполадок, а также дополнительную информацию, которая отсутствует в документации по продукту. Для просмотра технических замечаний перейдите на следующую Web-страницу, выберите адаптер из списка **Категория продукта** и щелкните на значке поиска: <http://www.ibm.com/support/search.wss?tc=SSMKUK&rs=695&rank=8&dc=DB520+D800+D900+DA900+DA800+DB560&dtm>.

Технический обзор WebSphere Adapter for Oracle E-Business Suite

Адаптер поддерживает интеграцию баз данных, доступных через программный интерфейс приложений (API) JDBC, с приложениями, выполняющимися в WebSphere Process Server или WebSphere Enterprise Service Bus. Адаптер предоставляет исходящую и входящую обработку в Java 2 Platform, Enterprise Edition (J2EE) Connector Architecture (JCA) и интеграцию с компонентами Service Component Architecture (SCA).

Исходящая обработка позволяет приложению получить доступ или изменить данные в базе данных. Адаптер преобразует запрос из приложения в исходящую операцию, которую он выполняет для создания, извлечения, изменения или удаления данных в базе данных, или выполняет хранящуюся в базе данных программу. Обработка этих запросов приводит к созданию, извлечению, изменению или удалению строк в соответствующих таблицах базы данных. Адаптер также позволяет выполнить хранимые процедуры или хранимые функции, которые определены в базе данных, и определенные пользователем операторы SELECT, INSERT, UPDATE и DELETE. Можно использовать адаптер для интеграции нескольких приложений с одной базой данных.

рис. 1 на стр. 3 представляет обзор потока исходящей обработки. Приложение, выполняющееся в WebSphere Process Server или WebSphere Enterprise Service Bus, вызывает службу в исходящем модуле, которая отправляет запрос в адаптер для обработки одного или нескольких бизнес-объектов. Адаптер использует API JDBC для подключения к серверу базы данных и доступа к таблицам и другим объектам базы данных.

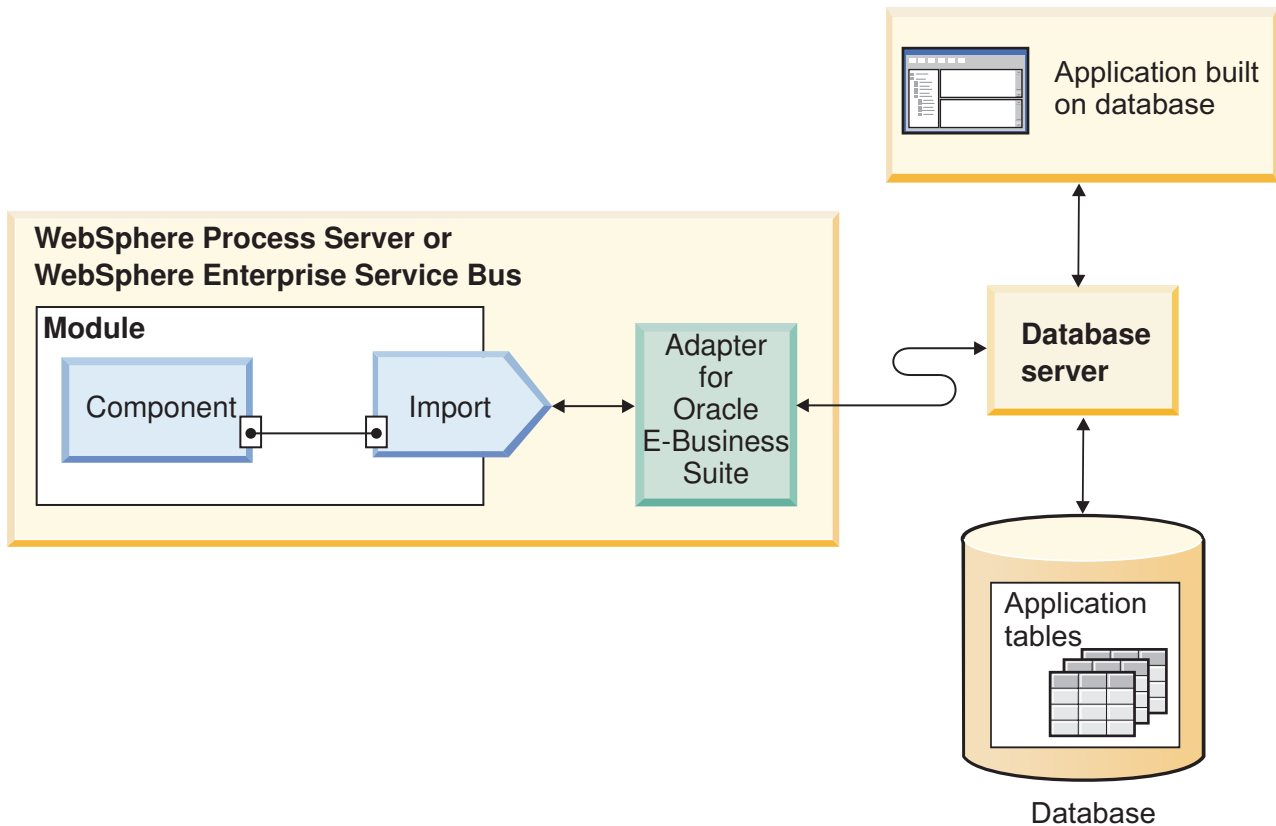


Рисунок 1. Обработка исходящих запросов

Входящая обработка позволяет приложению получать уведомления при изменении объектов в базе данных. Например, приложение может быть извещено, когда строки создаются, изменяются или удаляются в выбранных таблицах.

рис. 2 на стр. 4 представляет обзор потока входящей обработки. Приложение базы данных изменяет таблицы в базе данных. Изменение вызывает триггер или другой автоматический механизм для обновления хранилища событий с информацией об изменении. Периодически адаптер опрашивает хранилище событий, извлекает и обрабатывает события, а затем доставляет их в экспорт модуля, который является частью приложения, которое выполняется в WebSphere Process Server или WebSphere Enterprise Service Bus.

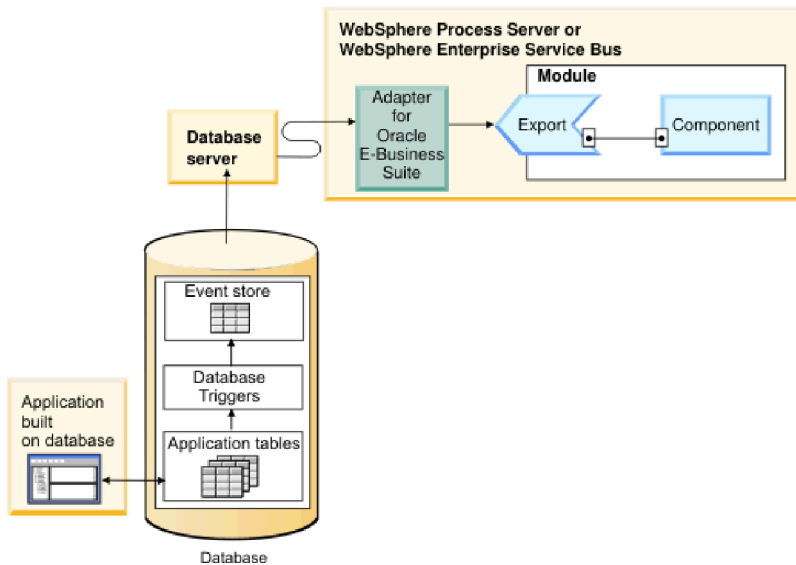


Рисунок 2. Обработка входящих событий

Адаптер может обрабатывать события одним из следующих способов:

- Стандартная обработка события с помощью хранилища событий, которое заполняется приложением базы данных
- Пользовательская обработка события с помощью определенного пользователем запроса к базе данных

В процессе *стандартной обработки события*, когда данные изменяются в таблицах базы данных, соответствующие события вставляются в таблицу базы данных, называемую хранилищем событий, вместе с существенной информацией, такой как значения ключей. Для того чтобы собрать измененные данные, в соответствующих таблицах создаются триггеры, или используются другие методы, такие как Сбор измененных данных Oracle, предоставленный для баз данных Oracle. Адаптер опрашивает хранилище событий и извлекает пакет событий. События могут быть отфильтрованы по типу бизнес-объекта, системному времени и ИД коннектора. Адаптер использует каждое событие для составления схемы бизнеса или бизнес-объекта, который содержит бизнес-объект, измененный этим событием. Бизнес-объект или схема бизнеса затем отправляются в объекты экспорта, которые настроены для получения определенного бизнес-объекта.

В процессе *пользовательской обработки события* адаптер выполняет запрос, указанный пользователем в виде стандартного оператора SQL, хранимой процедуры или хранимой функции. Любое из этих действий возвращает набор результатов для данных, возвращаемых запросом. Каждая строка набора результатов соответствует строке хранилища событий. Адаптер составляет бизнес-объект для каждого события и доставляет его в объекты экспорта (также называемые конечными точками), настроенные для (или подписанные на) определенный бизнес-объект.

И для стандартной, и для пользовательской обработки события можно указать частоту опроса событий адаптером и количество событий, извлекаемых за каждый период опроса.

Обработка исходящих запросов

Когда компоненту приложения необходимо обратиться к базе данных с запросом на предмет существования записи или с запросом на извлечение или изменение данных в

базе данных, адаптер выполняет функции коннектора между компонентом приложения и базой данных. Адаптер предоставляет набор стандартных исходящих операций, которые обрабатывают либо бизнес-объекты записи нового значения, либо бизнес-объекты поправок. Кроме того, при обработке исходящих запросов адаптер поддерживает как локальные, так и распределенные (XA) транзакции.

В модели бизнес-объектов адаптера применяются два вида бизнес-объектов для внесения изменений: запись нового значения и поправка. Бизнес-объект *записи нового значения* содержит полное состояние бизнес-объекта после внесения всех требуемых изменений. Бизнес-объект *поправки* содержит только значения ключей и данные, которые должны быть изменены. Бизнес-объекты поправок используются только в операциях, обновляющих бизнес-объекты.

Поддерживаемые операции

В таблице Табл. 1 перечислены исходящие операции, поддерживаемые для каждого типа бизнес-объекта, с указанием, поддерживает ли данная операция обработку записей нового значения или поправок.

Таблица 1. Исходящие операции, поддерживаемые для разных типов бизнес-объектов

Поддерживаемый бизнес-объект	Операция	Поддержка записи нового значения	Поддержка поправки
Таблицы Представления Синонимы - Псевдонимы	Create	Да	Неприменима
	Update	Да	Неприменима
	Delete	Да	Неприменима
	Retrieve	Неприменима	Неприменима
	RetrieveAll	Неприменима	Неприменима
	ApplyChanges	Да	Да
	Exists	Неприменима	Неприменима
Хранимые процедуры	Execute	Неприменима	Неприменима
Запросы	RetrieveAll	Неприменима	Неприменима

Управление транзакциями

При обработке исходящих запросов адаптер поддерживает как локальные, так и распределенные транзакции (транзакции XA). Транзакция в адаптере представляет собой изолированное взаимодействие с базой данных. Транзакция может состоять из нескольких операций с базой данных, которые выполняются как атомарный блок. Одновременно выполняющиеся операции от других пользователей базы данных не оказывают на эти операции никакого влияния.

Адаптер поддерживает транзакции только в том случае, если они поддерживаются сервером базы данных. Поддерживаются транзакции двух типов - транзакции XA и локальные транзакции:

- *Локальная транзакция* - это транзакция, в которой компонент определяет запуск и завершение транзакции с одной базой данных. Применяется протокол одноэтапной фиксации.
- *Транзакция XA* может охватывать несколько неоднородных баз данных. В этом случае применяется протокол глобальной, или двухфазной, фиксации.

транзакции XA

Адаптер поддерживает транзакции XA для обработки исходящих запросов. Настройте для адаптера поддержку транзакций XA одним из следующих способов:

- Укажите источник данных JNDI, поддерживающий транзакции XA, с помощью свойства DataSourceJNDIName
- Укажите источник данных XA с помощью свойства XADataSourceName

В свойстве DataSourceJNDIName указывается источник данных, созданный в WebSphere Process Server или WebSphere Enterprise Service Bus. Это имя представляет источник данных XA или источник данных пула соединений. Адаптер будет обрабатывать транзакции XA, если определить источник данных JNDI, который поддерживает транзакции XA на сервере, и указать этот источник при настройке адаптера. Кроме того, адаптер будет обрабатывать транзакции XA при работе с источником данных XA.

Исходящие операции

Компоненты приложения используют операции для выполнения таких действий как извлечение из базы данных. Адаптер предоставляет некоторые исходящие операции. Предоставлена подробная информация о способе обработки адаптером бизнес-объектов для каждой из поддерживаемых операций.

Операция может быть выполнена с помощью стандартного оператора SQL, предоставленного адаптером, или определенной вами хранимой процедуры. Можно выполнить хранимую процедуру для выполнения операции или совершить пользовательскую обработку до или после операции. В каждом бизнес-объекте можно настроить способ выполнения операции.

Операция Create:

Операция Create создает строки в таблицах базы данных в соответствии с бизнес-объектом в запросе. Если бизнес-объект - иерархический, то операция Create выполняет рекурсивный обход всех уровней иерархии этого бизнес-объекта, создавая строки, соответствующие каждому бизнес-объекту в иерархии.

Для обработки операции Create адаптер выполняет следующие действия:

1. Рекурсивно добавляет в базу данных каждый единичный дочерний бизнес-объект, содержащийся в родительском бизнес-объекте и принадлежащий ему. Другими словами, адаптер создает дочерний бизнес-объект и все дочерние бизнес-объекты, содержащиеся в данном объекте и его дочерних бизнес-объектах.
Если в определении бизнес-объекта указано, что атрибут представляет единичный дочерний бизнес-объект, и этот атрибут пуст, то адаптер игнорирует этот атрибут. Однако, если атрибут не представляет дочерний объект, как того требует определение бизнес-объекта, то адаптер возвращает ошибку, и обработка прекращается.
2. Извлекает каждый единичный дочерний бизнес-объект, содержащийся в родительском объекте, но не принадлежащий ему, и проверяет его существование. Если извлечь объект не удастся (что указывает на отсутствие дочернего объекта в базе данных), то адаптер возвращает ошибку и прекращает обработку. При успешном выполнении операции Retrieve адаптер рекурсивно обновляет дочерний бизнес-объект.

Примечание: Для правильной работы этого алгоритма в случае, когда дочерний бизнес-объект существует в базе данных, для атрибутов первичных ключей в дочерних бизнес-объектах должны быть правильно сформированы перекрестные

- ссылки в операциях Create. Если дочерний бизнес-объект не существует в базе данных приложения, то атрибуты первичных ключей не должны быть заданы.
3. Добавляет в базу данных бизнес-объект верхнего уровня, выполняя для этого следующие действия:
 - a. В качестве значения каждого внешнего ключа бизнес-объекта верхнего уровня задает значение первичного ключа соответствующего единичного дочернего бизнес-объекта. Поскольку значения в дочерних бизнес-объектах могут задаваться последовательностями или счетчиками базы данных или самой базой данных во время создания дочернего объекта, то это действие обеспечивает правильность значений внешних ключей в родительском объекте перед его добавлением в базу данных.
 - b. Генерирует новое уникальное значение ИД для каждого атрибута, который автоматически задается базой данных. Имя последовательности или счетчика базы данных хранится в информации уровня приложения для атрибута. Если с атрибутом связаны последовательность или счетчик базы данных, то любое значение, переданное сервером приложений, заменяется значением, сгенерированным адаптером.
 - c. Добавляет в базу данных бизнес-объект верхнего уровня.
 4. Обрабатывает все его множественные дочерние бизнес-объекты следующим образом:
 - a. Задает в каждом дочернем объекте значения внешних ключей, так чтобы они указывали на значения в соответствующих атрибутах первичных ключей в родительском объекте. Поскольку значения первичных ключей родительского объекта уже могли быть сгенерированы во время его создания, то это действие гарантирует, что значения внешних ключей во всех дочерних объектах будут правильными перед вставкой этих объектов в базу данных.
 - b. Добавляет все множественные дочерние объекты в базу данных.

Операция Retrieve:

Операция Retrieve извлекает из базы данных данные для иерархии бизнес-объектов.

Для обработки операции Retrieve адаптер выполняет следующие действия:

1. Удаляет из полученного бизнес-объекта верхнего уровня все дочерние бизнес-объекты. Иначе говоря, создается копия бизнес-объекта верхнего уровня без дочерних объектов.
2. Извлекает бизнес-объект верхнего уровня из базы данных.
 - Если при извлечении возвращается одна строка, обработка продолжается.
 - Если не возвращено ни одной строки (что указывает на отсутствие бизнес-объекта верхнего уровня в базе данных), то адаптер выдает ошибку `RecordNotFoundException`.
 - Если возвращается несколько строк, адаптер выдает ошибку `MultipleMatchingRecordsException`.

Операция Retrieve использует только первичный ключ. Другие столбцы игнорируются.

3. Рекурсивно извлекает все множественные дочерние бизнес-объекты.

Примечание: При заполнении массива бизнес-объектов уникальность не обеспечивается. Уникальность должна обеспечиваться базой данных. Если база данных возвращает одинаковые дочерние бизнес-объекты, то и адаптер возвращает одинаковые дочерние объекты.

4. Рекурсивно извлекает каждый единичный дочерний объект, независимо от того, принадлежит ли он родительскому объекту, или нет.

Примечание: Все единичные дочерние бизнес-объекты обрабатываются на основе их вхождения в бизнес-объект и до обработки родительского бизнес-объекта. Последовательность обработки не зависит от принадлежности, но от нее зависит тип обработки.

Извлечение пустых данных

Адаптер может извлечь из таблицы базы данных запись с пустым значением столбца. Например, в бизнес-объекте Customer могут существовать следующие столбцы: custid, ccode, fname и lname, где custid и ccode образуют составной ключ. Составные ключи - это первичные ключи, которые указывают на несколько атрибутов и используются для определения уникальности бизнес-объекта. Можно извлечь запись Customer, для которой ccode пуст. Адаптер генерирует для операции Retrieve следующий оператор SELECT:

```
select custid, ccode, fname, lname from customer where custid=?  
and ccode is null
```

Операция RetrieveAll:

Операция RetrieveAll используется адаптером для извлечения массива бизнес-объектов из базы данных. Применяемая адаптером процедура зависит от того, над какими объектами выполняется операция RetrieveAll: над бизнес-объектами таблиц базы данных или над объектами пользовательских запросов SQL.

Для бизнес-объектов таблиц базы данных

Критерии выбора для извлечения определяются всеми заполненными ключевыми и неключевыми атрибутами во входящем бизнес-объекте. В зависимости от выбранных атрибутов адаптер может извлечь из базы данных несколько строк для бизнес-объекта верхнего уровня. Используются все значения, указанные в бизнес-объекте верхнего уровня. Параметры в дочернем бизнес-объекте игнорируются. Если во входящем бизнес-объекте не заполнен ни один атрибут, из соответствующей таблицы базы данных извлекаются все строки.

Имя сгенерированного бизнес-объекта совпадает с именем таблицы базы данных. Например, таблица Customer из базы данных представляется как бизнес-объект с именем "Customer".

Для извлечения массива бизнес-объектов адаптер выполняет следующие действия:

1. Составляет бизнес-объект контейнера для всех извлеченных строк. Имя бизнес-объекта контейнера состоит из имени бизнес-объекта, к которому добавляется строка "Container".
2. Создает для каждой извлеченной строки бизнес-график верхнего уровня (если для модуля было дополнительно настроено применение бизнес-графиков). Имя бизнес-графика состоит из имени бизнес-объекта, к которому добавляется строка "BG".
3. Извлекает каждый из бизнес-графиков в контейнере с помощью операции Retrieve.

Структура объектов, возвращаемых операцией RetrieveAll, с бизнес-графиками и без них показана на следующих рисунках.

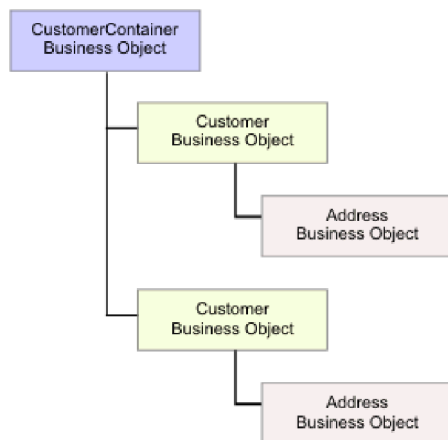


Рисунок 3. Структура бизнес-объекта, возвращаемого операцией RetrieveAll, без дополнительных бизнес-графиков

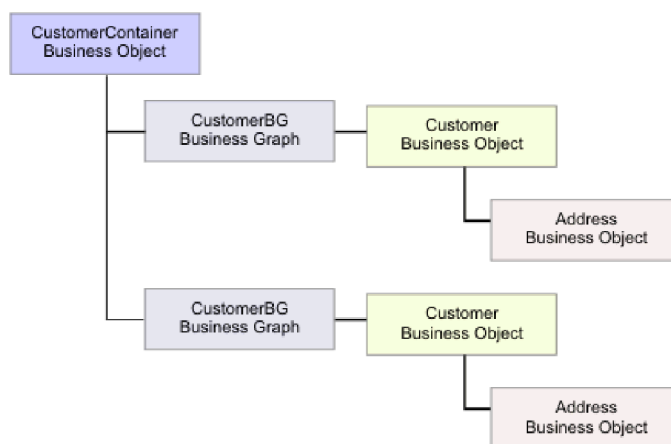


Рисунок 4. Структура бизнес-объекта, возвращаемого операцией RetrieveAll, с дополнительными бизнес-графиками

При выполнении операции RetrieveAll могут возникать следующие ошибки:

- RecordNotFoundException – Эта исключительная ситуация возникает, если для свойства фабрики управляемых соединений ErrorOnEmptyResultSet задано значение True, а в информационной системе предприятия не существует один или несколько заполненных бизнес-объектов из входящего объекта.
- MatchesExceededLimitException – Число совпадающих записей в базе данных превышает указанное в спецификации взаимодействия значение свойства Максимальное число записей для операции RetrieveAll. Фактическое число совпадений, обнаруженных адаптером в базе данных, содержится в атрибуте MatchCount сбоя, поэтому можно или увеличить максимальное число записей, или уточнить критерии поиска.

Примечание: Если значение свойства Максимальное число записей для операции RetrieveAll достаточно велико, то могут возникнуть неполадки из-за нехватки памяти (это зависит от размера и числа возвращаемых бизнес-объектов).

- EISSystemException – Одна или несколько неисправимых ошибок базы данных (информационной системы предприятия)

Для бизнес-объектов запросов

Операция RetrieveAll также поддерживается бизнес-объектами, создаваемыми для пользовательских операторов SELECT (бизнес-объектами запросов). ESW (мастер внешних служб) генерирует бизнес-объект запроса при выполнении пользовательского оператора SQL SELECT и создании иерархии бизнес-объектов запросов. Если дополнительно используются бизнес-графики, то иерархия будет выглядеть так, как показано на рис. рис. 5.

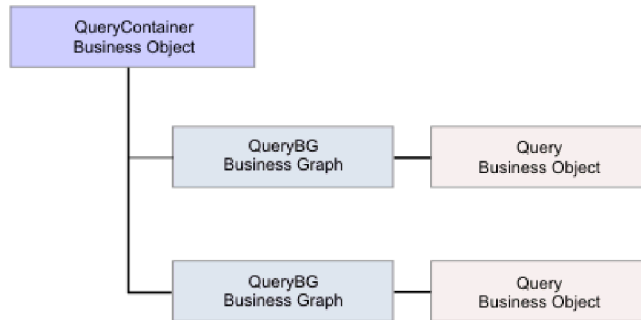


Рисунок 5. Бизнес-объекты пользовательских запросов

Если необязательные бизнес-графики не используются, то иерархия будет выглядеть так, как показано на рис. рис. 6.

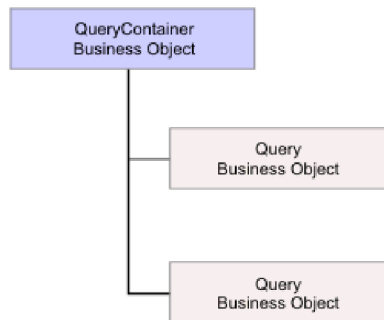


Рисунок 6. Бизнес-объекты пользовательских запросов

Для обработки бизнес-объекта запроса, сгенерированного ESW (мастер внешних служб) для пользовательского оператора SELECT, адаптер выполняет следующие действия:

1. Получает из бизнес-объекта запроса оператор SQL SELECT.
2. Определяет, указан ли в бизнес-объекте запроса динамический блок WHERE.
 - Если динамический блок WHERE есть, адаптер заменяет стандартный блок WHERE в операторе SELECT динамическим блоком WHERE.
 - Если динамического блока WHERE нет, адаптер заменяет параметры в операторе SELECT соответствующими значениями, указанными в бизнес-объекте запроса.
3. Выполняет оператор SELECT.
4. Получает возвращаемый набор результатов и заполняет значения бизнес-объекта запроса данными, извлеченными из базы данных, создавая бизнес-объект контейнера со структурой, показанной на рис. рис. 5.
5. Если для бизнес-объектов запросов определены какие-либо дочерние объекты, извлекает полную иерархию (полное извлечение) каждого бизнес-объекта запроса верхнего уровня в контейнере.

Примечание: Бизнес-объект запроса может быть только бизнес-объектом верхнего уровня. Для бизнес-объекта запроса не могут существовать дочерние бизнес-объекты запросов.

Извлечение объектов NULL

Адаптер может извлекать записи из таблицы базы данных, когда значение столбца равно NULL. Например, бизнес-объект Customer может иметь следующие столбцы: custid, ccode, fname и lname, где ccode не обязательно является первичным ключом. Можно извлечь все записи Customer, для которых столбец ccode равен NULL. Адаптер генерирует следующий запрос select для операции RetrieveAll:

```
select custid, ccode, fname, lname from customer where custid=?
and ccode is NULL
```

Операция Update:

В ходе операции Update исходный бизнес-объект сравнивается с бизнес-объектом, который извлекается из базы данных с использованием первичных ключей, указанных в исходном бизнес-объекте верхнего уровня.

При обновлении иерархического бизнес-объекта адаптер выполняет следующие действия:

1. Использует значения первичных ключей исходного бизнес-объекта для извлечения соответствующей сущности из базы данных. Извлеченный бизнес-объект - это точное представление текущего состояния данных в базе данных.

Если извлечь объект не удастся (что указывает на отсутствие бизнес-объекта верхнего уровня в базе данных), то адаптер возвращает исключительную ситуацию RecordNotFoundException, и обновление не выполняется.

При успешном извлечении адаптер сравнивает извлеченный бизнес-объект с исходным бизнес-объектом, чтобы определить, для каких дочерних бизнес-объектов необходимы изменения в базе данных. Однако, адаптер не сравнивает значения простых атрибутов исходного и извлеченного бизнес-объектов. Адаптер обновляет значения всех неключевых простых атрибутов.

Если все простые атрибуты в бизнес-объекте верхнего уровня представляют ключи, то адаптер не может генерировать запрос на обновление для бизнес-объекта верхнего уровня. В этом случае адаптер выдает предупреждение и продолжает обработку.

2. Рекурсивно обновляет все единичные дочерние объекты бизнес-объекта верхнего уровня.

Если значение параметра ownership равно true, и дочерний бизнес-объект содержится в исходном бизнес-объекте, но не в извлеченном бизнес-объекте, то адаптер рекурсивно создает в базе данных дочерний объект.

Адаптер обрабатывает единичные дочерние бизнес-объекты, которые содержатся в родительском объекте и принадлежат ему, одним из следующих способов:

- Если дочерний объект обнаруживается как в исходном, так и в извлеченном бизнес-объекте, то вместо обновления существующего дочернего объекта в базе данных адаптер удаляет его и создает новый.
- Если дочерний объект содержится в исходном бизнес-объекте, но не в извлеченном бизнес-объекте, то адаптер рекурсивно создает в базе данных дочерний объект.
- Если дочерний объект содержится в извлеченном бизнес-объекте, но не в исходном бизнес-объекте, то адаптер рекурсивно удаляет дочерний объект из базы данных.

В случае единичных дочерних объектов, которые содержатся в родительском объекте, но не принадлежат ему, адаптер пытается извлечь из базы данных каждый дочерний объект, существующий в исходном бизнес-объекте. При успешном извлечении адаптер заполняет этот дочерний бизнес-объект, но не обновляет его, поскольку адаптер никогда не изменяет единичные дочерние бизнес-объекты, которые содержатся в родительском бизнес-объекте, но не принадлежат ему. Если извлечь объект не удастся, возвращается исключительная ситуация `ObjectNotFound`.

3. Обновляет все простые атрибуты извлеченного бизнес-объекта, за исключением тех, для которых не указан соответствующий атрибут в исходном бизнес-объекте. Поскольку обновляемый бизнес-объект должен быть уникальным, адаптер выполняет проверку, чтобы убедиться, что в результате обрабатывается только одна строка. Если возвращается несколько строк, адаптер выдает ошибку.
4. Обрабатывает каждый множественный дочерний объект извлеченного бизнес-объекта одним из следующих способов:
 - Если дочерний объект существует в массивах как исходного, так и извлеченного бизнес-объекта, то адаптер рекурсивно обновляет его в базе данных.
 - Если дочерний объект существует в исходном массиве, но его нет в массиве извлеченного бизнес-объекта, то адаптер рекурсивно создает его в базе данных.
 - Если дочерний объект существует в массиве извлеченного бизнес-объекта, но отсутствует в исходном массиве, то адаптер рекурсивно удаляет его из базы данных, если только в информации уровня приложения для атрибута, представляющего этот дочерний объект в родительском объекте, не указано свойство `KeepRelationship` со значением `True`. В этом случае адаптер не удаляет дочерний объект из базы данных.

Пустые данные и операция Update

Адаптер может обновить в таблице базы данных запись с пустым значением столбца. Например, в бизнес-объекте `Customer` могут существовать следующие столбцы: `custid`, `ccode`, `fname` и `lname`, где `custid` и `ccode` образуют составной ключ. Составные ключи - это первичные ключи, которые указывают на несколько атрибутов и используются для определения уникальности бизнес-объекта. Запись `Customer`, для которой `ccode` пуст, можно обновить. Адаптер сгенерирует для операции `Update` следующий запрос на обновление:

```
update customer set fname=?, lname=? where custid=? and ccode is null
```

Операция ApplyChanges:

Операция `ApplyChanges` предоставляет поддержку как поправок, так и записей нового значения для изменения или удаления бизнес-объекта. Операция `ApplyChanges` доступна только при работе с бизнес-графиками.

Если в свойстве глагольной команды бизнес-графика указано имя операции, такое как `create`, `update` или `delete`, то адаптер выполняет обработку записи нового значения для операции `ApplyChanges`. Например, если в качестве глагольной команды указана `create`, адаптер обрабатывает операцию `ApplyChanges` так же, как операцию `Create`.

Если глагольная команда в бизнес-графике не задана, то для обновления бизнес-объекта адаптер использует `ChangeSummary`. В этой модели операция `ApplyChanges` отличается от операции `Update` в следующем:

- В операции `ApplyChanges` перед обновлением не выполняется операция `Retrieve`.
- Не выполняется сравнение между входящим бизнес-объектом и бизнес-объектом из базы данных.

- Все дочерние объекты обрабатываются на основе операции, заданной в ChangeSummary для каждого дочернего бизнес-объекта. Если операция для дочернего объекта не задана, адаптер возвращает ошибку.

При обновлении иерархического бизнес-объекта из ChangeSummary адаптер выполняет следующие действия. Обрабатываются только изменения из ChangeSummary.

1. Рекурсивно обрабатывает все единичные дочерние объекты родительского объекта. Если дочерний объект помечен в спецификации бизнес-объекта как обязательный, то он должен существовать во входящем объекте. В противном случае операция ApplyChanges не выполняется, и адаптер возвращает ошибку.
2. Всем внешним ключам в родительском объекте, указывающим на атрибуты единичных дочерних объектов, присваиваются соответствующие значения дочерних объектов. Это необходимо, поскольку единичные дочерние объекты уже могли быть добавлены в базу данных на предыдущих этапах, что привело к генерации новых значений последовательности.
3. Обновляет текущий объект, обрабатываемый с помощью оператора SQL UPDATE или хранимой процедуры. Обновляются все простые атрибуты отдельного бизнес-объекта. Адаптер не использует изменения уровня свойств, чтобы определить, какие атрибуты должны быть добавлены в оператор UPDATE; обновляются все атрибуты. Поскольку обновляемый объект должен быть уникальным, адаптер выполняет проверку, чтобы убедиться, что в качестве результата обрабатывается только одна строка. Если обрабатывается несколько строк, возвращается ошибка.
4. Всем внешним ключам во всех N дочерних объектах текущего объекта, указывающим на атрибуты родительского объекта, присваиваются соответствующие значения родительского объекта. Обычно для этих значений уже сформированы перекрестные ссылки во время отображения данных; однако, это может быть не так для новых дочерних объектов в контейнере с N объектами. Этот шаг обеспечивает правильность значений внешних ключей во всех N дочерних объектах перед обновлением этих объектов.
5. Обновляются все контейнеры с N объектами для текущего объекта.

При обработке дочерних объектов выбирается операция каждого дочернего объекта и выполняется соответствующая операция. В ApplyChanges над дочерним объектом допустимы операции Create, Delete и Update:

- Если в дочернем объекте найдена операция Create, и этот дочерний объект принадлежит родительскому объекту, то он создается в базе данных. Дочерние объекты, не принадлежащие родительскому объекту, извлекаются для проверки их существования в базе данных.
- Если в дочернем объекте найдена операция Delete, этот объект удаляется.
- Если в дочернем объекте найдена операция Update, этот объект обновляется в базе данных.

Операция Delete:

Операция Delete выполняется путем отсечения входящего бизнес-объекта и последующего извлечения полного бизнес-объекта из базы данных. Затем операция Delete выполняется рекурсивно над каждым бизнес-объектом в иерархии.

Операция Delete поддерживает физическое и логическое удаление, в зависимости от значения параметра StatusColumnName в информации уровня приложения бизнес-объекта. Если значение StatusColumnName определено, то адаптер выполняет операцию логического удаления, если не определено - операцию физического удаления.

Физическое удаление

При физическом удалении адаптер выполняет следующие действия:

- Рекурсивно удаляет все множественные дочерние бизнес-объекты.
- Удаляет бизнес-объект верхнего уровня.
- Рекурсивно удаляет все единичные дочерние бизнес-объекты, содержащиеся в родительском бизнес-объекте и принадлежащие ему.

Логическое удаление

При логическом удалении адаптер выполняет следующие действия:

- Выполняет обновление, в ходе которого атрибуту состояния бизнес-объекта присваивается значение, указанное в информации уровня приложения для бизнес-объекта. Адаптер гарантирует, что в результате обновляется только одна строка базы данных, в противном случае возвращается ошибка.
- Выполняет рекурсивное логическое удаление всех единичных дочерних объектов, содержащихся в дочернем объекте и принадлежащих ему, и всех множественных дочерних объектов. Единичные дочерние объекты, содержащиеся в родительском объекте, но не принадлежащие ему, не удаляются.

Пустые данные и операция Delete

Адаптер может удалить из таблицы запись с пустым значением столбца. Например, в бизнес-объекте Customer могут существовать следующие столбцы: custid, ccode, fname и lname, где custid и ccode образуют составной ключ. Составные ключи - это первичные ключи, которые указывают на несколько атрибутов и используются для определения уникальности бизнес-объекта. Запись Customer, для которой ccode пуст, можно удалить. Адаптер генерирует для операции Delete следующий запрос delete:
delete from customer where custid=? and ccode is null

Операция Execute:

Операция Execute позволяет выполнять хранимые процедуры и хранимые функции. ESW (мастер внешних служб) генерирует требуемый бизнес-объект хранимой процедуры, который соответствует определению хранимой процедуры или хранимой функции в базе данных. Для обработки бизнес-объекта хранимой процедуры адаптер использует операцию Execute.

Ниже рассматривается простой пример хранимой процедуры и бизнес-объекта, созданного на ее основе, а также перечисляются действия, выполняемые адаптером при обработке бизнес-объекта хранимой процедуры с помощью операции Execute.

Простой пример хранимой процедуры:

```
PROCEDURE testSP(IN int x,INOUT VARCHAR(10) msgSTR, OUT int status,  
                OUT struct outrec, OUT array retArr)
```

Эта процедура возвращает два набора результатов.

Вот пример бизнес-объекта, созданного на основе этой хранимой процедуры:

```
BOLevel ASI  
  SPName=testSP  
  ResultSet=true  
  MaxNumberOfResultSets=2  
  ReturnValue = propName  
  Возвращается, если хранимая процедура представляет собой функцию
```

Если возвращаемое значение - сложного типа, то это будет именем свойства, соответствующего дочернему бизнес-объекту (array/struct/resultset)
Определено только в том случае, если это функция

Свойства

x Type=IP
msgStr Type=IO
status Type=OP
outrec Type OP - Дочерний BO для outrec, ASI ChildBOType = struct
retarr Type OP - Дочерний BO с кардинальным числом n BO для retArr, ASI ChildBOType = array
childBOName1 - Дочерний BO для 1-го набора результатов, ASI ChildBOType = resultset
childBOName2 - Дочерний BO для 2-го набора результатов, ASI ChildBOType = resultset

Для обработки этой хранимой процедуры с помощью операции Execute адаптер выполняет следующие действия:

1. Создает следующий вызов хранимой процедуры: CALL testSP(x, msgStr, status, outrec, retArr).
2. Задает входные параметры x и msgStr для вызываемого оператора.
3. Выполняет вызываемый оператор.
4. Получает возвращаемое значение (если это функция); если значение скалярного типа, то задает значение в соответствующем атрибуте, если сложного (struct или array) - в дочернем бизнес-объекте.
5. Получает первый набор результатов и создает контейнер для ResultSet1.
6. Получает второй набор результатов и создает контейнер для ResultSet2.
7. Получает выходные параметры msgStr и status и задает соответствующие атрибуты бизнес-объекта.
8. Получает выходной параметр outrec и создает дочерний бизнес-объект на основе данных, возвращаемых в outrec. Если outrec - вложенный тип struct, то адаптер рекурсивно создает и сохраняет данные в иерархическом дочернем бизнес-объекте.
9. Получает выходной параметр retArr и создает множественный дочерний бизнес-объект на основе данных, возвращаемых в retArr. Если retArr - вложенный тип array, то адаптер рекурсивно создает и сохраняет данные в иерархическом дочернем бизнес-объекте.

Операция Exists:

Операция Exists определяет, содержит ли база данных записи, совпадающие с заданными в бизнес-объекте атрибутами.

В критериях выбора можно использовать как ключевые, так и неключевые атрибуты.

Примечание: Когда для поиска объектов таблицы в базе данных используется ESW (мастер внешних служб), можно выбрать несколько таблиц и добавить их к выбранным объектам в окне Поиск и выбор объектов. Однако, ESW (мастер внешних служб) не позволяет связать или соединить выбранные таблицы. Если задаче вашего бизнес-приложения потребуются бизнес-объект таблицы для выполнения операции Exists над соединенными таблицами, необходимо соединить таблицы в базе данных, чтобы создать представление соединенных таблиц. После того как представление создано, в нем можно выполнять поиск. В этом представлении должна поддерживаться операция Exists.

Для обработки операции Exists и отправки результатов на основе указанных атрибутов бизнес-объекта адаптер выполняет следующие действия:

1. Адаптер принимает бизнес-объект таблицы из точки импорта. Этот бизнес-объект может быть одноуровневым (простым, не содержащим дочерних бизнес-объектов) или иерархическим (сложным, содержащим один или несколько дочерних бизнес-объектов).

Если бизнес-объект иерархический, то адаптер компонует запрос только для бизнес-объекта верхнего уровня (отдельного бизнес-объекта на верхнем уровне иерархического бизнес-объекта).

Примечание: Входной бизнес-объект, который поддерживает операцию Exists, будет зависеть от типа бизнес-объекта. Операция Exists поддерживается не только бизнес-объектом таблицы, но и бизнес-объектом представлений, бизнес-объектом синонимов и бизнес-объектом псевдонимов.

2. Адаптер использует бизнес-объект таблицы для генерации оператора SQL SELECT, отправляемого на сервер.

Оператор SQL SELECT используется следующим образом:

```
select count(*) from TABLENAME where column1=? AND column2=?
```

Вот образец оператора SQL для рассматриваемого примера:

```
select count(*) from CUSTOMER where fname='John' AND lname='Smith'
```

В этом случае в операторе SQL содержатся *не являющиеся первичными ключами* атрибуты fname и lname с присвоенными значениями John и Smith, соответственно.

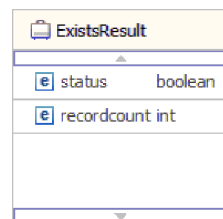
Информация об атрибутах из бизнес-объекта таблицы включается адаптером в блок *where* запроса SQL.

3. Сервер базы данных выполняет запрос SQL и отправляет результаты обратно адаптеру.
4. Адаптер получает от сервера базы данных результаты запроса SQL и задает атрибуты recordcount и status в бизнес-объекте **ExistsResults**.

Например, если операция Exists определила, что существуют две записи, совпадающие с атрибутом и значением, заданными в бизнес-объекте, то адаптер устанавливает status=true и recordcount=2.

Если запись с указанными атрибутами не найдена, выходному параметру status присваивается значение **false**, а выходному параметру recordcount - значение **0**.

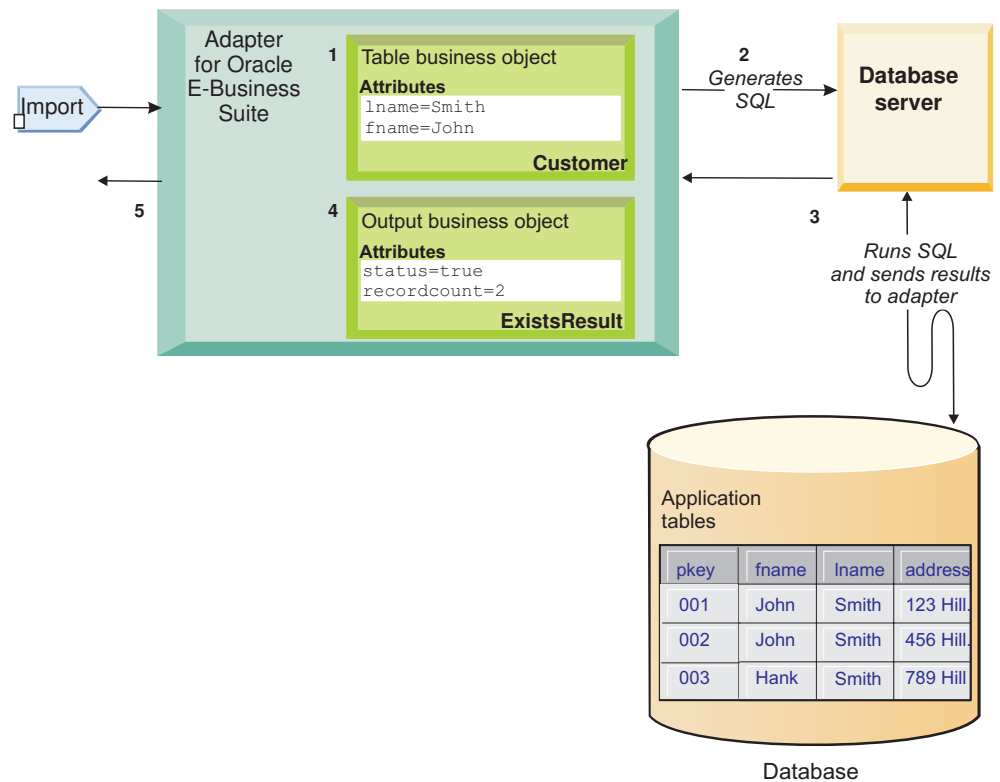
Структура бизнес-объекта ExistsResult, возвращаемого операцией Exists, показана на следующем рисунке.



ExistsResult	
status	boolean
recordcount	int

5. Адаптер возвращает вызывающей стороне бизнес-объект ExistsResult.

На следующем рисунке показано, каким образом адаптер обрабатывает бизнес-объект таблицы с помощью операции Exists.



Обработка входящих событий

Адаптер поддерживает управление входящими событиями с помощью доставки событий. Обрабатываются события из хранилища событий, которое заполняется или приложением базы данных, или в результате выполнения пользовательских запросов. Периодичность опроса и число записей, одновременно доставляемых адаптером в точку экспорта, задаются пользователем.

Адаптер выполняет опрос на предмет изменений, используя один из следующих методов:

- Обработку стандартных событий: адаптер проверяет события, сохраненные в хранилище событий приложением базы данных
- Обработку пользовательских событий: адаптер выполняет пользовательские запросы, хранимые процедуры или хранимые функции

Настроить обработку стандартных или пользовательских событий можно в ESW (мастер внешних служб) во время первоначальной настройки адаптера или позже с помощью административной консоли сервера, изменив свойства спецификации активации.

Объект базы данных, указанный в событии, загружается только после доставки уведомления в экспорт. В результате обнаружение и обработка ошибок загрузки откладывается до приема уведомления экспортом. Такой подход отличается от обработки событий в 6.0.2.x адаптера, где ошибки загрузки обнаруживаются до уведомления экспорта.

Обработка стандартных событий

При обработке стандартных событий адаптер предоставляет запросы SQL, которые спрашивают события и гарантируют, что событие доставлено ровно один раз.

При создании, обновлении или удалении записей в таблицах базы данных запускаются триггеры базы данных или такие инструменты как Oracle Change Data Capture (инструмент сбора данных об изменениях). Триггер или другой инструмент заносит запись события в *хранилище событий*, представляющее собой постоянный кэш, в котором записи событий хранятся до тех пор, пока опрашивающему адаптеру не удастся их обработать. Хранилище событий реализовано как таблица в одной базе данных вместе с пользовательскими таблицами, которые содержат объекты базы данных, доступные адаптеру.

Для того чтобы можно было получать события, генерируемые при определенных изменениях в таблицах базы данных, необходимо определить триггеры или настроить другие инструменты, которые отслеживали бы эти изменения.

Адаптер обеспечивает гарантированную однократную доставку, в результате которой каждое событие доставляется в точку экспорта один и только один раз. Если для модуля включена гарантированная однократная доставка, то для каждого события в хранилище событий задается ИД транзакции (XID). После того как событие получено для обработки, его значение XID в хранилище событий обновляется. Затем событие доставляется в соответствующую точку экспорта, после чего удаляется из хранилища событий. Событие невозможно обработать полностью, если его не удастся доставить из-за нарушения соединения с базой данных или остановки приложения. В этом случае значение в столбце XID указывает, что событие должно быть повторно обработано и отправлено в точку экспорта. После восстановления соединения с базой данных или повторного запуска адаптера в хранилище событий проверяются события со значением в столбце XID. В циклах опроса адаптер сначала обрабатывает эти события, а затем опрашивает другие события.

Адаптер может обрабатывать все события или отфильтровывать события по типу бизнес-объекта, метке времени или ИД коннектора. Для задания фильтра используются свойства `EventTypeFilter`, `FilterFutureEvents` или `AdapterInstanceEventFilter` спецификации активации. В свойстве `EventFilterType` задается список типов бизнес-объектов, разделенных запятыми. Обрабатываются только типы, перечисленные в этом списке. Если значение свойства не указано, фильтр не применяется, и обрабатываются все события. Если для свойства `FilterFutureEvents` указано значение `true`, то адаптер фильтрует события по метке времени и ИД коннектора. В каждом цикле опроса системное время сравнивается с меткой времени события. Если обнаруживается, что событие должно произойти в будущем, оно не обрабатывается, пока не наступит заданный момент времени. Если задано свойство спецификации активации `AdapterInstanceEventFilter`, то обрабатываются только события с ИД коннектора, указанным в этом свойстве.

Обработка пользовательских событий

При обработке пользовательских событий запросы SQL или хранимые процедуры, которые опрашивают события, задаются пользователем.

В случае обработки пользовательских событий пользователь определяет, какие события доставляются в точку экспорта, предоставляя запрос к базе данных (*пользовательский запрос события*), который должен выполняться адаптером вместо запроса SQL, используемого для опроса хранилища событий в случае обработки стандартных событий. Пользовательский запрос события должен выполнять всю необходимую фильтрацию. Для включения обработки пользовательских событий выберите соответствующую опцию в мастере или задайте свойство `EventQueryType` спецификации активации в административной консоли.

Для того чтобы при обработке пользовательских событий поддерживалась гарантированная однократная доставка, создайте хранилище стандартных событий для хранения значений XID. В этом хранилище адаптер будет хранить события, возвращаемые пользовательским запросом события, и обновлять их с помощью значений XID. Адаптер будет обрабатывать события тем же способом, что и в случае обработки стандартных событий. Не создавайте пользовательский запрос для обращения к хранилищу стандартных событий, поскольку в этой таблице временно хранятся события, когда для адаптера настроена гарантированная однократная доставка. Кроме того, в этом случае ситуации в хранилище событий не должны автоматически создаваться значения ИД события, так как адаптер заполняет ИД события значением, извлекаемым из пользовательского запроса в хранилище событий.

Для включения обработки пользовательских событий выберите дополнительную опцию в мастере при настройке модуля для работы с адаптером или задайте свойство `EventQueryType` спецификации активации.

Пользовательский запрос события

Для того чтобы задать выполняемый пользовательский запрос события, укажите в дополнительной опции мастера пользовательскую очередь событий или задайте свойство `CustomEventQuery` спецификации активации. Укажите один из следующих типов программ:

- Стандартные операторы SQL
- Хранимая процедура
- Хранимая функция

В качестве входного параметра для любой из этих программ указывается свойство `PollQuantity` спецификации активации (число опрашиваемых бизнес-объектов), которое адаптер предоставляет во время выполнения. В программе могут быть и другие входные параметры. Эти программы должны возвращать набор результатов, число записей в котором равно числу опрашиваемых бизнес-объектов, и который содержит следующие столбцы (в указанном порядке): `event_id`, `object_key`, `object_name` и `object_function`. На основе набора результатов адаптер создает объект события и обрабатывает события.

Стандартные операторы SQL

Можно задать оператор SQL `SELECT`, который выбирает события для обработки. Помимо входного параметра - числа опрашиваемых объектов в запросе могут быть и другие входные параметры.

Хранимая процедура

Пользовательский запрос может быть хранимой процедурой, в которую передается число опрашиваемых объектов в качестве входного параметра, а возвращается выходной параметр типа набора результатов. Для задания хранимой процедуры используйте следующий синтаксис:

```
call имя-процедуры (?, ?)
```

где *имя-процедуры* - это имя выполняемой хранимой процедуры. Первый параметр представляет число опрашиваемых объектов, а второй - набор результатов.

В хранимую процедуру могут передаваться другие входные параметры, которые задаются в самом операторе `call`, например:

```
call  
имя-процедуры (25, ?, ?)
```

Хранимая функция

Пользовательский запрос может быть хранимой функцией, которая принимает в качестве входного параметра число опрашиваемых объектов, а возвращает набор результатов. Для задания хранимой функции используйте следующий синтаксис:

```
? =  
call имя-функции (?)
```

где *имя-функции* - это имя хранимой функции, которая должна выполняться. Первый параметр представляет набор результатов, а второй - число опрашиваемых объектов.

В хранимую функцию могут передаваться другие входные параметры, которые задаются в самом операторе call, например:

```
? = call имя-функции  
(?, 'abc')
```

Пользовательские запросы на обновление и удаление

При обработке пользовательских событий можно также задать пользовательские запросы на обновление и удаление, которые должны выполняться после обработки каждого события. Применение *запроса на обновление* позволяет гарантировать, что запись базы данных не будет выбрана для обработки во время последующих циклов опроса. *Запрос на удаление* рекомендуется использовать в тех случаях, когда записи базы данных должны удаляться после обработки каждого события. И запрос на удаление, и запрос на обновление не являются обязательными.

Запросы на обновление и на удаление задаются, соответственно, с помощью свойств CustomUpdateQuery и CustomDeleteQuery спецификации активации. Эти запросы могут вводиться как стандартный оператор SQL, или хранимая процедура, или хранимая функция. Синтаксис пользовательского запроса на обновление или удаление такой же, как и для пользовательского запроса. В качестве входного параметра для запросов на обновление и удаление используется ИД события. Значение ИД события предоставляется адаптером во время выполнения. Для этих запросов также допустимы дополнительные входные параметры, которые задаются в самом синтаксисе запроса, как описано для пользовательского запроса события.

Хранилище событий

Хранилище событий - это постоянный кэш, в котором хранятся записи событий, пока не будут обработаны опрашивающим адаптером. Адаптер использует хранилище событий для отслеживания входящих запросов по мере их продвижения по системе. Всякий раз, когда создается, обновляется или удаляется запись базы данных, адаптер обновляет состояние события в хранилище событий. До тех пор, пока событие не будет доставлено в настроенную точку экспорта на сервере, состояние события непрерывно обновляется адаптером, что позволяет при необходимости восстановить событие.

Адаптер опрашивает записи событий в хранилище событий через равные промежутки времени. В каждом вызове опроса обрабатывается несколько событий. События обрабатываются в порядке возрастания приоритета и в порядке возрастания значения метки времени. В каждом цикле опроса адаптер захватывает несколько новых событий. Для каждого нового события адаптер извлекает значение, указанное в поле ключа объекта для события, а затем загружает бизнес-объект, который соответствует значению, указанному в поле имени объекта. После загрузки объекта

адаптер задает значения первичных ключей бизнес-объекта на основе значения, указанного в поле ключа объекта. После задания ключей адаптер извлекает объект на основе этих ключей. На основе извлеченной информации создается бизнес-объект или необязательный бизнес-график, которые публикуются в точке экспорта.

Если с операцией извлечения бизнес-объекта была связана хранимая процедура, то можно определить отображение между входными параметрами хранимой процедуры и атрибутами бизнес-объекта (как правило - первичными ключами). Если такое отображение определено, то адаптер задает входные параметры хранимой процедуры, вызывает ее и заполняет объект на основе результатов этой хранимой процедуры.

Если определено отображение между входными параметрами хранимой процедуры или функции и атрибутами бизнес-объекта (обычно с помощью первичных ключей) с помощью информации уровня приложения RetrieveSP, то адаптер задает входные параметры хранимой процедуры, вызывает ее и заполняет бизнес-объект на основе ее результатов.

Если столбец `object_function` содержит значение `Delete`, указывающее, что объект был удален, то объект из базы данных не извлекается. В объекте данных задаются ключи, и создается бизнес-объект или необязательный бизнес-график, который доставляется в точку экспорта.

В случае успешной публикации события запись удаляется из хранилища событий. В случае неудачи записи остаются в хранилище, а в столбец `event_status` помещается значение -1.

Формат и содержимое таблицы хранилища событий описываются в Табл. 2.

Таблица 2. Определение таблицы базы данных хранилища событий

Имя столбца	Тип	Описание
XID	Строка	Уникальный ID транзакции (XID) для гарантированной однократной доставки.
event_id	Число	Уникальный ID события, представляющий собой первичный ключ таблицы. Значение этого столбца может совпадать со значением столбца <code>object_key</code> .
object_key	Строка	Строка, содержащая ключи извлекаемой из хранилища событий записи. Значение <code>null</code> для этого столбца недопустимо. В качестве значения укажите одну или нескольких пар <i>ключ=значение</i> , разделенных точкой с запятой (;). Можно также указать через точку с запятой (;) только значения первичных ключей. В этом случае значения должны быть указаны в том же порядке, в котором определены первичные ключи в бизнес-объекте.
object_name	Строка	Имя бизнес-объекта или бизнес-графика. Бизнес-объект (или бизнес-объект внутри бизнес-графика) может быть иерархическим. Каждый бизнес-объект или бизнес-график относится к таблице или представлению. Значение <code>null</code> для этого столбца недопустимо.

Таблица 2. Определение таблицы базы данных хранилища событий (продолжение)

Имя столбца	Тип	Описание
object_function	Строка	Операция, соответствующая событию (Delete, Create, Update и т.д.). Значение null для этого столбца недопустимо.
event_priority	Число	Определяет приоритет события. Значение должно быть положительным целым числом. Значение null для этого столбца недопустимо.
event_time	Timestamp	Дата и время генерации события. Формат: мм/дд/гггг чч:мм:сс.
event_status	Число	Состояние события. Первоначально равно значению, соответствующему новому событию, и обновляется адаптером по мере обработки события. Состояние может иметь одно из следующих значений: <ul style="list-style-type: none"> • 0: новое событие. • 1: событие было доставлено в точку экспорта. • -1: при обработке события произошла ошибка. Значение null для этого столбца недопустимо.
event_comment	Строка	Любой комментарий, связанный с событием.
connector_ID	Строка	Уникальный идентификатор экземпляра адаптера, который будет получать указанное событие.

Бизнес-объекты

Бизнес-объект - это структура, которая состоит из данных, действия, выполняемого над данными, и дополнительных инструкций, если они есть, для обработки данных. WebSphere Adapter for Oracle E-Business Suite использует бизнес-объекты для представления таблиц и представлений в базе данных, а также результатов запросов к базе данных, хранимых процедур и хранимых функций. Бизнес-объекты могут также создать иерархию объектов из базы данных и сгруппировать несвязанные таблицы. Компонент связывается с адаптером с помощью бизнес-объектов.

Как адаптер использует бизнес-объекты

Интегрированное приложение использует бизнес-объекты для доступа к базе данных. Адаптер преобразовывает бизнес-объекты в исходящих запросах в вызовы API JDBC для доступа к базе данных. Для входящих событий адаптер преобразовывает данные в событиях в бизнес-объекты, которые возвращаются в приложение.

Адаптер использует бизнес-объекты для представления следующих типов объектов в базе данных:

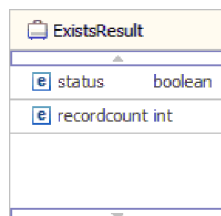
- Таблицы и представления
- Синонимы и псевдонимы
- Хранимые процедуры и хранимые функции

Бизнес-объекты запросов не представляют объектов базы данных. Бизнес-объекты запросов представляют пользовательский запрос SQL, выполняемый для базы данных.

Примечание: Перед использованием бизнес-объектов для представления перечисленных выше типов объектов убедитесь в том, что ключевые слова Java не используются для определения имен таблиц, представлений, хранимых процедур и функций.

Адаптеры используют некоторые бизнес-объекты для вывода. Они включают в себя:

- Бизнес-объект контейнера, который содержит вывод операции RetrieveAll.
- Бизнес-объект ExistsResult, который содержит вывод операции Exists.



Способ представления данных в бизнес-объектах

Для бизнес-объектов таблиц или представлений

Каждый столбец в таблице или представлении представлен простым атрибутом бизнес-объекта таблицы или представления. *Простой атрибут* - это атрибут, который представляет одиночное значение, такое как String, Integer или Date. Другие атрибуты представляют дочерний бизнес-объект или массив дочерних бизнес-объектов.

Простые атрибуты в одном бизнес-объекте не могут храниться в различных таблицах базы данных, однако, возможны следующие ситуации:

- Таблица базы данных может иметь больше столбцов, чем имеет атрибутов соответствующий бизнес-объект, то есть, некоторые столбцы в базе данных не представлены в бизнес-объекте. В проект должны быть включены только те столбцы, которые необходимы для обработки приложением бизнес-объекта.
- Бизнес-объект может иметь больше простых атрибутов, чем имеет столбцов соответствующая таблица базы данных, то есть, некоторые атрибуты бизнес-объекта не представлены в базе данных. Атрибуты, которые не представлены в базе данных, или не имеют информации, специфичной для приложения, и для них установлены значения по умолчанию, или они являются параметрами хранимых процедур или хранимых функций.
- Бизнес-объект может соответствовать представлению, которое охватывает несколько таблиц базы данных. Адаптер может использовать такой бизнес-объект при обработке событий, инициированных изменениями в базе данных, например, операциями Create, Update и Delete. Однако, при обработке запросов бизнес-объекта, адаптер может использовать такой бизнес-объект только для запросов Retrieve и RetrieveAll.

Бизнес-объект таблицы всегда имеет первичный ключ, даже если соответствующая таблица базы данных не имеет первичного ключа. Адаптер использует столбец, указанный в атрибуте первичного ключа, при извлечении бизнес-объектов таблицы. Адаптер поддерживает таблицы, которые имеют составные первичные ключи или несколько первичных ключей. Если таблица базы данных имеет один или несколько первичных ключей, мастер устанавливает свойство первичного ключа для этих столбцов в бизнес-объекте таблицы. Если таблица базы данных не имеет первичного ключа, ESW (мастер внешних служб) запрашивает информацию о первичном ключе при настройке этого бизнес-объекта. Укажите столбец, который содержит уникальные данные, такие как последовательность или столбец-идентификатор.

Бизнес-объекты таблиц и представлений поддерживают исходящие операции Create, Update, Delete, Retrieve, RetrieveAll, Exists и ApplyChanges. При выполнении операции Exists для иерархического бизнес-объекта таблицы, запрашивается только бизнес-объект верхнего уровня.

рис. 7 показывает бизнес-объект таблицы в редакторе бизнес-объекта. Бизнес-объект имеет атрибут для каждого столбца таблицы базы данных. Так как не имеет дочерних бизнес-объектов, все атрибуты являются простыми.

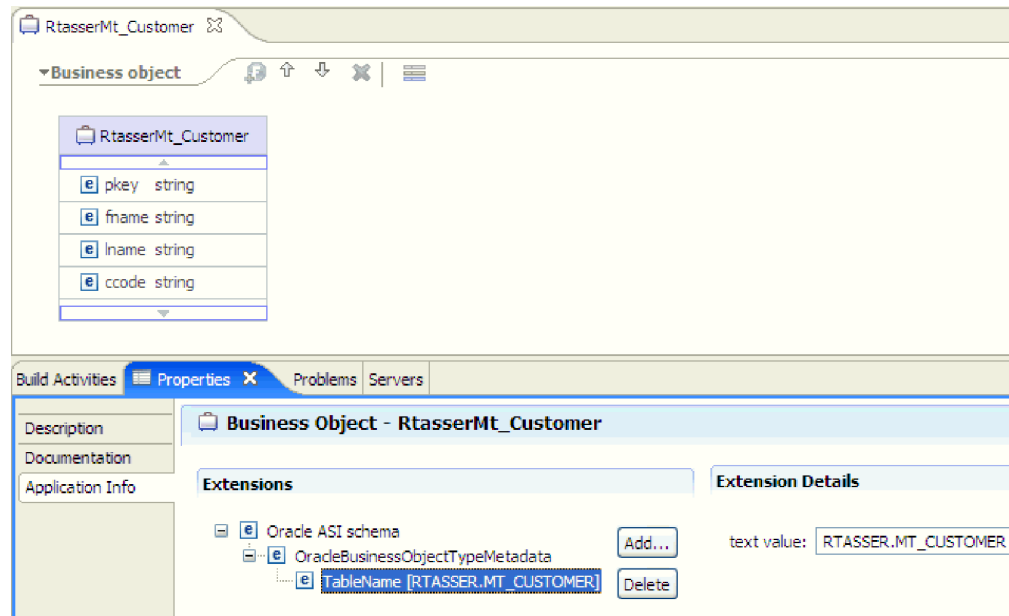


Рисунок 7. Бизнес-объект таблицы без дочерних бизнес-объектов.

рис. 8 на стр. 25 показывает бизнес-объект таблицы, которая имеет один дочерний бизнес-объект таблицы. Бизнес-объект имеет простые атрибуты для каждого столбца таблицы базы данных, а также сложный атрибут, указывающий на дочерний бизнес-объект.

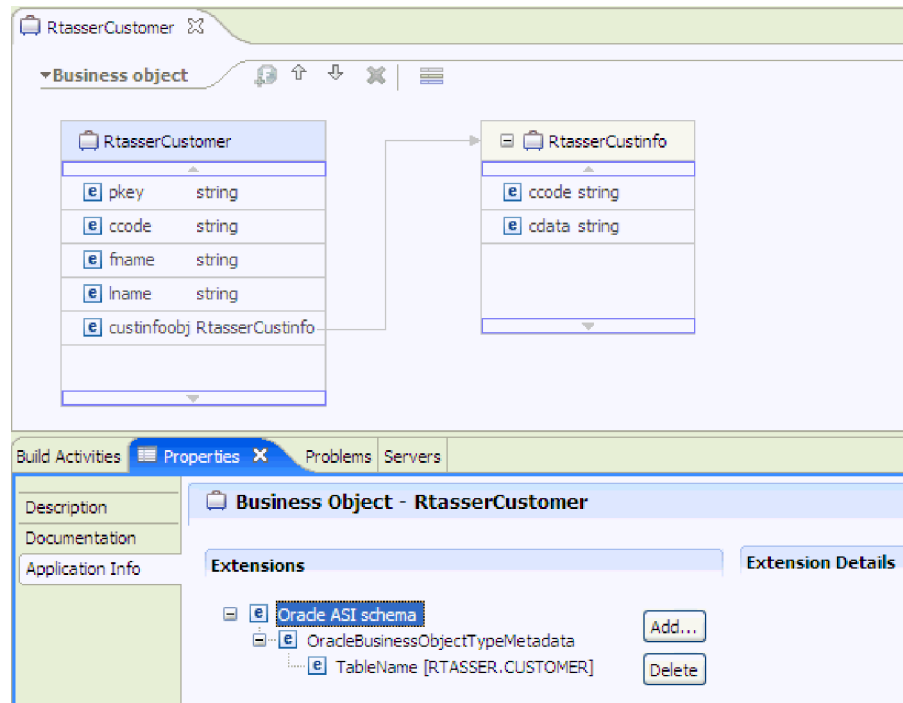


Рисунок 8. Бизнес-объект таблицы с одним дочерним бизнес-объектом.

Для бизнес-объектов хранимых процедур и функций

В бизнес-объекте для хранимой процедуры или хранимой функции все входные и выходные параметры хранимой процедуры или хранимой функции имеют соответствующие атрибуты в бизнес-объекте. Если любой из входных или выходных параметров имеет сложный тип, такой как массив или структура, то соответствующий атрибут бизнес-объекта является типом дочернего бизнес-объекта с дочерним бизнес-объектом, содержащим атрибуты массива или структуры. Если хранимая процедура возвращает набор результатов, то создается дочерний бизнес-объект, который содержит атрибуты возвращаемого набора результатов.

Бизнес-объекты для хранимых процедур и функций поддерживают исходящую операцию Execute.

Расположенная ниже панель Свойства показывает бизнес-объекты, сгенерированные из хранимой процедуры, которая имеет один входной тип и два выходных типа. Один из выходных параметров имеет тип данных Struct. ESW (мастер внешних служб) генерирует бизнес-объект для типа Struct и добавляет его как дочерний объект в родительский бизнес-объект. Для атрибута типа Struct в родительском бизнес-объекте информация, специфичная для приложения, ChildBOType установлена как Struct, чтобы указать, что он имеет этот тип.

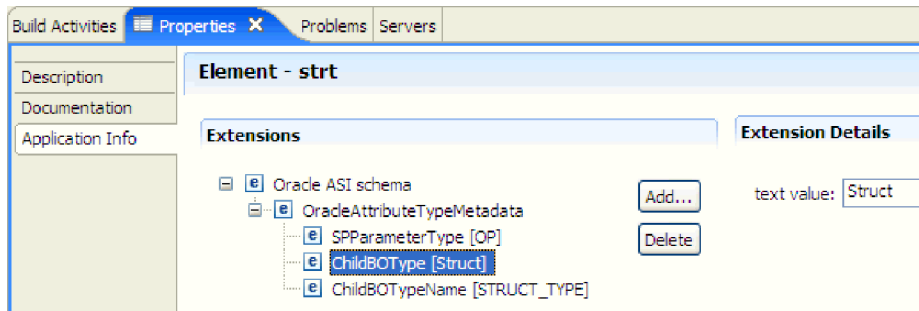


Рисунок 9. Тип дочернего бизнес-объекта с атрибутом структуры

На панели Свойства, показанной ниже, информация, специфичная для приложения ChildBOTypeName установлена в значение пользовательского типа Struct в базе данных.

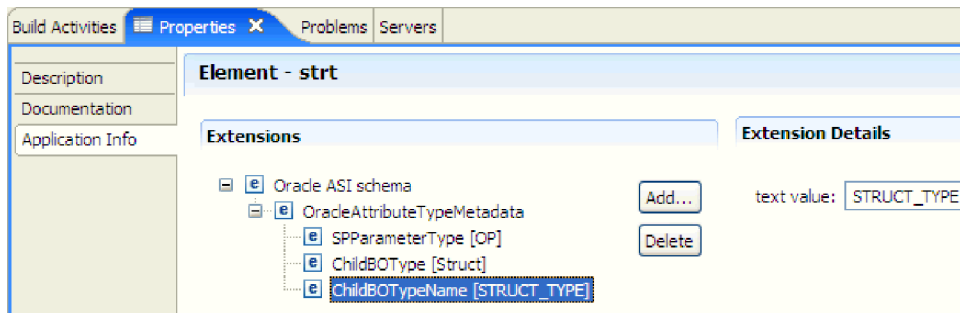


Рисунок 10. Тип дочернего бизнес-объекта установлен в тип Struct

Для бизнес-объектов запросов

Бизнес-объект для запроса базы данных определяет оператор SQL, выполняющий запрос, и параметры этого запроса. Бизнес-объект запроса поддерживает исходящую операцию RetrieveAll.

В качестве примера, предположим, что бизнес-объект запроса выполняет следующий оператор SELECT:

```
select C.pkey, C.fname,
A.city from customer C, address A
WHERE (C.pkey = A.custid) AND (C.fname like ?)
```

Вопросительный знак (?) указывает входной параметр запроса. Запрос может иметь несколько параметров, каждый из которых указан в операторе SELECT вопросительным знаком. Табл. 3 показывает атрибуты простого бизнес-объекта запроса. Бизнес-объект запроса имеет простые атрибуты для каждого столбца, который должен быть извлечен, простой атрибут для каждого параметра и “объект заменителя” для оператора WHERE запроса, который хранил оператор WHERE после подстановки параметров.

Таблица 3. Атрибуты бизнес-объекта запроса

Атрибут бизнес-объекта	Описание
pkey	Соответствует PKEY столбца базы данных в таблице Customer
fname	Соответствует FNAME столбца базы данных в таблице Customer
city	Соответствует CITY столбца базы данных в таблице Address

Таблица 3. Атрибуты бизнес-объекта запроса (продолжение)

Атрибут бизнес-объекта	Описание
parameter1	Параметр. Существует один параметр для каждого ? (вопросительного знака) в операторе SELECT. В операторе SELECT с несколькими параметрами, последующие параметры названы parameter2, parameter3 и так далее.
jdbcwhereclause	Объект заменителя для оператора WHERE

Следующий рисунок показывает бизнес-объект для простого запроса в редакторе бизнес-объектов.

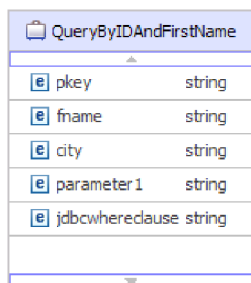


Рисунок 11. Атрибуты бизнес-объекта запроса

Этот рисунок показывает информацию, специфичную для приложения, для примера бизнес-объекта запроса. Информация, специфичная для приложения, SelectStatement содержит оператор SELECT.

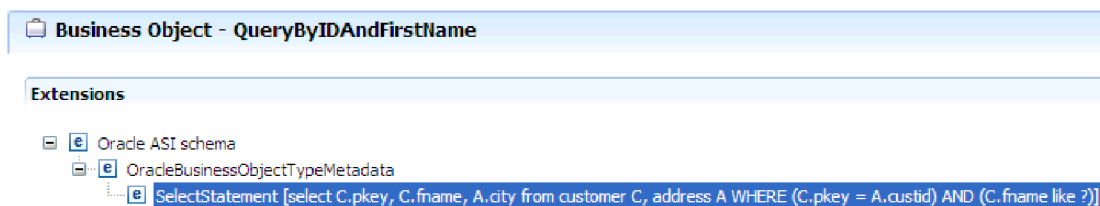


Рисунок 12. Оператор SELECT сохранен в информации, специфичной для приложения, бизнес-объекта

Бизнес-графики

В процессе настройки адаптера можно необязательно выбрать генерацию бизнес-графики. В 6.2 бизнес-графики требуются только в следующих ситуациях:

- Если необходимо использовать исходящую операцию ApplyChanges
- При добавлении бизнес-объектов в модуль, созданный с помощью более ранней версии WebSphere Integration Developer, чем 6.1.0

Если бизнес-графики существуют, то они обрабатываются, но глагольная команда игнорируется для всех операций, кроме ApplyChanges.

Способ создания бизнес-объектов

Бизнес-объекты создаются с помощью ESW (мастер внешних служб), запускаемого из WebSphere Integration Developer. Мастер подключается к базе данных, ищет объекты базы данных и показывает их. Вы выбираете объекты базы данных, для

которых вы хотите создать бизнес-объекты. Например, вы указываете, какие схемы вы хотите проверить. В этих схемах вы выбираете таблицы, представления, хранимые процедуры и функции, синонимы и псевдонимы. Кроме того, можно создать дополнительные бизнес-объекты. Например, можно создать бизнес-объект, представляющий результаты пользовательских операторов SELECT, INSERT, UPDATE или DELETE, который выполняется в базе данных. Мастер помогает построить иерархию бизнес-объектов, с помощью взаимосвязи родительский-дочерний.

После указания бизнес-объектов и определения из иерархии мастер генерирует бизнес-объекты для представления выбранных объектов. Он также генерирует другие артефакты, необходимые адаптеру.

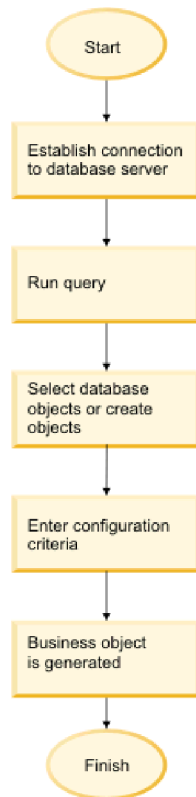


Рисунок 13. Способ создания бизнес-объектов

В некоторых случаях мастер не может полностью настроить взаимосвязь родительский-дочерний. Для этих взаимосвязей вы можете использовать редактор бизнес-объектов, запущенный из WebSphere Integration Developer, для изменения или завершения определения иерархии бизнес-объектов, созданной мастером. Более подробная информация об использовании редактора бизнес-объектов находится в WebSphere Integration Developer information center по следующей ссылке: <http://publib.boulder.ibm.com/infocenter/dmndhelp/v6rxmx/index.jsp>.

Иерархия бизнес-объектов

Определите взаимосвязи между таблицами базы данных с помощью отношений родительского и дочернего объектов и принадлежности данных в иерархических бизнес-объектах.

Бизнес-объекты могут быть или одноуровневыми, или иерархическими. Все атрибуты в одноуровневых бизнес-объектах являются простыми и представляют одну строку в

таблице базы данных. Иерархия может содержать связанные или несвязанные бизнес-объекты. Связанные бизнес-объекты имеют отношения типа родительский-дочерний, и могут как принадлежать, так и не принадлежать родительскому бизнес-объекту.

Термин *иерархический* относится к сложному бизнес-объекту и всем дочерним бизнес-объектам, содержащимся в нем на любом уровне. Термин *отдельный* относится к одному бизнес-объекту, не зависящему от дочерних бизнес-объектов, которые он может содержать, или от родительского бизнес-объекта, в котором он может содержаться. Отдельный бизнес-объект может задавать представление, которое охватывает несколько таблиц базы данных. Бизнес-объектом *верхнего уровня* называется отдельный бизнес-объект на верхнем уровне иерархии, для которого не существует родительского бизнес-объекта.

Иерархический бизнес-объект имеет атрибуты, которые представляют дочерний бизнес-объект, массив дочерних бизнес-объектов или комбинацию и тех, и других. Каждый дочерний бизнес-объект может, в свою очередь, содержать дочерний бизнес-объект или массив дочерних бизнес-объектов, и т.д.

Отношение с одним объектом возникает, когда атрибут в родительском бизнес-объекте представляет один дочерний бизнес-объект. В этом случае тип атрибута совпадает с типом дочернего бизнес-объекта. Адаптер поддерживает отношения с одним объектом, а также отношения с одним объектом и данные, не принадлежащие объекту.

Отношение с многими объектами возникает, когда атрибут в родительском бизнес-объекте представляет массив дочерних бизнес-объектов. В этом случае тип атрибута совпадает с типом дочерних бизнес-объектов.

Следующие типы отношений между бизнес-объектами позволяют отразить иерархию, представляющую таблицы базы данных:

- Отношения с одним объектом
- Отношения с одним объектом и данные, не принадлежащие объекту
- Отношения с многими объектами
- Дочерние бизнес-объекты с несколькими родительскими объектами

Для разных типов кардинальности отношение между родительским и дочерними бизнес-объектами описывается информацией уровня приложения в атрибутах ключей в бизнес-объекте, в котором хранится это отношение.

Отношения с одним объектом в бизнес-объектах:

В случае отношения с с одним объектом атрибут в родительском объекте представляет один дочерний бизнес-объект. В этом случае тип атрибута совпадает с типом дочернего бизнес-объекта. Адаптер поддерживает отношения с одним объектом, а также отношения с одним объектом и данные, не принадлежащие объекту.

Отношения с одним объектом

Как правило, бизнес-объект, который содержит единичный дочерний бизнес-объект, имеет минимум два атрибута, представляющие отношение. Тип одного атрибута совпадает с типом дочернего объекта. Другой атрибут - это простой атрибут, который содержит первичный ключ дочернего объекта в качестве внешнего ключа

родительского объекта. Число атрибутов внешних ключей родительского объекта совпадает с числом атрибутов первичных ключей дочернего объекта.

Типичное отношение с одним объектом показано на рис. рис. 14. В этом примере FKKey в объекте ParentBOName - это простой атрибут, содержащий первичный ключ дочернего объекта, а Child(1) (также в объекте ParentBOName) - атрибут, представляющий дочерний бизнес-объект.

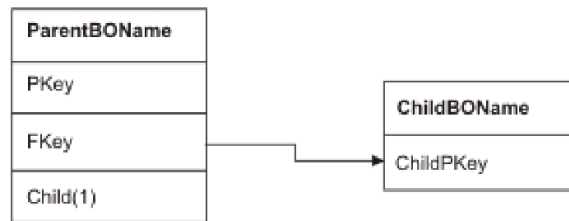


Рисунок 14. Типичное отношение с одним объектом

Поскольку внешние ключи, устанавливающие отношение, хранятся в родительском объекте, то для каждого родительского объекта может существовать только один дочерний бизнес-объект данного типа.

Родительский бизнес-объект содержит единичный принадлежащий ему дочерний бизнес-объект и единичный не принадлежащий ему дочерний бизнес-объект. Для отношений без принадлежности применяются таблицы поиска. Принадлежность указывается с помощью значения параметра Ownership в информации уровня приложения.

Отношения с одним объектом и данные, не принадлежащие объекту

Как правило, каждый родительский бизнес-объект владеет данными содержащегося в нем дочернего бизнес-объекта. Например, если каждый бизнес-объект Customer содержит один бизнес-объект Address, то при создании нового клиента (customer) новая строка вставляется как в таблицу Customer, так и в таблицу Address. Новый адрес уникален для нового клиента. Аналогично, при удалении клиента из таблицы Customer его адрес также удаляется из таблицы Address.

Возможны, однако, ситуации, когда несколько иерархических бизнес-объектов содержат одинаковые данные, которые не принадлежат ни одному из них. Предположим, например, что таблица базы данных Address содержит ссылку на таблицу поиска StateProvince. Поскольку таблица поиска обновляется редко и обслуживается независимо от данных об адресах, то создание или изменение сведений об адресах не влияет на информацию об области и районе в таблице поиска. Однако, для того чтобы вместе с бизнес-объектом Address можно было извлечь бизнес-объект StateProvince, StateProvince должен быть единичным дочерним объектом бизнес-объекта Address, а отношение должно быть определено без принадлежности данных.

Если в структуру базы данных включены таблицы поиска, то структура бизнес-объекта будет несколько отличаться от структуры базы данных. Это связано с тем, что адаптер извлекает данные только для бизнес-объекта таблицы и его дочерних бизнес-объектов таблиц. Для использования таблицы поиска необходимо

создать между таблицами отношение с одним объектом, в котором дочерний объект содержится в родительском, но не принадлежит ему. Несмотря на то, что таблица поиска StateProvince не является дочерней таблицей таблицы Address в базе данных, соответствующий бизнес-объект StateProvince является единственным дочерним объектом бизнес-объекта таблицы Address, поскольку каждый адрес содержит одну область или район. Однако, бизнес-объект StateProvince не “принадлежит” бизнес-объекту Address. При изменении адреса список областей и районов не изменяется.

Когда адаптер получает иерархический бизнес-объект с запросом на создание, удаление или обновление, то он не создает, не удаляет и не обновляет единичные дочерние бизнес-объекты, которые содержатся в родительском объекте, но не принадлежат ему. Для этих бизнес-объектов выполняются только операции Retrieve. Если адаптеру не удастся извлечь такой единичный бизнес-объект, то возвращается ошибка, и обработка прекращается; значения в бизнес-объект таблицы поиска не добавляются и не обновляются.

Денормализованные данные и данные без принадлежности

Помимо облегчения работы со статическими таблицами поиска, вложенность без принадлежности позволяет синхронизировать нормализованные и денормализованные данные.

Синхронизация нормализованных данных с денормализованными: В случае отношения без принадлежности можно создать или изменить данные при синхронизации данных нормализованного приложения с данными денормализованного приложения. Предположим, например, что в исходном нормализованном приложении данные хранятся в двух таблицах: А и В. Предположим далее, что в целевом денормализованном приложении все данные хранятся в одной таблице, так что каждый элемент А хранит избыточные данные В.

В этом случае для синхронизации данных таблицы В исходного приложения с целевым приложением необходимо инициировать событие таблицы А каждый раз, когда изменяются данные таблицы В. Кроме того, поскольку данные таблицы В хранятся в виде избыточных данных в таблице А, необходимо передать бизнес-объект для каждой строки в таблице А, который содержит измененные данные из таблицы В.

Примечание: Для того чтобы при обновлении денормализованных таблиц в результате одного обновления обновлялась только одна строка, а не несколько, убедитесь, что каждая запись имеет уникальный ключ. Если такого ключа не существует, адаптер выдаст ошибку, указывающую на обновление нескольких записей.

Синхронизация денормализованных данных с нормализованными данными: При синхронизации данных из исходного денормализованного приложения с целевым нормализованным приложением адаптер не создает, не удаляет и не обновляет данные, содержащиеся в нормализованном приложении, но не принадлежащие ему.

При синхронизации данных с нормализованным приложением адаптер игнорирует все единичные дочерние данные, содержащиеся в приложении, но не принадлежащие ему. Создание, удаление или изменение таких дочерних данных должно выполняться вручную.

Отношения с множеством объектов:

В случае отношения с множеством объектов атрибут в родительском бизнес-объекте представляет массив дочерних бизнес-объектов. Тип атрибута совпадает с типом дочернего бизнес-объекта. Внешний ключ, описывающий это отношение, хранится в дочернем объекте, за исключением случаев, когда единственный дочерний объект хранится в приложении. Тогда отношение "родительский-дочерний" хранится в родительском объекте.

Как правило, бизнес-объект, содержащий массив дочерних бизнес-объектов, имеет только один атрибут, представляющий отношение, и этот атрибут обычно является первичным ключом. Тип этого атрибута - массив того же типа, что и дочерние бизнес-объекты. Для того чтобы родительский объект мог содержать больше одного дочернего объекта, внешние ключи, устанавливающие отношение, хранятся в дочернем объекте.

Поэтому у каждого дочернего объекта есть минимум один простой атрибут, который содержит первичный ключ родительского объекта в качестве внешнего ключа. Дочерний объект имеет столько атрибутов внешних ключей, сколько атрибутов первичных ключей есть у родительского объекта.

Поскольку внешние ключи, устанавливающие отношение, хранятся в дочернем объекте, у каждого родительского объекта может быть произвольное число дочерних объектов.

Отношение с множеством объектов показано на рис. рис. 15. В этом примере ParentId в трех ChildBOName - это простой атрибут, содержащий первичный ключ родительского объекта, а Child1 в ParentBOName - это атрибут, который представляет массив дочерних бизнес-объектов.

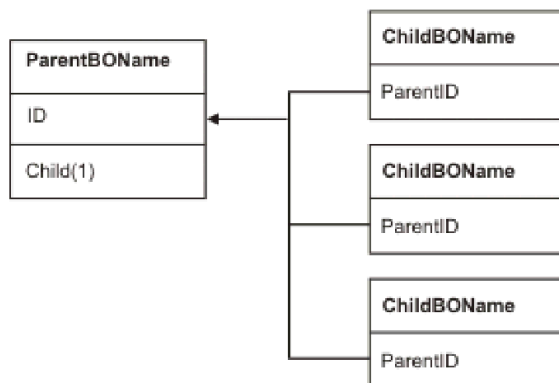


Рисунок 15. Отношение с множеством бизнес-объектов ($N > 1$)

Отношение с множеством бизнес-объектов может быть отношением с $N=1$. Некоторые приложения хранят одну дочернюю сущность, так что отношение родительский-дочерний хранится не в родительском, а в дочернем объекте. Другими словами, этот дочерний объект содержит внешний ключ, значение которого совпадает со значением, хранящимся в первичном ключе родительского объекта.

Приложения используют этот тип отношения, когда данные дочернего объекта не существуют независимо от его родительского объекта, и доступ к ним возможен только через родительский объект. Для таких данных дочернего объекта требуется,

чтобы родительский объект и значение его первичного ключа существовали до создания дочернего объекта и значения его внешнего ключа. Этот тип отношения показан на рисунке рис. 16.

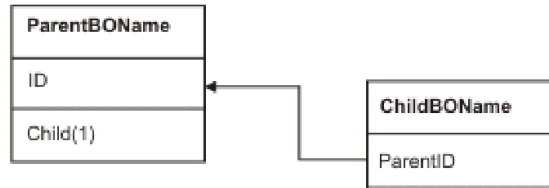


Рисунок 16. Отношение с множеством бизнес-объектов (N=1)

Таблицы базы данных с несколькими родительскими таблицами:

Если для дочерней таблицы в базе данных существует несколько родительских таблиц, необходимо с помощью редактора компоновки вручную настроить дополнительные родительские бизнес-объекты. ESW (мастер внешних служб) настраивает только один родительский объект.

Схема бизнес-объекта

Схема бизнес-объектов включает объекты базы данных, выбираемые с помощью ESW (мастер внешних служб). Каждый объект базы данных преобразуется в бизнес-объект верхнего уровня.

Схема определяет имена бизнес-объектов и информацию уровня приложения. Бизнес-объекты, их атрибуты и информация уровня приложения представляются в схеме следующим образом:

- Бизнес-объект преобразуется в определение сложного типа.
- Информация уровня приложения для бизнес-объекта содержится в аннотациях в сложном типе.
- Атрибуты бизнес-объекта преобразуются в определения типов элементов.
- Информация уровня приложения для каждого свойства в бизнес-объекте содержится в аннотациях для типов элементов.

Шаблон свойств уровня приложения для бизнес-объекта и для атрибутов определяется в схеме метаданных адаптера. Имя файла схемы - OracleEBSASI.xsd. Ссылка на этот шаблон содержится в аннотациях созданного для адаптера файла схемы.

Хранимые процедуры - Обзор

Хранимая процедура может быть бизнес-объектом, который выполняется модулем с помощью операции Execute; она может выполняться вместо стандартного оператора SQL для операции над любым бизнес-объектом, либо может выполнять дополнительные действия перед и после операции.

Хранимая процедура - это группа операторов SQL, которые образуют логический блок и выполняют конкретную задачу. Хранимая процедура инкапсулирует набор операций или запросов, выполняемых адаптером над объектом на сервере базы данных. Адаптер использует хранимые процедуры следующим образом:

- создает бизнес-объект хранимой процедуры для выполнения в базе данных

- расширяет операции бизнес-объекта путем замены оператора SQL, предоставленного для операции бизнес-объекта, или выполняет действия до или после выполнения операции

Бизнес-объект хранимой процедуры - Обзор

Можно создать бизнес-объект хранимой процедуры, который соответствует хранимой процедуре или хранимой функции в базе данных. Затем для выполнения хранимой процедуры над данными в базе данных можно использовать операцию Execute.

ESW (мастер внешних служб) позволяет создать бизнес-объекты хранимых процедур, которые выполняют хранимую процедуру или хранимую функцию. Для создания бизнес-объекта мастер проверяет хранимую процедуру или хранимую функцию в базе данных. Бизнес-объект хранимой процедуры содержит атрибут для каждого параметра.

Если атрибут параметра имеет простой тип данных, то существует атрибут для образца значения параметра. Образцы значений используются мастером при проверке хранимой процедуры перед ее сохранением. Адаптер использует результат хранимой процедуры для проверки параметров, получения максимального числа возвращаемых наборов результатов и для генерации дочерних бизнес-объектов на основе метаданных из этих наборов результатов. При проверке бизнес-объекта хранимой процедуры мастер автоматически создает иерархию бизнес-объектов хранимых процедур.

Если хранимая процедура имеет входные и выходные параметры или параметры возвращаемого значения со сложным типом данных, таким как Struct, Array или ResultSet, для такого параметра в мастере необходимо выбрать соответствующий тип данных и задать имя соответствующего пользовательского типа. Для параметров типа Struct или Array необходимо также задать имя соответствующего пользовательского типа, который хранится в свойстве SPComplexParameterTypeName.

Например, если вы создаете в базе данных один объект Struct с именем Struct_TEMP и задаете тип в качестве одного входного параметра, то для этого свойства необходимо задать значение Struct_TEMP. Это имя типа используется мастером для определения метаданных, которые должны создаваться для соответствующего дочернего бизнес-объекта. Если хранимая процедура возвращает ResultSet, то в свойстве MaxNumberOfResultSets необходимо задать число возвращаемых ею наборов результатов. Это значение представляет собой максимальное число возвращаемых наборов результатов, которые будут обрабатываться средой выполнения адаптера.

Во время поиска и во время выполнения WebSphere Adapter for Oracle E-Business Suite ожидает, что значение ResultSets, возвращаемое в результате выполнения хранимой процедуры, содержит столбцы с именами. Некоторые хранимые процедуры возвращают наборы результатов со столбцами без имени. Например, хранимая процедура с операторами SQL, аналогичными показанным в следующих примерах, будет возвращать ResultSets со столбцами без имени:

```
SELECT COUNT(*) FROM EMPLOYEE;  
SELECT 111,222,333 FROM CUSTOMER;
```

Oracle обрабатывает такие операторы SQL SELECT, присваивая столбцам таблицы в возвращаемом наборе результатов "фиктивные" имена. Например, для показанных выше операторов select это могут быть имена count(*) или d1, d2, d3, соответственно.

Если возвращаемый набор результатов содержит столбцы таблицы без имени (поскольку база данных не присваивала фиктивные имена), адаптер создает для таких столбцов фиктивные имена.

Фиктивные имена столбцов, сгенерированные или базой данных, или адаптером, присваиваются атрибутам бизнес-объекта хранимой процедуры.

Присвоение (адаптером или базой данных) фиктивных имен неименованным столбцам таблицы обеспечивает успешное выполнение хранимой процедуры во время поиска и во время выполнения.

Для бизнес-объектов хранимых процедур мастер поддерживает вложенные объекты Struct и Array и может поддерживать любое число уровней вложенной иерархии. Для всех таких вложенных объектов Struct и Array мастер может генерировать соответствующие дочерние бизнес-объекты.

Таблица 4. Свойства сложных типов данных для бизнес-объектов хранимых процедур

Свойство	Тип	Описание
SPComplexParameterType	String	Допустимо одно из следующих значений: Array ResultSet Struct
SPComplexParameterTypeName	String	Имя пользовательского типа. Это свойство является обязательным, если для свойства SPComplexParameterType указано значение Struct или Array.
MaxNumberOfResultSets	Integer	Максимальное число возвращаемых наборов результатов, обрабатываемых средой выполнения Adapter for Oracle E-Business Suite. Мастер создает такое количество бизнес-объектов.

Хранимые процедуры, используемые вместо операций или в дополнение к ним

Можно указать, что адаптер должен выполнять хранимую процедуру в базе данных перед, после или вместо операторов SQL для операции. Каждый объект может иметь различный набор хранимых процедур, используемых при каждой операции.

Для операций Create, Update, Delete, Retrieve или RetrieveAll адаптер может использовать простые операторы SQL. Имена столбцов в этих операторах SQL определяются на основе информации уровня приложения для атрибута. Для составления блока WHERE применяются значения ключей, указанные в бизнес-объекте. Каждый запрос охватывает только одну таблицу, если только он не опубликован в представлении. Однако, оператор SQL, предоставляемый адаптером, можно заменить или расширить с помощью хранимых процедур и хранимых функций.

Хранимая процедура или хранимая функция может вызываться адаптером в следующих случаях:

- Перед обработкой бизнес-объекта - для выполнения подготовительных процессов
- После обработки бизнес-объекта - для выполнения действий после операции
- Для выполнения набора операций над бизнес-объектом, вместо использования простого оператора Create, Update, Delete, Retrieve или RetrieveAll.

В бизнес-объекте с иерархической структурой для выполнения хранимой процедуры над всем бизнес-объектами необходимо отдельно связать хранимую процедуру с бизнес-объектом верхнего уровня и каждым дочерним бизнес-объектом. Если связать хранимую процедуру только с бизнес-объектом верхнего уровня, то хранимая процедура будет обрабатывать только бизнес-объект верхнего уровня - дочерние бизнес-объекты будут обрабатываться с помощью стандартного запроса SQL.

В таблице Табл. 5 перечислены элементы информации уровня приложения для хранимой процедуры, а также описывается их назначение и особенности использования. Подробные описания всех элементов приведены в разделах, следующих за таблицей. На рис. “Вид бизнес-объекта с определением хранимой процедуры” на стр. 39 показано определение хранимой процедуры для бизнес-объекта.

Таблица 5. Информация уровня приложения для хранимых процедур в бизнес-объектах таблиц и представлений

Описательное имя	Имя элемента	Назначение
Тип хранимой процедуры	StoredProcedureType	Определяет тип используемой хранимой процедуры и указывает, когда она вызывается (например, перед обработкой бизнес-объекта).
Имя хранимой процедуры	StoredProcedureName	Имя хранимой процедуры, связанной с соответствующим StoredProcedureType.
Набор результатов	ResultSet	Указывает, возвращает ли хранимая процедура набор результатов. Если возвращается набор результатов, то для текущего бизнес-объекта создается множественный дочерний объект с использованием значений, возвращаемых в строках этого набора результатов.
Параметры	Parameters	Каждый элемент Parameters описывает один параметр хранимой процедуры или хранимой функции.
Возвращаемое значение	ReturnValue	Указывает, что это вызов функции, а не вызов процедуры, поскольку значение возвращается хранимой процедурой.

Тип хранимой процедуры

Этот параметр определяет тип используемой хранимой процедуры и указывает, когда она вызывается (например, перед обработкой бизнес-объекта).

Таблица 6. Элемент Тип хранимой процедуры - Характеристики

Обязательный	Да
Значение по умолчанию	Нет

Таблица 6. Элемент Тип хранимой процедуры - Характеристики (продолжение)

Допустимые значения	<p>Допустимо одно из следующих значений:</p> <ul style="list-style-type: none"> • BeforeOperationSP • AfterOperationSP • OperationSP <p>где Operation - имя одной из следующих операций: Create, Update, Delete, Retrieve или RetrieveAll.</p>
Поддерживается преобразование двунаправленного текста	Нет
Тип свойства	String
Замечания об использовании	<p>Типы хранимых процедур, связанные с RetrieveAll, применимы только к бизнес-объектам верхнего уровня.</p> <p>Из свойства StoredProcedureType можно удалить любую выбранную информацию уровня приложения. Будут удалены также все соответствующие группы свойств информации уровня приложения для операции.</p>
Примеры	<ul style="list-style-type: none"> • CreateSP: выполняет операцию create • UpdateSP: выполняет операцию update • BeforeCreateSP: выполняется перед созданием бизнес-объекта • AfterCreateSP: выполняется после создания бизнес-объекта • AfterDeleteSP: выполняется после удаления бизнес-объекта

Имя хранимой процедуры

Имя хранимой процедуры, связанной с соответствующим StoredProcedureType.

Таблица 7. Элемент Имя хранимой процедуры - Характеристики

Обязательный	Да
Значение по умолчанию	Нет
Поддерживается преобразование двунаправленного текста	Да
Тип свойства	Строка

Набор результатов

Это значение определяет, возвращает ли хранимая процедура набор результатов. Если возвращается набор результатов, то для текущего бизнес-объекта создается множественный дочерний объект с использованием значений, возвращаемых в строках этого набора результатов.

Таблица 8. Элемент Набор результатов - Характеристики

Обязательный	Да
Значение по умолчанию	Нет

Таблица 8. Элемент Набор результатов - Характеристики (продолжение)

Допустимые значения	True False
Поддерживается преобразование двунаправленного текста	Нет
Тип свойства	Boolean
Замечания об использовании	Если хранимая процедура возвращает набор результатов, то после завершения работы с ESW (мастер внешних служб) убедитесь с помощью редактора бизнес-объектов, что значение этого атрибута равно true. Драйвер JDBC Oracle не всегда правильно возвращает это значение.

Параметры

Для каждого параметра хранимой процедуры или хранимой функции существует один элемент Parameters. Каждый элемент Parameters определяет имя и тип одного параметра.

Таблица 9. Элемент Параметры - Характеристики

Обязательный	Да
Значение по умолчанию	Нет
Содержимое	В элементе Parameters указывается следующая информация: <ul style="list-style-type: none"> • PropertyName: имя атрибута бизнес-объекта для передачи в качестве параметра. • Type: тип параметра, который может быть одним из следующих: <ul style="list-style-type: none"> – IP - только входной – OP - только выходной – IO - входной и выходной – RS - набор результатов
Поддерживается преобразование двунаправленного текста	Нет
Тип свойства	String
Замечания об использовании	Набор результатов может возвращаться только как выходной параметр. В этом случае один из параметров должен иметь тип RS, указывающий на набор результатов.

Возвращаемое значение

Указывает, что это вызов функции, а не вызов процедуры, поскольку возвращается значение.

Таблица 10. Элемент Возвращаемое значение - Характеристики

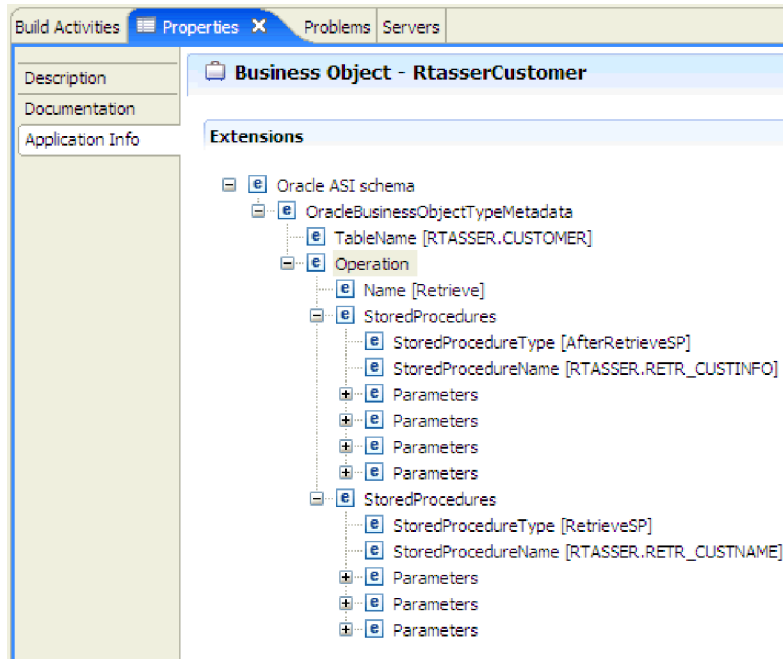
Обязательный	Нет
Значение по умолчанию	Нет
Допустимые значения	RS или имя атрибута бизнес-объекта или дочернего бизнес-объекта.

Таблица 10. Элемент Возвращаемое значение - Характеристики (продолжение)

Поддерживается преобразование двунаправленного текста	Нет
Тип свойства	String
Замечания об использовании	<p>Если для этого элемента задано значение RS, то возвращаемое значение представляет собой набор результатов и используется для создания контейнера с несколькими объектами, соответствующего данному бизнес-объекту. Если возвращаемое значение - имя атрибута, то оно присваивается данному конкретному атрибуту в бизнес-объекте. Если атрибут - это другой дочерний бизнес-объект, то адаптер возвращает ошибку.</p> <p>Если хранимая процедура, связываемая с бизнес-объектом, создаваемым на основе таблицы или представления, представляет собой функцию, то из этой хранимой процедуры возвращается значение. В информацию уровня приложения (ASI) для операции добавляется одно значение ASI ReturnValue. Существование этого параметра означает, что это вызов функции, а не вызов хранимой процедуры, поскольку значение возвращается функцией.</p> <p>Если значение этого параметра ASI - имя атрибута бизнес-объекта, то возвращаемое значение присваивается данному конкретному атрибуту в бизнес-объекте.</p> <p>Если значение - другой дочерний бизнес-объект, то среда выполнения адаптера возвращает ошибку.</p> <p>В общем, если возвращаемое значение имеет простой тип данных, то мастер позволяет связать с ним один атрибут бизнес-объекта, и в качестве значения этого элемента ASI используется имя этого атрибута. Но если возвращаемое значение представляет собой набор результатов, то мастер задает для этого элемента ASI значение RS.</p> <p>Примечание: Набор результатов может возвращаться как выходной параметр, а если это хранимая функция - как возвращаемое значение. В качестве типа выходного параметра задается значение RS, чтобы указать, что этот параметр используется для возврата набора результатов.</p>

Вид бизнес-объекта с определением хранимой процедуры

На следующем рисунке показано окно Свойства с бизнес-объектом customer, для которого существует связанная информация о типах хранимых процедур RetrieveSP и AfterRetrieveSP для операции Retrieve. Для извлечения бизнес-объекта таблицы вместо стандартного оператора SQL адаптер выполняет хранимую процедуру RTASSER.RETR_CUSTNAME. После извлечения бизнес-объекта адаптер выполняет хранимую процедуру RTASSER.RETR_CUSTINFO.



Хранимые функции - Обзор

База данных Oracle поддерживает не только хранимые процедуры, но и хранимые функции. Хранимые функции всегда возвращают значение, в остальном они аналогичны хранимым процедурам и точно так же поддерживаются адаптером.

Адаптер поддерживает хранимые функции, создаваемые пользователем с помощью оператора CREATE FUNCTION. Иногда функцию этого типа называют *пользовательской функцией* (UDF), однако этот термин более типичен, когда говорят о хранимой функции или процедуре Java, которые не поддерживаются адаптером.

Синтаксис вызова функции следующий:

? = call ИмяФункции список-параметров

Сравните его с синтаксисом вызова хранимой процедуры:

call ИмяХранимойПроцедуры список-параметров

Атрибут, содержащий возвращаемое значение, задается с помощью параметра ReturnValue информации уровня приложения для бизнес-объекта.

Бизнес-объект запроса - Обзор

Бизнес-объекты запросов выполняют пользовательский оператор SELECT в базе данных и возвращают совпадающие записи в бизнес-объектах.

ESW (мастер внешних служб) позволяет создавать бизнес-объекты запросов, которые выполняют пользовательские операторы SELECT в базе данных. В операторе SELECT вместо заменяемых параметров указывается ? (вопросительный знак). Затем мастер открывает область, в которой для каждого параметра можно указать тип данных и образец значения. Образец значения должен совпадать с данными в базе данных, так как результаты оператора SELECT используются мастером для создания бизнес-объекта запроса.

Перед тем как сохранить конфигурацию запроса в мастере, проверьте ее. При проверке мастер выполняет оператор SELECT с образцами значений. После получения набора результатов мастер анализирует метаданные для получения имен и типов всех столбцов. При этом для каждого столбца возвращаемого набора результатов создается соответствующий атрибут в бизнес-объекте запроса. Для каждого параметра блока WHERE мастер создает в бизнес-объекте запроса один атрибут **jdbewhereclause** и присваивает ему значение по умолчанию, представляющее собой блок WHERE. Эти атрибуты используются во время выполнения для генерации динамического блока WHERE, заменяющего блок WHERE по умолчанию.

Предположим, например, что указан следующий оператор SELECT:

```
select * from customer where fname=? and age=?
```

В этом блоке WHERE два параметра, обозначенные вопросительным знаком (?). Тип данных первого параметра - **string**; он совпадает с типом данных столбца fname. Второй параметр имеет тип данных **int**, совпадающий со столбцом age. Если в базе данных есть запись таблицы customer, в которой столбец fname содержит строку Mike, а столбец age - целое число 27, то эти значения можно указать в качестве образцов значений при настройке бизнес-объекта запроса. Мастер настроит бизнес-объект таким образом, чтобы он соответствовал возвращаемому набору результатов.

ESW (мастер внешних служб)

ESW (мастер внешних служб), входящий в состав WebSphere Integration Developer, позволяет выполнять поиск объектов в базе данных, генерировать бизнес-объекты запросов, а также создавать бизнес-объекты на основе выбранных объектов базы данных. Кроме того, мастер создает модуль и артефакты службы, позволяющие адаптеру работать в качестве компонента архитектуры компонентов служб (SCA).

Глава 2. Планирование реализации адаптера

Прежде, чем начать работу с WebSphere Adapter for Oracle E-Business Suite, необходимо досконально разобраться в среде сервера, в которой он будет работать. Ознакомьтесь с замечаниями по развертыванию адаптера в серверной среде и узнайте, как можно повысить быстродействие и готовность адаптера с помощью кластерной среды.

Перед тем, как начать

Для настройки и развертывания модели необходимо подробно ознакомиться с основными концепциями интеграции бизнес-процессов, особенностями работы Java Database Connectivity (JDBC) и продуктов базы данных в вашей среде, а также с функциями и возможностями WebSphere Integration Developer и WebSphere Process Server или WebSphere Enterprise Service Bus.

Для настройки и развертывания WebSphere Adapter for Oracle E-Business Suite необходимо понимание следующих концепций, а также опыт работы с перечисленными ниже инструментами и задачами:

- Бизнес-требования разрабатываемого решения.
- JDBC и продукты базы данных в вашей среде. Этот пункт включает проблемы доступа к данным, модели транзакций и соединения через неоднородные реляционные базы данных, запросы и Web-службы.
- Концепции интеграции бизнес-процессов, в том числе программная модель архитектуры компонентов служб (SCA).
- Возможности и требования сервера, на основе которого предполагается создать решение интеграции. Требуются навыки настройки и администрирования сервера хоста и работы с административной консолью для настройки определений свойств и фабрик соединений, а также для управления событиями.
- Инструменты и возможности, предоставляемые WebSphere Integration Developer. Требуются навыки применения этих инструментов для создания модулей, связывания и тестирования компонентов, а также выполнения других задач интеграции.

Поддержка защиты конфиденциальных данных в файлах протоколов и трассировки

Адаптер позволяет запретить отображение конфиденциальной информации в файлах протоколов и трассировки.

Файлы протоколов и трассировки адаптера могут содержать данные конфиденциальную информацию из Oracle DB. В некоторых случаях эти файлы доступны для просмотра несанкционированным пользователям. Например, сотрудник службы поддержки использует файлы протоколов и трассировки для устранения неполадок.

Для того чтобы обеспечить защиту данных в таких ситуациях, адаптер позволяет разрешить или запретить отображение конфиденциальных данных в файлах протоколов и трассировки. Эту функцию можно настроить с помощью ESW (мастер

внешних служб) или путем изменения значения свойства HideConfidentialTrace. Если это свойство включено, то адаптер заменяет конфиденциальные данные символами XXX.

Дополнительная информация приведена в разделе “Свойства фабрики управляемых соединений” на стр. 176.

Ниже перечислены типы информации, относящейся к категории конфиденциальной:

- Содержимое бизнес-объектов.
- Содержимое ключа объекта записи события.
- Имя пользователя, пароль, среда и роль
- URL, применяемый для подключения к Oracle DB.

Ниже перечислены типы информации, которые не считаются пользовательским данными:

- Содержимое записи события, не входящее в состав ключа объекта. Например, XID, ID события, имя бизнес-объекта и состояние события.
- Схемы бизнес-объектов.
- ID транзакций.
- Последовательности вызовов.

Защита

Адаптер используется запись данных аутентификации J2C или псевдонимы аутентификации, компонент защиты Java 2, для предоставления защиты по имени пользователя и паролю. Дополнительная информация о функциях защиты приведена в документации по WebSphere Process Server или WebSphere Enterprise Service Bus.

Идентификация пользователей

Адаптер поддерживает несколько способов указания имени пользователя и пароля, требуемых для подключения к Oracle DB. Ознакомьтесь с особенностями и ограничениями каждого способа, чтобы выбрать тот, который обеспечивает необходимый уровень защиты и удобства работы приложения.

Для интеграции адаптера в приложение имя пользователя и пароль указываются в следующих случаях:

- Когда ESW (мастер внешних служб) подключается к Oracle DB для получения или *обнаружения* информации об объектах и службах, к которым будет иметь доступ адаптер.
- Во время выполнения в WebSphere Process Server или WebSphere Enterprise Service Bus при подключении адаптера к Oracle DB для обработки исходящих запросов и входящих событий.

Идентификация в мастере

ESW (мастер внешних служб) запрашивает информацию о соединении для процесса поиска и использует ее в качестве значений по умолчанию для свойств адаптера, в которых указывается информация о соединении, применяемая во время выполнения. При работе мастера можно использовать другие имя пользователя и пароль, чем при развертывании приложения на сервере. Можно даже подключиться к другому Oracle DB, хотя схема должна быть одной и той же в обеих базах данных. Например, в ходе разработки и интеграции приложения, которое использует Adapter for Oracle

E-Business Suite, можно работать с тестовой базой данных с меньшим числом записей. При этом разработка будет идти быстрее, рабочая база данных не будет загружена, и никакие требования к конфиденциальности данных заказчиков не будут нарушены.

Мастер использует имя пользователя и пароль, указанные для процесса поиска, только во время поиска. Эти имя пользователя и пароль не будут доступны по завершении работы мастера.

Идентификация во время выполнения

Во время выполнения адаптер должен предоставить имя пользователя и пароль, чтобы подключиться к Oracle DB. Для того чтобы подключиться без участия пользователя, адаптер должен сохранить эти данные локально. В среде сервера есть несколько способов сохранения информации пользователей. ESW (мастер внешних служб) позволяет настроить любое из следующих средств получения информации о пользователе адаптером:

- Свойства адаптера
- Источник данных
- Псевдоним идентификации J2C

Если имя пользователя и пароль сохранены в свойствах адаптера, то эта информация будет доступна во время выполнения. Вы указываете это имя пользователя и пароль, когда вы настраиваете модуль с помощью ESW (мастер внешних служб). Этот способ, будучи самым простым, имеет важные ограничения. Свойства адаптера не зашифрованы, и пароль сохраняется открытым текстом в полях, к которым могут иметь доступ посторонние. Кроме того, при изменении пароля его необходимо обновить во всех экземплярах адаптера, которые обращаются к Oracle DB, включая адаптеры, встроенные в файлы EAR приложений, а также автономные адаптеры.

Применение источника данных позволяет воспользоваться соединением, уже установленным для другого приложения. Например, если к базе данных обращается несколько приложений с одинаковыми именем пользователя и паролем, то приложения можно развернуть с одним и тем же источником данных. Имя пользователя и пароль будут известны только тому, кто первым развернул приложение с этим источником данных или определил источник данных отдельно.

Применение данных идентификации (псевдонима идентификации J2C), созданных с помощью Службы идентификации Java (JAAS), входящей в состав средств защиты Java 2, является надежным и безопасным способом развертывания приложений. Администратор создает псевдоним идентификации, применяемый приложениями, которым требуется доступ к системе. Имя пользователя и пароль будут известны только администратору, который может внести необходимые изменения только в одном месте.

Варианты развертывания

Адаптер можно развернуть двумя способами: в составе развернутого приложений или в качестве автономного файла RAR. Выбор конкретного типа развертывания зависит от требований среды.

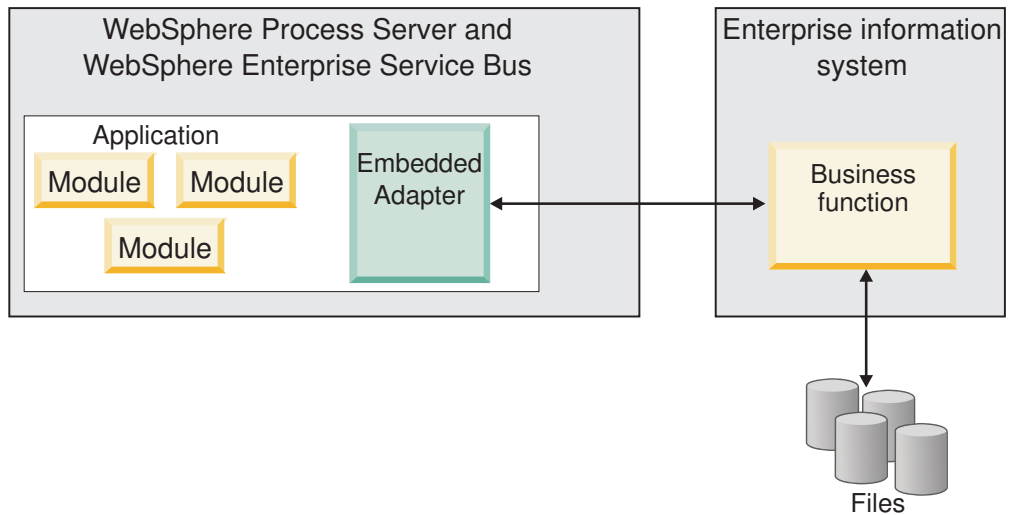
Опции развертывания описаны ниже:

- **С модулем для использования одним приложением.** Модуль, содержащий файлы адаптера, можно развернуть на любом сервере приложений. Если необходимо, чтобы адаптер использовался единственным модулем или несколькими модулями,

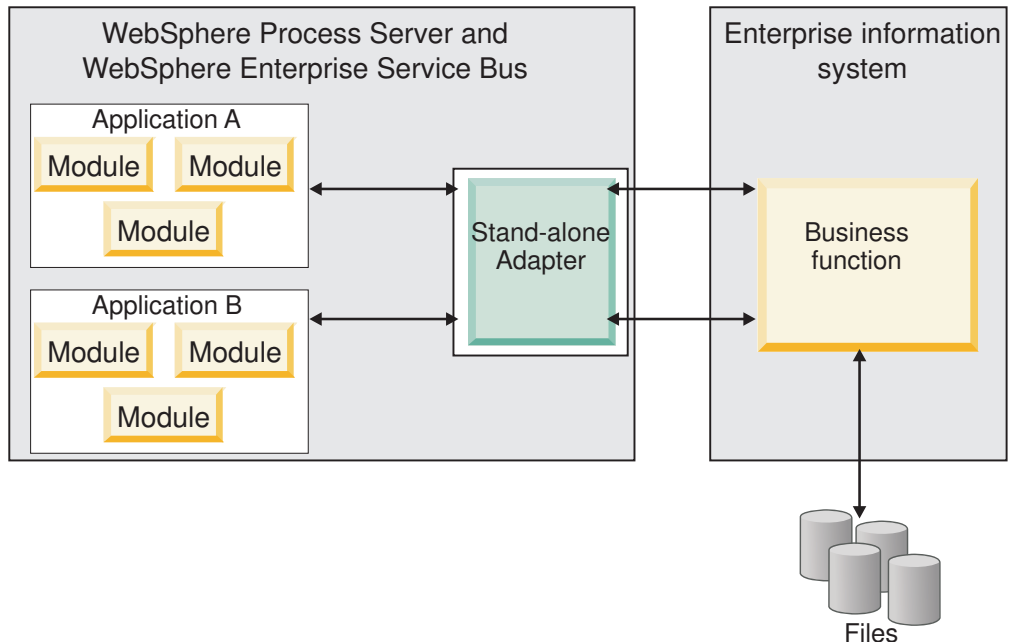
которые будут запускать различные версии адаптера, воспользуйтесь встроенным адаптером. Это позволит обновлять адаптер в одном модуле без риска дестабилизировать сменой версии адаптера другие модули.

- **На сервере для использования несколькими приложениями.** Если вы не включаете файлы адаптера в модуль, то вы должны установить их как автономный адаптер на каждом сервере приложений, на котором вы хотите запустить модуль. Автономный адаптер следует использовать в том случае, если несколько модулей могут применять одну версию адаптера и вы хотите управлять адаптером в центральном расположении. Применение автономного адаптера позволяет также сократить количество требуемых ресурсов.

Встроенный адаптер поставляется в составе файла EAR и доступен только приложению, вместе с которым он развертывается.



Автономный адаптер поставляется в отдельном файле RAR. После развертывания он доступен всем приложениям экземпляра сервера.



В ходе создания проекта приложения с помощью WebSphere Integration Developer можно выбрать способ упаковки адаптера: в файле EAR приложения или в отдельном файле RAR. Конкретный выбор зависит от способа применения адаптера в среде выполнения, а также свойств адаптера в административной консоли.

Выбор способа развертывания адаптер зависит от того, каким образом в дальнейшем планируется администрировать адаптер. Если требуется отдельная копия адаптера и не имеет значения, что после ее обновления может быть нарушена работа приложений, то адаптер можно развернуть в качестве автономного модуля.

Если планируется использовать несколько версий и важно обеспечить правильную работу приложений в случае обновления адаптера, то рекомендуется выбрать адаптер, встроенный в приложение. Такой подход позволяет связать версию адаптера с версией приложения и управлять ими как одним модулем.

Особенности встроенных адаптеров

Если вы планируете встроить адаптер в приложение, то обратите внимание на следующие особенности:

- Встроенный адаптер предусматривает изоляцию загрузчика классов. Загрузчик классов влияет на процесс создания пакетов приложений, а также работу таких приложений в среде выполнения. *Изоляция загрузчика классов* означает, что адаптер не может загружать классы из другого приложения или модуля. Изоляция загрузчика классов предотвращает взаимное влияние двух классов, названных подобным образом, из различных приложений.
- Каждое приложение, содержащее встроенный адаптер, администрируется отдельно от других приложений.

Особенности автономных адаптеров

Если вы планируете использовать автономный адаптер, то обратите внимание на следующие особенности:

- Автономные адаптеры не предусматривают изоляцию загрузчика классов. Так как автономные адаптеры не имеют изоляции загрузчика классов, выполняется только одна версия любого данного артефакта Java, а версия и последовательность этого артефакта не определены. Например, при использовании автономного адаптера существует только *одна* версия адаптера, *одна* версия класса основания адаптера (AFC) и *одна* версия JAR другой фирмы. Все автономные адаптеры обращаются к одной версии AFC; все экземпляры конкретного адаптера используют одну версию кода. Кроме того, все адаптеры используют одну и ту же версию внешней библиотеки.
- Обновление одного из общих артефактов отражается во всех использующих его приложениях. Например, если адаптер работает с сервером версии X, то после обновления приложения-клиента до версии Y исходное приложение может перестать работать.
- AFC совместим с предыдущими версиями, однако последняя версия AFC должна входить в состав всех автономно развертываемых файлов RAR. Если в пути к классам автономного адаптера указано несколько копий файла JAR, то одна из них выбирается случайным образом. По этой причине все файлы должны быть последней версии.

WebSphere Adapters в кластерных средах

Для повышения производительности и готовности адаптера можно развернуть модуль в кластере. Этот модуль копируется на все серверы, входящие в состав кластера, независимо от выбранного способа развертывания адаптера.

WebSphere Process Server или WebSphere Enterprise Service Bus, WebSphere Application Server Network Deployment и WebSphere Extended Deployment поддерживают кластерные среды. Кластеры - это группы серверов, управляемых вместе для распределения рабочей нагрузки, обеспечивающих высокий коэффициент готовности и масштабируемость. При установке кластера сервера создается профиль Администратора развертывания. Компонент HAManager, входящий в состав диспетчера развертывания, сообщает контейнеру Java 2 Platform, Enterprise Edition (J2EE) Connector Architecture (JCA) о необходимости активировать экземпляр адаптера. Контейнер JCA предоставляет среду выполнения для экземпляра адаптера. Дополнительная информация о создании кластеров приведена по следующему адресу: http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.nd.doc/info/ae/ae/trun_wlm_cluster_v61.html.

Using WebSphere Extended Deployment позволяет повысить производительность экземпляров адаптера в кластере. WebSphere Extended Deployment расширяет возможности WebSphere Application Server Network Deployment за счет применения динамического администратора полезной нагрузки вместо статического. Динамический администратор полезной нагрузки оптимизирует производительность экземпляров адаптера в кластере путем динамического распределения запросов. Такой подход предусматривает автоматический запуск и остановку серверов приложений в соответствии с текущим уровнем нагрузки, обеспечивая равномерную обработку запросов в системах с различными ресурсами и конфигурациями. Дополнительная информация о преимуществах WebSphere Extended Deployment находится по ссылке: <http://publib.boulder.ibm.com/infocenter/wxdinfo/v6r1/index.jsp>.

В кластерах экземпляры адаптеров обрабатывают как входящие события, так и исходящие запросы.

Высокая готовность для входящих событий

Обработка входящих событий управляется событиями, связанными с обновлением данных в Oracle DB. WebSphere Adapter for Oracle E-Business Suite отслеживает обновления путем опроса таблицы событий. Затем адаптер публикует событие в его конечной точке.

При развертывании модуля адаптера в кластере контейнер JCA проверяет свойство `enableHASupport` адаптера ресурса. Если для свойства `enableHASupport` указано значение `true` (значение по умолчанию), то все экземпляры адаптера зарегистрированы в HAManager со стратегией 1 из N. В рамках этой стратегии события опрашивает и получает только один экземпляр адаптера. Хотя другие экземпляры адаптера запущены в кластере, они находятся в спящем состоянии по отношению к активному событию, до тех пор пока активный экземпляр адаптера не завершит обработку события. Если по каким-либо причинам сервер, на котором запущена нить опроса, завершит свою работу, то активизируется экземпляр адаптера, выполняющийся на одном из резервных серверов.

Важное замечание: Не изменяйте значение свойства `enableHASupport`.

Высокая готовность для исходящих запросов

В кластерах экземпляры адаптера могут посылать исходящие запросы. Таким образом, если в среде несколько приложений взаимодействуют с одним и тем же адаптером WebSphere Adapter for Oracle E-Business Suite для исходящих запросов, для повышения производительности можно развернуть модуль в кластере. Кластер допускает одновременную обработку нескольких исходящих запросов, связанных с разными записями.

Если несколько исходящих запросов пытаются обработать одну и ту же запись, например, Адрес клиента, то функция управления нагрузкой WebSphere Application Server Network Deployment распределяет запросы между доступными экземплярами адаптеров в порядке получения. В результате операции по обработке таких исходящих запросов в кластере не отличаются от операций в средах с одним сервером: за один раз один экземпляр адаптера обрабатывает только один исходящий запрос. Дополнительная информация об управлении полезной нагрузкой приведена по следующему адресу: http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.nd.doc/info/ae/ae/trun_wlm.html.

Поддержка кэширования подготовленных операторов

WebSphere Adapter for Oracle E-Business Suite поддерживает кэширование подготовленных операторов сервером, что может сократить время, необходимое для выполнения исходящей или входящей операции или пакетных операций.

Адаптер использует *подготовленные операторы*, представляющие собой объекты Java, содержащие оператор SQL QUERY, который компилируется один раз и может выполняться многократно. Для оптимизации обработки подготовленных операторов сервер кэширует их. Если для адаптера должно выполняться кэширование подготовленного оператора, определите с помощью административной консоли источник данных и включите для него кэширование. Затем настройте адаптер для работы с этим источником данных, выбрав один из следующих способов:

- при первоначальной настройке адаптера с помощью ESW (мастер внешних служб) задайте имя JNDI источника данных
- задайте свойство DataSourceJNDIName в административной консоли

Глава 3. Примеры и учебники

На Web-сайте примеров и учебников управления бизнес-процессами доступны различные примеры и учебники, призванные облегчить работу с WebSphere Adapters.

Просмотреть примеры и учебники можно следующими способами:

- Со страницы приветствия, которая открывается при запуске WebSphere Integration Developer. Для просмотра примеров и руководств для WebSphere Adapter for Oracle E-Business Suite нажмите **Извлечь**. Затем совершите обзор показанных для выбора категорий.
- По следующему адресу: Business Process Management Samples and Tutorials <http://publib.boulder.ibm.com/bpcsamp/index.html>.

Глава 4. Настройка модуля для развертывания

Настройка адаптера с целью развертывания на сервере WebSphere Process Server или WebSphere Enterprise Service Bus предусматривает создание модуля, экспортируемого в качестве файла EAR в ходе развертывания адаптера, с помощью WebSphere Integration Developer. Затем указываются бизнес-объекты для поиска и система, в которой их следует искать.

Обзор процедуры настройки модуля

Перед тем, как приступить к работе с WebSphere Adapter for Oracle E-Business Suite в среде выполнения, необходимо настроить модуль. Подробное знакомство с этой задачей позволит выполнить ее наиболее эффективно.

Модуль можно настроить для использования адаптера с помощью WebSphere Integration Developer. На следующем рисунке показана последовательность задачи настройки, а затем приведены шаги, описывающие ее на высоком уровне. В следующих разделах приведены более подробные инструкции по выполнению каждого шага.

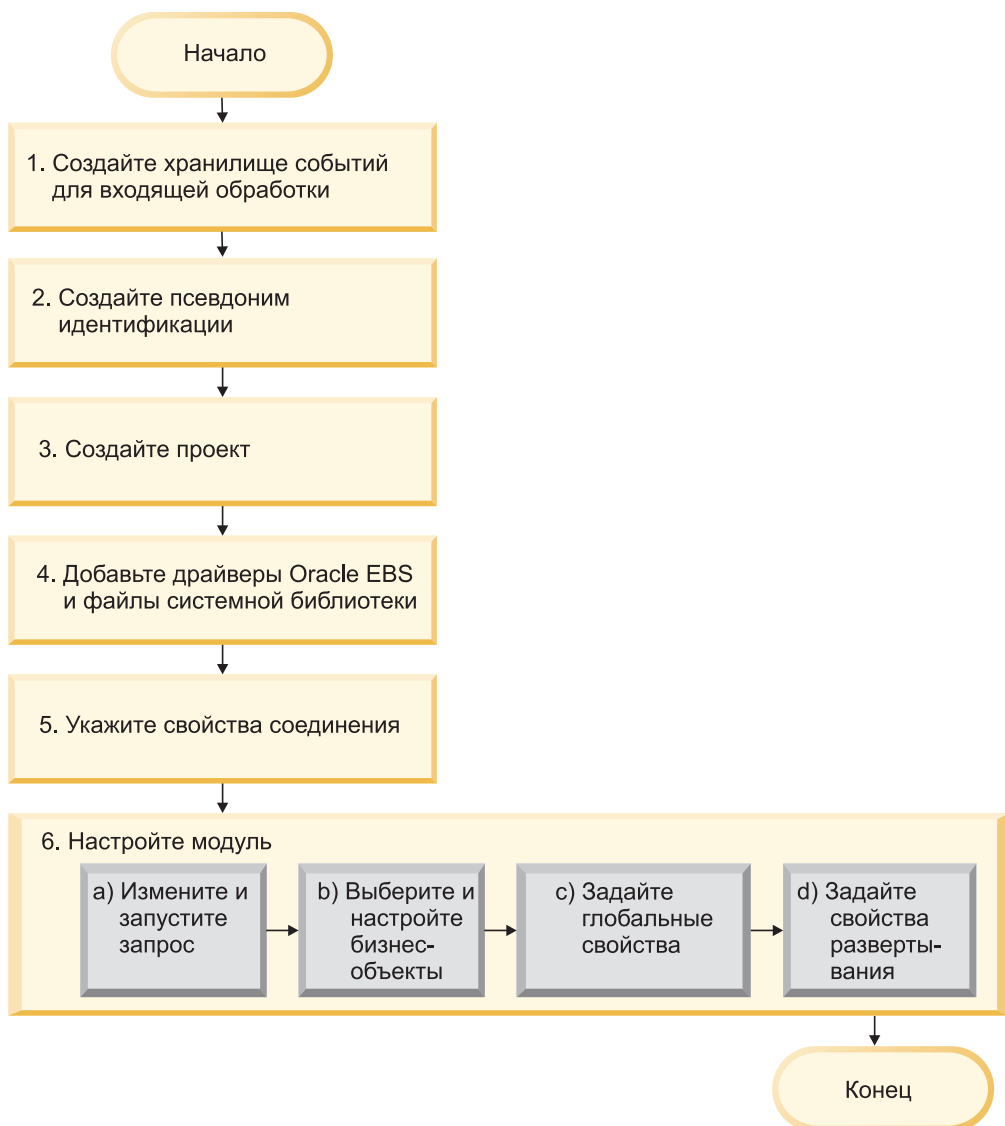


Рисунок 17. Обзор процедуры настройки модуля

Настройка модуля для развертывания

Эта задача состоит из следующих шагов:

1. Создайте хранилище событий для входящей обработки.
2. Создайте псевдоним идентификации для получения доступа к Oracle DB с использованием шифрованного пароля. Этот шаг необязательный, его необходимость зависит от стратегии обработки паролей и ИД. Этот шаг можно выполнить на сервере.
3. Создайте проект. Сперва запустите ESW (мастер внешних служб) в WebSphere Integration Developer, чтобы начать процесс создания и развертки модуля. Мастер создает проект, применяемый для организации связанных с модулем файлов.
4. Добавьте драйверы JDBC и исходные файлы библиотеки системы, необходимые для WebSphere Adapter for Oracle E-Business Suite для выполнения проекта. Эти зависимости также необходимы при экспорте модуля как файла EAR и при развертке файла EAR на сервере.

5. Укажите свойства соединения, необходимые ESW (мастер внешних служб) для подключения к Oracle DB с целью поиска объектов и служб.
6. Настройте модуль для входящей или исходящей обработки с помощью ESW (мастер внешних служб), который позволяет найти и выбрать бизнес-объекты и службы в Oracle DB, а также создать определения бизнес-объектов и связанные артефакты.
 - a. Измените и запустите запрос, который находит объекты базы данных, к которой у вас есть доступ.
 - b. Выберите и настройте бизнес-объекты для обработки входящих или исходящих запросов.
 - c. Задайте глобальные свойства операций.
 - d. Задайте свойства развертки, используемые адаптером для подключения к Oracle DB во время работы. Затем создайте службу с помощью ESW (мастер внешних служб) для сохранения нового модуля, который содержит настроенный(ые) бизнес-объект(ы), файл импорта или экспорта, а также интерфейс службы.

Создание хранилища событий

Необходимо создать хранилище событий в базе данных, перед тем как адаптер сможет обрабатывать входящие события. Можно установить триггеры в пользовательских таблицах для заполнения хранилища событий.

Описание задачи

Выполните эту задачу, только если требуется входящая обработка событий. Создайте хранилище событий в базе данных, которая содержит таблицы, для которых сгенерированы события.

Процедура

1. Создайте хранилище событий. Предоставлены следующие сценарии примеров создания хранилища событий для базы данных Oracle:
 - `ibm_websphere_event_table_create.sql`Эти файлы расположены в каталоге `каталог_установки_WID/ResourceAdapters/OracleEBS_версия/samples/scripts`, где `каталог_установки_WID` - это каталог установки для WebSphere Integration Developer, а `версия` указывает на версию адаптера, например, `6.2.0`.
2. Если необходимо, установите триггеры для пользовательских таблиц, чтобы при их изменении автоматически генерировались события, хранящиеся в хранилище событий. Кроме использования триггеров, можно заполнять хранилище событий с помощью пользовательского исходного кода SQL или хранимых процедур, которые не вызываются посредством триггеров. Вместо этого, они могут быть вызваны пакетной программой, которая выполняется периодически или запускается вручную.

Результаты

Хранилище событий доступно для обработки событий.

Создание псевдонимов идентификации

Псевдонимы идентификации - это компонент, который шифрует пароль, используемый адаптером для доступа к Oracle DB. Адаптер может использовать его для подключения к Oracle DB вместо использования ИД пользователя и пароля, сохраненных в свойстве адаптера.

Перед тем, как начать

Для того чтобы создать псевдоним идентификации, необходимо иметь доступ к административной консоли WebSphere Process Server или WebSphere Enterprise Service Bus. Необходимо также знать имя пользователя и пароль, используемые для подключения к Oracle DB.

Следующая процедура показывает, как получить доступ к административной консоли посредством WebSphere Integration Developer. Если вы используете административную консоль непосредственно (не через WebSphere Integration Developer), войдите в нее и пропустите шаг 2.

Описание задачи

Использование псевдонима идентификации избавляет от необходимости хранить пароль в текстовом виде в свойстве конфигурации адаптера, где он может быть видим другим.

Для того чтобы создать псевдоним идентификации, используйте следующую процедуру.

Процедура

1. Запустите AdminConsole.
Для того чтобы запустить AdminConsole посредством WebSphere Integration Developer, выполните следующие действия:
 - a. В проекции Бизнес-интеграция продукта WebSphere Integration Developer выберите вкладку **Серверы**.
 - b. Если сервер не показывает состояние **Запущен**, щелкните правой кнопкой мыши на его имени (например, **WebSphere Process Server**) и нажмите кнопку **Запустить**. Дождитесь, пока состояние сервера не будет **Запущен**.
 - c. Щелкните правой кнопкой мыши на имени сервера и выберите **Запустить административную консоль**.
 - d. Войдите в AdminConsole. Если административная консоль требует ИД пользователя и пароль, введите их и нажмите **Войти**. Если ИД пользователя и пароль не требуются, нажмите **Войти**.
2. В AdminConsole выберите **Защита** → **Администрирование защиты, приложения и инфраструктура**.
3. В разделе **Идентификация** выберите **Идентификация Java и служба проверки прав доступа** → **Данные идентификации J2C**.
4. Создайте псевдонимы идентификации
 - a. В показанном списке псевдонимов идентификации J2C нажмите **Создать**.
 - b. На вкладке **Конфигурация** введите имя псевдонима идентификации в поле **Псевдоним**.
 - c. Введите ИД пользователя и пароль, которые требуются для установки соединения с Oracle DB.
 - d. По выбору, введите описание псевдонима.

- e. Нажмите **ОК**.
Будет показан вновь созданный псевдоним.
Полное имя псевдонима содержит имя узла и указанное имя псевдонима идентификации. Например, если псевдоним создан на узле widNode с именем ProductionServerAlias, то его полным именем будет widNode/ProductionServerAlias. Это полное имя используется в последующих окнах конфигурации.
- f. Нажмите **Сохранить**.

Результаты

Создан псевдоним идентификации, который указывается при настройке свойств адаптера позже в мастере.

Создание проекта

Для начала процесса создания и развертывания модуля запускается ESW (мастер внешних служб) в WebSphere Integration Developer. Мастер создает проект коннектора, применяемый для организации связанных с модулем файлов.

Перед тем, как начать

Убедитесь в том, что собрана необходимая информация для установки соединения с Oracle DB. Например, требуется имя и IP-адрес Oracle DB, а также ИД пользователя и пароль для доступа к нему.

Описание задачи

Если существует проект, можно использовать его вместо создания нового. Выберите его перед запуском мастера.

Процедура

1. Для запуска ESW (мастер внешних служб) откройте проекцию Бизнес-интеграция продукта WebSphere Integration Developer и выберите **Файл → Создать → Внешняя служба**.
2. В окне Внешняя служба разверните узел **Адаптеры** и выберите **Oracle**.
3. Нажмите кнопку **Далее**.
4. В окне Выберите адаптер выберите **IBM WebSphere Adapter для Oracle (IBM : версия)**, где *версия* - это версия адаптера, который вы хотите использовать, например, 6.2.0.0
5. Нажмите кнопку **Далее**.
6. В окне Импорт адаптера примите имя проекта по умолчанию в поле **Проект коннектора** или введите другое имя.
7. В поле **Целевая среда выполнения** выберите тип сервера, на котором будет развернут модуль. Мастер создает артефакты, подходящие для этого сервера.
8. Нажмите кнопку **Далее**. Будет показано окно Требуемые файлы и библиотеки.

Результаты

Создан новый проект коннектора, который содержит файл RAR адаптера. Проект перечислен в проекции Интеграция бизнес-процессов.

Дальнейшие действия

Продолжите работу в ESW (мастер внешних служб). Следующим шагом является добавление файлов, специфичных для базы данных.

Добавление внешних зависимостей программного обеспечения

ESW (мастер внешних служб) требует копию некоторых файлов из Oracle E-Business Suite для возможности подключиться к ней. Укажите расположение JAR-файлов, которые содержат драйвер JDBC для Oracle и все необходимые файлы из стандартной библиотеки системы .

Перед тем, как начать

Для выполнения этой задачи необходимо выполнить ESW (мастер внешних служб) в WebSphere Integration Developer.

Описание задачи

В дополнение к этой задаче при настройке модуля необходимо также развернуть файлы в WebSphere Process Server или WebSphere Enterprise Service Bus.

Процедура

1. Получите файлы для драйвера JDBC Oracle или собственные библиотеки для программного обеспечения базы данных и операционной системы у администратора базы данных или на Web-сайте, посвященном программным продуктам для баз данных. Следующая таблица перечисляет Файлы драйвера JDBC Oracle, необходимые для программного обеспечения базы данных Oracle.

Таблица 11. Файлы драйвера JDBC для базы данных Oracle

База данных	Драйвер	Файлы драйвера JDBC	Внутренние системные библиотеки
Oracle	Простой драйвер	ojdbc14.jar	Нет

2. В окне Требуемые файлы и библиотеки укажите расположение файлов для драйвера JDBC Oracle, требуемое адаптером.
 - a. В окне **JAR-файлы драйвера JDBC Oracle** нажмите кнопку **Добавить** и выберите файлы драйвера JDBC Oracle.
 - b. Если вы используете драйвер Oracle JDBC type 2, нажмите кнопку **Добавить** в разделе **Системные библиотеки**, чтобы добавить собственные системные библиотеки, которые требуются для доступа к серверу базы данных. Если вы используете драйвер Oracle JDBC type 4, оставьте это поле пустым.
3. Нажмите кнопку **Далее**. Мастер показывает окно Направление обработки.

Результаты

Мастер имеет файлы, которые необходимы ему для подключения к серверу базы данных.

Продолжите работу в ESW (мастер внешних служб). Следующий шаг состоит в предоставлении информации, которая нужна мастеру для подключения к Oracle DB.

Установка свойств соединения для ESW (мастер внешних служб)

Для того чтобы ESW (мастер внешних служб) мог подключиться к экземпляру базы данных для поиска ее объектов, должны быть заданы свойства соединения.

Перед тем, как начать

Перед настройкой свойств соединения необходимо запустить ESW (мастер внешних служб).

Описание задачи

ESW (мастер внешних служб) требует эти свойства для подключения к базе данных для поиска и создания описания службы. Более подробная информация о свойствах находится в разделе “Свойства соединения для мастера” на стр. 167.

Процедура

1. В окне Направление обработки выберите **Исходящее** для передачи данных из объекта импорта службы в адаптер или **Входящее** для передачи данных из адаптера в объект экспорта службы, а затем нажмите **Далее**.
2. В окне Конфигурация поиска укажите свойства соединения, которые будут использоваться мастером для подключения к Oracle DB.
 - a. В списке программного обеспечения базы данных выберите продукт и версию. В области **Свойства** показаны поля, в которых указываются свойства соединения, специфичные для базы данных.
 - b. В поле **Тип драйвера JDBC** выберите тип драйвера JDBC, который вы хотите использовать.
 - c. В поле **ИД системы** укажите имя базы данных.
 - d. В поле **Имя хоста** укажите имя хоста или IP-адрес сервера базы данных. Если вы задаете IP-адрес в формате IPv6, заключите его в квадратные скобки ([]).
 - e. В поле **Номер порта** укажите номер порта для подключения к базе данных. Если вы выбрали именованный драйвер в поле **Тип драйвера JDBC**, мастер предоставляет значение по умолчанию для поля **Номер порта**. Если вы выбрали драйвер Другой, то номер порта недоступен.
 - f. Если вы выбрали именованный драйвер в поле **Тип драйвера JDBC**, мастер предоставляет значение по умолчанию для поля **Имя класса драйвера JDBC** и составляет значение для поля **URL базы данных** из других полей соединения. Если вы выбрали драйвер Другой, необходимо указать имя класса драйвера и URL базы данных (хотя часть URL базы данных может быть заполнена автоматически).
 - g. В поле **Дополнительные свойства соединения драйвера JDBC** укажите дополнительные свойства, устанавливаемые при подключении к базе данных. Укажите одну или несколько пар *имя:значение* через точку с запятой (;).
Например:
`loginTimeout:20;readOnly:true;securityMechanism:USER_ONLY_SECURITY`

Информация о соединении используется только для процесса поиска. Позже во время работы мастера можно будет указать другую информацию о соединении, которая будет использоваться во время выполнения.

3. В полях **Имя пользователя** и **Пароль** введите имя пользователя и пароль для подключения к базе данных из мастера. Это имя пользователя используется только в процессе поиска и не сохраняется. Позже во время работы мастера можно будет указать другое имя и пароль или другой метод идентификации, которые будут использоваться при во время выполнения.

4. В поле **Префикс для имен бизнес-объектов** введите строку, которая будет добавляться в начало имен бизнес-объектов. Это поле префикса не требует значения и может быть оставлено пустым.
5. Для того чтобы включить двунаправленного текста для адаптера во время выполнения:
 - a. Нажмите кнопку **Дополнительно**.
 - b. В области **Свойства Bidi** выберите переключатель **Преобразование Bidi**.
 - c. Установите свойства схемы упорядочения, направления текста, симметричного свопинга, перерисовки символов и формирования чисел для управления способом выполнения преобразования двунаправленного текста.
6. Для того чтобы изменить расположение файлов протокола мастера или объема информации, включенной в протоколы, выберите **Изменить свойства ведения протоколов для мастера**, а затем предоставьте следующую информацию:
 - В поле **Расположение вывода файла протокола** укажите расположение файла протокола для мастера.
 - В поле **Уровень ведения протоколов** укажите серьезность ошибок, которые вы хотите вносить в протокол.

Эта информация о протоколе используется только для мастера. Во время выполнения адаптер записывает сообщения об ошибках и информацию трассировки в стандартные файлы протокола и трассировки для сервера.

7. Нажмите кнопку **Далее**.

Если мастер генерирует исключительную ситуацию `com.ibm.adapter.framework.BaseException`, адаптер не может подключиться к серверу базы данных. Сообщение содержит дополнительную информацию о возможной причине неполадки. Кроме того, можно проверить протоколы, расположенные в каталоге, который указан в поле **Расположение вывода файла протокола**. Убедитесь в том, что информация о соединении верна.

Результаты

ESW (мастер внешних служб) подключается к базе данных и показывает окно Поиск и выбор объектов.

Дальнейшие действия

Продолжите работу в мастере. Следующим шагом является поиск в базе данных объектов, для которых мастер должен создать бизнес-объекты.

Настройка модуля для обработки исходящих запросов

Настройка модуля для обработки исходящих запросов предусматривает применение ESW (мастер внешних служб) в WebSphere Integration Developer для поиска и выбора бизнес-объектов и служб в Oracle DB, а также создания определений бизнес-объектов и связанных артефактов.

Поиск объектов базы данных

После настройки базы данных выполните запрос на поиск объектов базы данных. Откройте дерево найденных объектов, чтобы получить общее представление о структуре объектов Oracle DB. С помощью фильтров можно отобразить только объекты базы данных, которые вы хотите видеть. Определите количество бизнес-объектов, которое вы хотите создать для пользовательских запросов к базе данных.

Перед тем, как начать

Необходимо знать требование программы к данным для доступа к базе данных. Например, необходима следующая информация о базе данных:

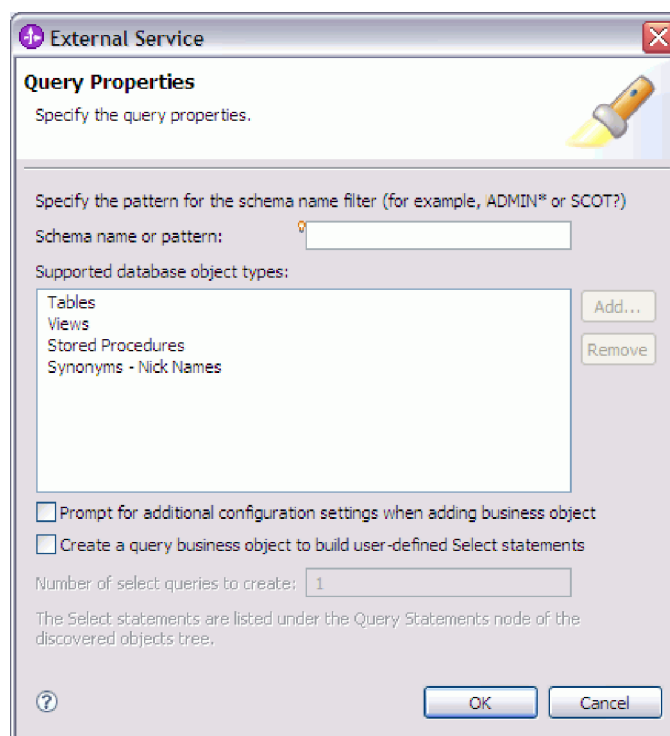
- Какие схемы необходимы модулю для доступа
- Какой тип объектов базы данных необходим для доступа в этих схемах
- К каким таблицам, представлениям, синонимам или псевдонимам, хранимым процедурам или функциям необходим доступ
- Сколько бизнес-объектов запросов необходимо создать, включая значения параметров и примеры значений базы данных для параметров

Описание задачи

Эта задача начинается в окне Поиск и выбор объектов ESW (мастер внешних служб).

Процедура

1. В окне Поиск и выбор объектов нажмите **Изменить запрос**. Будет показано окно Свойства запроса.



Используйте окно Свойства запроса для выполнения следующих задач:

- Уменьшить время поиска, ограничив подмножество схем базы данных
- Пропустить один или несколько типов объектов базы данных в поиске
- Позволить мастеру запрашивать информацию, специфичную для приложения, которая не может быть автоматически определена на основании информации в базе данных
- Укажите количество бизнес-объектов запросов, которое вы хотите создать

Ниже описаны действия для этих задач.

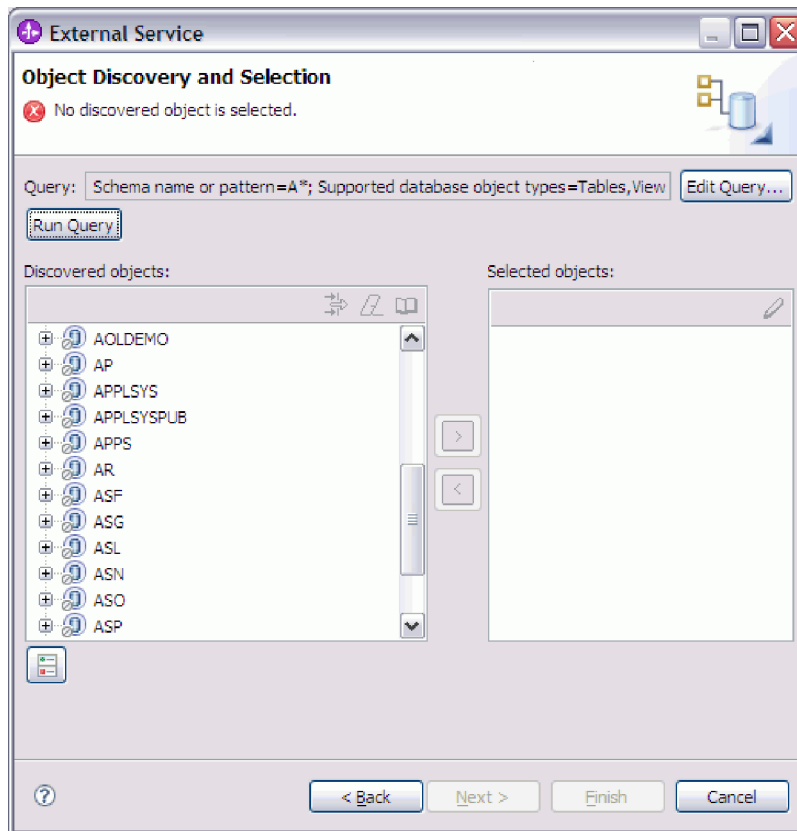
2. Для ограничения числа отображаемых схем баз данных введите имя или шаблон имени схемы в поле **Имя схемы или шаблон**. Знак вопроса или подчеркивание (? или _) заменяет один произвольный символ, звездочка или процент (* или %) -

несколько символов. Запрос возвратит только те схемы, имена которых начинаются с указанной строки или совпадают с ней. Если шаблон не указан, то отображаются все схемы из базы данных. Путем применения фильтра можно ускорить процесс поиска, если база данных содержит большое число схем.

3. Для того чтобы пропустить один или несколько типов объектов в поиске, выберите эти типы (таблицы, представления, хранимые процедуры или функции, синонимы или псевдонимы) в **Поддерживаемые типы объектов базы данных**, а затем нажмите кнопку **Удалить**. Если вы передумали, нажмите **Добавить** для возвращения типов объектов назад. Если необходим доступ только к определенным типам объектов базы данных, то для ускорения процесса поиска пропустите необязательные типы.
4. Выберите переключатель **Запрашивать дополнительные параметры при добавлении бизнес-объекта**. Затем при добавлении объекта базы данных в список создаваемых бизнес-объектов мастер автоматически предложит ввести настраиваемую информацию уровня приложения для объекта. Например, мастер может в пошаговом режиме создать иерархическую структуру бизнес-объектов. Если требуется создать иерархическую структуру, в которой бизнес-объект таблицы содержит два атрибута, которые связаны с атрибутами из двух других таблиц (два родительских бизнес-объекта), то настройку следует выполнить с помощью редактора сборки (запускается из WebSphere Integration Developer).

Важное замечание: Если этот переключатель не выбран, то мастер запрашивает только обязательную информацию. Настройку бизнес-объектов потребуется завершить с помощью редактора сборки.

5. Для того чтобы создать бизнес-объекты для выполнения пользовательских запросов к базе данных, выберите **Создать бизнес-объект запроса для построения пользовательских операторов выбора**, а затем введите количество бизнес-объектов запроса для создания. В этот момент указывается только количество бизнес-объектов. Мастер предложит указать имя и другие сведения о бизнес-объектах позже.
6. Нажмите **ОК** для сохранения изменений в запросе к базе данных.
7. В окне Поиск и выбор объектов нажмите **Выполнить запрос**, чтобы применить запрос для поиска объектов базы данных и создать шаблон для бизнес-объектов запроса. Результат выполнения обычного запроса показан на следующем рисунке.



Панель **Найденные объекты** перечисляет найденные объекты базы данных.

8. В списке **Найденные объекты** нажмите **+** (знак плюс), чтобы развернуть узел схемы, а затем разверните узлы **Таблицы**, **Представления**, **Хранимые процедуры** и **Синонимы - Псевдонимы** под ним для просмотра объектов базы данных, найденных мастером, или выполните поиск подмножества каждого типа объекта с помощью фильтрации.

Для того чтобы ограничить количество отображенных объектов определенного типа, выделите узел, не раскрывайте его. Щелкните на значке **Фильтрация**. Вопросительный знак или подчеркивание (? или _) соответствует одиночному символу, а звездочка или знак процента (* или %) соответствует нескольким символам. В узле показываются только типы объектов, такие как таблица или представление, которые начинаются с этой строки или соответствуют шаблону.

9. Щелкните на значке **+** (знак плюс), чтобы развернуть узел **Операторы запроса** и показать шаблон бизнес-объектов запроса.

Результаты

Мастер показывает объекты базы данных, к которым можно получить доступ с помощью адаптера, и шаблон бизнес-объектов запроса.

Дальнейшие действия

Продолжите работу в ESW (мастер внешних служб). Следующий шаг состоит в выборе объектов, которые вы хотите использовать в модуле, настройке каждого бизнес-объекта и создании иерархии бизнес-объектов.

Выбор и настройка бизнес-объектов

С помощью списка объектов баз данных, обнаруженных ESW (мастер внешних служб), продолжите выбор объектов баз данных, которые должны быть доступны модулю. Затем укажите информацию о конфигурации новых бизнес-объектов.

Описание задачи

Окно Поиск и выбор объектов позволяет выбрать и настроить объекты в любом порядке, с тем исключением, что необходимо выбрать и настроить родительскую таблицу перед выбором и настройкой дочерних таблиц. Кроме этого ограничения, вы можете добавлять объекты по-отдельности или группами. Можно смешивать объекты из различных узлов списка **Найденные объекты**. Например, можно выбрать несколько объектов таблиц, объект хранимой процедуры и оператор запроса, а затем добавить и настроить их одновременно.

Поток высокого уровня выбора и настройки бизнес-объектов выглядит следующим образом:

1. Выберите один или несколько объектов в списке **Найденные объекты** окна Поиск и выбор объектов.
2. Нажмите кнопку **>** (Добавить).
3. Откроется окно Свойства конфигурации.

- Если выбран отдельный объект, то откроется одно окно Свойства конфигурации.

В этом окне укажите пользовательские атрибуты и прочую информацию, которую мастер не может найти в базе данных, и нажмите кнопку **ОК** для сохранения конфигурации.

- Если выбрано несколько бизнес-объектов, то откроется окно Свойства конфигурации с отдельными страницами для каждого выбранного бизнес-объекта.

Последовательно щелкните на имени каждого объекта. Откроется окно, содержащее такую же информацию, как в случае выбора отдельного объекта.

Важное замечание: Не нажимайте кнопку **ОК** до тех пор, пока не будут заполнены страницы конфигурации всех объектов. Мастер запрещает закрытие окна до заполнения обязательных полей, однако окно можно закрыть перед указанием необязательных полей. Если необязательные поля не указаны, то после выхода из мастера их следует настроить с помощью редактора бизнес-объектов.

4. Мастер добавляет настроенный объект в список **Выбранные объекты**.

До закрытия мастера можно несколько раз выбирать и настраивать бизнес-объекты, необходимые для работы модуля. Однако с помощью мастера нельзя добавить объекты в существующие модули. Перед запуском мастера рекомендуется внимательно изучить требования программы, использующей бизнес-объекты.

Выбор и настройка таблиц, представлений и синонимов или псевдонимов

Для того чтобы выбрать и настроить таблицы, представления и синонимы или псевдонимы для использования в модуле, необходимо указать свойства конфигурации бизнес-объекта.

Перед тем, как начать

Для выполнения этой задачи требуется знание структуры данных в базе данных, а также объектов базы данных, с которыми должен взаимодействовать модуль. В частности требуется следующая информация:

- Структура таблиц, представлений и синонимов (псевдонимов), включая применяемые столбцы и атрибуты столбцов, такие как тип данных.
- Взаимосвязи между таблицами, включая число значений и принадлежность.

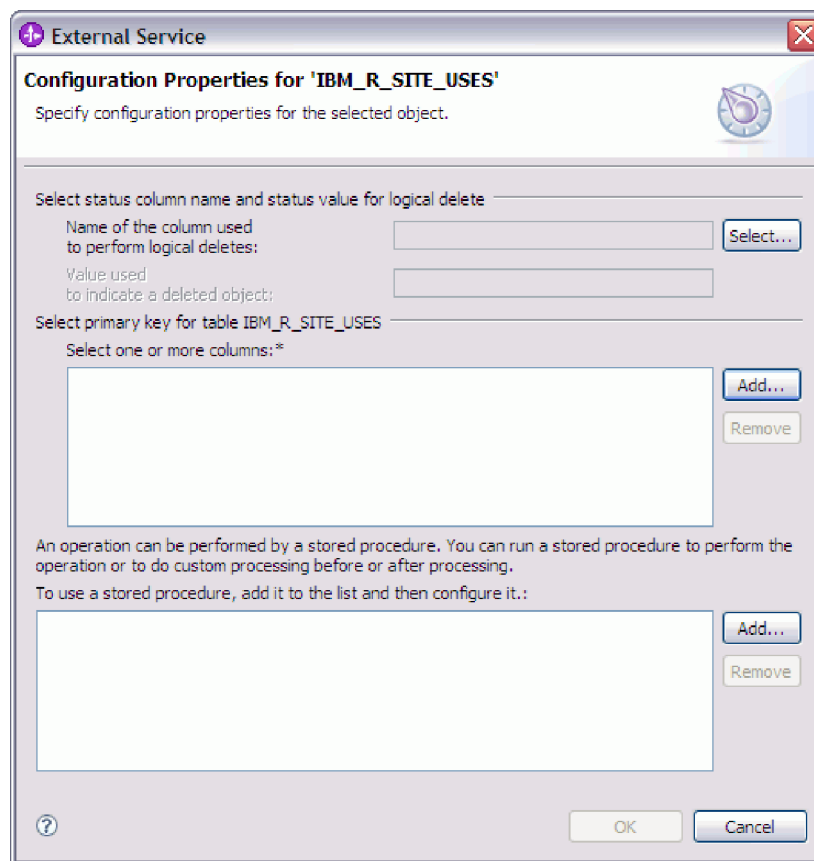
Описание задачи

Эта задача выполняется с помощью ESW (мастер внешних служб). Пользователь начинает в окне Поиск и выбор объектов и продолжает работу в окне Свойства конфигурации настраиваемого бизнес-объекта.

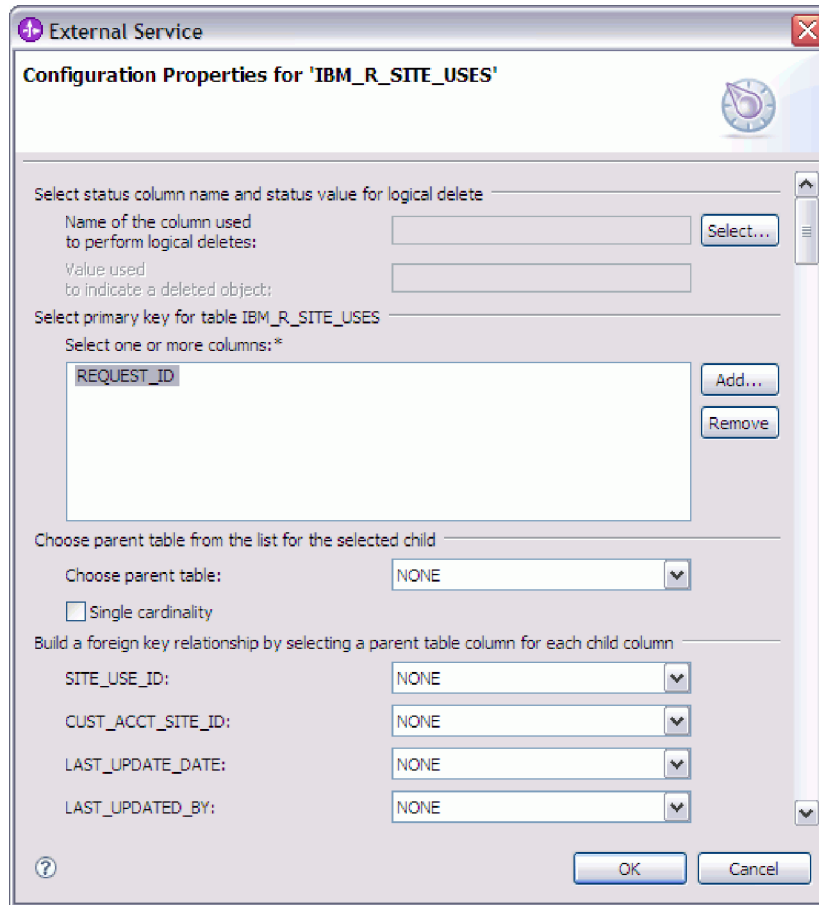
Процедура

1. В списке **Найденные объекты** окна Поиск и выбор объектов выберите одну или несколько таблиц, представлений или синонимов и нажмите кнопку **>** (Добавить) для добавления объектов в список **Выбранные объекты**.

На следующих рисунках показано окно Свойства конфигурации таблиц, представлений или синонимов. На первом рисунке показано окно для первой выбранной таблицы или группы таблиц.



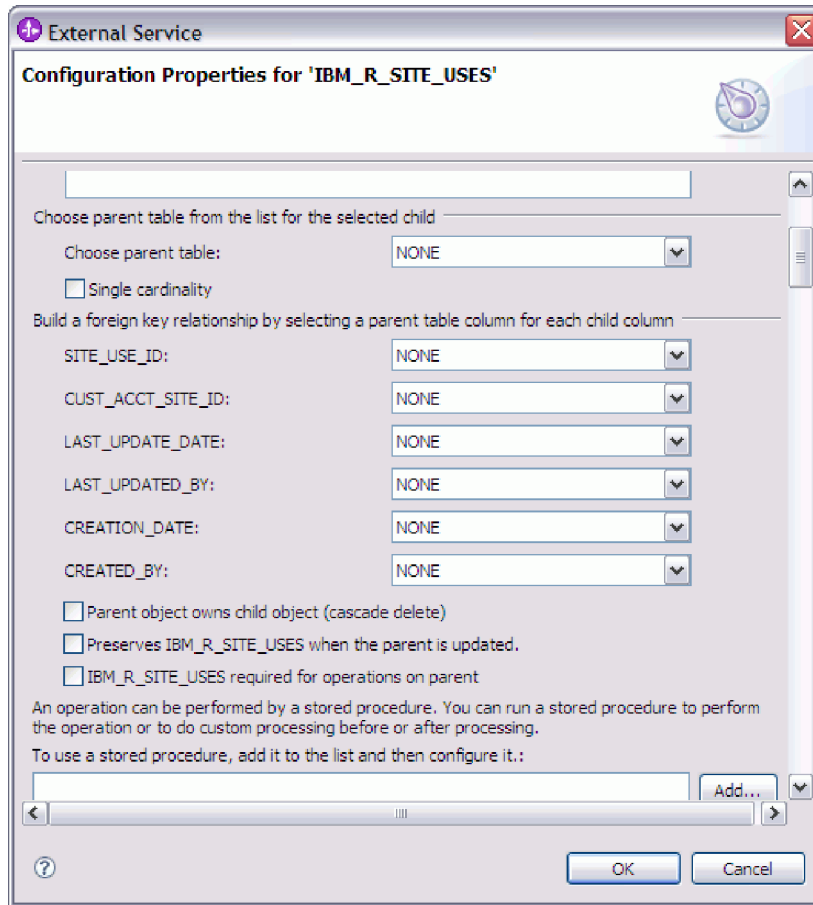
На следующем рисунке показано окно для последующих таблиц. После выбора и настройки по крайней мере одной таблицы в окне Свойства конфигурации для последующих таблиц отображается область, в которой можно задать иерархическую структуру таблиц.



В ходе настройки объекта варианты, требующие дополнительной настройки, могут отображать в окне дополнительные поля, требующие прокрутки окна. Перед нажатием кнопки **ОК** проверьте все поля в окне.

2. Если таблица содержит столбец, в котором указываются операции логического удаления:
 - a. Выберите имя столбца в поле **Имя столбца, используемого для выполнения операций логического удаления**.
 - b. В поле **Значение, указывающее на удаленный объект** укажите значение, указывающее на логическое удаление строки. Это значение можно получить у администратора базы данных.
3. Если показана область **Выберите первичный ключ для таблицы имя-таблицы**, то нажмите кнопку **Добавить**, выберите столбец первичного ключа и нажмите кнопку **ОК**. Если таблица содержит составной ключ, то можно выбрать несколько столбцов. Область **Выберите первичный ключ для таблицы имя-таблицы** отображается только в том случае, если таблица базы данных не содержит столбец первичного ключа. Каждый бизнес-объект таблицы должен содержать первичный ключ. Если первичный ключ задан в базе данных, то этот раздел окна не отображается.
4. При необходимости укажите взаимосвязь между бизнес-объектами.
Для создания иерархической структуры сначала настройте родительскую таблицу, затем вернитесь к окну Поиск и выбор объектов для выбора и настройки дочерних таблиц.

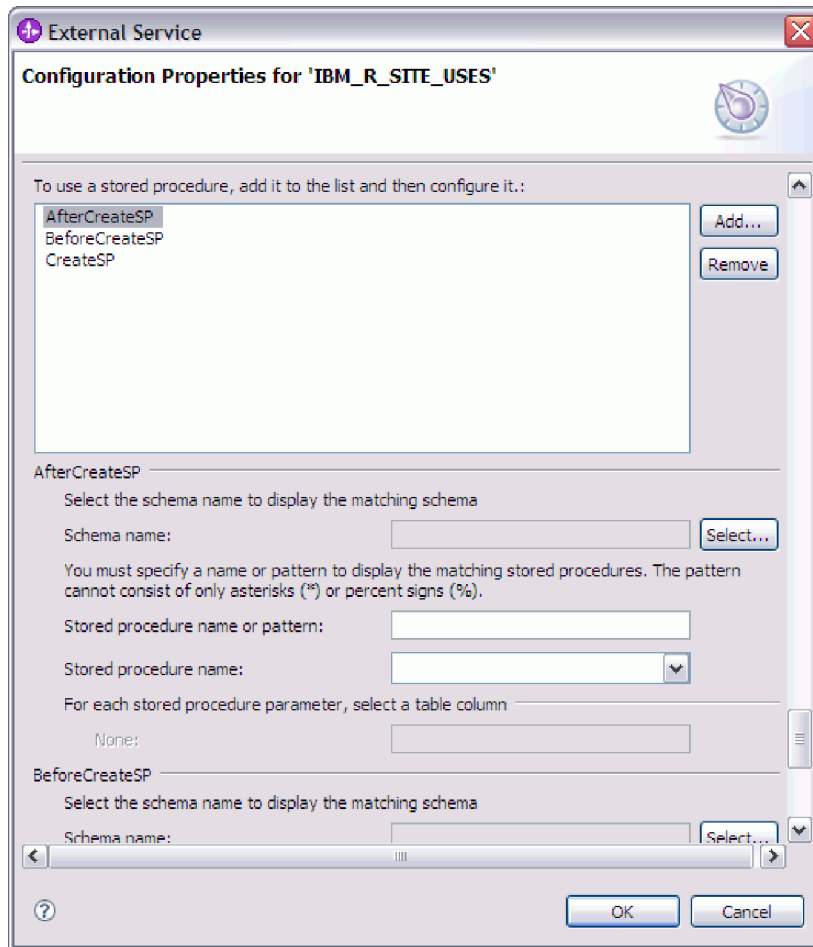
Настройте взаимосвязь с помощью окна Свойства конфигурации (см. следующий рисунок). Эти поля не отображаются в ходе настройки первой таблицы.



- a. В поле **Выберите родительскую таблицу** выберите имя родительской таблицы. Если родительская таблица не показана в списке, то она еще не настроена. Вернитесь и настройте родительскую таблицу перед настройкой дочерних объектов.
- b. Укажите число значений взаимосвязи:
 - Для взаимосвязи с одним значением выберите переключатель **Единственность**. В этом случае родительский объект может содержать только один дочерний бизнес-объект этого типа. Взаимосвязь с одним значением может представлять дочерний объект, таблицы поиска или равноправные объекты в базе данных.
 - Для взаимосвязи с несколькими значениями отмените выбор переключателя **Единственность**. В этом случае родительский объект может содержать массив дочерних бизнес-объектов указанного типа.
- c. Создайте взаимосвязь внешнего ключа между родительским и дочерним объектами, указав дочерние столбцы, выполняющие роль внешнего ключа в родительской таблице.
 - Если дочерний столбец не является внешним ключом, то выберите Нет.
 - Если дочерний столбец является внешним ключом, то выберите в родительской таблице столбец, соответствующий дочернему столбцу.

Примечание: Мастер позволяет настроить только одну родительскую таблицу. Если дочерняя таблица содержит несколько родительских таблиц, то после выхода из мастера их следует настроить с помощью редактора бизнес-объектов.

- d. Если дочерний объект принадлежит родительскому объекту, то дочерние объекты удаляются из базы данных вместе с родительским. Для указания такой взаимосвязи выберите переключатель **Родительскому объекту принадлежит дочерний объект (каскадное удаление)**. В противном случае отмените выбор этого переключателя во избежание удаления дочерних объектов, таких как таблицы поиска, вместе с родительскими объектами.
 - e. Если дочерние объекты не должны удаляться в ходе операции Update, выберите переключатель **Сохранять имя-дочерней-таблицы при обновлении родительского объекта**.
В ходе обновления родительской таблицы адаптер сравнивает входные дочерние бизнес-объекты с дочерними бизнес-объектами, возвращенными из базы данных. По умолчанию адаптер удаляет все возвращенные из базы данных дочерние бизнес-объекты, отсутствующие во входном бизнес-объекте.
 - f. По умолчанию операции над родительскими бизнес-объектами можно выполнять, не указывая дочерние бизнес-объекты. Для гарантированного изменения дочерних бизнес-объектов вместе с родительским бизнес-объектом выберите переключатель **имя-дочерней-таблицы требуется для обработки родительского объекта**.
5. Операцию можно выполнить с помощью стандартного оператора SQL, созданного адаптером, либо с помощью хранимых процедур или функций из базы данных. Для применения хранимых процедур и функций выполните следующие действия:
- a. Нажмите кнопку **Добавить**.
 - b. В окне **Добавить** выберите тип хранимой процедуры. Для каждой операции можно выбрать хранимую процедуру, выполняющую операцию, а также хранимые процедуры, выполняемые до и после операции. Например, для операции **Create** можно указать следующие хранимые процедуры: **CreateSP**, **BeforeCreateSP** и **AfterCreateSP**.
 - c. Нажмите кнопку **ОК**. В окне **Свойства конфигурации** будут показаны выбранные типы хранимых процедур и их параметры. Для просмотра параметров может потребоваться прокрутить окно вниз.



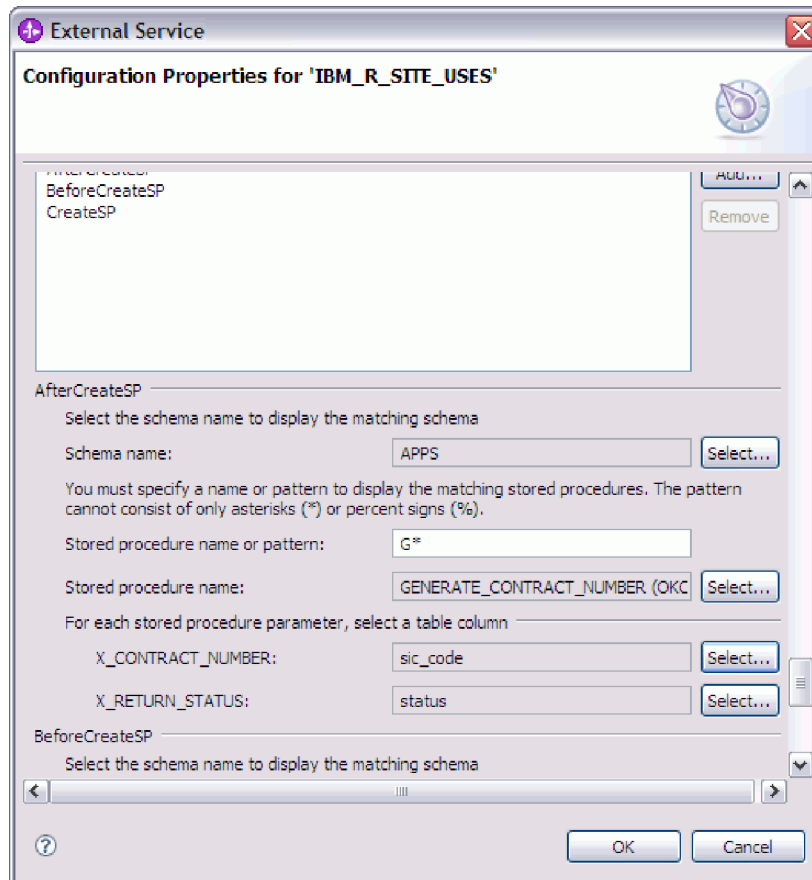
Примечание: В бизнес-объекте с иерархической структурой для выполнения хранимой процедуры над всем бизнес-объектами необходимо отдельно связать хранимую процедуру с бизнес-объектом верхнего уровня и каждым дочерним бизнес-объектом. Если связать хранимую процедуру только с бизнес-объектом верхнего уровня, то хранимая процедура будет обрабатывать только бизнес-объект верхнего уровня - дочерние бизнес-объекты будут обрабатываться с помощью стандартного запроса SQL.


6. Для каждого выбранного типа хранимой процедуры укажите имя хранимой процедуры в базе данных и настройте бизнес-объект.
 - a. В поле **Имя схемы** выберите имя схемы, содержащей хранимую процедуру.
 - 1) Нажмите кнопку **Выбрать**.
 - 2) В окне Выбрать значение выберите нужную схему.
 - 3) Нажмите **ОК**.
 - b. Укажите имя хранимой процедуры или функции.
 - 1) В поле **Имя хранимой процедуры или шаблон** введите имя хранимой процедуры или функции либо введите шаблон имени. Знак вопроса или подчеркивание (? или _) заменяет один произвольный символ, звездочка или процент (* или %) - несколько символов.
 - 2) В поле **Имя хранимой процедуры** выберите имя процедуры. Если список хранимых процедур содержит большое число записей, то рядом с полем

Имя хранимой процедуры отображается кнопка **Выбрать**. Нажмите кнопку **Выбрать**, для того чтобы выбрать имя хранимой процедуры или функции в окне **Выбрать**.

В окне **Свойства конфигурации** будет показана область, позволяющая настроить хранимую процедуру. Мастер автоматически создает список параметров путем проверки хранимой процедуры в базе данных.

- c. Для каждого параметра хранимой процедуры (слева) выберите столбец таблицы (справа) для передачи хранимой процедуры в соответствующий параметр. На следующем рисунке показано окно после настройки хранимой процедуры.



7. После заполнения всех полей в окне нажмите кнопку **ОК** для сохранения конфигурации бизнес-объекта. В окне **Поиск и выбор объектов** показаны таблица, представление, синоним и псевдоним.
8. Для изменения конфигурации объекта в списке **Выбранные объекты** выберите имя объекта и щелкните на значке  (**Изменить**).

В окне **Поиск и выбор объектов** продолжите выбор и настройку других типов бизнес-объектов. После завершения нажмите кнопку **Далее** для задания глобальных свойств.

Выбор и настройка хранимых процедур и функций

Для выбора и настройки бизнес-объектов, которые соответствуют хранимым процедурам и хранимым функциям, необходимо отфильтровать объекты базы данных и указать свойства конфигурации для объекта базы данных.


Перед тем, как начать

Для того чтобы выбрать и настроить бизнес-объекты для хранимых процедур и функций, необходимо понимать структуру данных в базе данных и знать, к каким объектам необходим доступ модуля. В частности, необходимо знать параметры, передаваемые в хранимую процедуру или функцию, к которой модуль должен иметь доступ.

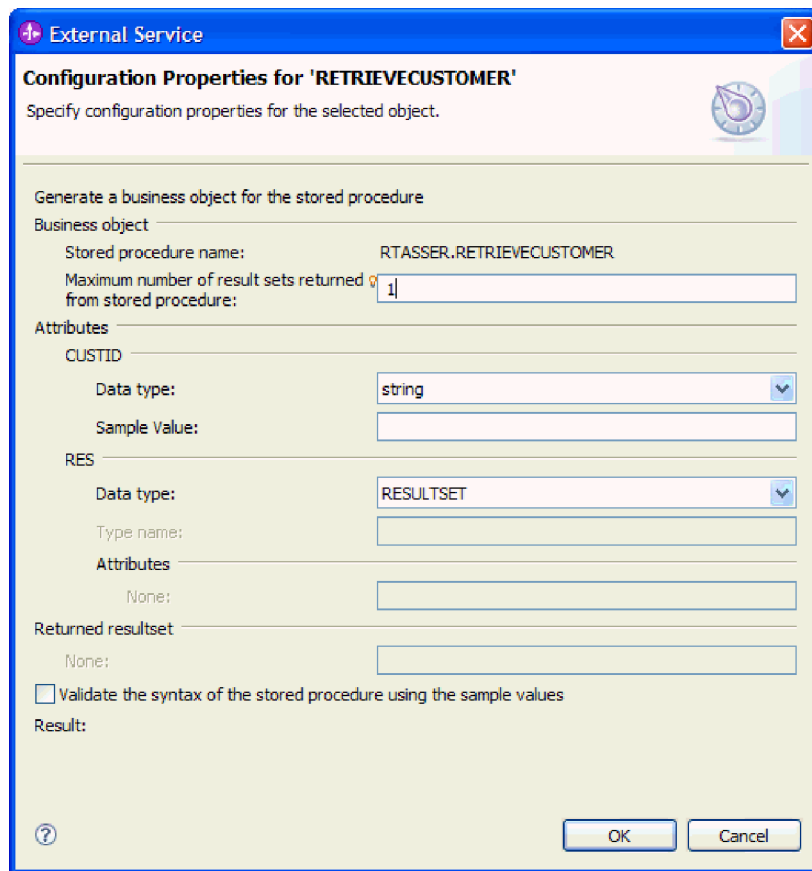
Описание задачи

Эта задача выполняется с помощью ESW (мастер внешних служб). Пользователь начинает в окне Поиск и выбор объектов и продолжает работу в окне Свойства конфигурации настраиваемого бизнес-объекта.

Процедура

1. В списке **Найденные объекты** окна Поиск и выбор объектов разверните узел схемы, содержащей нужную хранимую процедуру или функцию.
2. В окне Свойства фильтра укажите имя или шаблон имени по крайней мере в одном поле для фильтрации списка хранимых процедур.
 - a. Выберите **Хранимые процедуры** и щелкните на значке  (Изменить или создать фильтр), который расположен в верхней части панели **Найденные объекты**.
 - b. В окне Свойства фильтра введите имя или шаблон в поле **Имя объекта или шаблон**. Знак вопроса или подчеркивание (? или _) заменяет один произвольный символ, звездочка или процент (* или %) - несколько символов. В имени не учитывается регистр символов.
 - c. В поле **Имя каталога или шаблон** введите имя или шаблон. Знак вопроса или подчеркивание (? или _) заменяет один произвольный символ, звездочка или процент (* или %) - несколько символов.
 - d. Нажмите **ОК**. В узле Хранимые процедуры будут показаны все хранимые процедуры, соответствующие указанному фильтру.
3. В списке **Хранимые процедуры** выберите одну или несколько хранимых процедур и нажмите кнопку **>** (Добавить) для добавления объектов в список **Выбранные объекты**.

Хранимые процедуры из пакетов PL/SQL отображаются в формате *имя-хранимой-процедуры(имя-пакета)*. Например, если пакет EMP_MGMT содержит хранимую процедуру CREATE_DEPT, то в списке будет показано имя CREATE_DEPT(EMP_MGMT). В окне Свойства конфигурации: 'объект' перечислены атрибуты бизнес-объекта хранимой процедуры, в том числе имена и типы данных параметров хранимой процедуры, а также информация о возвращаемых наборах результатов.



4. Если хранимая процедура возвращает наборы результатов, то в поле **Максимальное число результатов, возвращаемых хранимой процедурой** должно быть указано максимальное число результатов. Мастер создает указанное число бизнес-объектов наборов результатов.

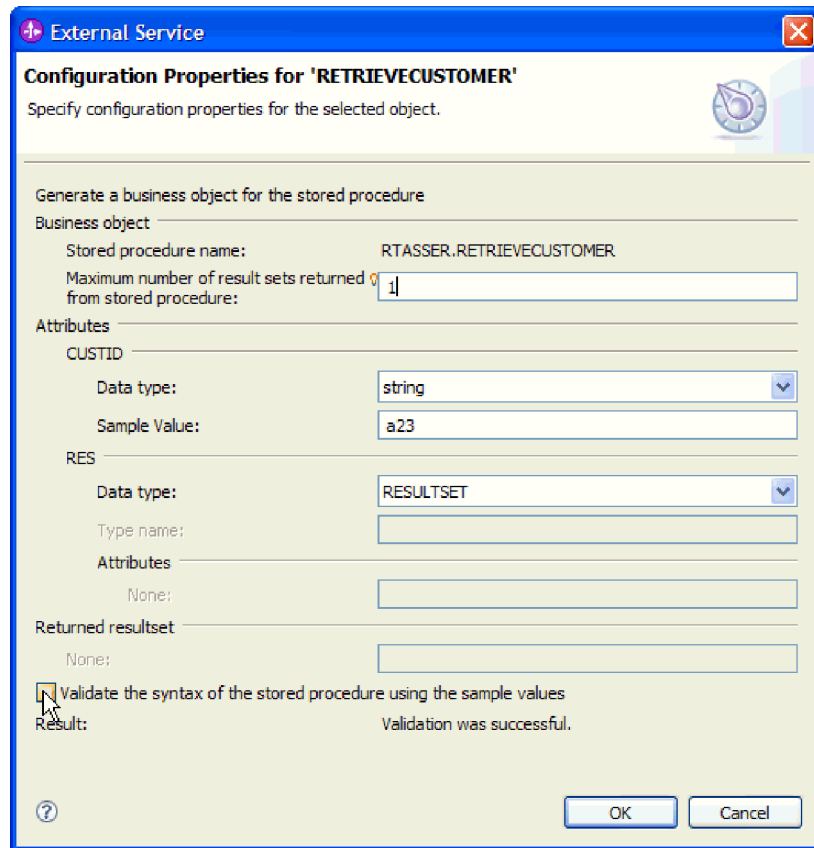
Примечание: Проверьте правильность числа результатов после проверки синтаксиса хранимой процедуры, поскольку драйвер Oracle не всегда возвращает наборы результатов. При необходимости укажите правильное число результатов после проверки и перед нажатием кнопки **ОК**. После завершения работы мастера можно дополнительно проверить параметр MaxNumOfRetRS из информации уровня приложения для бизнес-объекта хранимой процедуры.

5. Настройте каждый параметр:
 - a. Убедитесь, что в поле **Тип данных** показан правильный тип данных. Необходимо выбрать тип данных вручную.
 - b. Для атрибутов с простыми типами данных введите фактическое значение из базы данных в поле **Пример значения**. Например, если параметр передает фамилию клиента, то введите фамилию, которая существует в базе данных.
6. После настройки всех атрибутов выберите **Проверить синтаксис хранимой процедуры с помощью примеров значений**. Результат проверки отображается в области **Результат**.

Если в области **Результат** показано сообщение Сбой проверки, то указанная информация содержит ошибки. Исправьте определение в соответствии с информацией из следующего сообщения об ошибке. Проверьте правильность типа данных параметров и данных.


Файл .log, расположенный в папке .metadata рабочей области, содержит дополнительную информацию об этой неполадке.

Окно после проверки хранимой процедуры.



После отображения сообщения Проверка выполнена успешно нажмите кнопку **ОК** для сохранения определения бизнес-объекта хранимой процедуры.

Важное замечание: Если хранимая процедура или функция возвращает набор результатов, не нажимайте кнопку **ОК** до тех пор, пока не будет пройдена проверка. Мастер использует результаты, полученные в ходе проверки, для создания бизнес-объектов, в которых сохраняются результаты. Если проверка не будет пройдена, то адаптер не может вернуть набор результатов во время выполнения.

7. Для изменения конфигурации объекта в списке **Выбранные объекты** выберите имя объекта и щелкните на значке  (Изменить).

Результаты

Бизнес-объекты, настроенные для хранимых процедур и функций, отображаются в окне Поиск и выбор объектов.

В окне Поиск и выбор объектов продолжите выбор и настройку других типов бизнес-объектов. После завершения нажмите кнопку **Далее** для задания глобальных свойств.

Выбор и настройка бизнес-объектов запроса

Выберите и настройте бизнес-объекты запроса для пользовательских операторов SELECT, используемых в модуле.

Перед тем, как начать

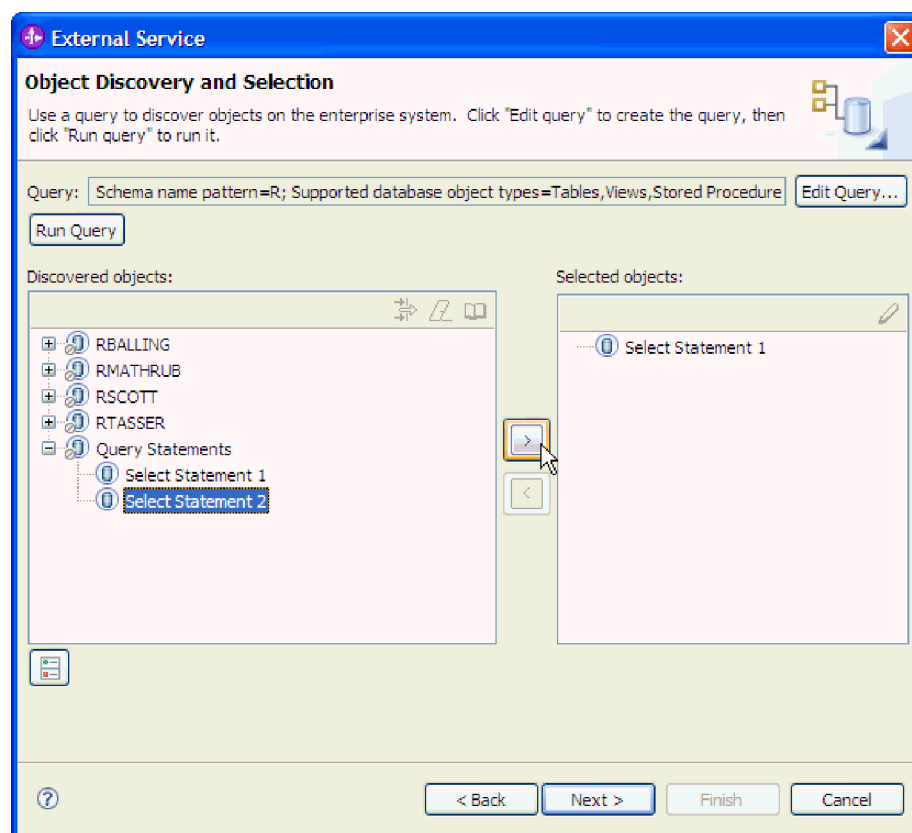
Для того чтобы настроить бизнес-объекты запроса, необходимо знать структуру данных в базе данных, включая таблицы и представления. Необходимо знать имя и тип данных столбцов, к которым должен иметь доступ модуль. Необходимо также иметь возможность написать операторы SQL SELECT.

Описание задачи

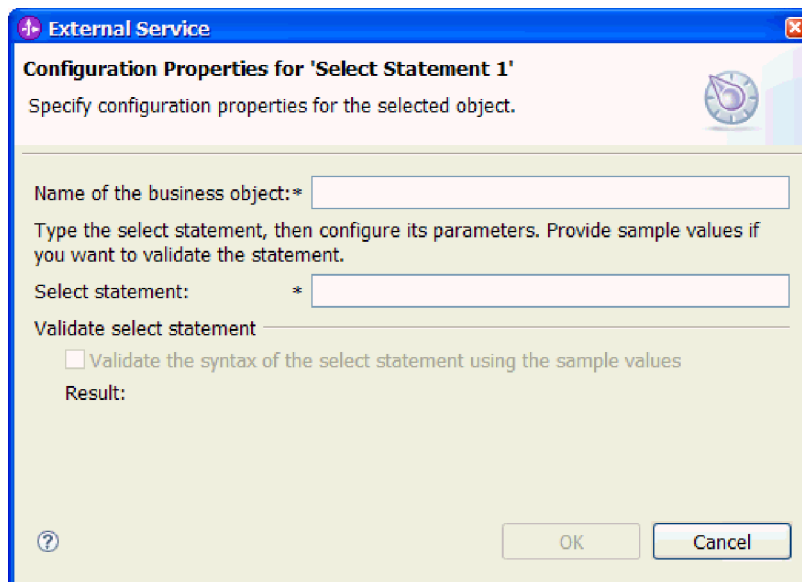
Эта задача выполняется с помощью ESW (мастер внешних служб). Пользователь начинает в окне Поиск и выбор объектов и продолжает работу в окне Свойства конфигурации настраиваемого бизнес-объекта.

Процедура

1. в списке **Найденные объекты** в окне Поиск и выбор объектов разверните узел **Операторы запроса**. Этот узел содержит шаблон объекта по имени **Оператор Select *n*** для каждого бизнес-объекта запроса, который запрошен в окне Свойства запроса. Например, если в окне указано количество из двух бизнес-объектов запроса, то список **Найденные объекты** содержит два шаблона объекта, как показано на следующем рисунке.

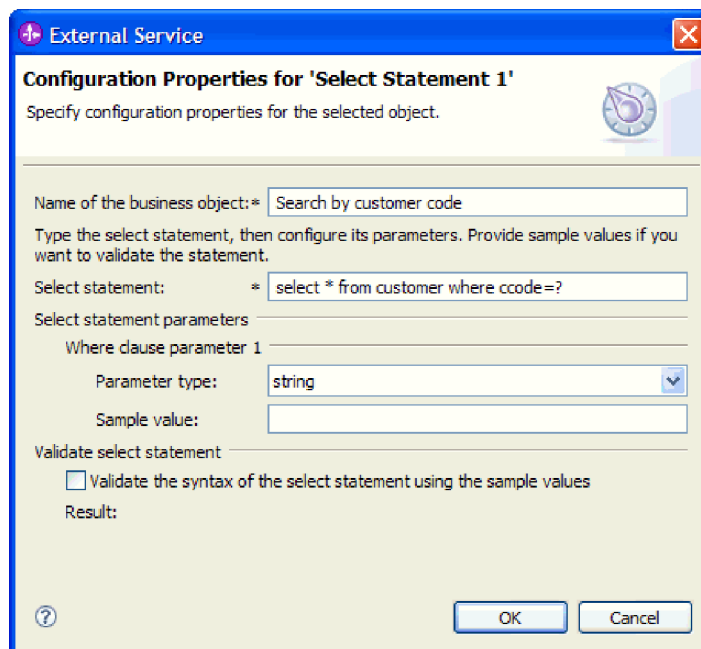


2. Выберите один или несколько шаблонов объектов и нажмите кнопку **>** (Добавить), чтобы добавить объекты в список **Выбранные объекты**. На следующем рисунке показано окно Свойства конфигурации, которое открывается при нажатии кнопки **>** (Добавить) для бизнес-объекта запроса.



3. В поле **Имя бизнес-объекта** введите имя бизнес-объекта. Имя может содержать пробелы и символы национальных языков.
4. В поле **Оператор Select** введите оператор SELECT, который вы хотите выполнить. Укажите каждый параметр с помощью вопросительного знака (?). Следующий пример оператора SELECT иллюстрирует гибкость бизнес-объекта запроса:
 - select * from customer where ccode=?
 - select * from customer where id=? and age=?
 - select * from customer where lname like ?
 - select C.pkey, C.fname, A.city from customer C, address A WHERE (C.pkey = A.custid) AND (C.fname like ?)

При вводе каждого символа ? разворачивается окно, показывая область, в которой определяется оператор WHERE для параметра. На следующем рисунке показано окно Свойства конфигурации для бизнес-объекта запроса, который содержит один параметр.



5. В поле **n** параметр оператора **Where** укажите информацию о каждом параметре оператора **SELECT**.
 - a. В поле **Тип параметра** выберите тип данных параметра.
 - b. В поле **Значение примера** введите значение примера для параметра.
 Например, для параметра, соответствующего столбцу, содержащему фамилию клиента, можно выбрать `string` в качестве типа данных и указать значение примера `Иванов`.
6. Нажмите **Проверить синтаксис оператора select с помощью значений примера**. Поле **Результат** показывает результат проверки.
 Если поле **Результат** показывает Проверка не выполнена, значит предоставлена неверная информация. Для того чтобы исправить определение, используйте сообщение об ошибке из сервера базы данных, которое следует за Проверка не выполнена. Проверьте синтаксис оператора **SELECT**, тип данных параметров и данные примера.
7. Когда будет показано сообщение Проверка выполнена успешно, нажмите **ОК** для сохранения определения бизнес-объекта запроса.

Результаты

Определенные бизнес-объекты запроса теперь перечислены в окне Поиск и выбор объектов.

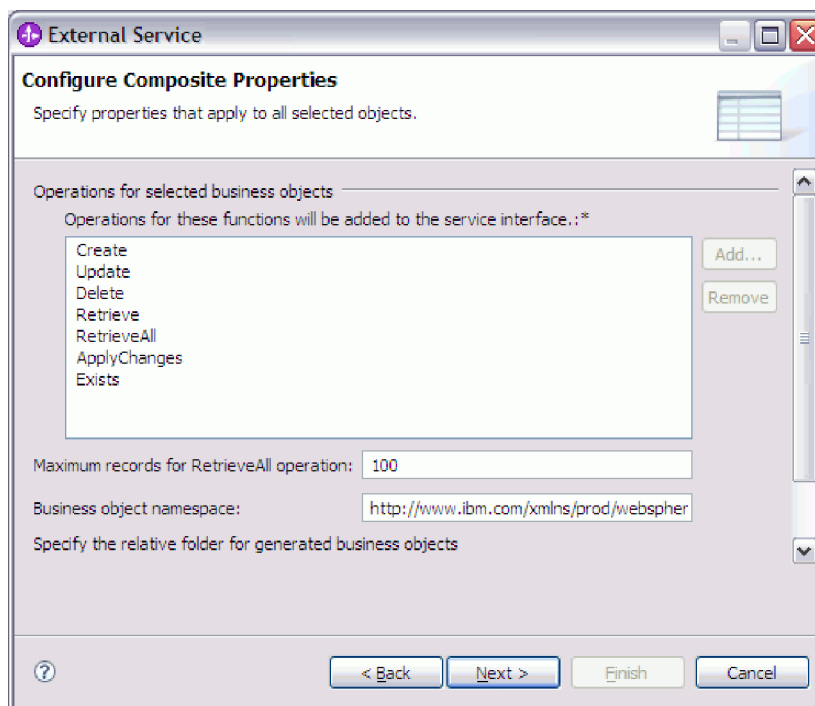
В окне Поиск и выбор объектов продолжите выбор и настройку других типов бизнес-объектов. После завершения нажмите кнопку **Далее** для задания глобальных свойств.

Установка глобальных свойств для операций

После выбора объектов базы данных в ESW (мастер внешних служб) необходимо указать свойства, применяемые ко всем бизнес-объектам.

Процедура

1. Если список **Выбранные объекты** в окне Поиск и выбор объектов содержит все бизнес-объекты, которые вы хотите использовать в приложении, нажмите кнопку **Далее**.
2. В окне Настроить составные свойства просмотрите список операций.
Это окно перечисляет все операции, которые поддерживаются адаптером для исходящих служб для всех бизнес-объектов, выбранных в предыдущем окне. Не все операции поддерживаются каждым бизнес-объектом. Например, бизнес-объекты запроса поддерживают только операцию RetrieveAll. Бизнес-объекты хранимых процедур поддерживают только операцию Execute.



3. Для того чтобы удалить операцию, которая не нужна, выберите ее имя и нажмите кнопку **Удалить**. Если вы передумали, нажмите **Добавить** и восстановите удаленную операцию.
4. В поле **Максимальное число записей для операций RetrieveAll** введите верхний предел числа записей, извлекаемых операцией RetrieveAll. Значение по умолчанию - 100. Более подробная информация об этом свойстве находится в разделе “Максимальное число записей для операции RetrieveAll” на стр. 186.

Примечание: Если операция RetrieveAll удалена на шаге 3, поле **Максимальное число записей для операций RetrieveAll** не будет доступно.

5. В окне **Пространство имен бизнес-объекта** примите пространство имен или введите полное имя другого пространства имен.
Пространство имен имеет префикс в имени бизнес-объекта для хранения логически разделенных схем бизнес-объектов.
6. По выбору, в поле **Папка** введите относительный путь для папки, в которой должны сохраняться сгенерированные бизнес-объекты.
7. Если вы хотите, чтобы схема бизнеса генерировалась для каждого бизнес-объекта, нажмите **Сгенерировать схему бизнеса для каждого бизнес-объекта**. Схемы бизнеса необходимы только в следующих ситуациях:
 - Если необходимо использовать операцию ApplyChanges

- При добавлении бизнес-объектов к модулю, созданному с помощью более ранней версии WebSphere Integration Developer, чем 6.1.x.

Примечание: Выберите этот параметр в случае добавления бизнес-объектов в модуль, созданный с помощью предыдущей версии WebSphere Integration Developer. В противном случае отключите интерфейс.

8. Нажмите кнопку **Далее**.

Результаты

Вы предоставили информацию, применяемую ко всем бизнес-объектам в модуле.

Дальнейшие действия

Продолжите работу в мастере. После этого необходимо указать сведения о развертывании, которые будут использоваться во время выполнения и данные для сохранения службы в виде модуля.

Установка свойств развертывания и генерация службы

После выбора и настройки бизнес-объектов для модуля следует настроить свойства, применяемые адаптером для подключения к базе данных. Мастер создаст модуль интеграции бизнес-процессов для сохранения всех артефактов и значений свойств.

Описание задачи

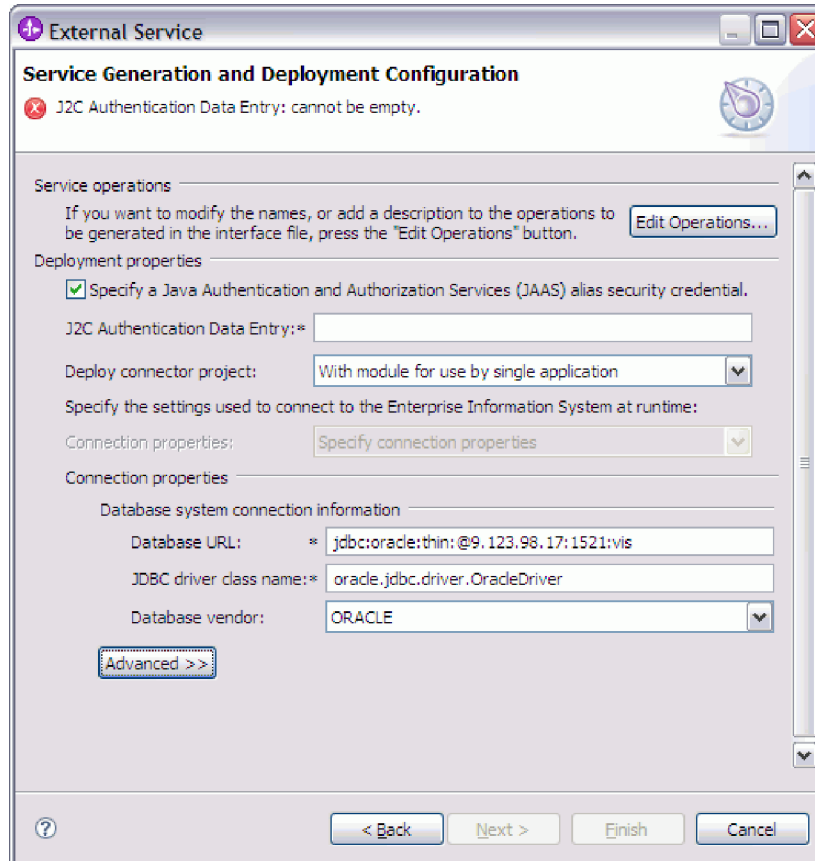
Эта задача выполняется с помощью окон Конфигурация создания и развертывания служб и Свойства расположения службы ESW (мастер внешних служб).

Свойства соединения в этой задаче инициализируются в соответствии со значениями, применяемыми мастером для подключения к базе данных. При необходимости здесь можно указать другие значения. Например, здесь можно указать драйвер Oracle JDBC вместо Oracle Thin.

Процедура

1. В окне Конфигурация создания и развертывания служб выберите **Изменить операции** для проверки имен или добавления описаний операций создаваемых

БИЗНЕС-ОБЪЕКТОВ.



2. Укажите способ получения идентификационных данных пользователя базы данных.

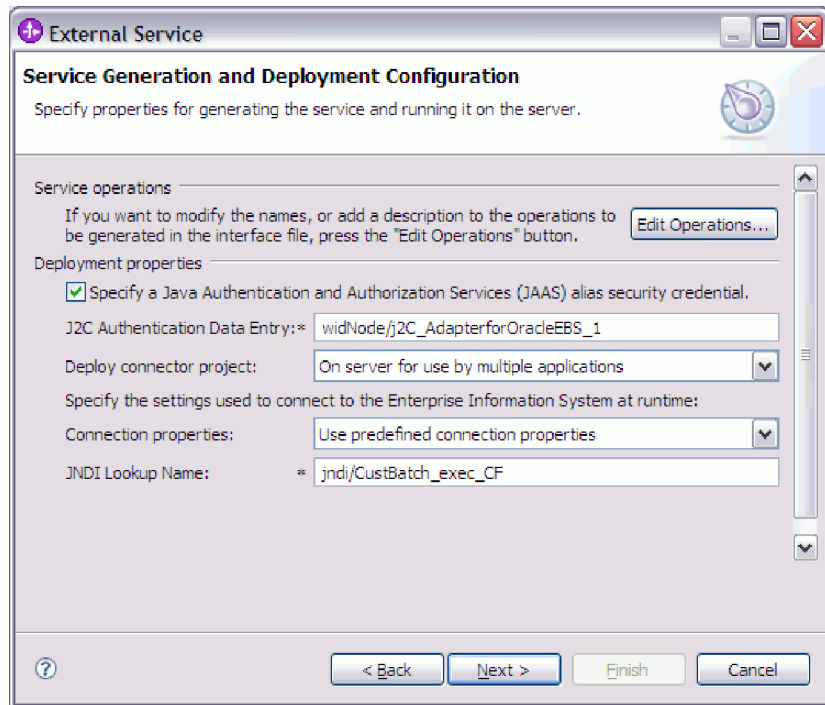
- Для применения псевдонима идентификации J2C выберите переключатель **Разрешение защиты псевдонима Службы идентификации Java (JAAS)** и введите имя псевдонима в поле **Запись данных идентификации J2C**.

Можно указать существующий псевдоним идентификации или создать новый перед развертыванием модуля. Имя обрабатывается с учетом регистра символов и содержит имя узла.

- Для применения существующего источника данных на сервер выполните следующие действия:
 - a. Отмените выбор переключателя **Разрешение защиты псевдонима Службы идентификации Java (JAAS)**.
 - b. Выберите **Дополнительно**.
 - c. Разверните **Альтернативные способы указания сведений о соединении**.
 - d. Заполните следующие поля:
 - **Имя JNDI Datasource**
 - **Имя источника данных ХА**
- Укажите идентификационные данные пользователя базы данных для сохранения в свойствах адаптера:
 - a. Отмените выбор переключателя **Разрешение защиты псевдонима Службы идентификации Java (JAAS)**.
 - b. Выберите **Дополнительно**.
 - c. В области **Свойства подключения к базе данных** введите значения в полях **Имя пользователя** и **Пароль**.

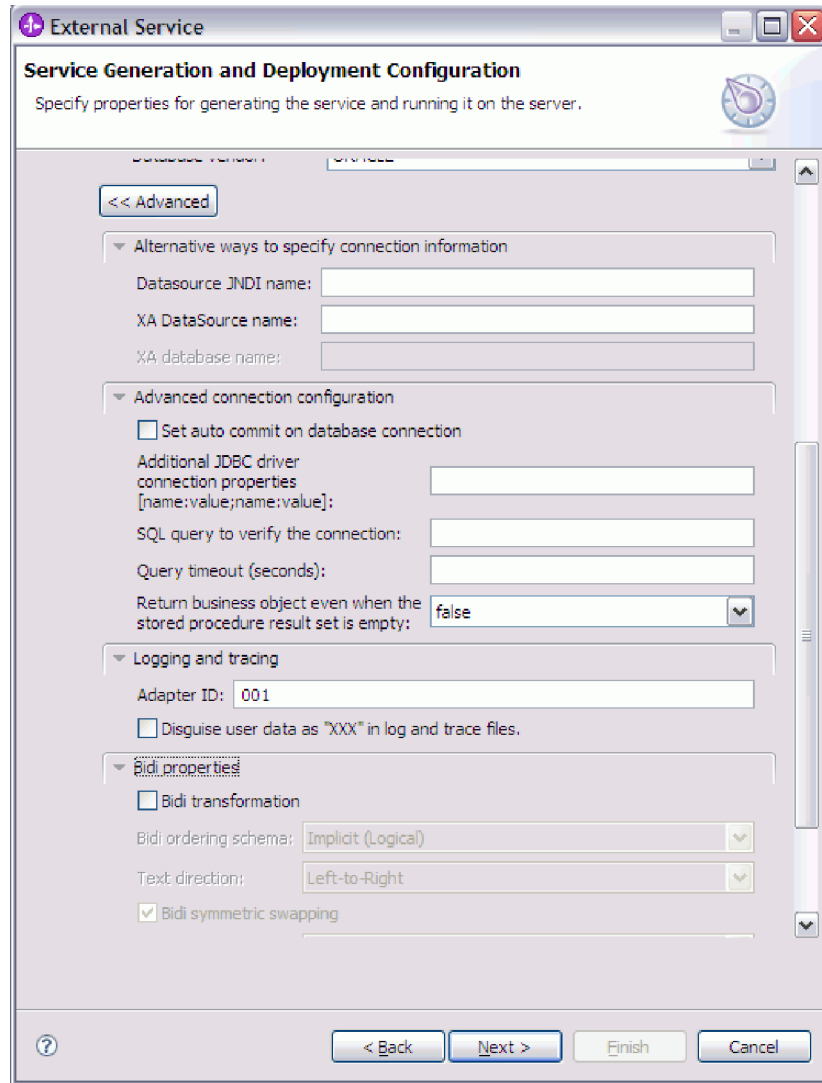
Примечание: Указанный пароль сохраняется в исходном виде в конфигурации адаптера. Его могут просмотреть несанкционированные пользователи.

3. В поле **Развернуть проект коннектора** укажите, включать ли файлы адаптера в модуль. Выберите одно из следующих значений:
 - **С модулем для использования одним приложением.** Модуль, содержащий файлы адаптера, можно развернуть на любом сервере приложений. Если необходимо, чтобы адаптер использовался единственным модулем или несколькими модулями, которые будут запускать различные версии адаптера, воспользуйтесь встроенным адаптером. Это позволит обновлять адаптер в одном модуле без риска дестабилизировать сменой версии адаптера другие модули.
 - **На сервере для использования несколькими приложениями.** Если вы не включаете файлы адаптера в модуль, то вы должны установить их как автономный адаптер на каждом сервере приложений, на котором вы хотите запустить модуль. Автономный адаптер следует использовать в том случае, если несколько модулей могут применять одну версию адаптера и вы хотите управлять адаптером в центральном расположении. Применение автономного адаптера позволяет также сократить количество требуемых ресурсов.
4. Если на предыдущем шаге было указано свойство На сервере - для нескольких адаптеров, то укажите способ задания свойств соединения.
 - В случае создания фабрики управляемых соединений или спецификации активации на сервере вручную или развертывания приложения, подключающегося в той же базе данных с помощью той же фабрики управляемых соединений или спецификации активации, допустимо повторное использование фабрики управляемых соединений или спецификации активации. Для этого следует указать имя соответствующего источника данных JNDI:
 - a. В поле **Свойства соединения** выберите Использовать предопределенные свойства соединения.
 - b. В поле **Имя поиска JNDI** введите имя JNDI источника данных для существующей фабрики управляемых соединений или спецификации активации.На следующем рисунке показаны типичные параметры для повторного использования фабрики управляемых соединений или спецификации активации в случае автономного развертывания адаптера.



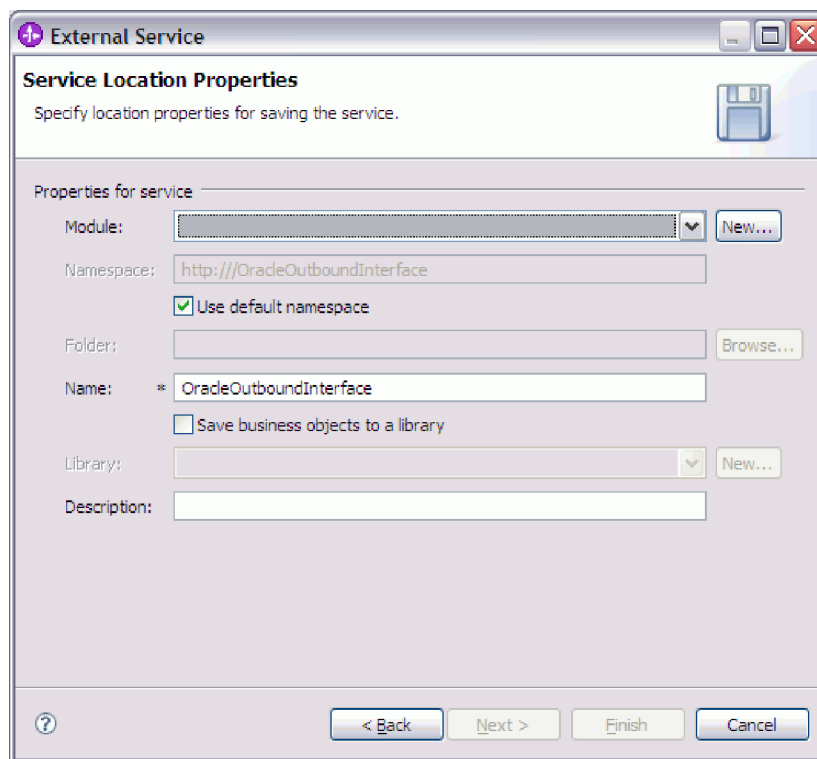
с. Нажмите кнопку **Далее** для завершения задачи.

- Если приложение подключается к базе данных от имени конкретного пользователя впервые или если идентификационными данными пользователя требуется управлять отдельно от других приложений, то выберите **Указать свойства соединения**.
5. Просмотрите и, если необходимо, измените значения требуемых свойств соединения. Поля инициализируются с информацией о подключении, указанной при запуске мастера. Можно изменить значения, чтобы указать другое имя пользователя и пароль во время выполнения. Можно также подключиться к другому Oracle DB, хотя имена схемы должны быть одинаковыми обеих баз данных. Формат свойств соединения специфичен для базы данных. Более подробная информация о о свойствах находится в разделе “Свойства фабрики управляемых соединений” на стр. 176.
 6. По выбору, укажите дополнительные свойства, щелкнув на **Дополнительные**. Разверните каждый из дополнительных разделов для просмотра свойств. На следующем рисунке показаны дополнительные свойства в окне Конфигурация создания и развертывания служб.



- Свойства, указанные в **Альтернативные способы указать информацию о соединении**, были описаны на предыдущем шаге.
- **Дополнительная конфигурация соединения**
 - a. Если вы хотите включить свойство AUTOCOMMIT для базы данных, выберите опцию **Установить автоматическую фиксацию для соединения базы данных**. Более подробная информация находится в разделе об этом свойстве находится в разделе “Автоматическая фиксация (AutoCommit)” на стр. 179.
 - b. Установите **Дополнительные свойства соединения с драйвером JDBC**. Более подробная информация находится в разделе об этом свойстве находится в разделе “Дополнительные свойства драйвера JDBC (DriverConnectionProperties)” на стр. 178.
 - c. Установите **Запрос SQL для проверки соединения**. Более подробная информация об этом свойстве находится в разделе “Запрос SQL для проверки соединения (PingQuery)” на стр. 184.
 - d. В поле **Тайм-аут запроса** введите интервал времени в секундах, в течение которого адаптер должен ждать ответа на запрос к базе данных. Более подробная информация находится в разделе об этом свойстве находится в разделе “Тайм-аут запроса (QueryTimeOut)” на стр. 183.

- e. Установите **Возвращать бизнес-объект, даже когда набор результатов хранимой процедура пуст**. Более подробная информация находится в разделе об этом свойстве находится в разделе “Возвращать бизнес-объект при получении пустого набора результатов хранимой процедуры (ReturnDummyBOForSP)” на стр. 184.
 - **Ведение протоколов и трассировка**
 - Если существует несколько экземпляров адаптера, установите значение ИД адаптера , которое уникально для этого экземпляра.
 - Если вы хотите скрыть потенциально секретную информацию о пользователе в файлах протокола и трассировки, выберите поле **Скрыть пользовательские данные в виде “XXX” файлах протокола и трассировки**.
7. Нажмите кнопку **Далее**. Будет показано окно Свойства расположения службы.



8. В окне Свойства расположения службы укажите имя модуля, который вы хотите создать. Это может быть новый или существующий модуль.
- Если предпочитаемое имя модуля показано в списке **Модуль**, выберите его.
- Важное замечание:** Если модуль содержит интерфейс или бизнес-объект, имя которого совпадает с настраиваемым интерфейсом или объектом, то исходный интерфейс или бизнес-объект будет заменен новой версией.
- В противном случае создайте новый модуль:
 - a. Нажмите кнопку **Создать**.
 - b. В окне Проект интеграции выберите **Создать проект модуля** и нажмите **Далее**.
 - c. В окне Модуль введите имя модуля. Например, OracleOutboundModule.
 - d. Если вы хотите, чтобы файлы описания службы (.import и .wsdl) были расположены в папке по умолчанию для модуля, оставьте выбранным поле

Использовать расположение по умолчанию. Если вы хотите указать другую папку в модуле, очистите опцию, а затем нажмите **Обзор**, чтобы найти другую папку для поля **Расположение**.

- e. Если вы хотите, чтобы модуль автоматически открывался в диаграмме сборки в WebSphere Integration Developer при закрытии мастера, выберите **Открыть диаграмму сборки модуля**. В противном случае очистите эту опцию.
 - f. Нажмите кнопку **Готово**, чтобы создать новый модуль.
9. Укажите пространство имен, которое вы хотите использовать для бизнес-объектов.
- Если вы хотите, чтобы бизнес-объекты в модуле использовали порожденное пространство имен по умолчанию, оставьте выбранной опцию **Использовать пространство имен по умолчанию**.
 - Для того чтобы указать другое пространство имен, очистите эту опцию и введите другое значение в поле **Пространство имен**.
10. По выбору, укажите папку в новом модуле, в которой будет храниться описание службы. В поле **Папка** введите имя папки или найдите существующую. Если имя папки не указано, то артефакты (файлы импорта, XSD и WSDL) сохраняются в корневой папке модуля, то есть папке с именем модуля.
11. В поле **Имя** примите имя импорта по умолчанию или введите другое имя.
12. По выбору, если вы хотите сохранить бизнес-объекты в библиотеке, где они могут быть использованы другими модулями, выберите **Сохранить бизнес-объекты в библиотеке** и укажите расположение библиотеки в поле **Библиотека**.
13. По выбору, в поле **Описание** введите описание модуля.
14. После завершения установки свойств нажмите **Готово**.
15. Если показано окно Модель изменена, нажмите **Да**.

Результаты

Мастер закрыт. Модуль создан в проекте и артефакты сгенерированы.

Дальнейшие действия

В некоторых случаях может потребоваться использовать редактор сборки для заполнения конфигурации. Затем можно протестировать и развернуть модуль.

Завершение настройки

В некоторых ситуациях, необходимо выполнить действия вручную для завершения настройки бизнес-объектов.

Описание задачи

Выполните эту задачу, когда необходимо настроить артефакты, сгенерированные мастером. Это может потребоваться в следующих ситуациях:

- Для установки параметра `CoruAttribute` для столбца, чтобы его значение совпадало с другим столбцом.
- Для добавления или удаления атрибутов из бизнес-объекта. Например, можно упростить бизнес-объект, удалив простые атрибуты, соответствующие столбцам базы данных, на которые не нужно ссылаться.

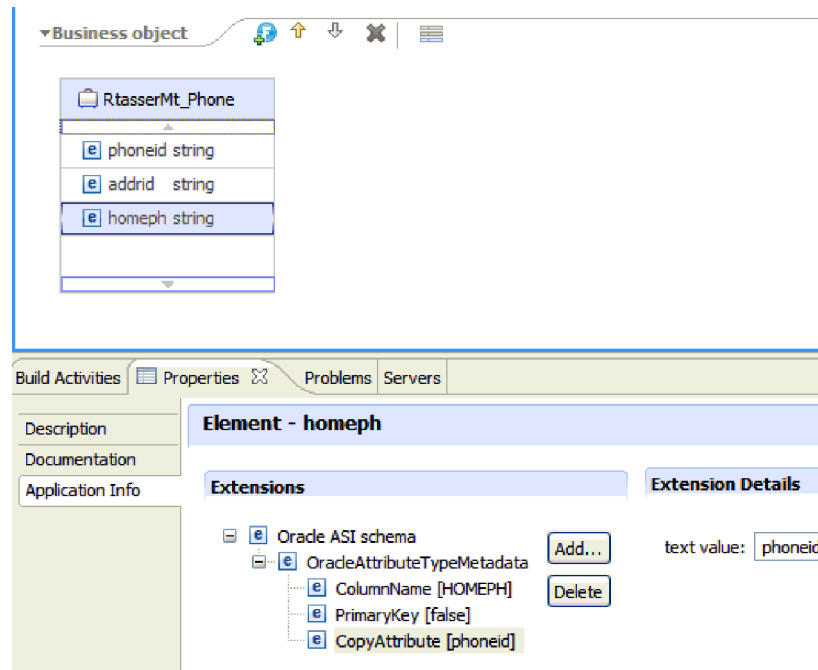
- Для настройки дополнительных родительских объектов бизнес-объекта таблицы, который имеет несколько родительских объектов. Мастер настраивает только один родительский объект для бизнес-объекта таблицы.

Этот раздел предоставляет подробные инструкции для установки параметра CopyAttribute бизнес-объекта таблицы. Другие изменения структур бизнес-объектов могут быть выполнены с помощью подобных приемов.

Параметр CopyAttribute содержится в свойствах атрибута для столбца, который вы хотите заполнить значениями и информацией для приложения из другого столбца. Например, если вы хотите, чтобы столбец контактная_информация новой строки в таблице содержал такое же значение, что и столбец электронная_почта, установите параметр CopyAttribute атрибута контактная_информация в электронная_почта. Для установки значения используется редактор сборки в WebSphere Integration Developer.

Процедура

1. В проекции Интеграция бизнес-процессов в WebSphere Integration Developer разверните модуль, разверните **Тип данных**, а затем найдите бизнес-объект таблицы. Именем бизнес-объекта является имя схемы базы данных плюс имя таблицы базы данных. Необязательное пространство имен может быть включено в начало имени.
2. Щелкните правой кнопкой мыши на имени бизнес-объекта и выберите **Открыть**. Редактор сборки показывает бизнес-объект, который имеет поле для каждого столбца.
3. В редакторе сборки выберите столбец, который должен соответствовать другому столбцу.
4. На панели Свойства выберите Информация приложения. Если панель Свойства не видна, щелкните правой кнопкой мыши на имени столбца и выберите **Показать в свойствах**.
5. Разверните узел **Схема ASI Oracle**, а затем разверните **OracleAttributeTypeMetadata**.
6. Щелкните правой кнопкой мыши на **OracleAttributeTypeMetadata** и выберите **Создать** → **oracleasi:CopyAttribute**.
7. Выберите свойство **CopyAttribute**.
8. В области Сведения о расширении установите текстовое значение равным имени столбца, который содержит информацию для копирования. Столбец может принадлежать текущему бизнес-объекту или его родительскому бизнес-объекту. Для копирования из столбца текущего бизнес-объекта установите значение равным имени столбца, например, phoneid. Для копирования из столбца родительского бизнес-объекта, добавьте к имени столбца префикс из двух точек (..), например, ..phone. Следующий рисунок показывает редактор сборки со свойством CopyAttribute, установленным как столбец текущей таблицы.



Результаты

Бизнес-объект настроен для использования свойства CopyAttribute для установки атрибута бизнес-объекта и свойств столбца базы данных на основании информации из другого столбца.

Дальнейшие действия

Теперь можно протестировать и развернуть модуль.

Настройка модуля для обработки входящих событий

Настройка модуля для обработки входящих событий предусматривает применение ESW (мастер внешних служб) в WebSphere Integration Developer для поиска и выбора бизнес-объектов и служб в Oracle DB, а также создания определений бизнес-объектов и связанных артефактов.

Поиск объектов базы данных

После настройки свойств соединения можно отправить запрос на поиск объектов базы данных. Откройте дерево найденных объектов, чтобы получить общее представление о структуре объектов Oracle DB. С помощью фильтров можно отобразить только объекты базы данных, которые вы хотите видеть.

Перед тем, как начать

Необходимо знать требование программы к данным для доступа к базе данных. Например, необходима следующая информация о базе данных:

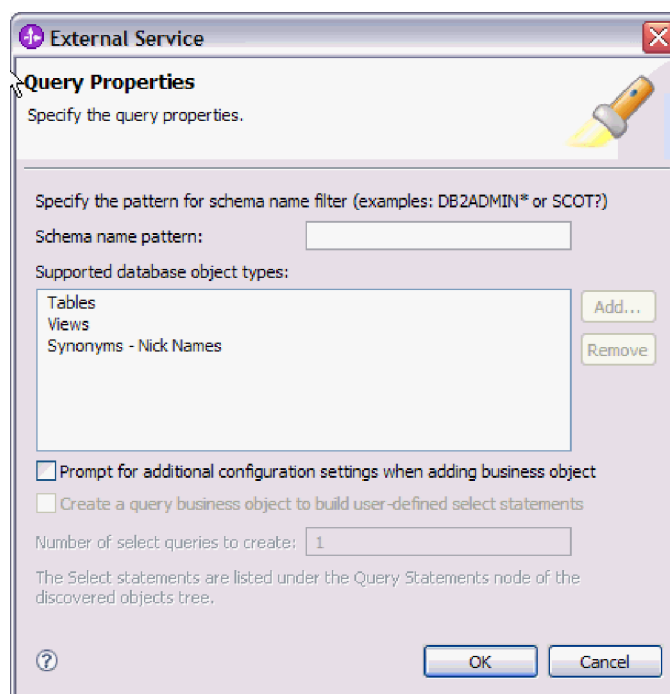
- Какие схемы необходимы модулю для доступа
- Какой тип объектов базы данных необходим для доступа в этих схемах

Описание задачи

Эта задача начинается в окне Поиск и выбор объектов ESW (мастер внешних служб).

Процедура

1. В окне Поиск и выбор объектов нажмите **Изменить запрос**. Будет показано окно Свойства запроса.



Обратите внимание на недоступную опцию **Создать бизнес-объект запроса для компоновки пользовательских операторов выбора**. Эта опция доступна только для исходящей обработки.

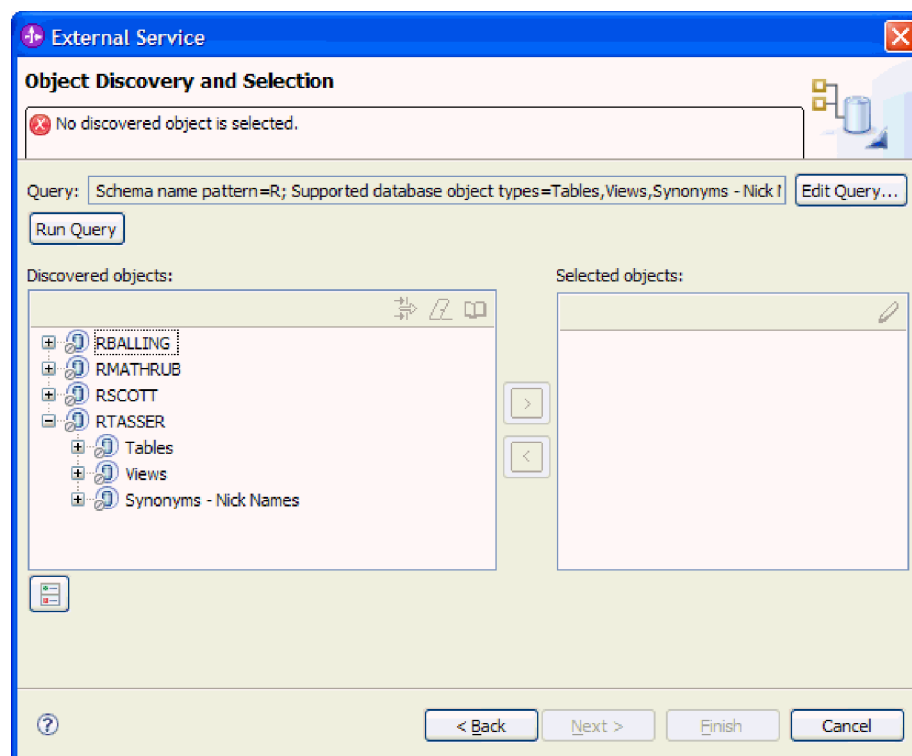
Используйте окно Свойства запроса для выполнения следующих задач:

- Уменьшить время поиска, ограничив подмножество схем базы данных
 - Пропустить один или несколько типов объектов базы данных в поиске
 - Позволить мастеру запрашивать информацию, специфичную для приложения, которая не может быть автоматически определена на основании информации в базе данных
2. Для ограничения числа отображаемых схем баз данных введите имя или шаблон имени схемы в поле **Имя схемы или шаблон**. Знак вопроса или подчеркивание (?) или _) заменяет один произвольный символ, звездочка или процент (* или %) - несколько символов. Запрос возвратит только те схемы, имена которых начинаются с указанной строки или совпадают с ней. Если шаблон не указан, то отображаются все схемы из базы данных. Путем применения фильтра можно ускорить процесс поиска, если база данных содержит большое число схем.
 3. Для того чтобы пропустить один или несколько типов объектов в поиске, выберите эти типы (таблицы, представления, синонимы или псевдонимы) в **Поддерживаемые типы объектов базы данных**, а затем нажмите кнопку **Удалить**. Если вы передумали, нажмите **Добавить** для возвращения типов объектов назад. Если база данных содержит типы объектов, к которым не нужен доступ, их пропуск может ускорить процесс поиска.
 4. Выберите переключатель **Запрашивать дополнительные параметры при добавлении бизнес-объекта**. Затем при добавлении объекта базы данных в список создаваемых бизнес-объектов мастер автоматически предложит ввести настраиваемую информацию уровня приложения для объекта. Например, мастер может в

пошаговом режиме создать иерархическую структуру бизнес-объектов. Если требуется создать иерархическую структуру, в которой бизнес-объект таблицы содержит два атрибута, которые связаны с атрибутами из двух других таблиц (два родительских бизнес-объекта), то настройку следует выполнить с помощью редактора сборки (запускается из WebSphere Integration Developer).

Важное замечание: Если этот переключатель не выбран, то мастер запрашивает только обязательную информацию. Настройку бизнес-объектов потребуется завершить с помощью редактора сборки.

5. Нажмите **ОК** для сохранения изменений в запросе.
6. В окне Поиск и выбор объектов нажмите **Выполнить запрос**, чтобы применить запрос для поиска объектов базы данных. Результат выполнения обычного запроса показан на следующем рисунке.



Панель **Найденные объекты** перечисляет найденные объекты. Таблицы, представления и синонимы/псевдонимы отсортированы по имени схемы.

7. В списке **Найденные объекты** нажмите **+** (знак плюс), чтобы развернуть узел схемы и узлы **Таблицы**, **Представления** и **Синонимы - Псевдонимы** под ним для просмотра объектов базы данных, найденных мастером.

Результаты

Мастер нашел объекты базы данных, к которым можно получить доступ с помощью адаптера.

Дальнейшие действия

Продолжите работу в ESW (мастер внешних служб). Следующий шаг состоит в выборе объектов, которые вы хотите использовать в модуле, настройке каждого бизнес-объекта и создании иерархии бизнес-объектов.

Выбор и настройка бизнес-объектов

С помощью списка объектов баз данных, обнаруженных ESW (мастер внешних служб), продолжите выбор объектов баз данных, которые должны быть доступны модулю. Затем укажите информацию о конфигурации новых бизнес-объектов.

Описание задачи

Окно Поиск и выбор объектов позволяет выбрать и настроить объекты в любом порядке, с тем исключением, что необходимо выбрать и настроить родительскую таблицу перед выбором и настройкой дочерних таблиц. Кроме этого ограничения, вы можете добавлять объекты по-отдельности или группами. Можно смешивать объекты из различных узлов списка **Найденные объекты**. Например, можно выбрать несколько объектов таблиц и представлений, объект хранимой процедуры, а затем добавить и настроить их одновременно.

Поток высокого уровня выбора и настройки бизнес-объектов выглядит следующим образом:

1. Выберите один или несколько объектов в списке **Найденные объекты** окна Поиск и выбор объектов.
2. Нажмите кнопку **>** (Добавить).
3. Откроется окно Свойства конфигурации.

- Если выбран отдельный объект, то откроется одно окно Свойства конфигурации.

В этом окне укажите пользовательские атрибуты и прочую информацию, которую мастер не может найти в базе данных, и нажмите кнопку **ОК** для сохранения конфигурации.

- Если выбрано несколько бизнес-объектов, то откроется окно Свойства конфигурации с отдельными страницами для каждого выбранного бизнес-объекта.

Последовательно щелкните на имени каждого объекта. Откроется окно, содержащее такую же информацию, как в случае выбора отдельного объекта.

Важное замечание: Не нажимайте кнопку **ОК** до тех пор, пока на будут заполнены страницы конфигурации всех объектов. Мастер запрещает закрытие окна до заполнения обязательных полей, однако окно можно закрыть перед указанием необязательных полей. Если необязательные поля не указаны, то после выхода из мастера их следует настроить с помощью редактора бизнес-объектов.

4. Мастер добавляет настроенный объект в список **Выбранные объекты**.

До закрытия мастера можно несколько раз выбирать и настраивать бизнес-объекты, необходимые для работы модуля. Однако с помощью мастера нельзя добавить объекты в существующие модули. Перед запуском мастера рекомендуется внимательно изучить требования программы, использующей бизнес-объекты.

Выбор и настройка таблиц, представлений и синонимов или псевдонимов

Выберите и настройте бизнес-объекты для таблиц, представлений и синонимов или псевдонимов для использования в модуле. Для входящей обработки ими являются бизнес-объекты, доставленные в событиях.

Перед тем, как начать

Для выполнения этой задачи требуется знание структуры данных в базе данных, а также объектов базы данных, с которыми должен взаимодействовать модуль. В частности требуется следующая информация:

- Структура таблиц, представлений и синонимов (псевдонимов), включая применяемые столбцы и атрибуты столбцов, такие как тип данных.
- Взаимосвязи между таблицами, включая число значений и принадлежность.

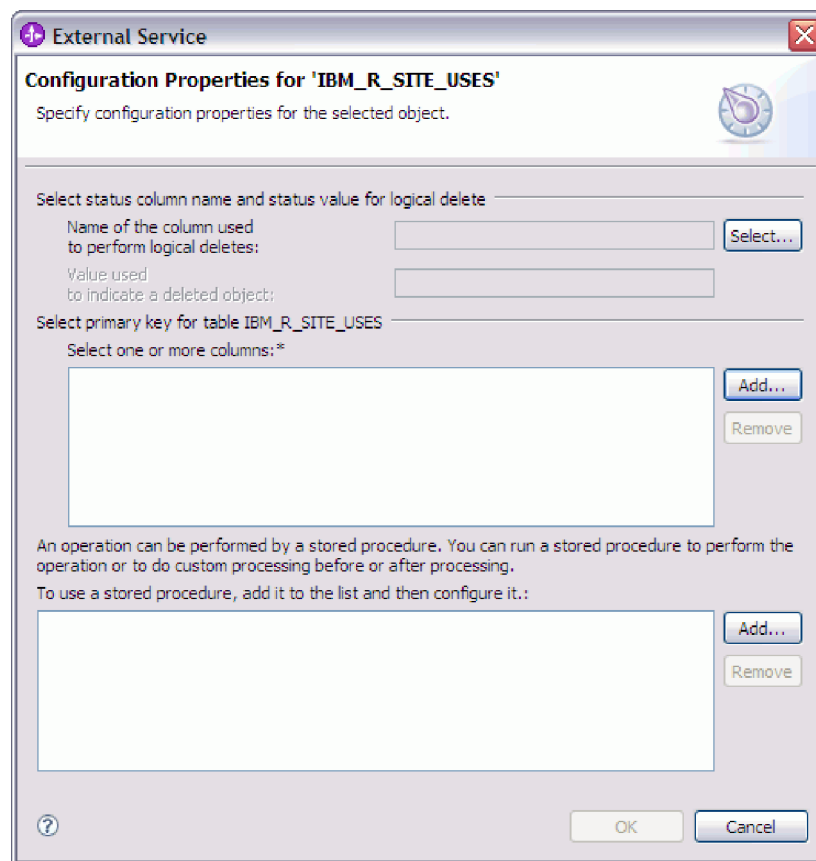
Описание задачи

Эта задача выполняется с помощью ESW (мастер внешних служб). Пользователь начинает в окне Поиск и выбор объектов и продолжает работу в окне Свойства конфигурации настраиваемого бизнес-объекта.

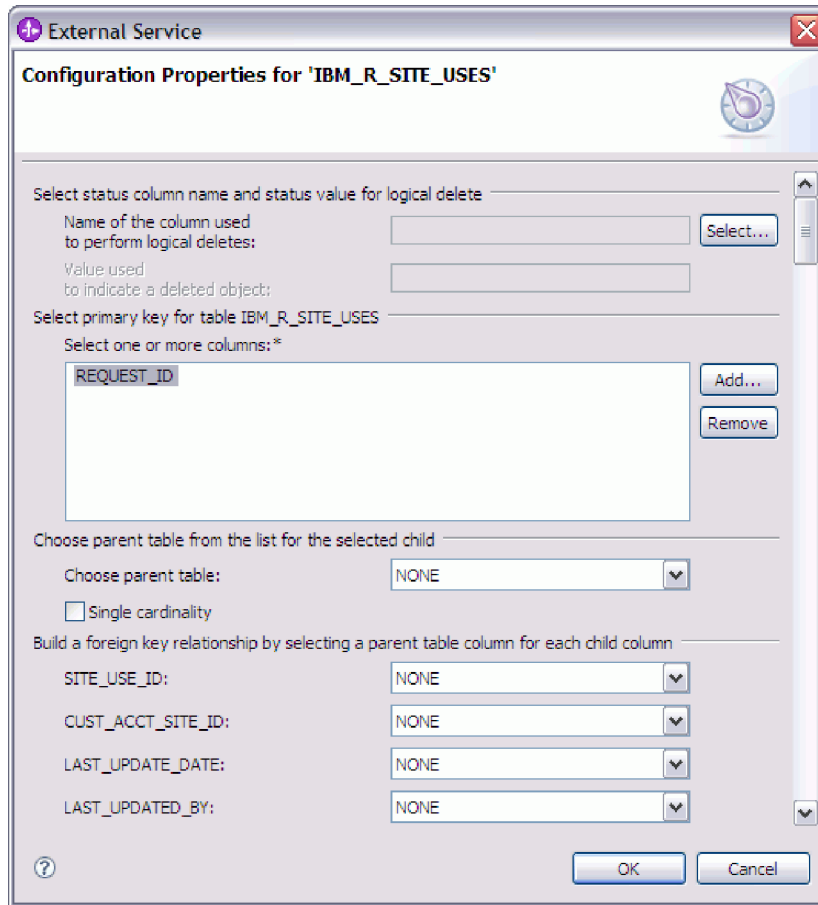
Процедура

1. В списке **Найденные объекты** окна Поиск и выбор объектов выберите одну или несколько таблиц, представлений или синонимов и нажмите кнопку **> (Добавить)** для добавления объектов в список **Выбранные объекты**.

На следующих рисунках показано окно Свойства конфигурации таблиц, представлений или синонимов. На первом рисунке показано окно для первой выбранной таблицы или группы таблиц.



На следующем рисунке показано окно для последующих таблиц. После выбора и настройки по крайней мере одной таблицы в окне Свойства конфигурации для последующих таблиц отображается область, в которой можно задать иерархическую структуру таблиц.

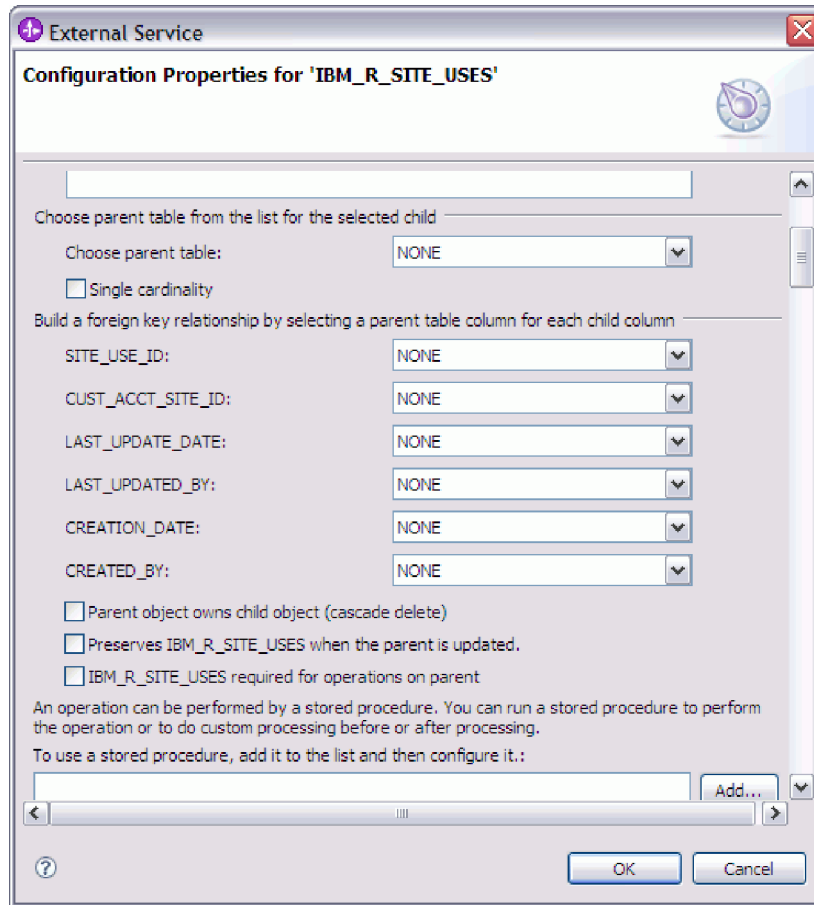


В ходе настройки объекта варианты, требующие дополнительной настройки, могут отображать в окне дополнительные поля, требующие прокрутки окна. Перед нажатием кнопки **ОК** проверьте все поля в окне.

2. Если таблица содержит столбец, в котором указываются операции логического удаления:
 - a. Выберите имя столбца в поле **Имя столбца, используемого для выполнения операций логического удаления**.
 - b. В поле **Значение, указывающее на удаленный объект** укажите значение, указывающее на логическое удаление строки. Это значение можно получить у администратора базы данных.
3. Если показана область **Выберите первичный ключ для таблицы имя-таблицы**, то нажмите кнопку **Добавить**, выберите столбец первичного ключа и нажмите кнопку **ОК**. Если таблица содержит составной ключ, то можно выбрать несколько столбцов. Область **Выберите первичный ключ для таблицы имя-таблицы** отображается только в том случае, если таблица базы данных не содержит столбец первичного ключа. Каждый бизнес-объект таблицы должен содержать первичный ключ. Если первичный ключ задан в базе данных, то этот раздел окна не отображается.
4. При необходимости укажите взаимосвязь между бизнес-объектами.

Для создания иерархической структуры сначала настройте родительскую таблицу, затем вернитесь к окну Поиск и выбор объектов для выбора и настройки дочерних таблиц.

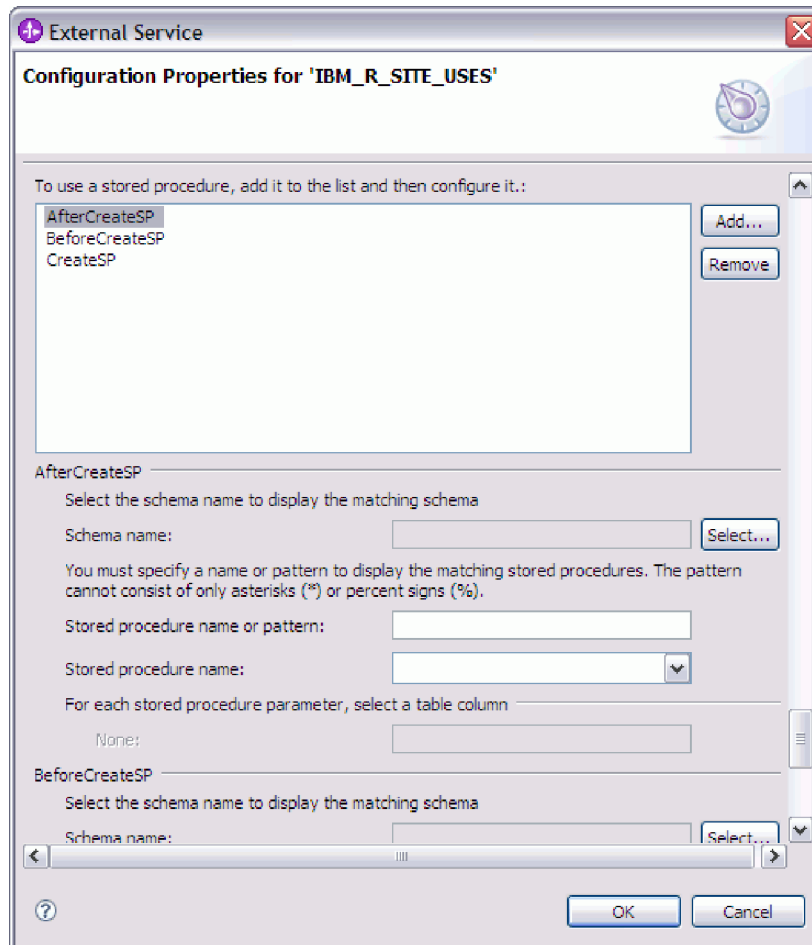
Настройте взаимосвязь с помощью окна Свойства конфигурации (см. следующий рисунок). Эти поля не отображаются в ходе настройки первой таблицы.



- a. В поле **Выберите родительскую таблицу** выберите имя родительской таблицы. Если родительская таблица не показана в списке, то она еще не настроена. Вернитесь и настройте родительскую таблицу перед настройкой дочерних объектов.
- b. Укажите число значений взаимосвязи:
 - Для взаимосвязи с одним значением выберите переключатель **Единственность**. В этом случае родительский объект может содержать только один дочерний бизнес-объект этого типа. Взаимосвязь с одним значением может представлять дочерний объект, таблицы поиска или равноправные объекты в базе данных.
 - Для взаимосвязи с несколькими значениями отмените выбор переключателя **Единственность**. В этом случае родительский объект может содержать массив дочерних бизнес-объектов указанного типа.
- c. Создайте взаимосвязь внешнего ключа между родительским и дочерним объектами, указав дочерние столбцы, выполняющие роль внешнего ключа в родительской таблице.
 - Если дочерний столбец не является внешним ключом, то выберите Нет.
 - Если дочерний столбец является внешним ключом, то выберите в родительской таблице столбец, соответствующий дочернему столбцу.

Примечание: Мастер позволяет настроить только одну родительскую таблицу. Если дочерняя таблица содержит несколько родительских таблиц, то после выхода из мастера их следует настроить с помощью редактора бизнес-объектов.

- d. Если дочерний объект принадлежит родительскому объекту, то дочерние объекты удаляются из базы данных вместе с родительским. Для указания такой взаимосвязи выберите переключатель **Родительскому объекту принадлежит дочерний объект (каскадное удаление)**. В противном случае отмените выбор этого переключателя во избежание удаления дочерних объектов, таких как таблицы поиска, вместе с родительскими объектами.
 - e. Если дочерние объекты не должны удаляться в ходе операции Update, выберите переключатель **Сохранять имя-дочерней-таблицы при обновлении родительского объекта**.
В ходе обновления родительской таблицы адаптер сравнивает входные дочерние бизнес-объекты с дочерними бизнес-объектами, возвращенными из базы данных. По умолчанию адаптер удаляет все возвращенные из базы данных дочерние бизнес-объекты, отсутствующие во входном бизнес-объекте.
 - f. По умолчанию операции над родительскими бизнес-объектами можно выполнять, не указывая дочерние бизнес-объекты. Для гарантированного изменения дочерних бизнес-объектов вместе с родительским бизнес-объектом выберите переключатель **имя-дочерней-таблицы требуется для обработки родительского объекта**.
5. Операцию можно выполнить с помощью стандартного оператора SQL, созданного адаптером, либо с помощью хранимых процедур или функций из базы данных. Для применения хранимых процедур и функций выполните следующие действия:
- a. Нажмите кнопку **Добавить**.
 - b. В окне **Добавить** выберите тип хранимой процедуры. Для каждой операции можно выбрать хранимую процедуру, выполняющую операцию, а также хранимые процедуры, выполняемые до и после операции. Например, для операции **Create** можно указать следующие хранимые процедуры: **CreateSP**, **BeforeCreateSP** и **AfterCreateSP**.
 - c. Нажмите кнопку **ОК**. В окне **Свойства конфигурации** будут показаны выбранные типы хранимых процедур и их параметры. Для просмотра параметров может потребоваться прокрутить окно вниз.



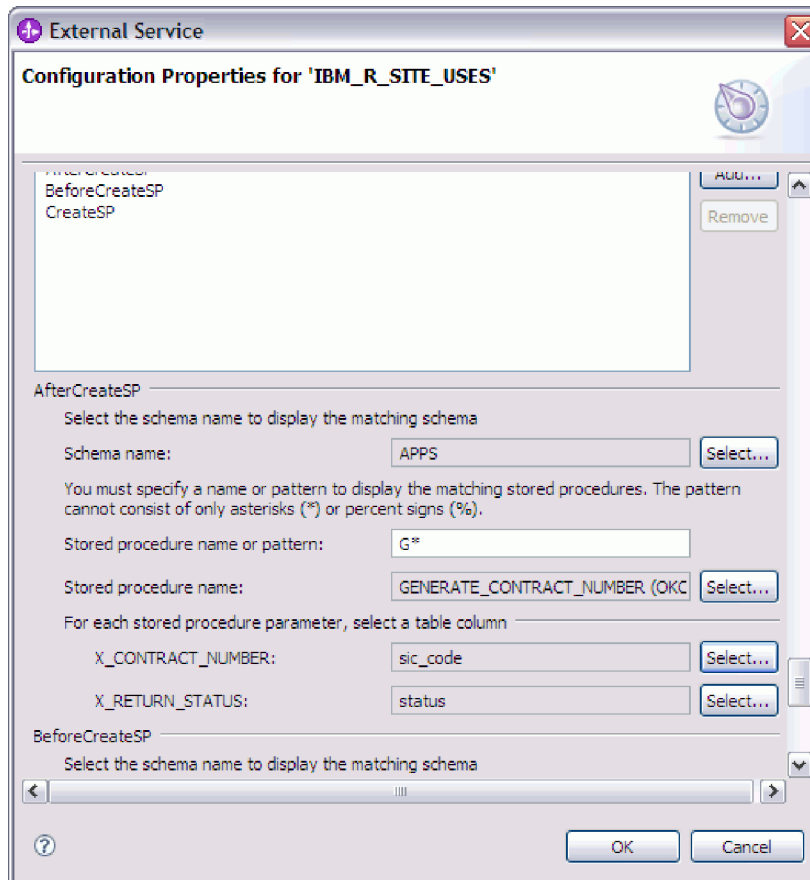
Примечание: В бизнес-объекте с иерархической структурой для выполнения хранимой процедуры над всем бизнес-объектами необходимо отдельно связать хранимую процедуру с бизнес-объектом верхнего уровня и каждым дочерним бизнес-объектом. Если связать хранимую процедуру только с бизнес-объектом верхнего уровня, то хранимая процедура будет обрабатывать только бизнес-объект верхнего уровня - дочерние бизнес-объекты будут обрабатываться с помощью стандартного запроса SQL.


6. Для каждого выбранного типа хранимой процедуры укажите имя хранимой процедуры в базе данных и настройте бизнес-объект.
 - a. В поле **Имя схемы** выберите имя схемы, содержащей хранимую процедуру.
 - 1) Нажмите кнопку **Выбрать**.
 - 2) В окне Выбрать значение выберите нужную схему.
 - 3) Нажмите **ОК**.
 - b. Укажите имя хранимой процедуры или функции.
 - 1) В поле **Имя хранимой процедуры или шаблон** введите имя хранимой процедуры или функции либо введите шаблон имени. Знак вопроса или подчеркивание (? или _) заменяет один произвольный символ, звездочка или процент (* или %) - несколько символов.
 - 2) В поле **Имя хранимой процедуры** выберите имя процедуры. Если список хранимых процедур содержит большое число записей, то рядом с полем

Имя хранимой процедуры отображается кнопка **Выбрать**. Нажмите кнопку **Выбрать**, для того чтобы выбрать имя хранимой процедуры или функции в окне **Выбрать**.

В окне **Свойства конфигурации** будет показана область, позволяющая настроить хранимую процедуру. Мастер автоматически создает список параметров путем проверки хранимой процедуры в базе данных.

- c. Для каждого параметра хранимой процедуры (слева) выберите столбец таблицы (справа) для передачи хранимой процедуры в соответствующий параметр. На следующем рисунке показано окно после настройки хранимой процедуры.



7. После заполнения всех полей в окне нажмите кнопку **ОК** для сохранения конфигурации бизнес-объекта. В окне **Поиск и выбор объектов** показаны таблица, представление, синоним и псевдоним.
8. Для изменения конфигурации объекта в списке **Выбранные объекты** выберите имя объекта и щелкните на значке  (**Изменить**).
9. После выбора и настройки всех необходимых бизнес-объектов нажмите кнопку **Далее** для установки глобальных свойств и настройки бизнес-объекта оболочки.

Дальнейшие действия

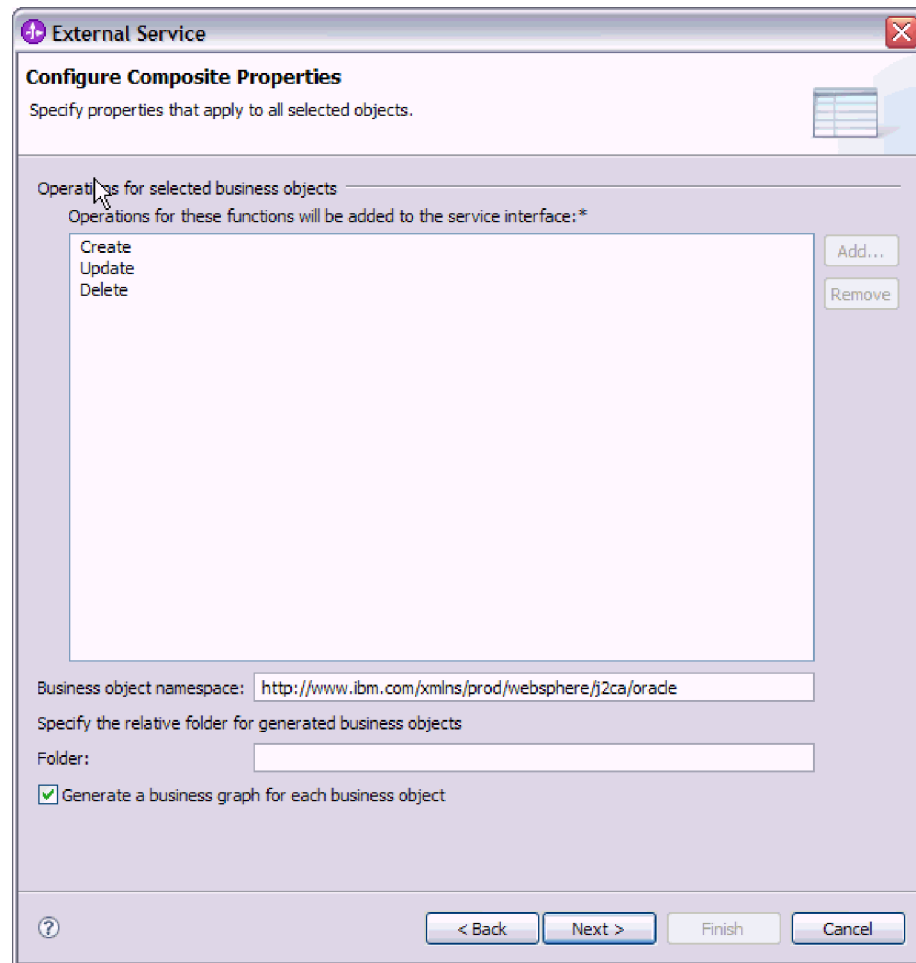
В окне **Поиск и выбор объекта** выберите и настройте другие типы бизнес-объектов.

Установка глобальных свойств для операций

После выбора объектов базы данных в ESW (мастер внешних служб) необходимо указать свойства, применяемые ко всем бизнес-объектам.

Процедура

1. Если список **Выбранные объекты** в окне Поиск и выбор объектов содержит все бизнес-объекты, которые вы хотите использовать в приложении, нажмите кнопку **Далее**.
2. В окне Настроить составные свойства просмотрите список операций. Этот список содержит операции, которые адаптер поддерживает для входящих служб. Включенные в список операции будут добавлены для всех бизнес-объектов, выбранных в предыдущем окне.
Указанные операции устанавливаются для всех сгенерированных бизнес-объектов.



3. Для того чтобы удалить операцию, которая не нужна, выберите ее имя и нажмите кнопку **Удалить**. Если вы передумали, нажмите **Добавить** и восстановите удаленную операцию.
4. В окне **Пространство имен бизнес-объекта** примите пространство имен или введите полное имя другого пространства имен.
Пространство имен имеет префикс в имени бизнес-объекта для хранения логически разделенных схем бизнес-объектов. Более подробная информация об этом свойстве находится в разделе "Пространство имен бизнес-объекта (BusinessObjectNameSpace)" на стр. 201.
5. По выбору, в поле **Папка** введите относительный путь для папки, в которой должны сохраняться сгенерированные бизнес-объекты.
6. Если вы хотите, чтобы схема бизнеса генерировалась для каждого бизнес-объекта, нажмите **Сгенерировать схему бизнеса для каждого бизнес-объекта**. Схемы бизнеса

нужны при добавлении бизнес-объектов к модулю, созданному с помощью более ранней версии WebSphere Integration Developer, чем 6.1.x.

Примечание: Необходимо выбрать эту опцию при добавлении бизнес-объектов к модулю, который был создан с помощью более ранней версии WebSphere Integration Developer. В противном случае необходимо переписать интерфейс.

7. По завершении нажмите **Далее**.

Результаты

Вы предоставили информацию, применяемую ко всем бизнес-объектам в модуле.

Дальнейшие действия

Продолжите работу в мастере. После этого необходимо указать сведения о развертывании, которые будут использоваться во время выполнения и данные для сохранения службы в виде модуля.

Установка свойств развертывания и генерация службы

После выбора и настройки бизнес-объектов для модуля следует настроить свойства, применяемые адаптером для подключения к базе данных. Мастер создаст модуль интеграции бизнес-процессов для сохранения всех артефактов и значений свойств.

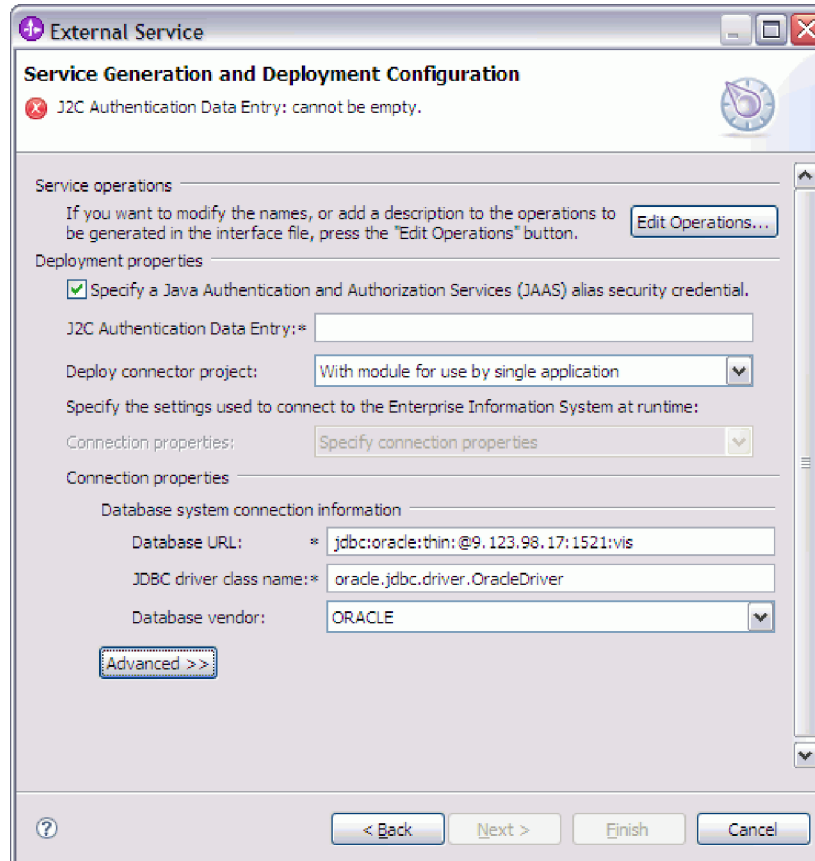
Описание задачи

Эта задача выполняется с помощью окон Конфигурация создания и развертывания служб и Свойства расположения службы ESW (мастер внешних служб).

Свойства соединения в этой задаче инициализируются в соответствии со значениями, применяемыми мастером для подключения к базе данных. При необходимости здесь можно указать другие значения. Например, здесь можно указать драйвер Oracle JDBC вместо Oracle Thin.

Процедура

1. В окне Конфигурация создания и развертывания служб выберите **Изменить операции** для проверки имен или добавления описаний операций создаваемых



2. Укажите способ получения адаптером имени пользователя базы данных и пароля во время выполнения.

- Для того чтобы использовать псевдоним идентификации J2C, выберите **Указать одноразовое разрешение защиты псевдонима Службы идентификации Java (JAAS)** и введите имя псевдонима в поле **Запись данных идентификации J2C**.

Можно указать существующий псевдоним идентификации или создать новый в любой момент перед развертыванием модуля. В имени учитывается регистр, и оно включает в себя имя узла.

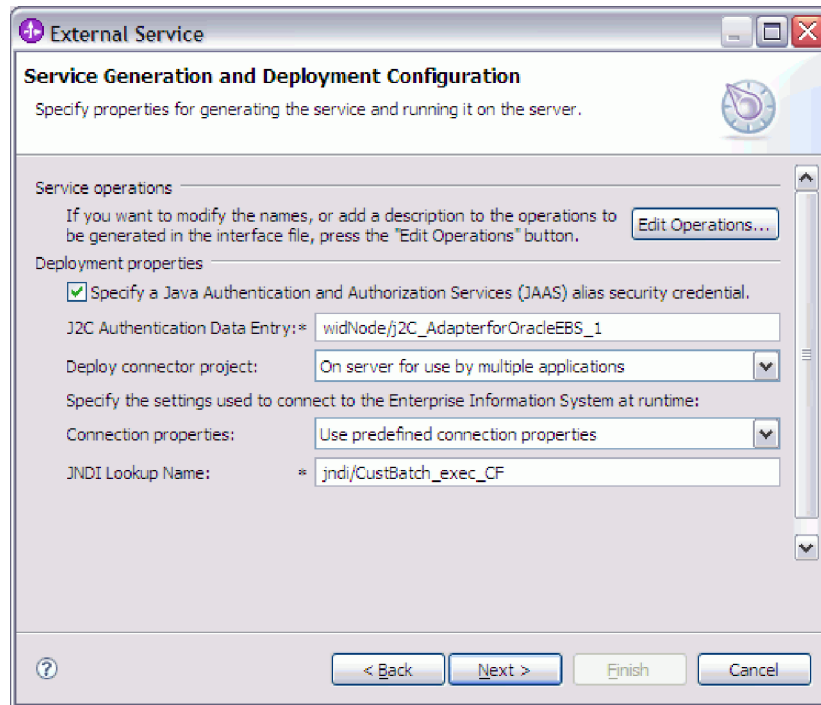
- Для того чтобы использовать имя пользователя и пароль, указанные в существующем источнике данных JNDI (Java Naming and Directory Interface) на сервере:
 - a. Очистите поле **Указать одноразовое разрешение защиты псевдонима Службы идентификации Java (JAAS)**.
 - b. Нажмите кнопку **Дополнительно**.
 - c. Разверните **Расширенная конфигурация соединения**.
 - d. В поле **Имя JNDI источника данных** введите имя существующего источника данных JNDI. Более подробная информация о находится в разделе “Имя JNDI источника данных (DataSourceJNDIName)” на стр. 203.
- Для того чтобы указать, что имя пользователя базы данных и пароль должны храниться в свойствах адаптера:
 - a. Очистите поле **Указать одноразовое разрешение защиты псевдонима Службы идентификации Java (JAAS)**.
 - b. Нажмите кнопку **Дополнительно**.

- с. В разделе **Информация о системном соединении с базой данных** введите **Имя пользователя** и **Пароль**. Более подробная информация о находитесь в разделах “Имя пользователя (UserName)” на стр. 216 и “Пароль (Password)” на стр. 211.

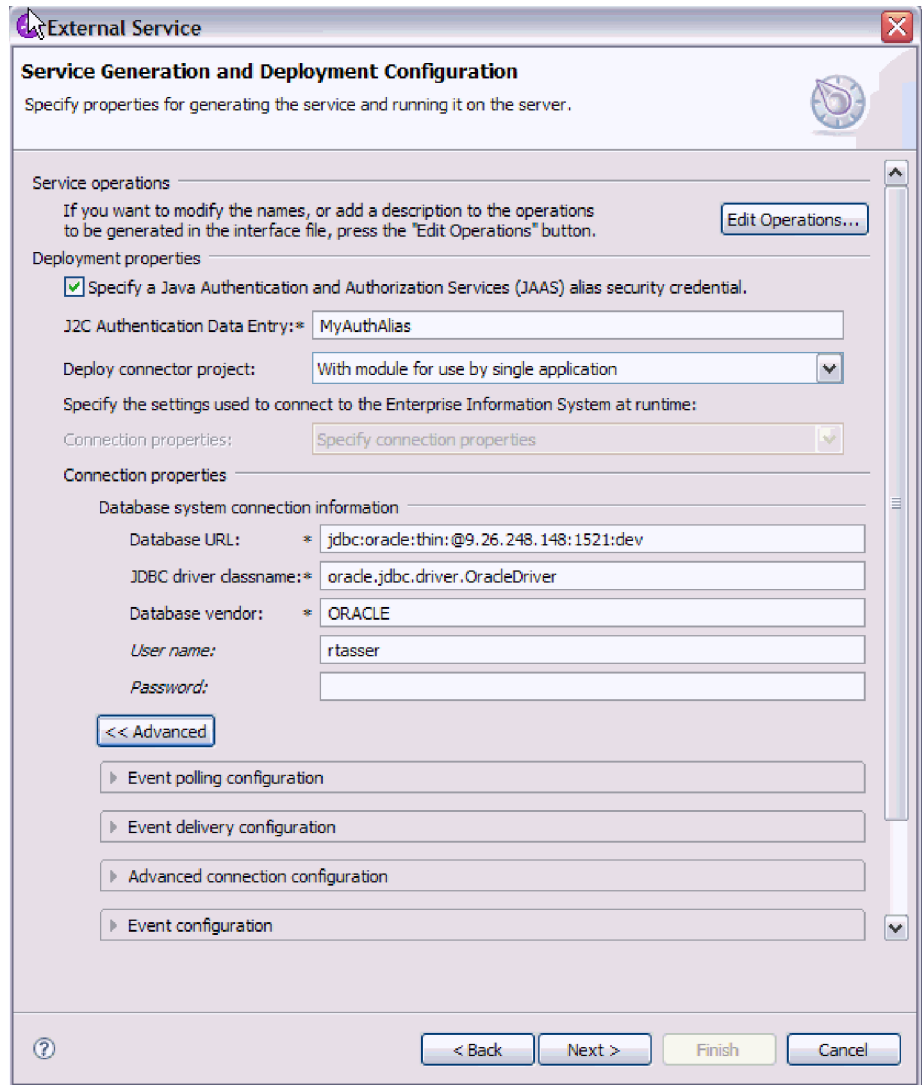
Примечание: Если пароль указан здесь, он сохраняется в текстовом виде в свойстве адаптера, видимом для неавторизованных пользователей.

3. В поле **Развернуть проект коннектора** укажите, включать ли файлы адаптера в модуль. Выберите одно из следующих значений:
- **С модулем для использования одним приложением.** Модуль, содержащий файлы адаптера, можно развернуть на любом сервере приложений. Если необходимо, чтобы адаптер использовался единственным модулем или несколькими модулями, которые будут запускать различные версии адаптера, воспользуйтесь встроенным адаптером. Это позволит обновлять адаптер в одном модуле без риска дестабилизировать сменой версии адаптера другие модули.
 - **На сервере для использования несколькими приложениями.** Если вы не включаете файлы адаптера в модуль, то вы должны установить их как автономный адаптер на каждом сервере приложений, на котором вы хотите запустить модуль. Автономный адаптер следует использовать в том случае, если несколько модулей могут применять одну версию адаптера и вы хотите управлять адаптером в центральном расположении. Применение автономного адаптера позволяет также сократить количество требуемых ресурсов.
4. Если на предыдущем шаге было указано свойство На сервере – для нескольких адаптеров, то укажите способ задания свойств соединения.
- В случае создания фабрики управляемых соединений или спецификации активации на сервере вручную или развертывания приложения, подключающегося в той же базе данных с помощью той же фабрики управляемых соединений или спецификации активации, допустимо повторное использование фабрики управляемых соединений или спецификации активации. Для этого следует указать имя соответствующего источника данных JNDI:
 - а. В поле **Свойства соединения** выберите Использовать предопределенные свойства соединения.
 - б. В поле **Имя поиска JNDI** введите имя JNDI источника данных для существующей фабрики управляемых соединений или спецификации активации.

На следующем рисунке показаны типичные параметры для повторного использования фабрики управляемых соединений или спецификации активации в случае автономного развертывания адаптера.



- с. Нажмите кнопку **Далее** для завершения задачи.
- Если приложение подключается к базе данных от имени конкретного пользователя впервые или если идентификационными данными пользователя требуется управлять отдельно от других приложений, то выберите **Указать свойства соединения**.
5. Просмотрите и, если необходимо, измените значения требуемых свойств соединения. Поля инициализируются с информацией о подключении, указанной при запуске мастера. Можно изменить значения, чтобы указать другое имя пользователя и пароль во время выполнения. Можно также подключиться к другому Oracle DB, хотя имена схемы должны быть одинаковыми обеих баз данных. Формат свойств соединения специфичен для базы данных. Более подробная информация о о свойствах находится в разделе “Свойства спецификации активации” на стр. 198.
 6. По выбору, укажите дополнительные свойства, щелкнув на **Дополнительные**. Разверните каждый из дополнительных разделов для просмотра свойств. На следующем рисунке показаны разделы дополнительных свойств в окне Конфигурация создания и развертывания служб.



- **Конфигурация опроса событий**

- В поле **Интервал между периодами опроса** введите количество миллисекунд, которое адаптер ждет между периодами опроса. Более подробная информация о находится в разделе “Интервал между периодами опроса (PollPeriod)” на стр. 212.
- В поле **Максимальное число событий в периоде опроса** введите количество событий для доставки в каждом периоде опроса. Более подробная информация о находится в разделе “Максимальное число событий в периоде опроса (PollQuantity)” на стр. 212.
- В поле **Интервал повторения при неудачном соединении** введите количество миллисекунд ожидания перед повторной попыткой подключения после неудачного подключения во время опроса. Более подробная информация о находится в разделе “Интервал повтора в случае сбоя соединения (RetryInterval)” на стр. 213.
- В поле **Количество попыток системного соединения** введите количество попыток подключения перед генерацией ошибки опроса. Более подробная информация о находится в разделе “Число попыток восстановить соединение с системой (RetryLimit)” на стр. 213.

- e. Если вы хотите, чтобы адаптер прекращал работу при возникновении ошибок опроса, выберите **Остановить адаптер при возникновении ошибки во время опроса**. Если эта опция не выбрана, адаптер заносит исключительную ситуацию в протокол, но продолжает работу. Более подробная информация о находится в разделе “Остановить адаптер при возникновении ошибки во время опроса (StopPollingOnError)” на стр. 215.
 - f. Необязательно можно выбрать **Восстанавливать соединение EIS при запуске**, если вы хотите, чтобы адаптер повторял попытку неудачного соединения при запуске. Более подробная информация о находится в разделе “Восстанавливать соединение EIS при запуске (RetryConnectionOnStartup)” на стр. 214
- **Конфигурация доставки событий**
 - a. В поле **Тип доставки** выберите метод доставки. Методы описаны в разделе “Тип доставки (DeliveryType)” на стр. 206.
 - b. Если вы хотите обеспечить доставку событий только один раз и только в один объект экспорта, выберите **Обеспечить однократную доставку**. Эта опция может снизить производительность, но не приводит к повторяющимся или пропущенным доставкам событий. Более подробная информация о находится в разделе “Гарантированная доставка событий (AssuredOnceDelivery)” на стр. 206.
 - c. По умолчанию адаптер обрабатывает все найденные им события при опросе. Если вы не хотите, чтобы он обрабатывал события с системным временем, которое позже текущего времени, выберите **Не обрабатывать события, которые имеют будущее системное время**. Более подробная информация о находится в разделе “Не обрабатывать события с системным временем в будущем (FilterFutureEvents)” на стр. 206.
 - d. В поле **Типы событий для обработки** введите список через запятую бизнес-объектов, для которых вы хотите обеспечить доставку. Оставьте это поле пустым, чтобы получать события всех типов бизнес-объектов.
Например, если вы хотите получать события только при изменении в базе данных таблиц Клиент и Заказ, но не других таблиц, установите для этого поля значение Клиент, Заказ.
Более подробная информация о находится в разделе “Типы обрабатываемых событий (EventTypeFilter)” на стр. 208.
 - e. В поле **Экземпляр адаптера для фильтрации событий** введите ИД коннектора, для которого вы хотите обеспечить доставку событий. Более подробная информация о находится в разделе “Экземпляр адаптера для фильтрации событий (AdapterInstanceEventFilter)” на стр. 199.
 - f. В поле **Ограничение повторов для неудачных событий** укажите количество попыток доставки события после неудачной доставки. Более подробная информация о находится в разделе “Количество повторов для недоставленных событий (FailedEventRetryLimit)” на стр. 209.
 - g. В разделе **Количество соединений для доставки событий** укажите минимальное и максимальное число соединений, используемых для доставки событий. Более подробная информация о находится в разделах “Минимальное число соединений (MinimumConnections)” на стр. 210 и “Максимальное число соединений (MaximumConnections)” на стр. 210.
 - **Дополнительная конфигурация соединения**
 - a. Свойство **Имя JNDI источника данных** уже обсуждалось ранее в этом разделе.
 - b. Установите **Дополнительные свойства соединения с драйвером JDBC**. Более подробная информация находится в разделе об этом свойстве находится в

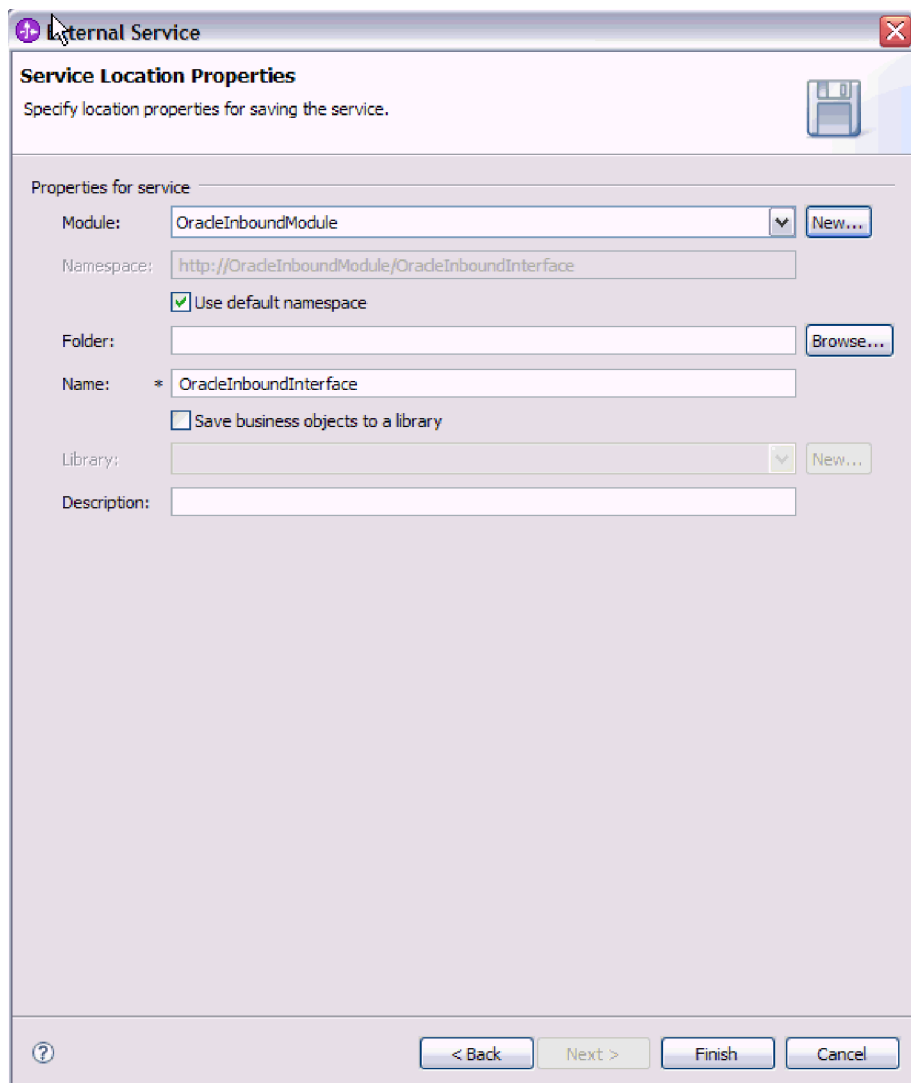
разделе “Дополнительные свойства драйвера JDBC (DriverConnectionProperties)” на стр. 200.

- c. Установите **Запрос SQL для проверки соединения**. Более подробная информация находится в разделе об этом свойстве находится в разделе “Запрос SQL для проверки соединения (PingQuery)” на стр. 211.
 - d. В поле **Тайм-аут запроса** введите интервал времени в секундах, в течение которого адаптер должен ждать ответа на запрос к базе данных. Более подробная информация находится в разделе об этом свойстве находится в разделе “Тайм-аут запроса (QueryTimeOut)” на стр. 212.
 - e. Установите **Возвращать бизнес-объект, даже когда набор результатов хранимой процедура пуст**. Более подробная информация находится в разделе об этом свойстве находится в разделе “Возвращать бизнес-объект при получении пустого набора результатов хранимой процедуры (ReturnDummyBOForSP)” на стр. 214.
- **Конфигурация события**
 - a. В поле **Упорядочить события по** укажите порядок, в котором события извлекаются и обрабатываются. Это список через запятую имен столбцов из таблицы событий вместе с ключевыми словами, которые задают порядок сортировки для каждого столбца. Слово *asc* задает возрастающий порядок, а *desc* - убывающий порядок. Более подробная информация о находится в разделе “Упорядочить события по (EventOrderBy)” на стр. 207.
 - b. В поле **Имя таблицы событий** примите имя по умолчанию для таблицы, которая содержит хранилище событий, или введите другое имя таблицы. Более подробная информация о находится в разделе “Имя таблицы событий (EventTableName)” на стр. 208.
 - c. В поле **Хранимая процедура, выполняемая перед опросом** укажите имя хранимой процедуры или хранимой функции, которая должна выполняться перед фактическим вызовом опроса. Более подробная информация о находится в разделе “Хранимая процедура, выполняемая перед опросом (SPBeforePoll)” на стр. 216.
 - d. В поле **Хранимая процедура, выполняемая после опроса** укажите имя хранимой процедуры или хранимой функции, которая должна выполняться после каждого цикла опроса. Более подробная информация о находится в разделе “Хранимая процедура, выполняемая после опроса (SPAAfterPoll)” на стр. 215.
 - e. В поле **Тип запроса события для обработки событий** выберите тип обработки событий, который вы хотите использовать:
 - Для использования стандартной обработки событий, предоставленной адаптером, выберите **Стандартная**.
 - Для того чтобы предоставить собственные запросы для настройки обработки событий, выберите **Пользовательская (динамическая)**. Если выбрана эта опция, заполните дополнительные поля, описанные в следующей таблице:

Поле	Значение	Дополнительная информация
Пользовательский запрос удаления	Имя запроса, хранимой процедуры или хранимой функции, выполняемой после обработки каждого события с целью удаления ненужных записей.	“Пользовательский запрос удаления (CustomDeleteQuery)” на стр. 201

Поле	Значение	Дополнительная информация
Пользовательский запрос событий	Имя запроса, хранимой процедуры или хранимой функции, выполняющей опрос событий.	“Пользовательский запрос событий (CustomEventQuery)” на стр. 202
Пользовательский запрос обновления	Имя запроса, хранимой процедуры или хранимой функции, выполняемой после обработки каждого события с целью предотвращения повторной загрузки события в следующем цикле обработки событий.	“Пользовательский запрос обновления (CustomUpdateQuery)” на стр. 202

- **Ведение протоколов и трассировка**
 - a. Если существует несколько экземпляров адаптера, установите значение ИД адаптера , которое уникально для этого экземпляра.
 - b. Если вы хотите скрыть потенциально секретную информацию о пользователе в файлах протокола и трассировки, выберите поле **Скрыть пользовательские данные в виде “XXX” файлах протокола и трассировки.**
7. Нажмите кнопку **Далее**. Будет показано окно Свойства расположения службы.



8. В окне Свойства расположения службы укажите имя модуля, который вы хотите создать. Это может быть новый или существующий модуль.
 - Если предпочитаемое имя модуля показано в списке **Модуль**, выберите его.

Важное замечание: Если модуль содержит интерфейс или бизнес-объект, имя которого совпадает с настраиваемым интерфейсом или объектом, то исходный интерфейс или бизнес-объект будет заменен новой версией.

- В противном случае создайте новый модуль:
 - a. Нажмите кнопку **Создать**.
 - b. В окне Проект интеграции выберите **Создать проект модуля** и нажмите **Далее**.
 - c. В окне Модуль введите имя модуля. Например, OracleInboundModule.
 - d. Если вы хотите, чтобы файлы описания службы (.export и .wsdl) были расположены в папке по умолчанию для модуля, оставьте выбранным поле **Использовать расположение по умолчанию**. Если вы хотите указать другую папку в модуле, очистите опцию, а затем нажмите **Обзор**, чтобы найти другую папку для поля **Расположение**.
 - e. Если вы хотите, чтобы модуль автоматически открывался в диаграмме сборки в WebSphere Integration Developer при закрытии мастера, выберите **Открыть диаграмму сборки модуля**. В противном случае очистите эту опцию.
 - f. Нажмите кнопку **Готово**, чтобы создать новый модуль.
9. Укажите пространство имен, которое вы хотите использовать для бизнес-объектов.
 - Если вы хотите, чтобы бизнес-объекты в модуле использовали пространство имен по умолчанию, оставьте выбранной опцию **Использовать пространство имен по умолчанию**.
 - Для того чтобы указать другое пространство имен, очистите эту опцию и введите другое значение в поле **Пространство имен**.
 10. По выбору, укажите папку в новом модуле, в которой будет храниться описание службы. В поле **Папка** введите имя папки или найдите существующую. Если имя папки не указано, то артефакты (файлы экспорта, XSD и WSDL) сохраняются в корневой папке модуля, то есть папке с именем модуля.
 11. В поле **Имя** примите имя импорта по умолчанию или введите другое имя.
 12. По выбору, если вы хотите сохранить бизнес-объекты в библиотеке, где они могут быть использованы другими модулями, выберите **Сохранить бизнес-объекты в библиотеке** и укажите расположение библиотеки в поле **Библиотека**.
 13. По выбору, в поле **Описание** введите описание модуля.
 14. После завершения установки свойств нажмите **Готово**.
 15. Если показано окно Модель изменена, нажмите **Да**.

Результаты

Мастер закрыт. Модуль создан в проекте и артефакты сгенерированы.

Дальнейшие действия

В некоторых случаях может потребоваться использовать редактор сборки для заполнения конфигурации. Затем можно протестировать и развернуть модуль.

Завершение настройки

В некоторых ситуациях, необходимо выполнить действия вручную для завершения настройки бизнес-объектов.

Описание задачи

Выполните эту задачу, когда необходимо настроить артефакты, сгенерированные мастером. Это может потребоваться в следующих ситуациях:

- Для установки параметра CopyAttribute для столбца, чтобы его значение совпадало с другим столбцом.
- Для добавления или удаления атрибутов из бизнес-объекта. Например, можно упростить бизнес-объект, удалив простые атрибуты, соответствующие столбцам базы данных, на которые не нужно ссылаться.
- Для настройки дополнительных родительских объектов бизнес-объекта таблицы, который имеет несколько родительских объектов. Мастер настраивает только один родительский объект для бизнес-объекта таблицы.

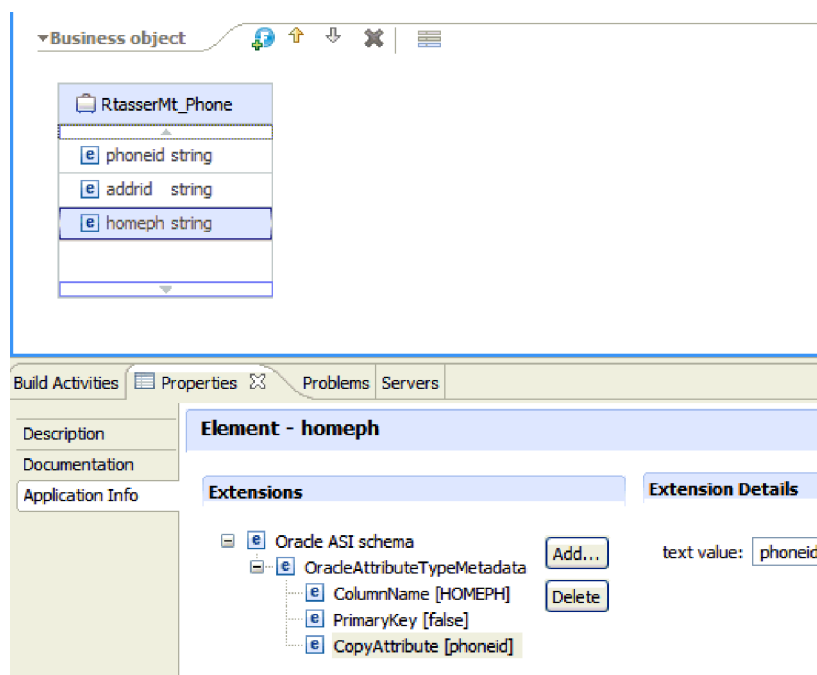
Этот раздел предоставляет подробные инструкции для установки параметра CopyAttribute бизнес-объекта таблицы. Другие изменения структур бизнес-объектов могут быть выполнены с помощью подобных приемов.

Параметр CopyAttribute содержится в свойствах атрибута для столбца, который вы хотите заполнить значениями и информацией для приложения из другого столбца. Например, если вы хотите, чтобы столбец контактная_информация новой строки в таблице содержал такое же значение, что и столбец электронная_почта, установите параметр CopyAttribute атрибута контактная_информация в электронная_почта. Для установки значения используется редактор сборки в WebSphere Integration Developer.

Процедура

1. В проекции Интеграция бизнес-процессов в WebSphere Integration Developer разверните модуль, разверните **Тип данных**, а затем найдите бизнес-объект таблицы. Именем бизнес-объекта является имя схемы базы данных плюс имя таблицы базы данных. Необязательное пространство имен может быть включено в начало имени.
2. Щелкните правой кнопкой мыши на имени бизнес-объекта и выберите **Открыть**. Редактор сборки показывает бизнес-объект, который имеет поле для каждого столбца.
3. В редакторе сборки выберите столбец, который должен соответствовать другому столбцу.
4. На панели Свойства выберите Информация приложения. Если панель Свойства не видна, щелкните правой кнопкой мыши на имени столбца и выберите **Показать в свойствах**.
5. Разверните узел **Схема ASI Oracle**, а затем разверните **OracleAttributeTypeMetadata**.
6. Щелкните правой кнопкой мыши на **OracleAttributeTypeMetadata** и выберите **Создать** → **oracleasi:CopyAttribute**.
7. Выберите свойство **CopyAttribute**.
8. В области Сведения о расширении установите текстовое значение равным имени столбца, который содержит информацию для копирования. Столбец может принадлежать текущему бизнес-объекту или его родительскому бизнес-объекту. Для копирования из столбца текущего бизнес-объекта установите значение равным имени столбца, например, phoneid. Для копирования из столбца родительского бизнес-объекта, добавьте к имени столбца префикс из двух точек

(..), например, ..phone. Следующий рисунок показывает редактор сборки со свойством CopyAttribute, установленным как столбец текущей таблицы.



Результаты

Бизнес-объект настроен для использования свойства CopyAttribute для установки атрибута бизнес-объекта и свойств столбца базы данных на основании информации из другого столбца.

Дальнейшие действия

Теперь можно протестировать и развернуть модуль.

Глава 5. Изменение свойств спецификации взаимодействия с помощью редактора сборки

Для того чтобы изменить свойства спецификации взаимодействия для модуля адаптера после генерации службы, используйте редактор сборки в WebSphere Integration Developer.

Перед тем, как начать

Для генерации службы адаптера необходимо использовать ESW (мастер внешних служб).

Описание задачи

После генерации службы адаптера вы можете изменить свойства спецификации взаимодействия. Свойства спецификации взаимодействия, которые необязательный, установлены на уровне метода для определенной операции определенного бизнес-объекта. Указанные значения появятся как значения по умолчанию во всех родительских бизнес-объектах, сгенерированных ESW (мастер внешних служб). Можно изменить свойства перед экспортом файла EAR. Эти свойства невозможно изменить после развертывания приложения.

Для того чтобы изменить свойства спецификации взаимодействия, выполните следующую процедуру.

Процедура

1. В проекции Интеграция бизнеса WebSphere Integration Developer раскройте имя модуля.
2. Разверните **Диаграмму сборки** и дважды щелкните мышью на интерфейсе.
3. Щелкните на интерфейсе в редакторе сборки. (Он показывает свойства модуля, если вы не щелкнули еще раз.)
4. Перейдите на вкладку **Свойства**. (Можно также щелкнуть правой кнопкой мыши на интерфейсе на диаграмме и выбрать **Показать в свойствах**.)
5. В разделе **Связывание** щелкните на **Связывания метода**. Показаны методы для интерфейса, по одному для каждой комбинации бизнес-объекта и операции.
6. Выберите метод, для которого нужно изменить свойство спецификации взаимодействия.
7. Измените свойство на вкладке **Общие**. Повторите этот шаг для каждого метода, свойство спецификации взаимодействия которого нужно изменить.

Результаты

Свойства спецификации взаимодействия, связанные с модулем адаптера, изменены.

Дальнейшие действия

Разверните модуль.

Глава 6. Развертывание модуля

Разверните модуль, чтобы поместить файлы, составляющие модуль и адаптер, в рабочую среду для работы или тестирования. В WebSphere Integration Developer интегрированные функции среды тестирования поддерживают WebSphere Process Server или WebSphere Enterprise Service Bus, в зависимости от профайлов среды выполнения, выбранных в процессе установки.

Среды развертывания

Существуют тестовая и рабочая среды, в которые можно развернуть модули и адаптеры.

В WebSphere Integration Developer, можно развернуть модули на одном или нескольких серверах в среде тестирования. Это является обычной практикой для выполнения и тестирования модулей интеграции бизнеса. Кроме того, модули можно экспортировать для развертывания на WebSphere Process Server или WebSphere Enterprise Service Bus в качестве файлов EAR с помощью AdminConsole или инструментов командной строки.

Развертывание модуля для тестирования

В WebSphere Integration Developer можно развернуть модуль, который включает в себя встроенный адаптер, в среде тестирования и работать с инструментами сервера, позволяющими выполнять такие задачи, как изменение конфигураций сервера, запуск и остановка серверов и тестирование исходного кода модуля на ошибки. Тестирование обычно выполняется на опциях интерфейса компонентов, что позволяет определить, правильно ли реализованы компоненты и связаны ссылки.

Описание задачи

Добавление внешних зависимостей

Зависимые файлы JAR должны быть добавлены в каталог библиотек или в файл EAR.

Описание задачи

Файлы JAR указываются в пути к классам. Зависимые библиотеки должны быть доступны на этапе развертывания модуля. Зависимые библиотеки можно сделать доступными двумя способами, один из которых доступен как для автономного, так и для встроенного развертывания, а другой - только для встроенного развертывания.

Добавление зависимостей внешнего программного обеспечения на сервер

Адаптеру необходимы некоторые файлы, установленные на сервере WebSphere Process Server или WebSphere Enterprise Service Bus, чтобы иметь возможность подключиться к Oracle DB.

Перед тем, как начать

Эту задачу выполнять не обязательно, если Oracle DB установлен в той же системе, что и WebSphere Process Server или WebSphere Enterprise Service Bus. Файлы уже доступны адаптеру.

Описание задачи

Адаптеру необходимы файлы драйвера JDBC Oracle, чтобы иметь возможность подключиться к нему.

Процедура

1. Необходимые файлы драйвера JDBC для программ базы данных и операционной системы можно получить у администратора базы данных или на Web-сайте, посвященном программным продуктам для баз данных.

В следующей таблице приведены файлы драйвера JDBC Oracle, необходимые для базы данных Oracle.

Таблица 12. Файлы драйвера JDBC для базы данных Oracle

База данных	Драйвер	Файлы драйвера JDBC	Внутренние системные библиотеки
Oracle	Простой драйвер	ojdbc14.jar	Нет

2. Скопируйте файлы на сервер.
 - В среде тестирования WebSphere Integration Developer скопируйте файлы в каталог `${установочный-каталог-WAS}/runtimes/bi_v62/lib/ext`.
 - В рабочей среде скопируйте файлы в каталог `${установочный-каталог-WAS}/lib/ext` сервера WebSphere Process Server или WebSphere Enterprise Service Bus.

Добавление внешних зависимостей для адаптера из комплекта

Необходимо скопировать зависимые файлы JAR в приложение EAR перед запуском приложений адаптера. Данный способ применим только в случае встроенного развертывания.

Описание задачи

Для получения необходимых файлов и их копирования в приложение EAR выполните следующие действия:

Процедура

1. Откройте рабочую область модуля и скопируйте файлы JAR в каталог. Например, для модуля `ModuleName` откройте рабочую область и скопируйте файлы JAR в каталог `ModuleNameApp/EarContent`.
2. Измените файл манифеста RAR адаптера (`manifest.mf`), содержащий список файлов JAR, необходимых для работы адаптера. Добавьте файлы JAR в следующем формате: `Class-Path: dependantjar1.jar, dependantjar2.jar`
3. Скопируйте исходные библиотеки в каталог `bin` и разверните приложение.

Результаты

Внешние библиотеки добавлены в среду выполнения.

Генерация и связывание целевого компонента для тестирования входящей обработки

Перед развертыванием в среде тестирования модуля, который включает в себя адаптер для входящей обработки, необходимо сгенерировать и связать целевой компонент. Этот целевой компонент служит *назначением*, которому адаптер отправляет события.

Перед тем, как начать

Необходимо сгенерировать модуль экспорта с помощью ESW (мастер внешних служб).

Описание задачи

Генерация и связывание целевого компонента для входящей обработки требуется только в среде тестирования. В этом нет необходимости при развертывании адаптера в рабочей среде.

Целевой компонент получает события. Вы *связываете* экспорт с целевым компонентом (соединяя два компонента) с помощью редактора сборки в WebSphere Integration Developer. Адаптер использует проводник для передачи данных события (из экспорта в целевой компонент).

Процедура

1. Создайте целевой компонент
 - a. В проекции Интеграция бизнес-процессов WebSphere Integration Developer разверните **Диаграмму сборки** и дважды щелкните мышью на компоненте экспорта. Если вы не изменили значение по умолчанию, именем компонента экспорта является имя адаптера + **InboundInterface**.
Интерфейс задает операции, которые могут быть вызваны, и передаваемые данные, такие как входные аргументы, значения возврата и исключения. **InboundInterface** содержит операции, требуемые адаптером для поддержки входящей обработки, и созданные при выполнении ESW (мастер внешних служб).
 - b. Создайте новый компонент, развернув **Компоненты**, выбрав **Компонент без типа**, и перетащив компонент на Диаграмму сборки.
Курсор изменится на значок расположения.
 - c. Щелкните на компоненте, чтобы он показывался на Диаграмме сборки.
2. Свяжите компоненты с помощью проводника.
 - a. Щелкните и перенесите компонент экспорта на новый компонент. Это нарисует проводник из компонента экспорта к новому компоненту, как показано на следующем рисунке:
 - b. Сохраните Диаграмму сборки. Выберите **Файл** → **Сохранить**.
3. Сгенерируйте реализацию для нового компонента.
 - a. Щелкните правой кнопкой мыши на новом компоненте и выберите **Сгенерировать реализацию**.

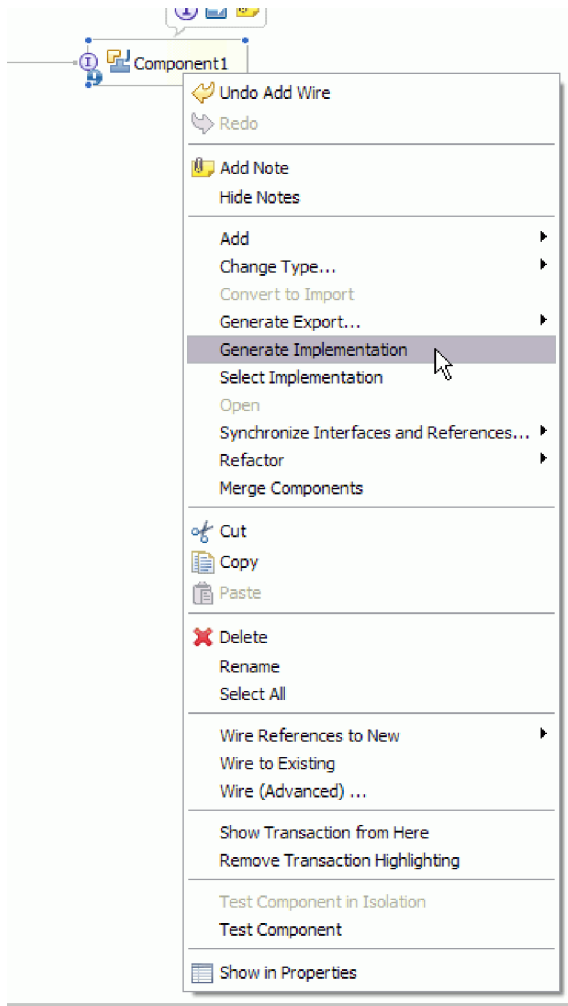


Рисунок 18. Генерация реализации Java

- b. Выберите (**пакет по умолчанию**) и нажмите **ОК**. Это создает конечную точку для входящего модуля.
Реализация Java показывается на отдельной вкладке.
- c. **Необязательно:** Добавьте операторы для печати объектов данных, полученных в конечной точке для каждого из ее методов.
- d. Сохраните файл, выбрав **Файл** → **Сохранить**.

Дальнейшие действия

Продолжите развертывание модуля для тестирования.

Подготовка к тестированию исходящих операций

Перед тестированием исходящей обработки модуля с помощью клиента тестирования WebSphere Integration Developer может потребоваться изменить некоторые из бизнес-объектов.

Описание задачи

Это шаг выполняется в клиенте тестирования WebSphere Integration Developer. Если он еще не открыт, откройте его из проекции Интеграция бизнес-процессов, щелкнув правой кнопкой мыши на имени проекта и выбрав **Тестирование** → **Модуль тестирования**.

- **Бизнес-объекты запросов**

Если бизнес-объект запроса был создан без оператора WHERE (например, он был определен с помощью такого оператора SELECT, как `Select * from Customer`), отмените установку атрибута `jdbcwhereclause` бизнес-объекта запроса перед тестированием в клиенте тестирования.

- **Бизнес-объекты таблицы, представления, синонима или псевдонима**

Перед тестированием операции RetrieveAll необходимо отменить установку всех атрибутов, значение которых не устанавливается в процессе тестирования.

- **Бизнес-объекты запросов**

Перед тестированием операции RetrieveAll необходимо отменить установку всех атрибутов, значение которых не устанавливается в процессе тестирования.

Добавление модуля на сервер

В WebSphere Integration Developer, можно добавить модули к одному или нескольким серверам в среде тестирования.

Перед тем, как начать

Если тестируемый модуль использует адаптер для выполнения входящей обработки, необходимо сгенерировать и связать *целевой компонент*, которому адаптер будет отправлять события.

Описание задачи

Для того чтобы протестировать модуль и его использование адаптера, необходимо добавить модуль к серверу.

Процедура

1. *Условие*: Если на **панели Серверы** не существует серверов, добавьте и определите новый сервер, выполнив следующие действия:
 - a. Поместите курсор на **панель Серверы**, щелкните правой кнопкой мыши и выберите **Создать** → **сервер**
 - b. В окне Определить новый сервер выберите тип сервера.
 - c. Настройте параметры сервера.
 - d. Нажмите кнопку **Готово**, чтобы опубликовать сервер.
2. Добавьте модуль к серверу.
 - a. Перейдите на панель Серверы. В окне WebSphere Integration Developer выберите **Окно** → **Показать панель** → **Сервер**
 - a. Запустите сервер. На вкладке Серверы на нижней правой панели экрана WebSphere Integration Developer щелкните правой кнопкой мыши на сервере и выберите **Запустить**.
3. Когда состояние сервера будет *Запущен*, щелкните правой кнопкой мыши на сервере и выберите **Добавить или удалить проекты**.
4. В окне Добавить или удалить проекты выберите проект и нажмите **Добавить**. Проект переместится из списка **Доступные проекты** в список **Настроенные проекты**.
5. Нажмите кнопку **Готово**. Это развернет модуль на сервере.

Вкладка Консоль на нижней правой панели показывает протокол в процессе добавления модуля к серверу.

Дальнейшие действия

Протестируйте функциональность модуля и адаптера.

Тестирование модуля для исходящей обработки с помощью клиента тестирования

Протестируйте собранный модуль и адаптер для исходящей обработки в клиенте тестирования WebSphere Integration Developer.

Перед тем, как начать

Сначала необходимо добавить модуль на сервер.

Описание задачи

Тестирование модуля обычно выполняется на опциях интерфейса компонентов, что позволяет определить, правильно ли реализованы компоненты и связаны ссылки.

Процедура

1. Выберите модуль для тестирования, щелкните правой кнопкой мыши на нем и выберите **Тестирование** → **Тестировать модуль**.
2. Информация о тестировании модуля с помощью клиента тестирования находится в разделе *Тестирование модулей и компонентов* в WebSphere Integration Developer Information Center.

Дальнейшие действия

Для того чтобы вы удовлетворены результатами тестирования модуля и адаптера, вы можете развернуть их в рабочей среде.

Развертывание модуля в рабочей среде

Развертывание модуля, созданного с помощью ESW (мастер внешних служб), на сервере WebSphere Process Server или WebSphere Enterprise Service Bus в рабочей среде выполняется в два этапа. Сначала модуль экспортируется на сервер WebSphere Integration Developer в качестве файла EAR. Затем файл EAR развертывается с помощью административной консоли WebSphere Process Server или WebSphere Enterprise Service Bus.

Описание задачи

Добавление зависимостей внешнего программного обеспечения на сервер

Адаптеру необходимы некоторые файлы, установленные на сервере WebSphere Process Server или WebSphere Enterprise Service Bus, чтобы иметь возможность подключиться к Oracle DB.

Перед тем, как начать

Эту задачу выполнять не обязательно, если Oracle DB установлен в той же системе, что и WebSphere Process Server или WebSphere Enterprise Service Bus. Файлы уже доступны адаптеру.

Описание задачи

Адаптеру необходимы файлы драйвера JDBC Oracle, чтобы иметь возможность подключиться к нему.

Процедура

1. Необходимые файлы драйвера JDBC для программ базы данных и операционной системы можно получить у администратора базы данных или на Web-сайте, посвященном программным продуктам для баз данных.

В следующей таблице приведены файлы драйвера JDBC Oracle, необходимые для базы данных Oracle.

Таблица 13. Файлы драйвера JDBC для базы данных Oracle

База данных	Драйвер	Файлы драйвера JDBC	Внутренние системные библиотеки
Oracle	Простой драйвер	ojdbc14.jar	Нет

2. Скопируйте файлы на сервер.
 - В среде тестирования WebSphere Integration Developer скопируйте файлы в каталог `${установочный-каталог-WAS}/runtimes/bi_v62/lib/ext`.
 - В рабочей среде скопируйте файлы в каталог `${установочный-каталог-WAS}/lib/ext` сервера WebSphere Process Server или WebSphere Enterprise Service Bus.

Установка файла RAR (только для модулей, использующих автономные адаптеры)

Если вы не хотите встраивать адаптер в модуль, а вместо этого хотите сделать его доступным всем развернутым приложениям в экземпляре сервера, вам потребуется установить адаптер в форме файла RAR на сервер приложений. Файл RAR - это файл архива Java (JAR), который используется для упаковки адаптера ресурса для архитектуры Java 2 Connector (J2C).

Перед тем, как начать

Необходимо установить для опция **Проект коннектора развертывания** значение **На сервере для использования несколькими адаптерами** в окне Конфигурация создания и развертывания служб программы ESW (мастер внешних служб).

Описание задачи

Установка адаптера в форме файла RAR открывает доступ к адаптеру всем компонентам приложения J2EE, выполняющимся в среде выполнения сервера.

Процедура

1. Запустите AdminConsole.
2. Выберите **Ресурсы** → **Адаптеры ресурсов** → **Адаптеры ресурсов**.
3. На странице Адаптеры ресурсов выберите **Установить RAR**.

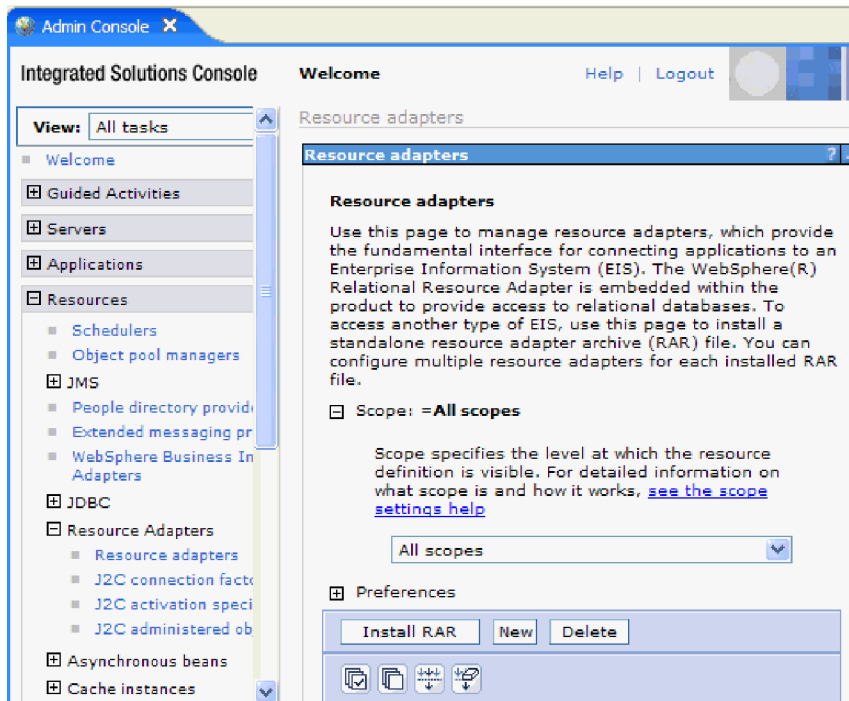


Рисунок 19. Кнопка Установить RAR на странице Адаптеры ресурсов

4. На странице Установить файл RAR нажмите кнопку **Обзор** и перейдите к файлу RAR для данного адаптера.

Файлы RAR обычно установлены в следующем расположении:

каталог_установки_WID/ResourceAdapters/имя_адаптера/deploy/адаптер.rar

5. Нажмите кнопку **Далее**.
6. На странице Ресурсы адаптера при необходимости измените имя адаптера и добавьте описание.
7. Нажмите **ОК**.
8. Нажмите **Сохранить** на врезке **Сообщения** в верхней части страницы.

Дальнейшие действия

Следующий шаг - это экспорт модуля как файла EAR, который можно развернуть на сервере.

Экспорт модуля в виде файла EAR

Экпортируйте модуль как файл EAR с помощью WebSphere Integration Developer. Создав файл EAR, вы можете сохранить все содержимое модуля и впоследствии развернуть его на сервере WebSphere Process Server или WebSphere Enterprise Service Bus.

Перед тем, как начать

Перед экспортом модуля в файл EAR необходимо создать модуль, взаимодействующий со службой. Этот модуль должен быть показан в проекции Бизнес-интеграция WebSphere Integration Developer.

Описание задачи

Для экспорта модуля в качестве файла EAR выполните следующие действия:

Процедура

1. Щелкните правой кнопкой на модуле и выберите **Экспортировать**.
2. В окне Выбор разверните **Java EE**.
3. Выберите **файл EAR** и нажмите **Далее**.
4. Необязательно: Выберите правильное приложение EAR. Имя приложения EAR составляется из имени модуля, в которому добавляется строка “App”.
5. Нажмите кнопку **Обзор** и выберите локальную папку для сохранения файла EAR.
6. По выбору, если вы хотите экспортировать исходные файлы, выберите **Экспорт исходных файлов**. Эта опция предоставляется для экспорта исходных файлов в дополнение к файлу EAR. Исходные файлы включают в себя файлы, связанные с компонентами Java, описаниями данных и т.д.
7. Для замены существующего файла выберите **Заменить существующий файл**.
8. Нажмите кнопку **Готово**.

Результаты

Содержимое модуля экспортировано в качестве файла EAR.

Установите модуль в AdminConsole. В результате модуль будет развернут на сервере WebSphere Process Server или WebSphere Enterprise Service Bus.

Установка файла EAR

Установка файла EAR - это последний этап процесса развертывания. При установке файла EAR и его запуске на сервере адаптер, включенный в модуль проекта, запускается как часть установленного приложения.

Перед тем, как начать

Перед установкой на сервере WebSphere Process Server или WebSphere Enterprise Service Bus модуль необходимо экспортировать в качестве файла EAR.

Описание задачи

Для установки файла EAR выполните следующие действия. Дополнительная информация о приложениях адаптеров в кластере приведена в справочной системе <http://www.ibm.com/software/webservers/appserv/was/library/>.

Процедура

1. Откройте WebSphere Process Server или WebSphere Enterprise Service Bus AdminConsole. Для этого щелкните на экземпляре сервера правой кнопкой мыши и выберите **Запустить AdminConsole**.
2. В окне AdminConsole выберите **Приложения** → **Установить новые приложения**.

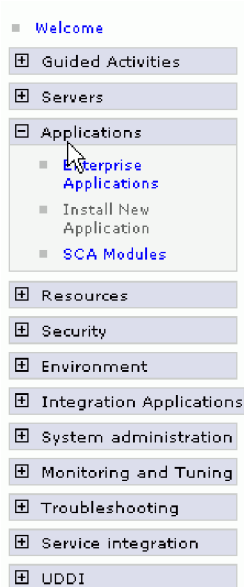


Рисунок 20. Окно Подготовка к установке приложения

3. Нажмите кнопку **Обзор**, выберите файл EAR и нажмите **Далее**. Имя файла EAR представляет собой имя модуля, к которому добавлена строка "App."
4. Необязательно: В случае развертывания в среде кластера выполните следующие действия.
 - a. В окне **Шаг 2: Связывание модулей с серверами** выберите модуль.
 - b. Выберите имя кластера.
 - c. Нажмите кнопку **Применить**.
5. Нажмите кнопку **Далее**. Откроется окно Обзор. Проверьте правильность параметров и нажмите кнопку **Готово**.
6. Необязательно: В случае применения псевдонима идентификации выполните следующие действия:
 - a. Разверните раздел **Защита** и выберите **Псевдонимы идентификации бизнес-интеграции**.
 - b. Выберите псевдоним, который требуется настроить. Для внесения изменений в конфигурации псевдонимов идентификации требуются права доступа администратора или оператора.
 - c. Необязательно: При необходимости укажите значение в поле **Имя пользователя**.
 - d. При необходимости укажите значение в поле **Пароль**.
 - e. При необходимости укажите значение в поле **Подтверждение пароля**.
 - f. Нажмите **ОК**.

Результаты

Проект развертывается. Открывается окно Приложения J2EE.

Дальнейшие действия

Настроить свойства и подготовить приложения к работе в кластере можно в административной консоли перед настройкой инструментов устранения неполадок.

Глава 7. Администрирование модуля адаптера

В случае автономного развертывания адаптера для запуска, остановки, отслеживания и устранения неполадок модуля адаптера применяется административная консоль. Модуль адаптера, встроенного в приложение, запускается и останавливается вместе с приложением.

Изменение свойств конфигурации для встроенных адаптеров

Для изменения свойств конфигурации после развертывания адаптера в качестве части модуля применяется административная консоль. Можно изменить свойства адаптера ресурса (применяются в ходе работы адаптера), фабрики управляемых соединений (применяются для обработки исходящих запросов), а также свойства спецификации активации (применяются для обработки входящих событий).

Настройка свойств адаптера ресурсов для встроенных адаптеров

Настройка свойств адаптера ресурсов для адаптера выполняется после развертывания модуля с помощью консоли администрирования. Выберите название свойства, которое необходимо настроить, и измените или укажите его значение.

Перед тем, как начать

Разверните модуль адаптера на сервере WebSphere Process Server или WebSphere Enterprise Service Bus.

Описание задачи

Другие свойства представляют собой стандартные параметры конфигурации, общие для всех адаптеров WebSphere.

Для того чтобы настроить свойства в административной консоли, выполните следующие действия:

Процедура

1. Запустите административную консоль.
2. В разделе **Приложения** выберите **Приложения организации**.
3. В списке **Приложения организации** щелкните на названии модуля адаптера, свойства которого необходимо изменить. Будет показана страница Конфигурация.

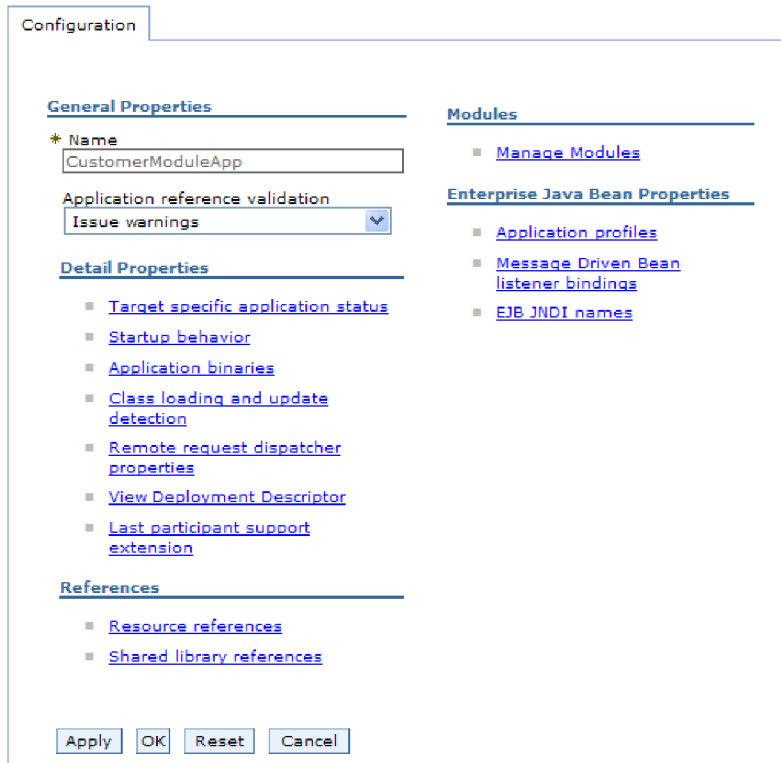


Рисунок 21. Выбор опции Управление модулями на вкладке Конфигурация

4. В разделе **Модули** щелкните на записи **Управление модулями**.
5. Щелкните на **IBM WebSphere Adapter for Oracle E-Business Suite**.
6. В списке **Дополнительные свойства** щелкните на **Адаптер ресурса**.
7. На следующей странице в списке **Дополнительные свойства** выберите **Пользовательские свойства**.
8. Для каждого свойства, которое требуется изменить, выполните следующие действия.

Примечание: Дополнительная информация об этих свойствах приведена в разделе “Свойства адаптера ресурсов” на стр. 172.

- a. Щелкните на имени свойства. Будет показана страница **Конфигурация** для выбранного свойства. Например, если выбрать свойство **logNumberOfFiles**, то будет показана следующая страница:

The image shows a configuration window titled 'Configuration' with a sub-section 'General Properties'. It contains several input fields: 'Scope' with the value 'widNode', a 'Required' checkbox which is unchecked, 'Name' with 'logNumberOfFiles', 'Value' with '1', an empty 'Description' text area, and a 'Type' dropdown menu set to 'java.lang.String'. At the bottom, there are four buttons: 'Apply', 'OK', 'Reset', and 'Cancel'.

Рисунок 22. Вкладка Конфигурация для свойства `logNumberOfFiles`

- б. Измените или укажите значение в поле **Значение**.
Можно изменить число в поле **Значение** и добавить описание свойства.
 - с. Нажмите **ОК**.
9. Щелкните на ссылке **Сохранить** в поле **Сообщения** в верхней части окна.

Результаты

Свойства адаптера ресурса, связанные с модулем адаптера, изменены.

Настройка свойств фабрики управляемых соединений (J2C) для встроенных адаптеров

Настройка свойств фабрики управляемых соединений для адаптера выполняется после развертывания модуля с помощью консоли администрирования. Выберите название свойства, которое необходимо настроить, и измените или укажите его значение.

Перед тем, как начать

Разверните модуль адаптера на сервере WebSphere Process Server или WebSphere Enterprise Service Bus.

Описание задачи

Свойства фабрики управляемых соединений необходимы для настройки экземпляра целевого сервера Oracle DB.

Примечание: Название свойств в административной консоли: "свойства фабрики управляемых соединений J2C".

Для того чтобы настроить свойства в административной консоли, выполните следующие действия:

Процедура

1. Запустите административную консоль.
2. В разделе **Приложения** выберите **Приложения организации**.
3. В списке **Приложения J2EE** щелкните на названии модуля адаптера, свойства которого необходимо изменить.
4. В разделе **Модули** щелкните на записи **Управление модулями**.

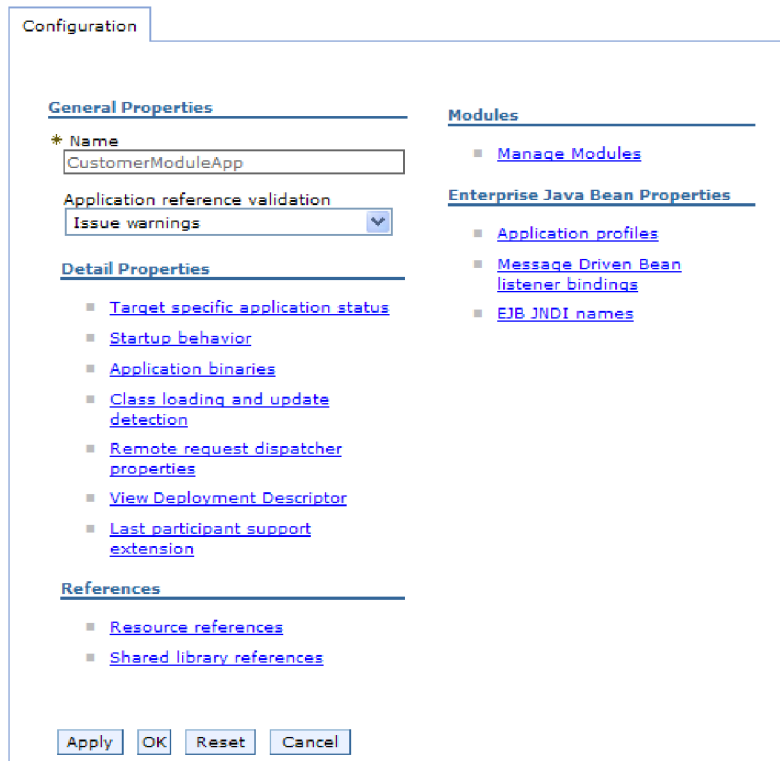


Рисунок 23. Выбор опции Управление модулями на вкладке Конфигурация

5. Щелкните на **IBM WebSphere Adapter for Oracle E-Business Suite**.
6. В списке **Дополнительные свойства** выберите **Адаптер ресурса**.
7. На следующей странице щелкните в списке **Дополнительные свойства** выберите **Фабрики соединений J2C**.
8. Щелкните на имени фабрики соединений, связанной с модулем адаптера.
9. В списке **Дополнительные свойства** выберите **Пользовательские свойства**.
Другие свойства - это свойства фабрики соединений J2C, уникальные для адаптера Adapter for Oracle E-Business Suite. Свойства пула соединений и дополнительные свойства настраиваются в ходе разработки собственного адаптера.
10. Для каждого свойства, которое требуется изменить, выполните следующие действия.

Примечание: Дополнительная информация об этих свойствах приведена в разделе “Свойства фабрики управляемых соединений” на стр. 176.

- a. Щелкните на имени свойства.

- b. Измените или укажите значение в поле **Значение**.
- c. Нажмите **ОК**.

11. Щелкните на ссылке **Сохранить** в поле **Сообщения** в верхней части окна.

Результаты

Свойства фабрики управляемых соединений, связанной с модулем адаптера, изменены.

Настройка свойств спецификации активации для встроенных адаптеров

Настройка свойств спецификации активации для адаптера выполняется после развертывания модуля с помощью консоли администрирования. Выберите название свойства конечной точки сообщения, которое необходимо настроить, и измените или укажите его значение.

Перед тем, как начать

Разверните модуль адаптера на сервере WebSphere Process Server или WebSphere Enterprise Service Bus.

Описание задачи

Свойства спецификации активации служат для настройки конечной точки для обработки входящих запросов.

Для того чтобы настроить свойства в административной консоли, выполните следующие действия:

Процедура

1. Запустите административную консоль.
2. В разделе **Приложения** выберите **Приложения организации**.
3. В списке **Приложения организации** щелкните на названии модуля адаптера, свойства которого необходимо изменить.
4. В разделе **Модули** щелкните на записи **Управление модулями**.

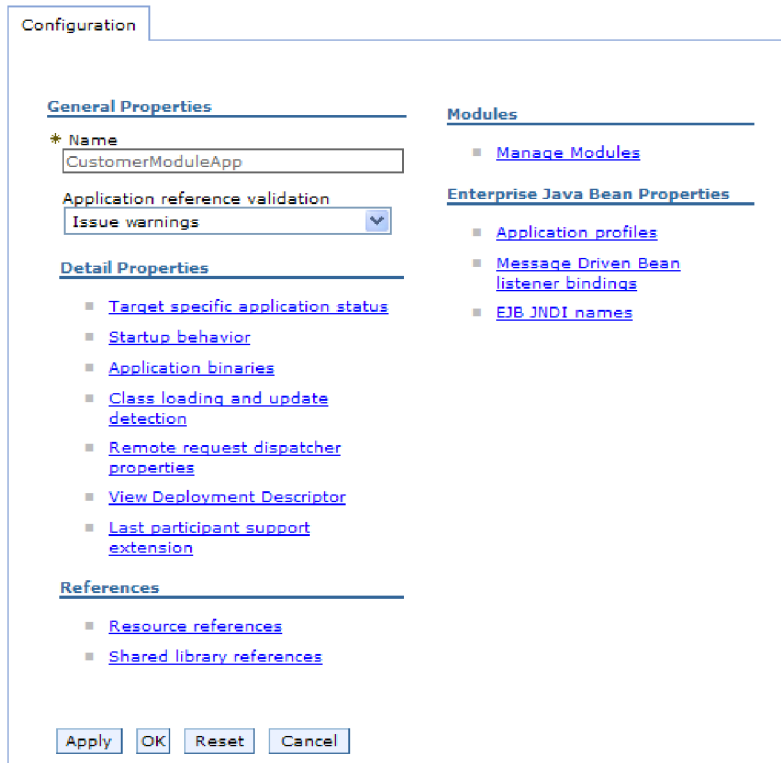


Рисунок 24. Выбор опции Управление модулями на вкладке Конфигурация

5. Щелкните на **IBM WebSphere Adapter for Oracle E-Business Suite**.
6. В списке **Дополнительные свойства** щелкните на **Адаптер ресурса**.
7. На следующей странице щелкните в списке **Дополнительные свойства** на **Спецификация активации J2C**.
8. Щелкните на имени спецификации активации, связанной с модулем адаптера.
9. В списке **Дополнительные свойства** щелкните на **Пользовательские свойства спецификации активации J2C**.
10. Для каждого свойства, которое требуется изменить, выполните следующие действия.

Примечание: Дополнительная информация об этих свойствах приведена в разделе “Свойства спецификации активации” на стр. 198.

- a. Щелкните на имени свойства.
 - b. Измените или укажите значение в поле **Значение**.
 - c. Нажмите **ОК**.
11. Щелкните на ссылке **Сохранить** в поле **Сообщения** в верхней части окна.

Результаты

Свойства спецификации активации, связанные с модулем адаптера, изменены.

Изменение свойств конфигурации для автономных адаптеров

Для задания свойств конфигурации после установки автономного адаптера применяется административная консоль. Вы предоставляете общую информацию об адаптере, а затем устанавливаете свойства адаптера ресурса (которые используются для операции общего адаптера). Если адаптер будет использоваться для исходящих операций, создается фабрика соединений, а затем для нее устанавливаются свойства. Если адаптер будет использоваться для входящих операций, создается спецификация активации, а затем для нее устанавливаются свойства.

Настройка свойств адаптера ресурсов для автономных адаптеров

Настройка свойств адаптера ресурса для автономного адаптера после его установки в WebSphere Process Server или WebSphere Enterprise Service Bus выполняется с помощью административной консоли. Выберите название свойства, которое необходимо настроить, и измените или укажите его значение.

Перед тем, как начать

Установите адаптер на сервере WebSphere Process Server или WebSphere Enterprise Service Bus.

Описание задачи

Другие свойства представляют собой стандартные параметры конфигурации, общие для всех адаптеров WebSphere.

Для того чтобы настроить свойства в административной консоли, выполните следующие действия:

Процедура

1. Запустите административную консоль.
2. Выберите **Ресурсы** → **Адаптеры ресурсов** → **Адаптеры ресурсов**.
3. На странице Адаптеры ресурсов выберите **IBM WebSphere Adapter for Oracle E-Business Suite**.
4. В списке **Дополнительные свойства** выберите **Пользовательские свойства**.
5. Для каждого свойства, которое требуется изменить, выполните следующие действия.

Примечание: Дополнительная информация об этих свойствах приведена в разделе “Свойства адаптера ресурсов” на стр. 172.

- a. Щелкните на имени свойства.
- b. Измените или укажите значение в поле **Значение**.

Например, если вы выберете **logNumberOfFiles**, вы увидите следующую страницу:

The image shows a configuration dialog box with the following fields and controls:

- Configuration** (tab)
- General Properties** (section header)
- * Scope**: widNode
- Required**
- Name**: logNumberOfFiles
- Value**: 1
- Description**: (empty text area)
- Type**: java.lang.String
- Buttons: **Apply**, **OK**, **Reset**, **Cancel**

Рисунок 25. Вкладка Конфигурация для свойства logNumberOfFiles

Можно изменить число в поле **Значение** и добавить описание свойства.

с. Нажмите **ОК**.

6. Нажмите **Сохранить** на врезке **Сообщения** в верхней части страницы.

Результаты

Свойства адаптера ресурса, связанные с адаптером, изменены.

Настройка свойств фабрики управляемых соединений (J2C) для автономных адаптеров

Настройка свойств фабрики управляемых соединений для автономного адаптера после его установки в WebSphere Process Server или WebSphere Enterprise Service Bus выполняется с помощью административной консоли. Выберите название свойства, которое необходимо настроить, и измените или укажите его значение.

Перед тем, как начать

Установите адаптер на сервере WebSphere Process Server или WebSphere Enterprise Service Bus.

Описание задачи

Свойства фабрики управляемых соединений необходимы для настройки экземпляра целевого сервера Oracle DB.

Примечание: Название свойств в административной консоли: "свойства фабрики управляемых соединений J2C".

Для того чтобы настроить свойства в административной консоли, выполните следующие действия:

Процедура

1. Запустите административную консоль.
2. Выберите **Ресурсы** → **Адаптеры ресурсов** → **Адаптеры ресурсов**.
3. На странице Адаптеры ресурсов выберите **IBM WebSphere Adapter for Oracle E-Business Suite**.
4. В списке **Дополнительные свойства** выберите **Фабрики соединений J2C**.
5. Если вы собираетесь использовать существующую фабрику соединений, то перейдите к шагу выбора фабрики соединений из списка.

Примечание: Если вы выбрали **Использовать предопределенные свойства соединения** при использовании ESW (мастер внешних служб) для настройки модуля адаптера, не требуется создавать фабрику соединений.

При создании фабрики соединений выполните следующие действия:

- a. Нажмите кнопку **Создать**.
- b. В разделе **Общие свойства** вкладки **Конфигурация** введите имя фабрики соединений. Например, можно указать AdapterCF.
- c. Введите значение для **Имя JNDI**. Например, можно указать com/eis/AdapterCF.
- d. Выберите псевдоним аутентификации из списка **Псевдоним аутентификации, управляемый компонентом**.
- e. Нажмите **ОК**.
- f. Нажмите **Сохранить** на врезке **Сообщения** в верхней части страницы.
Созданная фабрика соединений будет показана в списке.

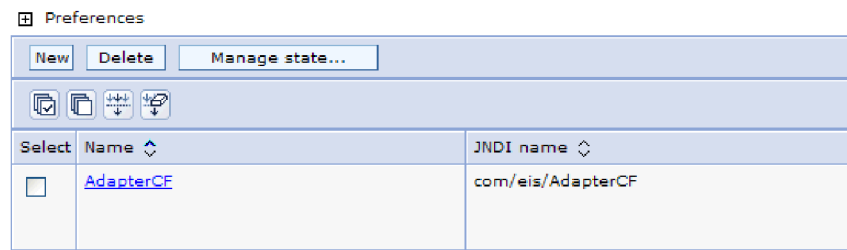


Рисунок 26. Пользовательские фабрики соединений для адаптера ресурса

6. Выберите одну из списка фабрик соединений.
7. В списке **Дополнительные свойства** выберите **Пользовательские свойства**.
Другие свойства - это свойства фабрики соединений J2C, уникальные для адаптера Adapter for Oracle E-Business Suite. Свойства пула соединений и дополнительные свойства настраиваются в ходе разработки собственного адаптера.
8. Для каждого свойства, которое требуется изменить, выполните следующие действия.

Примечание: Дополнительная информация об этих свойствах приведена в разделе “Свойства фабрики управляемых соединений” на стр. 176.

- a. Щелкните на имени свойства.
- b. Измените или укажите значение в поле **Значение**.

- c. Нажмите **ОК**.
9. После завершения установки свойств нажмите **Применить**.
10. Нажмите **Сохранить** на врезке **Сообщения** в верхней части окна.

Результаты

Свойства фабрики управляемых соединений связаны с установленным адаптером.

Настройка свойств спецификации активации для автономных адаптеров

Настройка свойств спецификации активации для автономного адаптера после его установки в WebSphere Process Server или WebSphere Enterprise Service Bus выполняется с помощью административной консоли. Выберите название свойства конечной точки сообщения, которое необходимо настроить, и измените или укажите его значение.

Перед тем, как начать

Установите адаптер на сервере WebSphere Process Server или WebSphere Enterprise Service Bus.

Описание задачи

Свойства спецификации активации служат для настройки конечной точки для обработки входящих запросов.

Для того чтобы настроить свойства в административной консоли, выполните следующие действия:

Процедура

1. Запустите административную консоль.
2. Выберите **Ресурсы** → **Адаптеры ресурсов** → **Адаптеры ресурсов**.
3. На странице Адаптеры ресурсов выберите **IBM WebSphere Adapter for Oracle E-Business Suite**.
4. В списке **Дополнительные свойства** щелкните на **Спецификации активации J2C**.
5. Если вы собираетесь использовать существующую спецификацию активации, перейдите к шагу выбора спецификации активации из списка.

Примечание: Если вы выбрали **Использовать предопределенные свойства соединения** при использовании ESW (мастер внешних служб) для настройки модуля адаптера, не требуется создавать спецификацию активации.

При создании спецификации активации выполните следующие действия:

- a. Нажмите кнопку **Создать**.
- b. В разделе **Общие свойства** вкладки **Конфигурация** введите имя спецификации активации. Например, можно указать AdapterAS.
- c. Введите значение для **Имя JNDI**. Например, можно указать com/eis/AdapterAS.
- d. Выберите псевдоним аутентификации из списка **Псевдоним аутентификации**.
- e. Выберите тип получателя запросов сообщения.
- f. Нажмите **ОК**.
- g. Нажмите **Сохранить** на врезке **Сообщения** в верхней части страницы.

Показывается вновь созданная спецификация активации.

6. Выберите одну из списка спецификаций активации.
7. В списке **Дополнительные свойства** щелкните на **Пользовательские свойства спецификации активации J2C**.
8. Для каждого свойства, которое требуется установить, выполните следующие действия.

Примечание: Дополнительная информация об этих свойствах приведена в разделе “Свойства спецификации активации” на стр. 198.

- a. Щелкните на имени свойства.
 - b. Измените или укажите значение в поле **Значение**.
 - c. Нажмите **ОК**.
9. После завершения установки свойств нажмите **Применить**.
 10. Нажмите **Сохранить** на врезке **Сообщения** в верхней части страницы.

Результаты

Свойства спецификации активации связаны с установленным адаптером.

Запуск приложения, использующего адаптер

Для запуска приложения, которое использует адаптер, следует использовать административную консоль. По умолчанию приложение автоматически запускается вместе с сервером.

Описание задачи

Используйте эту процедуру для запуска приложения, которое использует встроенный или автономный адаптер. Для приложения, которое использует встроенный адаптер, он запускается вместе с приложением. Для приложения, которое использует автономный адаптер, он запускается вместе с сервером приложений.

Процедура

1. В административной консоли выберите **Приложения** → **Приложения J2EE**.

Примечание: Административная консоль помечена “Консоль интегрированных решений”.

2. Включите переключатель рядом с приложением, которое необходимо запустить. Имя приложения - это имя установленного файла EAR без расширения .EAR.
3. Нажмите кнопку **Запустить**.

Результаты

Состояние приложения изменится на **Запущено**, и в верхней части административной консоли появится сообщение о запуске приложения.

Остановка приложения, использующего адаптер

Для остановки приложения, которое использует адаптер, следует использовать административную консоль. По умолчанию приложение автоматически останавливается вместе с сервером.

Описание задачи

Используйте эту процедуру для остановки приложения, которое использует встроенный или автономный адаптер. Для приложения, которое использует встроенный адаптер, он останавливается вместе с приложением. Для приложения, которое использует автономный адаптер, он останавливается вместе с сервером приложений.

Процедура

1. В административной консоли выберите **Приложения** → **Приложения J2EE**.

Примечание: Административная консоль помечена “Консоль интегрированных решений”.

2. Включите переключатель рядом с приложением, которое необходимо остановить. Имя приложения - это имя установленного файла EAR без расширения .EAR.
3. Нажмите кнопку **Остановить**.

Результаты

Состояние приложения изменится на Остановлено, и в верхней части административной консоли появится сообщение об остановке приложения.

Отслеживание производительности с помощью системы сбора статистики

Система сбора статистики (PMI) - это функция административной консоли, обеспечивающая динамический мониторинг производительности компонентов рабочей среды, включая Adapter for Oracle E-Business Suite. PMI собирает статистику производительности, такую как среднее время ответа и общее число запросов, для различных компонентов сервера и сохраняет ее в виде иерархической структуры. Полученные данные можно просмотреть с помощью программы Tivoli Performance Viewer, которая представляет собой утилиту мониторинга с графическим интерфейсом, интегрированную в административную консоль WebSphere Process Server.

Описание задачи

Мониторинг производительности адаптера можно обеспечить путем сбора следующих данных:

- Обработка исходящих запросов.
- Извлечение входящих событий из таблицы событий.
- Доставка входящих событий в конечные точки.

Перед тем, как приступить к настройке PMI для мониторинга адаптера, необходимо задать уровень трассировки и запустить события для сбора статистики.

Дополнительную информацию об особенностях применения PMI в среде адаптеров можно найти на Web-сайте WebSphere Application Server: <http://www.ibm.com/software/webservers/appserv/was/library/>.

Настройка системы сбора статистики

Систему сбора статистики (PMI) можно настроить для сбора статистики производительности адаптера, такой как среднее время ответа и общее число запросов. Настроив PMI, вы сможете отслеживать производительность адаптера с помощью программы Tivoli Performance Viewer.

Перед тем, как начать

Перед тем, как приступить к настройке РМІ для мониторинга адаптера, необходимо задать уровень трассировки и запустить события для сбора статистики.

1. Для включения трассировки и получения данных о событиях необходимо указать один из следующих уровней трассировки: fine, finer, finest или all. После *=info добавьте двоеточие и строку, например:

```
*=info: WBILocationMonitor.CEI.ResourceAdapter.  
*=finest: WBILocationMonitor.LOG.ResourceAdapter.*=finest:
```

Инструкции по заданию уровня трассировки приведены в разделе “Включение трассировки в инфраструктуре событий общего формата (CEI)” на стр. 136.

2. Создайте по крайней мере один исходящий запрос или входящее событие для настройки собранной статистики.

Процедура

1. Включите РМІ для адаптера.
 - a. В административной консоли разверните **Мониторинг и точная настройка** и выберите **Система сбора статистики (РМІ)**.
 - b. В списке серверов щелкните на имени применяемого сервера.
 - c. Выберите вкладку Конфигурация, затем выберите переключатель **Включить систему сбора статистики (РМІ)**.
 - d. Выберите переключатель **Пользовательская статистика**, для того чтобы разрешить настройку собираемой статистики.

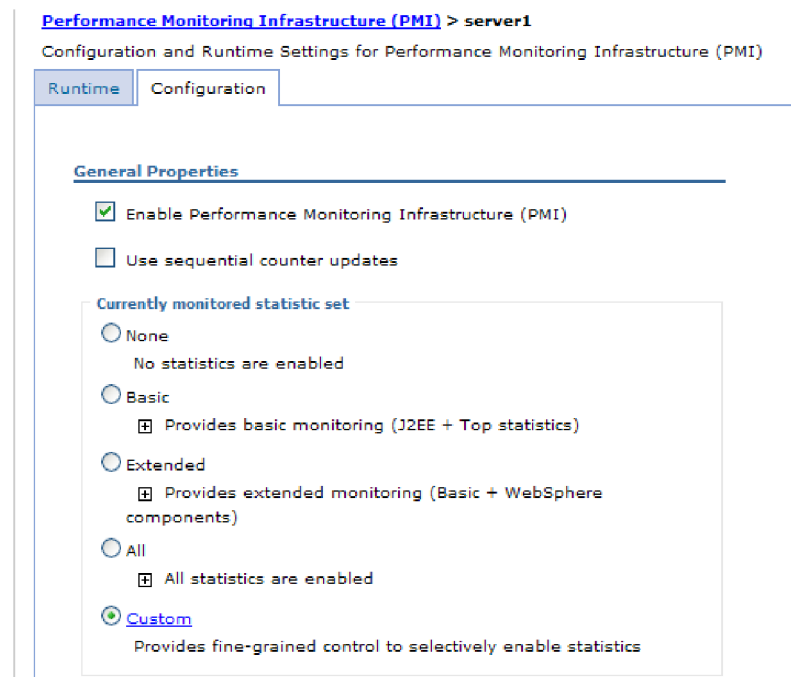


Рисунок 27. Включение системы сбора статистики

- e. Нажмите кнопку **Применить** или **ОК**.
 - f. Нажмите **Сохранить**. Система сбора статистики включена.
2. Настройте РМІ для адаптера.

- В административной консоли разверните **Мониторинг и точная настройка** и выберите **Система сбора статистики (PMI)**.
- В списке серверов щелкните на имени применяемого сервера.
- Выберите переключатель **Пользовательская статистика**.
- Перейдите на вкладку **Рабочая среда**. На следующем рисунке показана вкладка Рабочая среда.

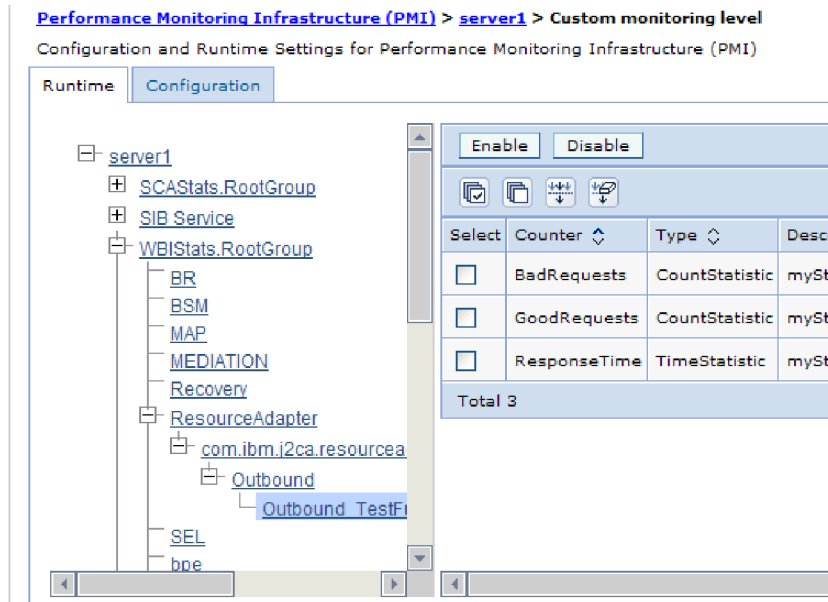


Рисунок 28. Вкладка Рабочая среда применяется для настройки PMI

- Выберите **WBISStats.RootGroup**. Это submodule PMI для данных, собранных в корневой группе. В данном примере применяется корневая группа WBISStats.
- Выберите **ResourceAdapter**. Это submodule для данных, собранных для адаптеров JCA.
- Щелкните на имени адаптера и выберите процессы для мониторинга.
- В правой панели укажите статистику для сбора, выбрав соответствующие переключатели, и нажмите кнопку **Включить**.

Результаты

Система PMI настроена для работы с адаптером.

Дальнейшие действия

Теперь можно посмотреть статистику производительности адаптера.

Просмотр статистики производительности

Статистику производительности адаптера можно посмотреть с помощью программы Tivoli Performance Viewer. Tivoli Performance Viewer - это утилита мониторинга с графическим интерфейсом, интегрированная в административную консоль WebSphere Process Server.

Перед тем, как начать

Настройте систему сбора статистики для работы с адаптером.

Процедура

1. В административной консоли разверните **Мониторинг и точная настройка**, затем разверните **Просмотр статистики производительности** и выберите **Текущие операции**.
2. Выберите сервер в списке.
3. В разделе сервера разверните **Модули производительности**.
4. Выберите **WBIStatsRootGroup**.
5. Выберите **ResourceAdapter** и имя модуля адаптера.
6. При наличии нескольких процессов выберите переключатели рядом с процессами, статистику которых требуется просмотреть.

Результаты

Статистика отображается в правой панели. Кнопки **Показать график** и **Показать таблицу** позволяют просмотреть данных в виде графика и таблицы соответственно. На следующем рисунке показан график статистики производительности адаптера.

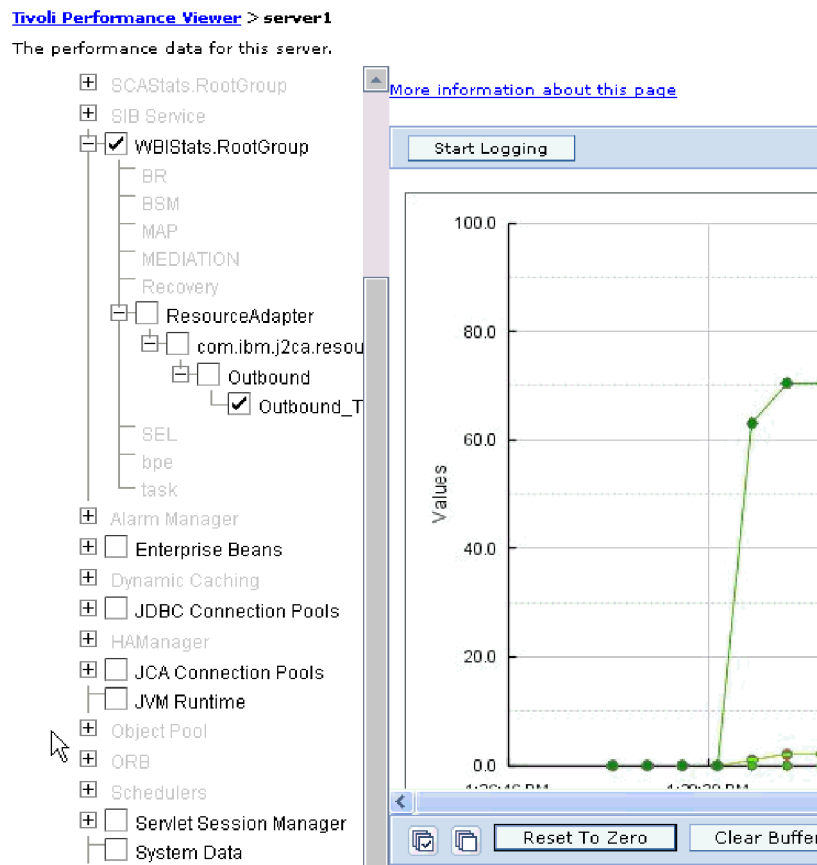


Рисунок 29. График статистики производительности адаптера

Включение трассировки в инфраструктуре событий общего формата (CEI)

Адаптер может использовать встроенный в сервер компонент Общая инфраструктуру событий для сообщения о важных бизнес-событиях, таких как запуск или остановка цикла опроса. Данные события могут быть записаны в базу данных или файл протокола трассировки в зависимости от параметров конфигурации.

Описание задачи

Процедура

1. На административной консоли выберите **Устранение неполадок**.
2. Выберите **Протоколы и трассировка**.
3. Выберите сервер в списке.
4. На врезке **Изменение уровня подробности протокола** щелкните на имени базы данных CEI (например, WBIEventMonitor.CEI.ResourceAdapter.*) или файле протокола трассировки (например, WBIEventMonitor.LOG.ResourceAdapter.*), в который адаптер должен записывать данные события.
5. Выберите уровень подробности данных о бизнес-событиях, которые адаптер должен записывать в базу данных или файл протокола трассировки, и (необязательно) укажите уровень дискретности, связанный с сообщениями и трассировками.
 - **Без протокола.** Выключает ведение протокола.
 - **Только сообщения.** Адаптер сообщает о событиях.
 - **Все сообщения и трассировки.** Адаптер сообщает сведения о событиях.
 - **Уровни сообщения и трассировки.** Параметры для управления степенью подробности сообщений адаптера о полезной нагрузке бизнес-объекта, связанной с событием. Если вы хотите настроить уровень подробности, выберите одно из следующего:
 - Точный.** Адаптер сообщает о событиях, но не о полезной нагрузке бизнес-объекта.
 - Более точный.** Адаптер сообщает о событиях и о полезной нагрузке бизнес-объекта.
 - Самый точный.** Адаптер сообщает о событиях и о полезной нагрузке бизнес-объекта.
6. Нажмите **ОК**.

Результаты

Ведение протоколов включено. Можно просмотреть записи CEI в файле протокола трассировки или с помощью программы просмотра Событий общего формата в административной консоли.

Глава 8. Устранение неполадок и поддержка

Распространенные методы устранения неполадок и информация по их самостоятельному устранению, помогающие быстро справиться со сложностями.

Поддержка Анализатор протоколов и данных трассировки

Адаптер создает файлы протоколов и трассировки, доступные для просмотра с помощью Анализатор протоколов и данных трассировки.

Анализатор протоколов и данных трассировки позволяет фильтровать файлы протоколов и трассировки с целью изоляции сообщений и данных трассировки, относящихся к адаптеру. Кроме того, в программе просмотра протоколов выделяются сообщения и данные трассировки адаптера.

ИД компонента адаптера, предназначенный для фильтрации и выделения, состоит из символов 0EBSRA и идентификатора адаптера. Например, если свойство ИД адаптера содержит значение 001, то ИД компонента равен 0EBSRA001.

В случае применения нескольких экземпляров адаптера первые семь символов ИД адаптера должны быть уникальными для каждого экземпляра, для того чтобы информацию из файлов протоколов и трассировки можно было сопоставить с конкретным экземпляром адаптера. Такой подход также обеспечивает уникальность ИД компонентов экземпляров адаптера. Для того чтобы показать, каким образом длина ИД адаптера влияет на фильтрацию файлов протоколов и трассировки, предположим, что для двух экземпляров WebSphere Adapter for Oracle E-Business Suite в свойстве ИД адаптера указаны значения 001 и 002. ИД компонентов этих экземпляров (0EBSRA001 и 0EBSRA002) достаточно коротки, чтобы оставаться уникальными, т.е. они позволяют отличать экземпляры друг от друга. Однако экземпляры с более длинными ИД адаптера нельзя отличить друг от друга. Например, предположим, что для двух экземпляров в свойстве ИД адаптера указаны значения Instance01 и Instance02. Пользователь не сможет отдельно проверить информацию из файлов протоколов и трассировки для каждого экземпляра адаптера, поскольку ИД компонента усекается до 0EBSRAInstanc.

В случае обработки исходящих запросов свойство ИД адаптера доступно как на уровне адаптера ресурса, так и на уровне фабрики управляемых соединений. В ходе изменения ИД адаптера после настройки адаптера для обработки исходящих запросов с помощью ESW (мастер внешних служб) следует обратить внимание на согласованность значений свойств в конфигурациях адаптера ресурса и фабрики управляемых соединений во избежание несогласованной маркировки записей протоколов и трассировки. Это замечание неприменимо в случае обработки входящих событий, поскольку свойство ИД адаптера доступно только на уровне адаптера ресурса.

Дополнительная информация об ИД адаптера приведена в разделе “ИД адаптера (AdapterID)” на стр. 173. Дополнительная информация о Анализатор протоколов и данных трассировки приведена в разделе <http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/org.eclipse.hyades.log.ui.doc.user/concepts/cltaviews.htm>.

Настройка протоколов и трассировки

Настройте необходимые свойства протоколов и трассировки в соответствии с текущими требованиями. Для контроля за состоянием обработки событий активируйте функцию ведения протокола адаптера. Измените имена файлов трассировки и протокола адаптера, чтобы отличить их от других файлов протокола и трассировки.

Описание задачи

Настройка свойств протокола

Для активации функции ведения протоколов и настройки свойств вывода протокола, включая расположение, уровень детализации и формат вывода протокола, используйте административную консоль.

Описание задачи

Для ведения протокола адаптера необходимо указать точки мониторинга событий компонента, уровень подробностей о событии и формат вывода для записи событий в протокол. С помощью административной консоли можно выполнить следующие задачи:

- Включать или выключать функцию ведения протокола определенного события
- Указывать уровень детализации протокола
- Выбирать место хранения файлов протокола и указывать их количество
- Задавать формат вывода протоколов

Если вы зададите формат вывода для Анализатора протокола, вы сможете открывать вывод трассировки с помощью Анализатора протокола, который представляет из себя приложение, входящее в состав Process Server. Это может пригодиться при установке отношения данных трассировки двух разных процессов сервера, так как в данном случае открывается доступ к функции слияния Анализатора протокола.

За дополнительной информацией об управлении Process Server, включая компоненты служб и точки событий, обратитесь к документации по Process Server.

Конфигурацию протокола можно изменить статически или динамически. Статические изменения конфигурации активируются при запуске или перезапуске сервера приложений. Динамические изменения конфигурации, то есть изменения, вносимые в процессе работы, вступают в силу немедленно.

При создании протокола на основе данных конфигурации устанавливается уровень его детализации. Если данные конфигурации недоступны для данного протокола, уровень его детализации наследуется от его родителя. При отсутствии данных конфигурации для родительского протокола проверяется его родитель и далее по структуре дерева, пока не будет обнаружено ненулевое значение. Изменения уровня протокола распространяются на его дочерние элементы, которые, в свою очередь, при необходимости распространяют их на свои дочерние элементы.

Для того чтобы активировать функцию ведения протоколов и задать свойства вывода протокола, выполните следующие шаги.

Процедура

1. В панели навигации административной консоли выберите **Серверы** → **Серверы приложений**.

2. Выберите имя сервера.
3. В разделе **Устранение неполадок** выберите **Протоколы и трассировка**.
4. Выберите **Изменить уровень детализации протокола**.
5. Укажите, когда изменения вступят в силу:
 - Для статического изменения конфигурации перейдите на вкладку **Конфигурация**.
 - Для динамического изменения конфигурации перейдите на вкладку **Среда выполнения**.
6. Выберите пакеты, уровень ведения протоколов которых требуется изменить. Имена пакетов WebSphere Adapters начинаются с **com.ibm.j2ca.***:
 - Для базового компонента адаптера выберите **com.ibm.j2ca.base.***.
 - Для базового компонента адаптера и всех развернутых адаптеров выберите **com.ibm.j2ca.***.
 - Для общего базового компонента WebSphere Adapter for JDBC и WebSphere Adapter for Oracle E-Business Suite выберите **com.ibm.j2ca.dbadapter.core.***.
 - Только для Adapter for Oracle E-Business Suite выберите пакет **com.ibm.j2ca.oracleebs.***.
7. Выберите уровень протокола.

Уровень ведения протоколов	Описание
Fatal	Дальнейшее выполнение задачи невозможно или произошел сбой в компоненте.
Серьезный	Дальнейшее выполнение задачи невозможно, но компонент функционирует. Этот уровень ведения протоколов включает в себя условия, указывающие на вероятность критической ошибки, например, ситуации нехватки ресурсов.
Warning	Возникла предполагаемая ошибка, либо может возникнуть серьезная ошибка. На этом уровне ведения протокола также регистрируются события, указывающие на усугубление ситуации, например возможную утечку ресурсов.
Audit	Произошло значительное событие, влияющее на состояние сервера или ресурсы.
Info	Задача выполняется. На этом уровне ведения протокола сохраняется общая информация о ходе выполнения задачи.
Config	Сообщается состояние конфигурации или факты ее изменения.
Detail	Подзадача выполняется. На этом уровне сохраняется общая информация о ходе выполнения подзадачи.

8. Нажмите кнопку **Применить**.
9. Нажмите **ОК**.
10. Для вступления изменений статической конфигурации в силу остановите и перезапустите Process Server.

Результаты

Начиная с этого момента, записи протокола для выбранных компонентов адаптера будут содержать указанный уровень информации.

Изменение имен файлов протокола и трассировки

Для того чтобы отделить данные протокола и трассировки адаптера от других процессов, с помощью административной консоли измените имена файлов. По умолчанию данные протокола для всех процессов и приложений на сервере Process Server записываются в файл SystemOut.log, а данные трассировки - в файл trace.log.

Перед тем, как начать

Файлы протокола и трассировки можно переименовать в любой момент после развертывания модуля адаптера на сервере приложений.

Описание задачи

Имена файлов протоколов и трассировки можно изменить статически или динамически. Статические изменения вступают в силу при запуске или перезапуске сервера приложений. Динамические изменения, то есть изменения, вносимые в процессе работы, применяются немедленно.

Файлы протокола и трассировки находятся в папке *каталог-установки/profiles/имя-профайла/logs/имя-сервера*.

Для того чтобы изменить имена файлов протокола и трассировки, выполните указанную ниже процедуру.

Процедура

1. На панели навигации административной консоли выберите **Приложения > Приложения J2EE**.
2. В списке приложений J2EE выберите имя приложение адаптера. Имеется в виду имя файла EAR адаптера без расширения .ear. Например, если файлу EAR присвоено имя Accounting_OutboundApp.ear, щелкните на **Accounting_OutboundApp**.
3. В списке Модули на вкладке Конфигурация выберите **Управление модулями**.
4. В списке модулей выберите IBM WebSphere Adapter for Oracle E-Business Suite.
5. На вкладке Конфигурация в разделе Дополнительные свойства выберите **Адаптер ресурса**.
6. На вкладке Конфигурация в разделе Дополнительные свойства выберите **Пользовательские свойства**.
7. В таблице Пользовательские свойства укажите новые имена файлов.
 - a. Выберите **logFilename** для изменения имени файла протокола или **traceFilename** для изменения имени файла трассировки.
 - b. На вкладке Конфигурация введите новое имя в поле **Значение**. Имя файла протокола по умолчанию - SystemOut.log; имя файла трассировки по умолчанию - trace.log.
 - c. Нажмите кнопку **Применить** или **ОК**. Изменения сохраняются в локальной системе.
 - d. Для сохранения изменений в главной конфигурации на сервере воспользуйтесь одним из следующих способов:
 - **Статическое изменение:** Остановите и перезапустите сервер. Внесенные изменения применяются после остановки и запуска сервера.
 - **Динамическое изменение:** Щелкните на ссылке **Сохранить**, расположенной в поле Сообщение над таблицей Пользовательские свойства. Снова нажмите кнопку **Сохранить**. Внесенные изменения применяются немедленно.

Поддержка FFDC

Адаптер поддерживает сбор данных независимого сбоя (FFDC), что обеспечивает постоянную запись неполадок и значительных ошибок программного обеспечения во время выполнения WebSphere Process Server или WebSphere Enterprise Service Bus.

Компонент FFDC выполняется в фоновом режиме и собирает события и ошибки, произошедшие во время выполнения. Компонент предоставляет средства для связи неполадок друг с другом, что позволяет программному обеспечению связать эффекты неполадок с их причинами и облегчить локализацию корневой причины неполадки. Собранные данные могут использоваться для определения обработки исключения, произошедшего во время выполнения адаптера.

При возникновении неполадки адаптер записывает сообщения об исключении и данные контекста в файл протокола, расположенный в каталоге *корневой-каталог-установки/profiles/profile/logs/ffdc*.

Более подробная информация о сборе данных независимого сбоя (FFDC) находится в документации по WebSphere Process Server или WebSphere Enterprise Service Bus.

Сбои бизнеса

Адаптер поддерживает бизнес-сбои, т.е. ожидаемые и объявляемые в описании исходящей службы или при импорте исключительные ситуации. Бизнес-сбои могут происходить в заранее известных точках бизнес-процесса в результате нарушения бизнес-правил или ограничений.

Хотя WebSphere Process Server или WebSphere Enterprise Service Bus поддерживают другие виды сбоев, адаптер генерирует только бизнес-сбои, которые в данной документации называются просто *сбои*. Не все исключения становятся сбоями. Сбои создаются только для тех ошибок, которые можно исправить без завершения работы приложения. Например, адаптер создает сбой при получении исходящего бизнес-объекта, в котором отсутствуют требуемые данные, или в случае обнаружения отдельных ошибок в ходе обработки исходящих запросов.

Бизнес-объекты сбоев

ESW (мастер внешних служб) создает бизнес-объект для каждого сбоя, который может сгенерировать адаптер. Кроме того, мастер создает бизнес-объект расширенного набора WBIFault, который имеет общую для всех сбоев информацию, такую как атрибуты сообщения, errorCode и primarySetKey, как показано в рис. 30.

WBIFault	
message	string
errorCode	string
primaryKeySet	PrimaryKeyPairType []

Рисунок 30. Структура бизнес-объекта WBIFault

Некоторые сбои содержат атрибут matchCount, предоставляющий дополнительную информацию об ошибке. Для других сбоев WBIFault содержит всю информацию, которая необходима для управления сбоем.

WebSphere Adapter for Oracle E-Business Suite включает сбои. Установка сбоев вручную не требуется. Адаптер предоставляет следующие бизнес-объекты сбоев, которые создаются мастером:

- **IntegrityConstraintFault**
При обработке операции Create, Update или Execute (отличной от хранимой процедуры) адаптер выбрасывает ошибку ограничения целостности данных, если Oracle DB выбрасывает исключение SQLException нарушения ограничения целостности данных. Например, если внешний ключ не найден, адаптер выбрасывает эту ошибку.
- **MatchesExceededLimitFault**
В ходе обработки операции RetrieveAll адаптер создает этот сбой, если число записей, возвращенных запросом базы данных, превышает ограничение, заданное в свойстве MaxRecords спецификации взаимодействия.
Для увеличения числа возвращаемых записей увеличьте значение свойства MaxRecords для операции RetrieveAll в спецификации взаимодействия.
Бизнес-объект, связанный с этим сбоем, содержит только одно свойство matchCount, которое представляет собой строку, содержащую число совпадений.
- **MissingDataFault**
Адаптер создает этот сбой, если бизнес-объект, переданный исходящей операции, содержит не все требуемые атрибуты.
- **MultipleMatchingRecordsFault**
Адаптер создает этот сбой в ходе обработки операции Retrieve или Update, если для указанных ключей запрос возвращает несколько записей. Бизнес-объект, связанный с этим сбоем, содержит только одно свойство matchCount, которое представляет собой строку, содержащую число совпадений.
- **ObjectNotFoundFault**
Этот сбой может возникнуть при выполнении операции Create или Update. При выполнении операции Create адаптер извлекает дочерний объект с одиночным количеством элементов, если принадлежностью является false для этого дочернего объекта. Если извлечение дочернего объекта не возвращает результатов, выбрасывается этот сбой.
- **RecordNotFoundFault**
Адаптер создает этот сбой в ходе извлечения данных, если в базе данных не найдены записи для указанных ключей. Этот сбой может возникать в операциях Delete, Update, Retrieve и RetrieveAll.
- **UniqueConstraintFault**
Адаптер создает этот сбой в ходе обработки операции Create или Update, если в Oracle DB возникнет исключительная ситуация SQLException, связанная с нарушением ограничения.

XAResourceNotAvailableException

Если в Process Server процессов содержится несколько сообщений об исключительной ситуации com.ibm.ws.Transaction.XAResourceNotAvailableException, для исправления этой неполадки удалите протоколы транзакций.

Признак:

При запуске адаптера в протоколе Process Server постоянно появляется запись о следующей исключительной ситуации:

```
com.ibm.ws.Transaction.XAResourceNotAvailableException
```


Неполадка:

Ресурс был удален, пока Process Server выполнял или отзывал транзакцию для этого ресурса. При запуске адаптер пытается восстановить транзакцию, но это ему не удается, потому что ресурс был удален.

Решение:

Для устранения этой неполадки выполните следующие действия:

1. Остановите Process Server.
2. Удалите протокол транзакции, в котором содержится запись о ней. Для определения транзакции воспользуйтесь данными трассировки исключительной ситуации. В результате сервер больше не будет предпринимать попытки по восстановлению этих транзакций.

Примечание: В среде тестирования или разработки в общем случае можно удалить все протоколы транзакций. В WebSphere Integration Developer удалите файлы и подкаталоги из каталога протокола транзакции *установочный-каталог-сервера\profiles\профайл\tranlog*.

В рабочей среде удалите только транзакции, представляющие события, которые уже не нужно обрабатывать. Это можно сделать, заново установив адаптер, связав его с первоначальной базой событий и удалив только те транзакции, которые больше не нужны. Кроме того, транзакции можно удалить из файла log1 или log2 в следующем каталоге:

установочный-каталог-сервера\profiles\профайл\tranlog\узел\wps\сервер\transaction\tranlog

3. Запустите Process Server.

Ресурсы для самостоятельного устранения неполадок

Ресурсы для самостоятельного устранения неполадок, предоставляемые службой поддержки программного обеспечения IBM, содержат техническую документацию и самую актуальную информацию по поддержке, а также дают возможность загрузить инструменты и исправления и предотвратить возникновение неполадок в WebSphere Adapters. Ресурсы для самостоятельного устранения неполадок помогут диагностировать неполадки в адаптере и обратиться в службу поддержки программного обеспечения IBM.

Web-сайт службы поддержки

На Web-сайте поддержки WebSphere Adapters (<http://www.ibm.com/software/integration/wbiadapters/support/>) приведены ссылки на ресурсы следующих типов, посвященные обучению, работе и устранению неполадок WebSphere Adapters:

- Предупреждения относительно продукта
- Техническая информация, в частности, справочная система Information Center по продуктам, руководства, справочники IBM Redbooks и информационные бюллетени.
- Учебные материалы
- Комментарии

Рекомендованные исправления

Список рекомендованных исправлений доступен в следующем расположении:
<http://www.ibm.com/support/docview.wss?fdoc=aimadp&rs=695&uid=swg27010397>

Комментарии

Технические комментарии содержат последнюю документацию по Adapter for Oracle E-Business Suite, в том числе следующие разделы:

- Неполадки и способы их устранения
- Ответы на часто задаваемые вопросы
- Инструкции по установке, настройке, работе и устранению неполадок адаптера
- *IBM Software Support Handbook*

Список технических комментариев по WebSphere Adapters приведен по следующему адресу:

<http://www.ibm.com/support/search.wss?tc=SSMKUK&rs=695&rank=8&dc=DB520+D800+D900+DA900+DA800+DB560&dtm>

Модуль для IBM Support Assistant

Вместе с Adapter for Oracle E-Business Suite поставляется модуль для программы IBM Support Assistant, которая представляет собой бесплатную локальную рабочую среду обслуживания программного обеспечения. Модуль поддерживает динамическую трассировку. Дополнительная информация по установке модуля IBM Support Assistant и работе с ним приведена по следующему адресу:

<http://www.ibm.com/software/support/isa/>

Устранение обычных неполадок

Описаны некоторые из неполадок, которые могут возникнуть в базе данных при выполнении WebSphere Adapter for Oracle E-Business Suite, вместе со способом их устранения и обходными путями. Эти неполадки и способы их устранения являются дополнительными к тем, которые документированы как технические заметки на Web-сайте поддержки программного обеспечения.

Полный список технических замечаний о WebSphere Adapters находится в <http://www.ibm.com/support/search.wss?tc=SSMKUK&rs=695&rank=8&dc=DB520+D800+D900+DA900+DA800+DB560&dtm>.

RecordNotFoundException для операции RetrieveAll в клиенте тестирования

Неполадка

При выполнении операции RetrieveAll в клиенте тестирования WebSphere Integration Developer генерируется исключительная ситуация RecordNotFoundException, когда из запроса ожидаются данные. Генерируется следующее сообщение:
RecordNotFoundException: Запись не найдена в EIS.

Причина

Это исключение может возникнуть, если в предложении WHERE для оператора SELECT не установлены все атрибуты бизнес-объекта. Оставить атрибут пустым, что является значением по умолчанию, это не то же самое, что явно отменить установку значения.

Исправление

В клиенте тестирования установите значения атрибутов, которые требуются для <unset>. Повторите операцию RetrieveAll. Если исключительная ситуация генерируется снова, вероятно, в базе данных не существует соответствующих записей.

Типы данных CLOB размером 4К или более не могут быть вставлены в базы данных Oracle 9i или 10g

Неполадка

Возникает следующая исключительная ситуация при вставке значений CLOB (большой символьный объект) размером 4К или более в базы данных Oracle 9i или 10g:

- Oracle 9i: ResourceAdapt E
com.ibm.j2ca.dbadapter.core.runtime.DBOperationHandler.
executePreparedCUDStatement CWYDB0301E: Не удалось выполнить операцию базы данных, возникла исключительная ситуация SQL по следующей причине: Больше нет данных для чтения из сокета.
- Oracle 10g: ResourceAdapt E
com.ibm.j2ca.dbadapter.core.runtime.DBOperationHandler.
executePreparedCUDStatement CWYDB0301E: Не удалось выполнить операцию базы данных, возникла исключительная ситуация SQL по следующей причине: ORA-01460: запрошено нереализованное или неподходящее преобразование.

Причина

Вы используете старый драйвер, который не поддерживает CLOB размером больше 4К.

Исправление

Используйте драйвер OracleThin из Oracle 10.1.0.2 или более позднего выпуска.

Некоторые из сгенерированных бизнес-объектов не имеют атрибутов для объектов базы данных Oracle

Неполадка

Некоторые бизнес-объекты, которые сгенерированы из объекта базы данных Oracle, не имеют атрибутов для столбцов таблицы.

Причина

При некоторых условиях драйвер JDBC Oracle не возвращает информацию о столбце для объекта базы данных. Следующие дефекты зарегистрированы в данный момент в Oracle для следующих неполадок:

- 2281705. DATABASEMETADA.GETCOLUMNS не возвращает лежащую в основе таблицу, если существует синоним

- 2696213. JDBC GETPROCEDURECOLUMNS не возвращает столбцы для синонима процедуры

Кроме того, информация о столбце не будет возвращена, если используется частный синоним, ссылающийся на объект в другой схеме.

Исправление

Для таблиц, которые имеют синоним, сгенерируйте бизнес-объект с помощью синонима для таблицы.

Для синонимов процедур сгенерируйте бизнес-объект с помощью исходной процедуры, на которой основан синоним.

Для частных синонимов, которые ссылаются на объект в другой схеме, или используйте исходную таблицу, или создайте синоним в текущей схеме.

Исключения ResourceException при исходящей обработке

Если возникла исключительная ситуация ResourceException, проверьте поле корневой причины. Обычные неполадки имеют следующие корневые причины:

- Исключительная ситуация SQLException

Если исключительная ситуация SQLException включает в себя текст Неверный ИД пользователя или пароль, значит для исходящего соединения указан неверный ИД пользователя или пароль.

Например:

```
javaax.resource.ResourceException: ORA-01017; неверное имя пользователя или пароль; вход запрещен.
```

- Исключительная ситуация ConnectException

Если текст исключительной ситуации ConnectException включает в себя такие слова как нет доступа или невозможно установить соединение, значит сервер базы данных не работает, или существует сетевая неполадка, которая не позволяет установить соединение.

Например:

```
java.sql.SQLException: Исключительная ситуация ввода/вывода: Сетевому адаптеру не удалось установить соединение
```

Исключительная ситуация ResourceException при входящей обработке

Эта исключительная ситуация указывает на то, что существует повторяющаяся неполадка при подключении к базе данных. Для опроса событий адаптер должен подключиться к базе данных. Если не удастся установить соединение, то адаптер ждет настраиваемый интервал времени перед повторной попыткой. Адаптер повторяет попытку настраиваемое число раз перед завершением опроса. Когда адаптер завершает опрос, он генерирует исключительную ситуацию ResourceException.

Неполадка UniqueConstraintViolation, неполадка MultiMatchingRecordsException

Нарушение загрузчика классов при запуске ESW (мастер внешних служб)

Неполадка

Невозможно использовать ESW (мастер внешних служб) после использования соединения с базой данных в проекции Данные. В конце второй панели мастера генерируется следующая исключительная ситуация:

```
com.ibm.adapter.framework.api.ImportException  
Причина: нарушено ограничение загрузки класса (класс:  
oracle/jdbc/driver/OracleConnection  
метод: getWrapper()Loracle/jdbc/OracleConnection;) at pc:0
```

Ошибка происходит в обеих следующих ситуациях:

- Когда вы устанавливаете соединение с базой данных с помощью ESW (мастер внешних служб), возникает ошибка при попытке подключиться к базе данных из проекции Данные.
- Когда вы устанавливаете соединение с базой данных посредством проекции Данные, возникает ошибка при попытке подключиться к базе данных с помощью ESW (мастер внешних служб).

Причина

Ошибка возникает, потому что проекция Данные и мастер используют свои собственные загрузчики классов. После загрузки в проекцию Данные DLL, которая является собственной библиотекой, используемой драйвером JDBC, она не может быть загружена снова в мастер. JVM имеют внутреннее ограничение, которое позволяет только одному загрузчику классов загружать собственные библиотеки в любой момент времени. Поэтому, если загрузчик классов A загружает DLL B, то никакой другой загрузчик классов не может загрузить DLL B, пока загрузчик классов A не будет освобожден и мусор не будет собран. Так как вы фактически не можете управлять сбором мусора, это обычно означает, что если вы хотите загрузить DLL B с помощью другого загрузчика классов, вам необходимо перезапустить JVM. Это ограничение известно и документировано для WebSphere Application Server.

Исправление

Единственное решение состоит в перезапуске WebSphere Integration Developer при возникновении этой ошибки.

Ошибка закрытого соединения возникает при использовании XA с Oracle 10g

Неполадка

Когда Adapter for Oracle E-Business Suite используется для выполнения транзакции XA с помощью Oracle 10g, адаптер генерирует исключительную ситуацию закрытого соединения: `javax.resource.ResourceException: Closed Connection`.

Причина

Это известная неполадка драйвера базы данных Oracle 10g. Следующий дефект зарегистрирован в Oracle для этой неполадки: 3488761 Ошибка закрытого соединения из `OracleConnection.getConnection()` - драйверы 10G.

Исправление

Дефект исправлен в драйвере Oracle 10g выпуска 2. В качестве обходного пути можно использовать драйверы Oracle 9i JDBC Thin для подключения к базе данных для транзакций XA.

Ошибка при запуске транзакции в Oracle

Неполадка

Когда Adapter for Oracle E-Business Suite используется для выполнения транзакции XA с помощью базы данных Oracle, генерируется следующая ошибка: WTRN0078E: Попытка администратора транзакций запустить ресурс транзакции привела к ошибке. Код ошибки - XAER_RMERR.

Причина

Для поддержки сервером базы данных Oracle транзакций XA должны быть выполнены некоторые команды.

Исправление

Должны быть выполнены два сценария, которые находятся в каталоге Oracle. Эта операция, скорее всего, должна быть выполнена администратором базы данных Oracle, потому что для этого необходимо войти в Oracle как пользователь SYSOPER или SYSDBA, чтобы иметь необходимые права доступа для выполнения этих сценариев. Сценарии:

```
<ORACLE_HOME>javavm\install  
file: initxa.sql  
file: initjvm.sql
```

Сценарий initxa.sql настраивает базу данных для XA. При его успешном выполнении база данных настраивается для XA. Сценарий может выполняться успешно при первой попытке. К сожалению, он вероятно не будет выполнен успешно, так как недостаточно свободной памяти базы данных.

Для того чтобы исправить эту неполадку, выполните сценарий initjvm.sql. Вероятно, он также не будет выполнен успешно, но при этом будут указаны параметры, требующие настройки. Параметры хранятся в следующем файле:

```
<ORACLE_HOME>\database  
file: init<DATABASE_SID>.ora
```

Табл. 14 показывает два параметра, которые обычно необходимо увеличить. Определенная конфигурация базы данных может потребовать настройки других параметров.

Таблица 14. Обычные размеры параметров

Имя параметра	Минимальное значение
java_pool_size	12000000
shared_pool_size	24000000

Более подробное рассмотрение столбца транзакции (XID) в таблице событий

Если адаптер настроен для гарантированной единичной доставки, используйте столбец состояния в столбце XID для определения того, обработано ли событие:

- Если столбец XID содержит 0, значит событие еще не принято для обработки.
- Если столбец XID содержит ИД транзакции (то есть, он не содержит 0), значит адаптер начал процесс обработки события, которая еще не завершена. Эту комбинацию можно увидеть при сбросе адаптера или сервера приложений в процессе обработки события. Администратор транзакций выполнит или операцию COMMIT, или операцию ROLLBACK, для этих транзакций в процессе восстановления.

Обработка неожиданных результатов из оператора SQL запроса

Получив неожиданные результаты из запроса, включите трассировку и просмотрите информацию о запросе SQL в протоколе. Включение трассировки особенно полезно при работе в клиенте тестирования для того, чтобы убедиться в том, что вы не забыли *отменить установку* всех ненужных атрибутов. Также полезно включить трассировку, чтобы определить, правильно ли заполнен входной бизнес-объект.

Глава 9. Справочная информация

Подробная информация о бизнес-объектах, свойствах адаптера (свойствах поиска служб предприятия, свойствах адаптера ресурсов, свойствах фабрики управляемых (J2C) соединений, свойствах спецификации активации и свойствах спецификации взаимодействия), сообщениях и связанная информация о продукте предоставлена для справки.

Информация о бизнес-объекте

Бизнес-объект - это структура, которая содержит информацию, специфичную для приложения (метаданные), о способе обработки адаптером бизнес-объекта, а также об операции, которая должна быть выполнена для бизнес-объекта. Имя бизнес-объекта генерируется ESW (мастер внешних служб) в соответствии с соглашением об именах для адаптера.

Атрибуты бизнес-объектов

Атрибуты бизнес-объектов определяют содержимое бизнес-объекта и состоят из списка столбцов в объекте базы данных.

Бизнес-объект - это просто контейнер для данных, указанных в атрибутах. Каждый атрибут имеет имя, тип, количество элементов и несколько других свойств. ESW (мастер внешних служб) устанавливает имя атрибута в имя столбца. Адаптер добавляет количество атрибутов, тип и информацию, специфичную для приложения. Структура данных в базе данных определена бизнес-объектом, но данные в базе данных - это атрибуты бизнес-объекта.

Табл. 15 перечисляет свойства атрибута бизнес-объекта и описывает их интерпретацию и параметры.

Таблица 15. Свойства атрибута

Свойства	Интерпретация и параметры
Количество элементов	<p>Целое число, указывающее количество элементов бизнес-объекта. Каждый атрибут бизнес-объекта, который представляет дочерний бизнес-объект или массив дочерних бизнес-объектов, имеет значение одинарный или множественный (неограниченное целое) соответственно.</p> <p>В одинарных и множественных взаимосвязях, отношение между родительским и дочерними бизнес-объектами описываются информацией, специфичной для приложения, в атрибуте ключа бизнес-объекта.</p>
Внешний ключ	<p>При извлечении массивов дочерних бизнес-объектов, количество элементов которых равно n, внешние ключи используются в предложении WHERE оператора SELECT.</p> <p>Операция RetrieveAll переопределяет использование ключей и внешних ключей.</p> <p>Примечание: Адаптер не поддерживает указание атрибута, который представляет дочерний бизнес-объект как внешний ключ.</p>
Имя	<p>Это свойство представляет уникальное имя атрибута, если это простой атрибут, или имя бизнес-объекта, если это дочерний бизнес-объект.</p>

Таблица 15. Свойства атрибута (продолжение)

Свойства	Интерпретация и параметры
MinOccurs MaxOccurs	Если столбец не является первичным ключом и не допускает нулевые значения, то атрибуты MinOccurs и MaxOccurs являются обязательными, и их значения не меньше 1.
Первичный ключ	<p>Указывает на то, является ли атрибут первичным ключом. По крайней мере один простой атрибут в каждом бизнес-объекте должен быть указан как первичный ключ.</p> <p>Если свойство первичного ключа установлено в значение true для простого атрибута, то адаптер добавляет этот атрибут в предложение WHERE оператора SELECT и в предложения SQL UPDATE, которые он генерирует при обработке бизнес-объекта. Операция RetrieveAll переопределяет использование первичных и внешних ключей.</p> <p>Примечание: Адаптер не поддерживает указание атрибута, который представляет дочерний бизнес-объект или массив дочерних бизнес-объектов, как атрибута первичного ключа.</p>
Обязательный	Указывает, должен ли атрибут содержать значение. Если это свойство имеет значение true для контейнера, количество элементов которого является одинарным (1), то требуется требует, чтобы родительский бизнес-объект содержал дочерний бизнес-объект для этого атрибута. Бизнес-объекты, которые передаются в адаптер для операций Create, Update и Delete, должны также содержать дочерний бизнес-объект. Количество элементов является одинарным (1) для простых атрибутов и множественным (n) для атрибутов контейнера. Адаптер не позволяет выполнить операцию Create, если бизнес-объект не имеет верного значения или значения по умолчанию для требуемого атрибута. Она также не выполняется, если данные не доступны для извлечения из базы данных для этого объекта.
Тип	<p>Для простых атрибутов это свойство задает тип, такой как Integer, String, Date, Timestamp, Boolean, Double или Float. Поддерживаемые типы для простых атрибутов и их преобразования в тип Oracle объекта базы данных описаны в Табл. 16.</p> <p>Для атрибутов, которые указывают дочерний бизнес-объект, это свойство задает имя бизнес-объекта.</p>

Тип каждого объекта базы данных, возвращенный как метаданные Oracle, преобразуется в тип атрибута бизнес-объекта, как указано в Табл. 16. Адаптером поддерживаются только перечисленные типы Oracle. Все столбцы с типами, которые не перечислены, не добавляются в бизнес-объект. Создается информационное сообщение, объясняющее неполадку, например, Столбец с именем xxxx в таблице с именем уууу имеет неподдерживаемый тип и не будет добавлен в бизнес-объект.

Примечание: Если при генерации бизнес-объекта для хранимых процедур или таблиц, метаданные Oracle не сопоставлены такому же типу атрибута бизнес-объекта, измените тип данных атрибута вручную в файле XSD для бизнес-объекта.

Таблица 16. Тип столбца метаданных Oracle и типы атрибутов бизнес-объекта

Тип столбца метаданных Oracle	Тип атрибута бизнес-объекта
CHAR LONG VARCHAR2	Строка
NUMBER	Строка

Таблица 16. Тип столбца метаданных Oracle и типы атрибутов бизнес-объекта (продолжение)

Тип столбца метаданных Oracle	Тип атрибута бизнес-объекта
TIMESTAMP DATE	Строка
FLOAT	Double
BLOB	hexBinary
CLOB	Строка
NCHAR NVARCHAR2	Строка
RAW LONG RAW	hexBinary

Информация уровня приложения для атрибутов

Информация уровня приложения (ASI) для атрибутов бизнес-объектов различается в зависимости от того, является ли атрибут простым, или представляет дочерний объект или массив дочерних бизнес-объектов. Информация уровня приложения для атрибута, представляющего дочерний объект, определяется тем, где хранится отношение родительского и дочернего объекта: в дочернем или в родительском объекте.

Информация уровня приложения для простых атрибутов

Информация уровня приложения для простых атрибутов содержит несколько параметров и их значения. Единственный обязательный параметр для простого атрибута - это имя столбца. Информация уровня приложения для простых атрибутов описывается в таблице Табл. 17.

Таблица 17. Информация уровня приложения для простых атрибутов

Параметр	Тип	Описание	Значение по умолчанию
BLOB	Boolean	Указывает, имеет ли столбец базы данных, соответствующий данному атрибуту, тип данных BLOB. При отображении данных BLOB адаптер выводит число байтов в шестнадцатеричном формате. Тип атрибута - hexBinary. Если значение этого параметра равно True, то тип данных столбца - BLOB.	Нет
ByteArray	Boolean	Указывает, является ли тип данных столбца двоичным. Если значение равно True, то адаптер считывает и записывает двоичные данные в базу данных и передает их в виде строки на сервер приложений. Адаптер задает двоичные данные в бизнес-объекте. Тип атрибута - hexBinary.	False
ChildBOType	String	Если атрибут - сложный тип данных, то информация уровня приложения позволяет задать фактический тип: <ul style="list-style-type: none"> • Struct • Array • ResultSet 	Нет

Таблица 17. Информация уровня приложения для простых атрибутов (продолжение)

Параметр	Тип	Описание	Значение по умолчанию
ChildBOTypeName	String	Если значение информации уровня приложения ChildBOType равно Struct или Array, то значение данного параметра представляет имя пользовательского типа. В этом значении учитывается регистр символов.	
CLOB	Boolean	Указывает, имеет ли столбец базы данных, соответствующий данному атрибуту, тип данных CLOB. Это значение применимо только к атрибутам типа String. Если значение этого параметра равно True, то тип данных столбца - CLOB. Атрибут CLOB имеет тип String, длина которого используется для определения длины CLOB.	Нет
ColumnName	String	Имя столбца базы данных, соответствующего этому атрибуту. Это единственный обязательный параметр.	Нет
CopyAttribute	String	Пользовательское значение, указывающее на имя другого атрибута из того же бизнес-объекта или родительского бизнес-объекта. Если значение, заданное в информации уровня приложения, указывает на имя другого атрибута в том же самом бизнес-объекте, то перед тем как добавить бизнес-объект в базу данных во время операции Create, адаптер использует значение другого атрибута для задания значения данного атрибута (в котором определена информация уровня приложения). Например, если столбец contact новой строки таблицы должен содержать то же значение, что и столбец email, укажите в параметре CopyAttribute атрибута contact значение email. Значение не может указывать на атрибут в дочернем бизнес-объекте, но может указывать на атрибут в родительском бизнес-объекте. В этом случае перед именем добавляются две точки. Например, можно указать ссылку на атрибут ccode родительского бизнес-объекта в виде ..ccode. Если информация уровня приложения не содержит этот параметр, то адаптер не копирует значение из другого атрибута, а использует значение текущего атрибута.	Нет

Таблица 17. Информация уровня приложения для простых атрибутов (продолжение)

Параметр	Тип	Описание	Значение по умолчанию
DateType	String	<p>Указывает, что соответствующий элемент представляет собой дату, время или метку времени. Укажите одно из следующих значений:</p> <ul style="list-style-type: none"> • дату • время • метку времени <p>Значение атрибута типа DateType задается в одном из следующих форматов:</p> <ul style="list-style-type: none"> • дата - гггг-ММ-дд • время - чч:мм:сс • метка времени - гггг-ММ-дд чч:мм:сс.нннннннн <p>Примечание: Адаптер использует значение системного времени из базы данных. Подробную информацию о методе Timestamp можно найти на Web-сайте Sun по адресу http://java.sun.com/j2se/1.5.0/docs/api/, указав в строке поиска <i>Timestamp</i>.</p>	Нет
FixedChar	Boolean	<p>Указывает, имеет ли атрибут постоянную длину, если тип столбцов в таблице - не VARCHAR, а CHAR. Например, если значение этого параметра равно true, то при обращении с запросом к базе данных адаптер дополняет пробелами значение атрибута, связанного со столбцом типа CHAR, до максимальной длины атрибута.</p> <p>Этот параметр должен обновляться вручную в файле XSD для бизнес-объекта. Для изменения файла XSD откройте бизнес-объект в редакторе XML или текстовом редакторе. Необходимо внести два изменения:</p> <ol style="list-style-type: none"> 1. Удалите строку <code>type="string"</code>, добавленную по умолчанию к тегу <code><element></code> атрибута объекта. 2. Добавьте новый раздел <code><simpletype></code> перед тегом <code></element></code>, как показано в следующем примере: <pre> <simpletype> <restriction base="string"> <maxLength value="10"/> </restriction> </simpletype> </pre> <p>Сохраните определение объекта и убедитесь в отсутствии ошибок проверки в файле XSD после его обновления.</p> <p>Пользуясь этой таблицей, просмотрите пример параметра FixedChar в файле XSD бизнес-объекта.</p>	false

Таблица 17. Информация уровня приложения для простых атрибутов (продолжение)

Параметр	Тип	Описание	Значение по умолчанию
ForeignKey	String	<p>Значение этого свойства определяется тем, где хранится отношение родительский-дочерний: в родительском или в дочернем бизнес-объекте.</p> <p>Если отношение хранится в родительском объекте, то значение содержит как тип дочернего бизнес-объекта, так и имя атрибута дочернего объекта, применяемого в качестве внешнего ключа (<i>Child_BO_name/Child_Property_Name</i>).</p> <p>Если отношение хранится в дочернем объекте, то значение содержит только имя атрибута родительского объекта, используемого в качестве внешнего ключа.</p> <p>Если атрибут не является внешним ключом, не включайте этот параметр в информацию уровня приложения.</p>	Нет
OrderBy	String	<p>Если указано значение этого параметра, и атрибут находится в дочернем бизнес-объекте, то адаптер использует значение этого атрибута в блоке ORDER BY запросов на извлечение.</p> <p>Адаптер может извлекать дочерние бизнес-объекты или в восходящем (ASC), или в нисходящем (DESC) порядке. Если этот параметр не включен в информацию уровня приложения, то адаптер не задает порядок извлечения.</p>	Нет
PrimaryKey	Boolean	<p>Если столбец, связанный с этим атрибутом, представляет собой первичный ключ в соответствующей таблице базы данных, то значение PrimaryKey равно True.</p>	Нет
SPParameterType	String	<p>Задаст тип хранимой процедуры.</p> <p>Возможные значения:</p> <ul style="list-style-type: none"> • IP (только ввод) • OP (только вывод) • IO (ввод и вывод) • RS (набор результатов) 	Нет
UniqueIdentifier (UID)	String	<p>Этот параметр используется адаптером для генерации уникального ИД бизнес-объекта. Поддерживается генерация последовательностей и столбцов идентификации.</p> <p>Формат этого параметра следующий:</p> <p>UID=AUTO <i>имя-последовательности</i></p> <p>Для последовательности в атрибуте UID указывается имя последовательности.</p> <p>Для столбца идентификации в атрибуте UID задается значение AUTO.</p> <p>Если уникальный ИД для атрибута не требуется, не включайте этот параметр в информацию уровня приложения.</p>	Нет

Формат информации уровня приложения для атрибута показан в следующем примере раздела файла XSD:

Пример раздела файла XSD

```
<element name="pkey" nillable="true"
  minOccurs="0" maxOccurs="1">
  <annotation xml:space="preserve">
    <appinfo
      source="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/
metadata">
      <oracleasi:OracleAttributeTypeMetadata
xmlns:oracleasi="http://www.ibm.com/xmlns/prod/websphere/j2ca/
oracle/metadata">
        <oracleasi:ColumnName>PKEY</oracleasi:ColumnName>
        <oracleasi:PrimaryKey>true</oracleasi:PrimaryKey>
        <oracleasi:FixedChar>true</oracleasi:FixedChar>
        </oracleasi:OracleAttributeTypeMetadata>
      </appinfo>
    </annotation>
    <simpleType>
      <restriction base="string">
        <maxLength value="10"/>
      </restriction>
    </simpleType>
  </element>
  <element name="ccode" type="string" nillable="true"
minOccurs="0" maxOccurs="1">
    <annotation xml:space="preserve">
      <appinfo
source="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/metadata">
      <oracleasi:OracleAttributeTypeMetadata
xmlns:oracleasi="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/
metadata">
        <oracleasi:ColumnName>CCODE</oracleasi:ColumnName>
        <oracleasi:PrimaryKey>false</oracleasi:PrimaryKey>
        <oracleasi:ForeignKey>custinfoobj/ccode</oracleasi:ForeignKey>
      </oracleasi:OracleAttributeTypeMetadata>
    </appinfo>
  </annotation>
  </element>
  <element name="fname" type="string" nillable="true"
minOccurs="0" maxOccurs="1">
    <annotation xml:space="preserve">
      <appinfo
source="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/metadata">
      <oracleasi:OracleAttributeTypeMetadata
xmlns:oracleasi="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/
metadata">
        <oracleasi:ColumnName>FNAME</oracleasi:ColumnName>
        <oracleasi:PrimaryKey>false</oracleasi:PrimaryKey>
      </oracleasi:OracleAttributeTypeMetadata>
    </appinfo>
  </annotation>
  </element>
  <element name="lname" type="string" nillable="true"
minOccurs="0" maxOccurs="1">
    <annotation xml:space="preserve">
      <appinfo
source="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/metadata">
      <oracleasi:OracleAttributeTypeMetadata
xmlns:oracleasi="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/
```

```

metadata">
<oracleasi:ColumnName>LNAME</oracleasi:ColumnName>
<oracleasi:PrimaryKey>false</oracleasi:PrimaryKey>
</oracleasi:OracleAttributeTypeMetadata>
</appinfo>
</annotation>
      </element>
      <element name="custinfoobj"
        type="rtassercustinfo:RtasserCustinfo"
nillable="true" minOccurs="0"
        maxOccurs="1">
        <annotation xml:space="preserve">
<appinfo
source="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/metadata">
<oracleasi:OracleAttributeTypeMetadata
xmlns:oracleasi="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/
metadata">
<oracleasi:Ownership>true</oracleasi:Ownership>
<oracleasi:KeepRelationship>false</oracleasi:KeepRelationship>
</oracleasi:OracleAttributeTypeMetadata>
</appinfo>
</annotation>
      </element>

```

Пример параметра FixedChar в файле XSD бизнес-объекта

```

<element name="pkey" nillable="true"
  minOccurs="0" maxOccurs="1">
  <annotation xml:space="preserve">
<appinfo
  source="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/
metadata">
  <oracleasi:OracleAttributeTypeMetadata
  xmlns:oracleasi="http://www.ibm.com/xmlns/prod/websphere/j2ca/
oracle/metadata">
  <oracleasi:ColumnName>PKEY</oracleasi:ColumnName>
  <oracleasi:PrimaryKey>true</oracleasi:PrimaryKey>
  <oracleasi:FixedChar>true</oracleasi:FixedChar>
  </oracleasi:OracleAttributeTypeMetadata>
  </appinfo>
  </annotation>
  <simpleType>
  <restriction base="string">
  <maxLength value="10"/>
  </restriction>
  </simpleType>
  </element>

```

Информация уровня приложения для атрибутов дочернего бизнес-объекта типа

В противоположность простым атрибутам, для сложных атрибутов, относящихся к дочерним бизнес-объектам, используются два параметра информации уровня приложения. При задании этой информации необходимо указать параметры, перечисленные в таблице Табл. 18 на стр. 159.

Таблица 18. Информация уровня приложения для атрибутов дочернего бизнес-объекта типа

Параметр	Тип	Описание	Значение по умолчанию
KeepRelationship	Boolean	Значение True запрещает удаление дочернего бизнес-объекта во время операции Update.	Нет
Ownership	Boolean	Этот параметр указывает на принадлежность дочернего бизнес-объекта родительскому бизнес-объекту. Если указано значение True, то над дочерним бизнес-объектом разрешено выполнять операции Create, Update и Delete. Если значение равно False, то к дочернему объекту не применимы никакие обновления. При создании родительского объекта проверяется существование дочернего объекта, чтобы обеспечить сохранение целостности взаимосвязей в базе данных.	Нет

Пример принадлежности в файле XSD бизнес-объекта

```

<element name="addressobj"
          type="rtasseraddress:RtasserAddress"
          nillable="true"
          minOccurs="0"
          maxOccurs="unbounded">
  <annotation xml:space="preserve">
    <appinfo
      source="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/metadata">
      <oracleasi:OracleAttributeTypeMetadata
        xmlns:oracleasi="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/metadata">
        <oracleasi:Ownership>true</oracleasi:Ownership>
      </oracleasi:OracleAttributeTypeMetadata>
    </appinfo>
  </annotation>
</element>

<element name="custinfoobj"
          type="rtassercustinfo:RtasserCustinfo"
          nillable="true"
          minOccurs="1"
          maxOccurs="1">
  <annotation xml:space="preserve">
    <appinfo
      source="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/metadata">
      <oracleasi:OracleAttributeTypeMetadata
        xmlns:oracleasi="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/metadata">
        <oracleasi:Ownership>>false</oracleasi:Ownership>
      </oracleasi:OracleAttributeTypeMetadata>
    </appinfo>
  </annotation>
</element>

```

Ниже приведен пример файла определения XSD для единичного и множественного дочерних бизнес-объектов. Элемент custInfoObj - это единичный дочерний бизнес-объект, а addressObj - множественный дочерний бизнес-объект.

Другой пример файла XSD единичного и множественного дочерних бизнес-объектов

```

<element name="addressobj"
          type="rtasseraddress:RtasserAddress"
          nillable="true"
          minOccurs="0"
          maxOccurs="unbounded">
  <annotation xml:space="preserve">
    <appinfo
      source="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/metadata">

```



```

<oracleasi:OracleAttributeTypeMetadata
xmlns:oracleasi="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/
metadata">
<oracleasi:Ownership>true</oracleasi:Ownership>
</oracleasi:OracleAttributeTypeMetadata>
</appinfo>
</annotation>
</element>

<element name="custinfoobj"
          type="rtassercustinfo:RtasserCustinfo"
nillable="true"          minOccurs="0"
          maxOccurs="1">
  <annotation xml:space="preserve">

<appinfo
source="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/metadata">
<oracleasi:OracleAttributeTypeMetadata
xmlns:oracleasi="http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/
metadata">
<oracleasi:Ownership>false</oracleasi:Ownership>
</oracleasi:OracleAttributeTypeMetadata>
</appinfo>
</annotation>
</element>

```

Информация уровня приложения для операций

Информация уровня приложения используется адаптером для выполнения таких операций, как извлечение информации из базы данных и ее обновление. Адаптер извлекает и обновляет таблицы базы данных с помощью запросов SQL, хранимых процедур или хранимых функций, указанных в бизнес-объектах.

При добавлении в бизнес-объекты хранимых процедур или хранимых функций задайте информацию уровня приложения (ASI) в соответствии с таблицей Табл. 19.

Таблица 19. Информация уровня приложения для операции

Элемент ASI операции для параметров хранимой процедуры	Задается мастером	Описание
Parameters	Да	Список параметров хранимой процедуры.
PropertyName	Да	Задаёт имя атрибута бизнес-объекта, который выбирается.
ResultSet	Нет	Если хранимая процедура возвращает набор результатов, задайте для этого параметра в определении бизнес-объекта значение True.

Таблица 19. Информация уровня приложения для операции (продолжение)

Элемент ASI операции для параметров хранимой процедуры	Задается мастером	Описание
ReturnValue	Да	Если хранимая процедура возвращает значение, этот параметр содержит одно из следующих значений: <ul style="list-style-type: none"> Строку RS. Это значение указывает, что процедура возвращает набор результатов, который используется для создания контейнера с множеством элементов, соответствующего данному бизнес-объекту. Имя атрибута бизнес-объекта. Это значение указывает, что процедура возвращает значение, которое во время выполнения должно быть присвоено конкретному атрибуту бизнес-объекта. Если атрибут представляет собой другой дочерний бизнес-объект, адаптер возвращает ошибку.
StoredProcedure	Да	Содержит имя хранимой процедуры.
StoredProcedureType	Да	Выбирается из списка типов. Информация о допустимых типах хранимых процедур приведена в разделе “Тип хранимой процедуры” на стр. 36.
Type	Да	Задаёт тип параметра хранимой процедуры. Возможные значения: <ul style="list-style-type: none"> IP (только ввод) OP (только вывод) IO (ввод и вывод) RS (набор результатов)

Информация уровня приложения для бизнес-объектов

Информация уровня приложения в определениях бизнес-объектов предоставляет адаптеру инструкции по обработке бизнес-объектов, относящиеся к конкретному приложению. Для генерации запросов на выполнение операций Create, Update, Retrieve и Delete адаптер анализирует информацию уровня приложения из бизнес-объекта либо из его атрибутов или операций.

Информация уровня приложения для бизнес-объектов таблицы и представления

Информация уровня приложения на уровне бизнес-объекта используется для указания имени соответствующей таблицы базы данных и для задания сведений, необходимых для выполнения физической или логической операции Delete.

ESW (мастер внешних служб) задает для атрибута ASI TableName значение в формате *ИмяСхемы.ИмяТаблицы*. Мастер запрашивает информацию, необходимую для выполнения физической или логической операции Delete, а затем задает информацию уровня приложения для бизнес-объекта, перечисленную в Табл. 20.

Таблица 20. Информация уровня приложения (ASI) для бизнес-объекта таблицы

Информация уровня приложения	Тип	Описание
TableName	String	Имя таблицы базы данных, соответствующей этому бизнес-объекту.

Таблица 20. Информация уровня приложения (ASI) для бизнес-объекта таблицы (продолжение)

Информация уровня приложения	Тип	Описание
StatusColumnName	String	Указывает, каким образом удаляются данные из таблицы: логически или физически. Если параметр StatusColumnName не задан, выполняется физическое удаление данных. Если параметр задан, то он определяет имя столбца, который указывает на логическое удаление строки. Этот параметр задается при выборе объекта таблицы в ESW (мастер внешних служб). Этот параметр применим как к операции Update, так и к операции Delete.
StatusValue	String	Значение, указывающее на логическое удаление столбца. Это значение задается при выборе объекта таблицы в ESW (мастер внешних служб).

Для того чтобы проиллюстрировать, как адаптер определяет, какое удаление (логическое или физическое) должно выполняться в ответ на запрос на выполнение операции Update или Delete, предположим, что для бизнес-объекта Customer задана информация уровня приложения, показанная в таблице Табл. 21.

Таблица 21. Пример параметров информации уровня приложения для бизнес-объекта таблицы

Информация уровня приложения	Значение
TableName	customer
StatusColumnName	status
StatusValue	deleted

Предположим, что адаптер получает запрос на удаление клиента. Поскольку в ASI бизнес-объекта содержится параметр StatusColumnName, адаптер выполняет логическое удаление. Для этого строка “deleted”, указанная в параметре StatusValue, помещается в столбец status, который указан в параметре StatusColumnName.

В результате адаптер выполняет следующий оператор SQL:

```
UPDATE customer set status = 'deleted' where pkey = . . . .
```

Однако, если параметр StatusColumnName не задан, записи таблицы customer удаляются физически. Адаптер выполняет следующий оператор SQL:

```
DELETE from customer where pkey = . . . .
```

Информация уровня приложения для бизнес-объектов хранимых процедур

Для бизнес-объектов, основанных на хранимых процедурах, ESW (мастер внешних служб) задает для параметра SPName ASI уровня бизнес-объекта значение в виде *ИмяСхемы + ИмяХранимойПроцедуры*. Задается информация уровня приложения на уровне бизнес-объекта, перечисленная в таблице Табл. 22 на стр. 163. На основе входных и выходных параметров хранимой процедуры создаются атрибуты бизнес-объекта. Если хранимая процедура имеет одно возвращаемое значение, то создается соответствующий атрибут бизнес-объекта. Если возвращаемое значение

или любой из входных или выходных параметров имеют сложный тип данных, то мастер создает для них дочерние бизнес-объекты.

В ESW (мастер внешних служб) может поддерживаться поиск объектов базы данных для вложенных структур и массивов. Если эти дочерние объекты генерируются на основе возвращаемых наборов результатов, то их имена задаются в виде *Префикс + ИмяСхемы + ИмяХранимойПроцедуры + RetRS + Число*. Например, если хранимая процедура возвращает два набора результатов, то мастер создает для них два дочерних бизнес-объекта. Их имена будут, соответственно, *Префикс + ИмяСхемы + ИмяХранимойПроцедуры + RetRS1* и *Префикс + ИмяСхемы + ИмяХранимойПроцедуры + RetRS2*.

Если дочерние бизнес-объекты создаются на основе входных и выходных параметров со сложным типом данных ResultSet, Struct или Array, им присваиваются имена вида *Префикс+ИмяСхемы+ИмяХранимойПроцедуры+ИмяПараметра*. Для дочерних бизнес-объектов, соответствующих вложенным структурам или массивам, имена будут иметь следующий вид:

Префикс+ИмяСхемы+ИмяХранимойПроцедуры+ИмяПараметра+ИмяСтолца.

Таблица 22. Информация уровня приложения (ASI) для бизнес-объектов, основанных на хранимых процедурах

Информация уровня приложения	Тип	Описание
SPName	String	Имя хранимой процедуры или хранимой функции
ResultSet	Boolean	Указывает, будет ли хранимая процедура или хранимая функция возвращать набор результатов. Если значение равно true, то хранимая процедура возвращает один или несколько наборов результатов, если false, набор результатов не возвращается.
MaxNumberOfRetRS	String	Максимальное число возвращаемых наборов результатов, обрабатываемых средой выполнения адаптера
ReturnValue	String	Укажите имя соответствующего атрибута бизнес-объекта, если хранимая процедура возвращает значение. Если возвращается простой тип данных, то атрибут также представляет собой простой тип данных. Если возвращается набор результатов, то атрибут указывает на дочерний бизнес-объект.

Информация уровня приложения для бизнес-объектов запроса

Для бизнес-объектов запросов существует один параметр ASI уровня бизнес-объекта, как показано в таблице Табл. 23.

Таблица 23. Информация уровня приложения (ASI) для бизнес-объектов запроса

Информация уровня приложения	Тип	Описание
SelectStatement	String	Полный оператор SELECT, выполняющий запрос. Этот оператор задается в ESW (мастер внешних служб).

Кроме того, поскольку все бизнес-объекты - верхнего уровня, ESW (мастер внешних служб) создает для них бизнес-графики. Имя бизнес-графика состоит из имени бизнес-объекта, к которому добавляются буквы "BG." Например, для бизнес-объекта

OracleSchema1Customer будет создан бизнес-график с именем OracleSchema1CustomerBG. Операции, заданные в бизнес-объекте, заданы также и в бизнес-графике.

При создании мастером бизнес-объекта хранимой процедуры при необходимости создается и дочерний бизнес-объект типа ResultSet, Struct или Array. Создание отношений "родительский-дочерний" между бизнес-объектами таблиц выполняется вручную с помощью редактора бизнес-объектов.

Мастер обрабатывает бизнес-объекты на основе синонимов или псевдонимов точно так же, как бизнес-объекты на основе таблиц и представлений, даже когда синоним представляет хранимую процедуру.

Соглашение об именах

Когда ESW (мастер внешних служб) генерирует бизнес-объект, он предоставляет имя бизнес-объекта, которое соответствует соглашению об именах для адаптера. Обычно, имя бизнес-объекта указывает на его структуру.

Когда ESW (мастер внешних служб) создает имена бизнес-объектов, он заменяет все специальные символы, кроме знака подчеркивания () в имени бизнес-объекта на U с последующим номером символа в Unicode. Например, имя бизнес-объекта для таблицы Order_Item в базе данных - это Order_Item. Имя бизнес-объекта для таблицы Shipping-Address - это ShippingU45Address.

Имена бизнес-объектов не имеют семантического значения для адаптера или базы данных, то есть, они не несут ни информации, ни смысла. Если одно имя заменяется другим, поведение адаптера остается прежним.

Имена бизнес-объектов могут содержать метаданные, специфичные для базы данных. Имя может использовать строку, такую как Oracle или %AppName%, в качестве префикса для удобства различения между двумя типами бизнес-объектов: характерных для приложения и общих. Остальная часть имени может описывать таблицу или хранимую процедуру, которую представляет бизнес-объект. Например, если определение бизнес-объекта сгенерировано для таблицы Employee в приложении базы данных, таком как Human Resources (HR), то соответствующим именем бизнес-объекта будет HREmployee.

Для бизнес-объектов, которые не соответствуют объектам базы данных, например, бизнес-объектов для запросов к базе данных, если им присвоено такое же имя, как у бизнес-объекта таблицы или хранимой процедуры, то к концу этого имени добавляется число для различения объектов.

Глобализованные символы поддерживаются в именах бизнес-объектов.

Можно переименовать бизнес-объекты с помощью функциональности рефакторинга в WebSphere Integration Developer. Дополнительная информация находится в документации по адресу: WebSphere Integration Developer.

следующая таблица описывает соглашения об именах, которые используются мастером для бизнес-объектов.

Таблица 24. Соглашение об именах бизнес-объектов

Элемент	Соглашение об именах
Схема бизнеса	Схема бизнеса, которая содержит родительский бизнес-объект, именуется с помощью содержимого бизнес-объекта с добавлением строки “BG”. Например, схема бизнеса, которая содержит бизнес-объект SalesCustomer называется SalesCustomerBG.
Бизнес-объекты для: <ul style="list-style-type: none"> • Таблиц • Представлений • Хранимых процедур procedures • Хранимых функций • Синонимов и псевдонимов 	<p>Для тех бизнес-объектов, которые основаны на таблицах, представлениях, хранимых процедурах, синонимах и псевдонимах, ESW (мастер внешних служб) генерирует имя в виде <i>Префикс + Имя-схемы + Имя-объекта</i>, где:</p> <ul style="list-style-type: none"> • <i>Префикс</i> - это значение, указанное в свойстве соединения с внешней службой Prefix. Префикс не является обязательным, и если он не указан, то никакого префикса не добавлено к имени бизнес-объекта. • <i>Имя-схемы</i> - это имя схемы, к которой принадлежит объект. • <i>Имя-объекта</i> - это имя таблицы, представления, хранимой процедуры, хранимой функции или синонима/антонима. Число добавляется к имени, если необходимо отличить бизнес-объект от другого бизнес-объекта с таким же именем. <p>Например, после добавления префикса Campaign12 для таблицы Customer в схеме Sales именем бизнес-объекта является Campaign12SalesCustomer.</p>
Бизнес-объекты запросов	<p>Для бизнес-объектов запросов ESW (мастер внешних служб) генерирует имя в виде <i>Префикс + QueryBOName</i>, где:</p> <ul style="list-style-type: none"> • <i>Префикс</i> - это префикс, указанный в мастере. Префикс не является обязательным, и если он не указан, то никакого префикса не добавлено к имени бизнес-объекта. • <i>QueryBOName</i> - это значение, указанное при настройке бизнес-объекта в мастере. Число добавляется к имени, если необходимо отличить бизнес-объект от другого бизнес-объекта с таким же именем.

Свойства конфигурации исходящего соединения

WebSphere Adapter for Oracle E-Business Suite поддерживает несколько категорий свойств исходящего соединения, которые настраиваются с помощью ESW (мастер внешних служб) при создании объектов и служб. Обратите внимание, что в отличие от свойств адаптера ресурса и фабрики управляемых соединений, которые можно изменить после развертывания модуля на сервере WebSphere Process Server или WebSphere Enterprise Service Bus с помощью WebSphere Integration Developer или AdminConsole, свойства соединения для ESW (мастер внешних служб) нельзя изменить после развертывания.

Руководство по описанию свойств

Свойства, используемые для настройки WebSphere Adapter for Oracle E-Business Suite, подробно описаны в таблицах, которые включены во все разделы свойств конфигурации, таких как свойства адаптера ресурса, свойства фабрики управляемых соединений и т.д. Для того чтобы помочь вам использовать эти таблицы, здесь приведена информация о каждой строке.

Следующая таблица объясняет значение каждой строки, которая может быть показана в таблице для свойства конфигурации.

Строка	Описание
Обязательное	<p>Для обеспечения работы адаптера в обязательном поле (свойстве) должно быть указано значение. В некоторых случаях для обязательных полей ESW (мастер внешних служб) предусмотрены значения по умолчанию.</p> <p>Удаление значения по умолчанию из обязательного поля в ESW (мастер внешних служб) <i>не изменит значения по умолчанию</i>. Когда обязательное поле не содержит никакого значения, ESW (мастер внешних служб) обработает поле с помощью присвоенного значения по умолчанию, и это значение по умолчанию будет также показано в административной консоли.</p> <p>Возможные значения: Да и Нет.</p> <p>Кроме того, свойство может быть обязательным только в том случае, если в другом свойстве указано конкретное значение. Такие зависимости указываются в таблице. Например,</p> <ul style="list-style-type: none"> • Да, если для свойства EventQueryType указано значение Dynamic • Для для баз данных Oracle
Допустимые значения	Перечислены возможные значения свойства.
Значение по умолчанию	<p>Предопределенное значение, устанавливаемое ESW (мастер внешних служб). Для обязательного свойства следует принять значение по умолчанию или указать собственное значение. Если значение по умолчанию не предусмотрено, то в таблице будет указано Значение по умолчанию отсутствует.</p> <p>Слово Нет допустимо в качестве значения по умолчанию; оно не указывает на отсутствие значения по умолчанию.</p>
Единицы измерения	Задает единицы измерения свойства, например, килобайты или секунды.
Тип свойства	<p>Описывает тип свойства. Поддерживаются следующие типы свойств:</p> <ul style="list-style-type: none"> • Boolean • String • Integer
Применение	<p>Описывает особенности применения свойства. Например, ниже приведен пример описания свойства в документации:</p> <p>Для Rational Application Developer for WebSphere Software 6.40 и более ранних версий:</p> <ul style="list-style-type: none"> • Пароль должен содержать только прописные буквы • Длина пароля должна составлять 8 символов <p>Для Rational Application Developer for WebSphere Software после версии 6.40:</p> <ul style="list-style-type: none"> • В пароле не учитывается регистр символов • Пароль может содержать до 40 символов. <p>В этом разделе перечислены связанные свойства и рассмотрены типы условных взаимосвязей.</p>
Пример	<p>Содержит примеры значений свойства, например:</p> <p>"Например, если для свойства Язык указано значение JA (японский), то то для свойства Номер кодовой страницы необходимо указать значение 8000".</p>
Локализуемое	<p>Глобализованные свойства поддерживают национальные языки, т.е. значения можно указывать на национальных языках.</p> <p>Возможные значения: Да и Нет.</p>

Строка	Описание
Поддерживает двунаправленный текст	Указывает, поддерживается ли свойство в двунаправленной обработке (bidi). Двунаправленная обработка работает с данными, содержащими как текст, написанный слева направо (URL или путь к файлу), так и справа налево (арабский или иврит) в одном и том же файле. Возможные значения: Да и Нет .

Свойства соединения для мастера

Свойства соединения с внешней службой используются для установки соединения между ESW (мастер внешних служб), средством создания бизнес-объектов, и базой данных.

Свойства внешней службы указывают такие данные, как конфигурация соединения, свойства двунаправленного преобразования и опции ведения протоколов для мастера. После установки соединения мастер может найти в базе данных метаданные, которые ему необходимы для создания бизнес-объектов. Некоторые из свойств, предоставленных мастеру для поиска объектов в базе данных, используются как начальные значения для свойств среды выполнения, которые задаются позже в мастере. Они включают в себя свойства адаптера ресурса, фабрики управляемых соединений и спецификации активации.

Свойства соединения для ESW (мастер внешних служб) описаны в следующей таблице. Более подробное описание каждого свойства предоставлено в разделе, который следует за таблицей. Информация о том, как читать таблицу свойств, находится в разделе “Руководство по описанию свойств” на стр. 165.

Таблица 25. Свойства соединения для ESW (мастер внешних служб)

Имя свойства в мастере	Описание
Дополнительные свойства соединения драйвера JDBC	Дополнительные свойства для подключения к базе данных с помощью драйвера JDBC, применяемые совместно со свойствами UserName и Password.
“База данных” на стр. 168	Имя и версия программного обеспечения управления базой данных, к которой адаптер будет получать доступ
URL базы данных	URL базы данных, применяемый для подключения к базе данных
“Имя хоста” на стр. 169	Имя хоста или IP-адрес сервера базы данных
Имя класса драйвера JDBC	Имя класса драйвера JDBC
“Тип драйвера JDBC” на стр. 170	Тип используемого драйвера JDBC
Пароль	Пароль для соответствующего имени пользователя
“Номер порта” на стр. 171	Номер порта для подключения к экземпляру базы данных
“Префикс имен бизнес-объектов” на стр. 171	Префикс, добавляемый к имени бизнес-объекта
ИД системы	Имя ИД системы (SID) базы данных
Имя пользователя	Имя пользователя базы данных для подключения к ней

ESW (мастер внешних служб) использует свойства двунаправленного соединения для применения правильного двунаправленного преобразования данных, передаваемых в информационную систему предприятия.

Дополнительные свойства соединения драйвера JDBC

Содержит дополнительную информацию для подключения к базе данных с помощью драйвера JDBC.

Таблица 26. Дополнительные свойства соединения драйвера JDBC

Обязательное	Нет
Допустимые значения	Свойства подключения к базе данных задаются на уровне конкретных баз данных.
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Применяются совместно со свойствами UserName и Password для настройки соединения адаптера с базой данных. Укажите свойства в следующем формате: <i>имя:значения</i> . В качестве разделителя применяется точка с запятой (;).
Пример	Следующее значение этого свойства задает тайм-аут входа в систему: loginTimeout:20; ConnectionRetryCount:5; ConnectionRetryDelay:5
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

База данных

Это свойство указывает программное обеспечение, которое управляет базой данных, к которой адаптер получает доступ.

Таблица 27. Сведения о программном обеспечении базы данных

Строка	Описание
Обязательное	Да
Допустимые значения	Программное обеспечение базы данных Oracle по имени и номеру версии.
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	ESW (мастер внешних служб) использует значение этого свойства для установки значений по умолчанию и генерации списков опций, специфичных для этой базы данных, для других свойств. Например, если выбрано Oracle 10, поле класса драйвера JDBC в мастере показывает только те драйверы JDBC, которые поддерживаются этой версией базы данных Oracle.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

URL базы данных

Это свойство задает URL, специфичный для драйвера JDBC, для создания соединения с базой данных.

Таблица 28. Сведения об URL базы данных

Обязательное	Да
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Это значение специфично для программного обеспечения базы данных и используемого драйвера JDBC. Если ваш сервер базы данных поддерживает IPv6, можно указать часть имени хоста URL базы данных в формате IPv6. Заключите IP-адрес в квадратные скобки ([]).
Примеры	Ниже приведены типичные значения серверов базы данных Oracle. jdbc:oracle:thin:@9.26.248.148:1521:dev
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Имя хоста

Это свойство задает имя хоста или IP-адрес сервера базы данных.

Таблица 29. Имя хоста - Сведения

Строка	Описание
Обязательное	Да
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Это имя хоста или IP-адрес сервера базы данных. Если сервер базы данных поддерживает IPv6, можно указать имя хоста в формате IPv6.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Имя класса драйвера JDBC

Это свойство задает имя класса драйвера JDBC.

Таблица 30. Сведения об имени класса драйвера JDBC

Строка	Описание
Обязательное	Да
Допустимые значения	Возможные значения зависят от типа и версии базы данных. Мастер показывает список известных драйверов.
Значение по умолчанию	Значение по умолчанию зависит от типа и версии базы данных.
Тип свойства	Строка

Таблица 30. Сведения об имени класса драйвера JDBC (продолжение)

Строка	Описание
Формат	Хотя мастер показывает имя класса по умолчанию для выбранного типа драйвера JDBC, можно при необходимости ввести имя другого класса. Если выбрано Другое для значения драйвера JDBC, то значение по умолчанию не предоставляется, и необходимо ввести имя класса. Имя класса должно содержаться в файлах драйвера JDBC, предоставленных при запуске мастера.
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Тип драйвера JDBC

Это свойство задает тип используемого драйвера JDBC.

Таблица 31. Сведения о типе драйвера JDBC

Строка	Описание
Обязательное	Да
Допустимые значения	Возможные значения зависят от типа и версии базы данных. Мастер показывает список известных драйверов.
Значение по умолчанию	Значение по умолчанию зависит от типа и версии базы данных.
Тип свойства	Строка
Формат	Хотя в действительности вы выбираете или тип 2, или тип 4 (универсальный) для драйвера, каждая система базы данных имеет свое собственное имя для драйвера. Мастер показывает список драйверов, известных для каждой системы базы данных. Если ваш драйвер в нем не указан, выберите Другой. Информация в этом поле должна быть согласована с файлами драйвера JDBC, предоставленными при запуске мастера.
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Пароль (Пароль)

Это свойство задает пароль для имени пользователя базы данных.

Таблица 32. Пароль - Сведения

Строка	Описание
Обязательное	Да
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Пароль связан с именем пользователя, введенным для подключения к базе данных с целью поиска объектов.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Номер порта

Это свойство задает номер порта для экземпляра базы данных.

Таблица 33. Сведения о номере порта

Обязательное	Да
Значение по умолчанию	Значение по умолчанию специфично для базы данных и инициализируется мастером, если выбран именованный драйвер в поле Тип драйвера JDBC . Если для типа драйвера выбрано Другой, то значение по умолчанию не предоставляется.
Тип свойства	Строка
Формат	Это номер порта для подключения к экземпляру базы данных. Это свойство не доступно, если для типа драйвера JDBC выбрано значение Другой.
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Префикс имен бизнес-объектов

Префикс, добавляемый к имени бизнес-объекта.

Таблица 34. Сведения о префиксе

Обязательное	Нет
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Используйте префикс для удобства различения между типами бизнес-объектов.
Пример	Можно указать префикс <code>Oracle</code> для общих бизнес-объектов и <code>%AppName%</code> для бизнес-объектов, специфичных для приложения.
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

ИД системы

Это свойство задает имя ИД системы (SID) базы данных.

Таблица 35. Сведения об ИД системы

Обязательное	Да
Значение по умолчанию	Значение по умолчанию специфично для базы данных.
Тип свойства	Строка
Формат	ИД системы (SID) - это имя базы данных.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Имя пользователя (UserName)

Имя пользователя используется для подключения к базе данных.

Таблица 36. Имя пользователя - Сведения

Обязательное	Да
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Имя пользователя вводится для подключения к базе данных с целью поиска объектов.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Свойства адаптера ресурсов

Свойства адаптера ресурсов управляют общей работой адаптера, например, указывая пространство имен для бизнес-объектов. Свойства адаптера ресурсов устанавливаются с помощью ESW (мастер внешних служб) при настройке адаптера. После развертывания адаптера используйте административную консоль для изменения этих свойств.

Следующая таблица перечисляет свойства адаптера ресурсов и их назначение. Полное описание каждого свойства предоставлено в разделе, который следует за таблицей. Информация о том, как читать таблицу свойств, находится в разделе 1 "Руководство по описанию свойств" на стр. 165.

Таблица 37. Свойства адаптера ресурсов для Adapter for Oracle E-Business Suite

Имя		Описание
В мастере	В административной консоли	
ИД адаптера	AdapterID	Это свойство задает экземпляр адаптера для обработки событий PMI, а также для ведения протоколов и трассировки.
Производитель базы данных	DatabaseVendor	Тип базы данных, применяемой адаптеров для специальной обработки.
Скрыть пользовательские данные в виде "XXX" в файлах протокола и трассировки	HideConfidentialTrace	Позволяет вместо пользовательских данных заносить в файлы протоколов и трассировки символы X.
Тайм-аут запроса	QueryTimeout	Максимальное время выполнения запросов SQL (в секундах)
Возвращать бизнес-объект, даже когда набор результатов хранимой процедура пуст	ReturnDummyBOForSP	Применяется для возврата выходных параметров, даже если набор результатов пустой.
(Не доступно)	enableHASupport	Не используйте это свойство.
(Не доступно)	LogFileMaxSize	Устаревшее
(Не доступно)	LogFilename	Устаревшее
(Не доступно)	LogNumberOfFiles	Устаревшее

Таблица 37. Свойства адаптера ресурсов для Adapter for Oracle E-Business Suite (продолжение)

Имя		Описание
В мастере	В административной консоли	
Запрос SQL для проверки соединения	PingQuery	Запрос SQL, используемый для проверки надежности соединения с базой данных.
(Не доступно)	TraceFileMaxSize	Устаревшее
(Не доступно)	TraceFileName	Устаревшее
(Не доступно)	TraceNumberOfFiles	Устаревшее

ИД адаптера (AdapterID)

Задаёт конкретный экземпляр адаптера.

Таблица 38. ИД адаптера - Сведения

Обязательное	Да
Значение по умолчанию	001
Тип свойства	String
Применение	<p>Задаёт экземпляр адаптера в файлах протоколов и трассировки, а также помогает идентифицировать экземпляр адаптера в ходе мониторинга адаптеров. ИД адаптера вместе с идентификатором уровня адаптера (OEBSRA) образуют имя компонента, применяемое инструментом Анализатор протоколов и данных трассировки. Например, если свойство ИД адаптера содержит значение 001, то ИД компонента равен OEBSRA001.</p> <p>В случае применения нескольких экземпляров адаптера первые семь символов ИД адаптера должны быть уникальными для каждого экземпляра, для того чтобы информацию из файлов протоколов и трассировки можно было сопоставить с конкретным экземпляром адаптера. Такой подход также обеспечивает уникальность ИД компонентов экземпляров адаптера.</p> <p>Для того чтобы показать, каким образом длина ИД адаптера влияет на фильтрацию файлов протоколов и трассировки, предположим, что для двух экземпляров WebSphere Adapter for Oracle E-Business Suite в свойстве ИД адаптера указаны значения 001 и 002. ИД компонентов этих экземпляров (OEBSRA001 и OEBSRA002) достаточно коротки, чтобы оставаться уникальными, т.е. они позволяют отличать экземпляры друг от друга. Однако экземпляры с более длинными ИД адаптера нельзя отличить друг от друга. Например, предположим, что для двух экземпляров в свойстве ИД адаптера указаны значения Instance01 и Instance02. Пользователь не сможет отдельно проверить информацию из файлов протоколов и трассировки для каждого экземпляра адаптера, поскольку ИД компонента усекается до OEBSRAInstanc.</p> <p>В случае обработки входящих событий значение этого свойства указывается на уровне адаптера ресурса. В случае обработки исходящих запросов оно может быть указано как на уровне адаптера ресурса, так и на уровне фабрики управляемых соединений. После настройки адаптера для обработки исходящих запросов с помощью ESW (мастер внешних служб) можно задать свойство адаптера ресурса и фабрики управляемых соединений независимо друг от друга. В ходе настройки этих свойств с помощью редактора сборки WebSphere Integration Developer или административной консоли следует обратить внимание на согласованность значений свойств во избежание несогласованной маркировки записей протоколов и трассировки.</p>
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Производитель базы данных (DatabaseVendor)

Задает тип базы данных. Тип определяется в соответствии с производителем базы данных.

Таблица 39. Производитель базы данных - Сведения

Обязательное	Да
Допустимые значения	Oracle
Значение по умолчанию	ORACLE
Тип свойства	Строка
Формат	Отдельные операторы SQL требуют специальной обработки, которая зависит от типа базы данных. Например, типы данных Struct и Array в Oracle требуют специальной обработки. Это свойство задает RDBMS, которая определяет тип базы данных. Укажите Oracle в качестве значения, соответствующего производителю базы данных.
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Маскировать пользовательские данные как "XXX" в файлах протоколов и трассировки (HideConfidentialTrace) - Свойство

Позволяет заменять пользовательские данные в файлах протоколов и трассировки на символы X во избежание несанкционированного доступа к конфиденциальной информации.

Таблица 40. Маскировать пользовательские данные как "XXX" в файлах протоколов и трассировки - Сведения

Обязательное	Нет
Допустимые значения	True False
Значение по умолчанию	False
Тип свойства	Булевское
Применение	Если указано значение True, то вместо пользовательских данных адаптер заносит в файлы протоколов и трассировки символы X. В случае обработки входящих событий значение этого свойства указывается на уровне адаптера ресурса. В случае обработки исходящих запросов оно может быть указано как на уровне адаптера ресурса, так и на уровне фабрики управляемых соединений. После настройки адаптера для обработки исходящих запросов с помощью ESW (мастер внешних служб) можно задать свойства адаптера ресурса и фабрики управляемых соединений независимо друг от друга. В ходе настройки этих свойств с помощью редактора сборки WebSphere Integration Developer или административной консоли следует обратить внимание на согласованность значений свойств во избежание несогласованной маркировки записей протоколов и трассировки.
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Включить поддержку высокой готовности (enableHASupport)

Не используйте это свойство. Его значение должно быть равно true.

Тайм-аут запроса (QueryTimeout)

Максимальное время выполнения запросов SQL (в секундах).

Таблица 41. Тайм-аут запроса - Сведения

Обязательное	Нет
Значение по умолчанию	Значение по умолчанию отсутствует
Единицы измерения	Секунды
Тип свойства	Integer
Формат	Если время выполнения запроса превышает значение тайм-аута, то база данных выбрасывает исключение SQL. Связанное сообщение заносится в файл протокола. Если значение не указано, то тайм-аут не отслеживается.
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Возвращать бизнес-объект при получении пустого набора результатов хранимой процедуры (ReturnDummyBOForSP)

Применяется для возврата выходных параметров, даже если набор результатов пустой.

Таблица 42. Возвращать бизнес-объект при получении пустого набора результатов хранимой процедуры - Сведения

Обязательное	Нет
Допустимые значения	True False
Значение по умолчанию	False
Тип свойства	Булевское
Формат	Операция RetrieveSP возвращает набор результатов. Если набор результатов пуст и в свойстве ReturnDummyBOForSP указано значение False, то бизнес-объекты не создаются и выходные параметры вызова процедуры будут недоступны для загрузки. Если в свойстве ReturnDummyBOForSP указано значение True, то создается фиктивный бизнес-объект, содержащий значения соответствующих атрибутов.
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Запрос SQL для проверки соединения (PingQuery)

Запрос SQL, используемый для проверки надежности соединения с базой данных.

Таблица 43. Запрос проверки связи - Сведения

Обязательное	Нет
Тип свойства	Строка
Значение по умолчанию	Значение по умолчанию отсутствует
Формат	Содержит запрос SQL, применяемый для проверки соединения с базой данных. Адаптер выполняет проверку связи каждый раз при получении исключительной ситуации SQLException в ходе выполнения исходящих операций. Адаптер не выполняет попыток восстановления соединения. Если база данных недоступна, то адаптер отправляет уведомление контейнеру. За восстановление соединений с базой данных отвечает администратор пула соединений.
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Свойства фабрики управляемых соединений

Свойства фабрики управляемых соединений используются адаптером во время выполнения для создания экземпляра исходящего соединения в Oracle DB.

Свойства фабрики управляемых соединений устанавливаются с помощью ESW (мастер внешних служб) в процессе настройки адаптера. Их можно изменить с помощью редактора сборки WebSphere Integration Developer или после развертывания с помощью административной консоли WebSphere Process Server или WebSphere Enterprise Service Bus.

В следующей таблице перечислены и описаны свойства фабрики управляемых соединений. Полное описание каждого свойства предоставлено в разделе, который следует за таблицей. Информация о том, как читать таблицу свойств, находится в разделе “Руководство по описанию свойств” на стр. 165.

Примечание: ESW (мастер внешних служб) ссылается на эти свойства как на свойства фабрики управляемых соединений, в то время как административная консоль ссылается на них как на свойства фабрики соединений J2C.

Таблица 44. Свойства фабрики управляемых соединений для Adapter for Oracle E-Business Suite

Имя свойства		Описание
В мастере	В административной консоли	
ИД адаптера	AdapterID	Это свойство задает экземпляр адаптера для обработки событий PMI, а также для ведения протоколов и трассировки.
Дополнительные свойства драйвера JDBC	DriverConnectionProperties	Дополнительные свойства для подключения к базе данных с помощью драйвера JDBC, применяемые совместно со свойствами UserName и Password.
Автоматическая фиксация	AutoCommit	Значение AutoCommit, используемое при соединении
Имя JNDI источника данных	DataSourceJNDIName	Имя источника данных JNDI для установления соединения с базой данных.
URL базы данных	DatabaseURL	URL базы данных, применяемый для подключения к базе данных

Таблица 44. Свойства фабрики управляемых соединений для Adapter for Oracle E-Business Suite (продолжение)

Имя свойства		Описание
В мастере	В административной консоли	
Производитель базы данных	DatabaseVendor	Тип базы данных, применяемой адаптеров для специальной обработки.
Скрыть пользовательские данные в виде "XXX" в файлах протокола и трассировки	HideConfidentialTrace	Позволяет вместо пользовательских данных заносить в файлы протоколов и трассировки символы X.
Класс драйвера JDBC	JDBCDriverClass	Имя класса драйвера JDBC, применяемого для подключения к базе данных.
Пароль	Пароль	Пароль, связанный с именем пользователя.
Тайм-аут запроса	QueryTimeOut	Максимальное время выполнения запросов SQL (в секундах)
Возвращать бизнес-объект, даже когда набор результатов хранимой процедура пуст	ReturnDummyBOForSP	Применяется для возврата выходных параметров, даже если набор результатов пустой.
Запрос SQL для подключения	PingQuery	Запрос SQL, используемый для проверки надежности соединения с базой данных.
Имя пользователя	Имя пользователя	Имя пользователя базы данных.
Имя источника данных XA	XADataSourceName	Имя источника данных XA, используемое для установки соединения с базой данных для (распределенных) транзакций XA

ИД адаптера (AdapterID)

Задает конкретный экземпляр адаптера.

Таблица 45. ИД адаптера - Сведения

Обязательное	Да
Значение по умолчанию	001
Тип свойства	String

Таблица 45. ИД адаптера - Сведения (продолжение)

Применение	<p>Задаёт экземпляр адаптера в файлах протоколов и трассировки, а также помогает идентифицировать экземпляр адаптера в ходе мониторинга адаптеров. ИД адаптера вместе с идентификатором уровня адаптера (OEBSRA) образуют имя компонента, применяемое инструментом Анализатор протоколов и данных трассировки. Например, если свойство ИД адаптера содержит значение 001, то ИД компонента равен OEBSRA001.</p> <p>В случае применения нескольких экземпляров адаптера первые семь символов ИД адаптера должны быть уникальными для каждого экземпляра, для того чтобы информацию из файлов протоколов и трассировки можно было сопоставить с конкретным экземпляром адаптера. Такой подход также обеспечивает уникальность ИД компонентов экземпляров адаптера.</p> <p>Для того чтобы показать, каким образом длина ИД адаптера влияет на фильтрацию файлов протоколов и трассировки, предположим, что для двух экземпляров WebSphere Adapter for Oracle E-Business Suite в свойстве ИД адаптера указаны значения 001 и 002. ИД компонентов этих экземпляров (OEBSRA001 и OEBSRA002) достаточно коротки, чтобы оставаться уникальными, т.е. они позволяют отличать экземпляры друг от друга. Однако экземпляры с более длинными ИД адаптера нельзя отличить друг от друга. Например, предположим, что для двух экземпляров в свойстве ИД адаптера указаны значения Instance01 и Instance02. Пользователь не сможет отдельно проверить информацию из файлов протоколов и трассировки для каждого экземпляра адаптера, поскольку ИД компонента усекается до OEBSRAInstanc.</p> <p>В случае обработки входящих событий значение этого свойства указывается на уровне адаптера ресурса. В случае обработки исходящих запросов оно может быть указано как на уровне адаптера ресурса, так и на уровне фабрики управляемых соединений. После настройки адаптера для обработки исходящих запросов с помощью ESW (мастер внешних служб) можно задать свойства адаптера ресурса и фабрики управляемых соединений независимо друг от друга. В ходе настройки этих свойств с помощью редактора сборки WebSphere Integration Developer или административной консоли следует обратить внимание на согласованность значений свойств во избежание несогласованной маркировки записей протоколов и трассировки.</p>
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Дополнительные свойства драйвера JDBC (DriverConnectionProperties)

Содержит дополнительную информацию для подключения к базе данных с помощью драйвера JDBC.

Таблица 46. Дополнительные свойства соединения драйвера JDBC

Обязательное	Нет
Допустимые значения	Свойства подключения к базе данных задаются на уровне конкретных баз данных.
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	<p>Применяются совместно со свойствами UserName и Password для настройки соединения адаптера с базой данных.</p> <p>Укажите свойства в следующем формате: <i>имя:значения</i>. В качестве разделителя применяется точка с запятой (;).</p>
Пример	<p>Следующее значение этого свойства задаёт тайм-аут входа в систему:</p> <pre>loginTimeout:20; ConnectionRetryCount:5; ConnectionRetryDelay:5</pre>

Таблица 46. Дополнительные свойства соединения драйвера JDBC (продолжение)

Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Автоматическая фиксация (AutoCommit)

Это свойство указывает, установлено ли значение AutoCommit для соединения.

Таблица 47. Сведения об автоматической фиксации

Обязательное	Нет
Допустимые значения	True False
Значение по умолчанию	False
Тип свойства	Булевское
Формат	Это свойство игнорируется для транзакций XA (распределенных)
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Имя JNDI источника данных (DataSourceJNDIName)

Имя источника данных JNDI для установления соединения с базой данных.

Таблица 48. Имя JNDI источника данных - Сведения

Обязательное	Нет
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка

Таблица 48. Имя JNDI источника данных - Сведения (продолжение)

Формат	<p>Позволяет указать имя JNDI источника данных в WebSphere Process Server или WebSphere Enterprise Service Bus с информацией о подключении к целевой базе данных.</p> <p>Для повышения производительности входящих и исходящих операций укажите имя источника данных, поддерживающего кэширование подготовленных операторов.</p> <p>Если также заданы свойства UserName и Password, то они переопределяют имя пользователя и пароль, указанные на уровне источника данных.</p> <p>Свойства, отвечающие за подключение к базе данных, применяются в следующем порядке:</p> <ol style="list-style-type: none"> 1. Если задано свойство DataSourceJNDIName, то адаптер использует его для подключения к базе данных. Если также заданы свойства UserName и Password, то они переопределяют имя пользователя и пароль, указанные на уровне источника данных. 2. Если свойство DataSourceJNDIName не задано, то адаптер подключается с помощью свойства XADataSourceName (если оно задано). Свойство DataSourceJNDIName представляет источник данных XA или пула соединений. Если источник данных JNDI, указанный на сервере с поддержкой транзакций XA, задан в ходе настройки адаптера, то адаптер принимает участие в транзакциях XA. Кроме того, для поддержки транзакций XA можно указать свойство XADataSourceName. 3. Если свойства DataSourceJNDIName и XADataSourceName не указаны, то адаптер использует свойства DatabaseURL, JDBCDriverClass, UserName и Password для установления соединения. <p>Не следует путать свойство Имя JNDI источника данных с именем JNDI фабрики управляемых соединений или спецификации активации на сервере. Ниже рассмотрены важные различия между типами имен JNDI:</p> <ul style="list-style-type: none"> • Имя JNDI источника данных <ul style="list-style-type: none"> – Задаст соединение с базой данных – Применяется вместо сохранения идентификационных данных пользователя в свойствах адаптера – Сохраняется в свойствах адаптера • Имя JNDI фабрики управляемых соединений или спецификации активации <ul style="list-style-type: none"> – Соединение с фабрикой управляемых соединений или спецификацией активации на сервере – Применяется вместо указания значения каждого свойства фабрики управляемых соединений или спецификации активации в мастере – Сохраняется в качестве целевого сервера в файле импорта
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

URL базы данных (DatabaseURL)

Это свойство задает URL, специфичный для драйвера JDBC, для создания соединения с базой данных.

Таблица 49. URL базы данных - Сведения

Обязательное	<p>Да, если заданы следующие свойства:</p> <ul style="list-style-type: none"> • DataSourceJNDIName • XADataSourceName
--------------	---

Таблица 49. URL базы данных - Сведения (продолжение)

Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	<p>В окне ESW (мастер внешних служб) составьте URL базы данных, заполнив соответствующие поля. Например, URL базы данных Oracle состоит из ИД системы (SID), имени хоста, и номера порта базы данных. В административной консоли введите полный URL базы данных.</p> <p>Если сервер базы данных поддерживает IPv6, то имя хоста можно указать в формате IPv6.</p> <p>Свойства, отвечающие за подключение к базе данных, применяются в следующем порядке:</p> <ol style="list-style-type: none"> 1. Если задано свойство DataSourceJNDIName, то адаптер использует его для подключения к базе данных. Если также заданы свойства UserName и Password, то они переопределяют имя пользователя и пароль, указанные на уровне источника данных. 2. Если свойство DataSourceJNDIName не задано, то адаптер подключается с помощью свойства XADataSourceName (если оно задано). Свойство DataSourceJNDIName представляет источник данных XA или пула соединений. Если источник данных JNDI, указанный на сервере с поддержкой транзакций XA, задан в ходе настройки адаптера, то адаптер принимает участие в транзакциях XA. Кроме того, для поддержки транзакций XA можно указать свойство XADataSourceName. 3. Если свойства DataSourceJNDIName и XADataSourceName не указаны, то адаптер использует свойства DatabaseURL, JDBCDriverClass, UserName и Password для установления соединения. <p>Если в качестве имени хоста указывается адрес IPv6, то его следует заключить в квадратные скобки ([]).</p>
Примеры	<p>В следующем примере показаны типичные URL баз данных:</p> <pre>jdbc:oracle:thin:@9.26.248.148:1521:dev</pre>
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Производитель базы данных (DatabaseVendor)

Задаёт тип базы данных. Тип определяется в соответствии с производителем базы данных.

Таблица 50. Производитель базы данных - Сведения

Обязательное	Да
Допустимые значения	Oracle
Значение по умолчанию	ORACLE
Тип свойства	Строка
Формат	<p>Отдельные операторы SQL требуют специальной обработки, которая зависит от типа базы данных. Например, типы данных Struct и Array в Oracle требуют специальной обработки. Это свойство задаёт RDBMS, которая определяет тип базы данных.</p> <p>Укажите Oracle в качестве значения, соответствующего производителю базы данных.</p>
Локализуемое	Нет

Таблица 50. Производитель базы данных - Сведения (продолжение)

Поддерживает двунаправленный текст	Нет
------------------------------------	-----

Маскировать пользовательские данные как "XXX" в файлах протоколов и трассировки (HideConfidentialTrace) - Свойство

Позволяет заменять пользовательские данные в файлах протоколов и трассировки на символы X во избежание несанкционированного доступа к конфиденциальной информации.

Таблица 51. Маскировать пользовательские данные как "XXX" в файлах протоколов и трассировки - Сведения

Обязательное	Нет
Допустимые значения	True False
Значение по умолчанию	False
Тип свойства	Булевское
Применение	<p>Если указано значение True, то вместо пользовательских данных адаптер заносит в файлы протоколов и трассировки символы X.</p> <p>В случае обработки входящих событий значение этого свойства указывается на уровне адаптера ресурса. В случае обработки исходящих запросов оно может быть указано как на уровне адаптера ресурса, так и на уровне фабрики управляемых соединений. После настройки адаптера для обработки исходящих запросов с помощью ESW (мастер внешних служб) можно задать свойства адаптера ресурса и фабрики управляемых соединений независимо друг от друга. В ходе настройки этих свойств с помощью редактора сборки WebSphere Integration Developer или административной консоли следует обратить внимание на согласованность значений свойств во избежание несогласованной маркировки записей протоколов и трассировки.</p>
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Имя класса драйвера JDBC (JDBCDriverClass)

Имя класса драйвера JDBC, применяемого для подключения к базе данных.

Таблица 52. Имя класса драйвера JDBC - Сведения

Обязательное	Да, если не задано свойство DataSourceJNDIName
Допустимые значения	Значение, указанное в ходе настройки подключения к экземпляру базы данных.
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка

Таблица 52. Имя класса драйвера JDBC - Сведения (продолжение)

Формат	<p>В ходе работы с ESW (мастер внешних служб) класс драйвера JDBC указан, если выбрана общая комбинация базы данных и драйвера (например, драйверы Type 4 для последних версий Oracle). Для драйверов Type 2 необходимо указать имя класса базы данных вручную.</p> <p>Если выбран драйвер Type 2 или базовый драйвер, то необходимо ввести имя класса драйвера JDBC.</p> <p>В административной консоли введите имя драйвера.</p> <p>Если задано свойство DataSourceJNDIName, то это свойство игнорируется.</p>
Примеры	<p>Имя класса драйвера JDBC отображается в ESW (мастер внешних служб) и административной консоли. Ниже приведены примеры свойств класса драйвера JDBC в ESW (мастер внешних служб) и административной консоли. ESW (мастер внешних служб):</p> <ul style="list-style-type: none"> Для подключения к базе данных Oracle 10 с помощью драйвера Type 4 выберите Oracle Thin Driver. <p>Административная консоль:</p> <p>Драйвер Oracle Thin JDBC oracle.jdbc.driver.OracleDriver</p>
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Пароль (Password)

Пароль для имени пользователя базы данных.

Таблица 53. Пароль - Сведения

Обязательное	<p>Нет. Если задан псевдоним идентификации или свойство DataSourceJNDIName, то пароль указывать необязательно.</p> <p>Если заданы псевдоним идентификации, свойство DataSourceJNDIName и свойство Пароль, то преимуществом обладает свойство Пароль.</p>
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Значение этого свойства переопределяет пароль, указанный для источника данных на сервере с помощью псевдонима идентификации или свойства DataSourceJNDIName.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Тайм-аут запроса (QueryTimeout)

Максимальное время выполнения запросов SQL (в секундах).

Таблица 54. Тайм-аут запроса - Сведения

Обязательное	Нет
Значение по умолчанию	Значение по умолчанию отсутствует

Таблица 54. Тайм-аут запроса - Сведения (продолжение)

Единицы измерения	Секунды
Тип свойства	Integer
Формат	Если время выполнения запроса превышает значение тайм-аута, то база данных выбрасывает исключение SQL. Связанное сообщение заносится в файл протокола. Если значение не указано, то тайм-аут не отслеживается.
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Возвращать бизнес-объект при получении пустого набора результатов хранимой процедуры (ReturnDummyBOForSP)

Применяется для возврата выходных параметров, даже если набор результатов пустой.

Таблица 55. Возвращать бизнес-объект при получении пустого набора результатов хранимой процедуры - Сведения

Обязательное	Нет
Допустимые значения	True False
Значение по умолчанию	False
Тип свойства	Булевское
Формат	Операция RetrieveSP возвращает набор результатов. Если набор результатов пуст и в свойстве ReturnDummyBOForSP указано значение False, то бизнес-объекты не создаются и выходные параметры вызова процедуры будут недоступны для загрузки. Если в свойстве ReturnDummyBOForSP указано значение True, то создается фиктивный бизнес-объект, содержащий значения соответствующих атрибутов.
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Запрос SQL для проверки соединения (PingQuery)

Запрос SQL, используемый для проверки надежности соединения с базой данных.

Таблица 56. Запрос проверки связи - Сведения

Обязательное	Нет
Тип свойства	Строка
Значение по умолчанию	Значение по умолчанию отсутствует

Таблица 56. Запрос проверки связи - Сведения (продолжение)

Формат	Содержит запрос SQL, применяемый для проверки соединения с базой данных. Адаптер выполняет проверку связи каждый раз при получении исключительной ситуации SQLException в ходе выполнения исходящих операций. Адаптер не выполняет попыток восстановления соединения. Если база данных недоступна, то адаптер отправляет уведомление контейнеру. За восстановление соединений с базой данных отвечает администратор пула соединений.
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Имя пользователя (UserName)

Имя пользователя базы данных.

Таблица 57. Имя пользователя - Сведения

Обязательное	Нет. Если задан псевдоним идентификации или свойство DataSourceJNDIName, то имя пользователя указывать необязательно. Если заданы псевдоним идентификации, свойство DataSourceJNDIName и свойство Имя пользователя, то преимуществом обладает свойство Имя пользователя.
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Значение этого свойства переопределяет имя пользователя, указанное для источника данных на сервере с помощью псевдонима идентификации или свойства DataSourceJNDIName.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Имя источника данных XA (XADataSourceName)

Это свойство задает имя источника данных XA, используемое для установки соединения с базой данных для транзакций XA (распределенных)

Таблица 58. Сведения об имени источника данных XA

Обязательное	Нет
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Это свойство используется для установки соединения XA с базой данных Oracle. Если задано свойство DataSourceJNDIName, то это свойство игнорируется.
Примеры	Обычное значение для базы данных Oracle: oracle.jdbc.xa.client.OracleXADataSource
Локализуемое	Нет

Таблица 58. Сведения об имени источника данных XA (продолжение)

Поддерживает двунаправленный текст	Нет
------------------------------------	-----

Свойства спецификации взаимодействия

Свойства спецификации взаимодействия, или InteractionSpec, управляют взаимодействием операции. ESW (мастер внешних служб) задает свойства спецификации взаимодействия в ходе настройки адаптера. Обычно изменять эти свойства не требуется. Однако пользователь может изменить некоторые свойства исходящих операций. Например, можно увеличить значение свойства указания взаимодействия, задающее максимальное число записей для возврата операций RetrieveAll если операции RetrieveAll не возвращают полные сведения. Для изменения этих свойств после развертывания приложения используйте редактор компоновки в WebSphere Integration Developer. Свойства сохраняются в связывании методов точки импорта.

Табл. 59 перечисляет и описывает установленное свойство спецификации взаимодействия. Информация о том, как читать таблицы сведений свойств в следующих разделах, находится в разделе “Руководство по описанию свойств” на стр. 165.

Таблица 59. Свойство спецификации взаимодействия для Adapter for Oracle E-Business Suite

Имя свойства	Описание
“Максимальное число записей для операции RetrieveAll”	Максимальное число наборов результатов для возвращения при выполнении операции RetrieveAll

Максимальное число записей для операции RetrieveAll

Это свойство задает максимальное число возвращаемых записей для операции RetrieveAll.

Таблица 60. Сведения о максимальном числе записей для операции RetrieveAll

Обязательное	Да
Значение по умолчанию	100
Формат	Если количество соответствий в базе данных превышает значение этого свойства, то адаптер выбрасывает исключение MatchesExceededLimitException и неполадку MatchesExceededLimitFault. Если операция RetrieveAll не возвращает всех записей, увеличьте это значение. Если недостаточно памяти, уменьшите это значение.
Тип свойства	Целое число
Глобализованное	Нет
Поддерживает двунаправленный текст	Нет

Свойства конфигурации входящего соединения

WebSphere Adapter for Oracle E-Business Suite поддерживает несколько категорий свойств входящего соединения, которые настраиваются с помощью ESW (мастер внешних служб) при создании объектов и служб. Можно изменить свойства спецификации активации и адаптера ресурсов после развертывания модуля с помощью WebSphere Integration Developer или AdminConsole, но свойства соединения для ESW (мастер внешних служб) невозможно изменить после развертывания.

Руководство по описанию свойств

Свойства, используемые для настройки WebSphere Adapter for Oracle E-Business Suite, подробно описаны в таблицах, которые включены во все разделы свойств конфигурации, таких как свойства адаптера ресурса, свойства фабрики управляемых соединений и т.д. Для того чтобы помочь вам использовать эти таблицы, здесь приведена информация о каждой строке.

Следующая таблица объясняет значение каждой строки, которая может быть показана в таблице для свойства конфигурации.

Строка	Описание
Обязательное	<p>Для обеспечения работы адаптера в обязательном поле (свойстве) должно быть указано значение. В некоторых случаях для обязательных полей ESW (мастер внешних служб) предусмотрены значения по умолчанию.</p> <p>Удаление значения по умолчанию из обязательного поля в ESW (мастер внешних служб) <i>не изменит значения по умолчанию</i>. Когда обязательное поле не содержит никакого значения, ESW (мастер внешних служб) обработает поле с помощью присвоенного значения по умолчанию, и это значение по умолчанию будет также показано в административной консоли.</p> <p>Возможные значения: Да и Нет.</p> <p>Кроме того, свойство может быть обязательным только в том случае, если в другом свойстве указано конкретное значение. Такие зависимости указываются в таблице. Например,</p> <ul style="list-style-type: none">• Да, если для свойства EventQueryType указано значение Dynamic• Для баз данных Oracle
Допустимые значения	Перечислены возможные значения свойства.
Значение по умолчанию	<p>Предопределенное значение, устанавливаемое ESW (мастер внешних служб). Для обязательного свойства следует принять значение по умолчанию или указать собственное значение. Если значение по умолчанию не предусмотрено, то в таблице будет указано Значение по умолчанию отсутствует.</p> <p>Слово Нет допустимо в качестве значения по умолчанию; оно не указывает на отсутствие значения по умолчанию.</p>
Единицы измерения	Задаёт единицы измерения свойства, например, килобайты или секунды.
Тип свойства	<p>Описывает тип свойства. Поддерживаются следующие типы свойств:</p> <ul style="list-style-type: none">• Boolean• String• Integer

Строка	Описание
Применение	<p>Описывает особенности применения свойства. Например, ниже приведен пример описания свойства в документации:</p> <p>Для Rational Application Developer for WebSphere Software 6.40 и более ранних версий:</p> <ul style="list-style-type: none"> • Пароль должен содержать только прописные буквы • Длина пароля должна составлять 8 символов <p>Для Rational Application Developer for WebSphere Software после версии 6.40:</p> <ul style="list-style-type: none"> • В пароле не учитывается регистр символов • Пароль может содержать до 40 символов. <p>В этом разделе перечислены связанные свойства и рассмотрены типы условных взаимосвязей.</p>
Пример	<p>Содержит примеры значений свойства, например:</p> <p>"Например, если для свойства Язык указано значение JA (японский), то то для свойства Номер кодовой страницы необходимо указать значение 8000".</p>
Локализуемое	<p>Глобализованные свойства поддерживают национальные языки, т.е. значения можно указывать на национальных языках.</p> <p>Возможные значения: Да и Нет.</p>
Поддерживает двунаправленный текст	<p>Указывает, поддерживается ли свойство в двунаправленной обработке (bidi). Двунаправленная обработка работает с данными, содержащими как текст, написанный слева направо (URL или путь к файлу), так и справа налево (арабский или иврит) в одном и том же файле.</p> <p>Возможные значения: Да и Нет.</p>

Свойства соединения для мастера

Свойства соединения с внешней службой используются для установки соединения между ESW (мастер внешних служб), средством создания бизнес-объектов, и базой данных.

Свойства внешней службы указывают такие данные, как конфигурация соединения, свойства двунаправленного преобразования и опции ведения протоколов для мастера. После установки соединения мастер может найти в базе данных метаданные, которые ему необходимы для создания бизнес-объектов. Некоторые из свойств, предоставленных мастеру для поиска объектов в базе данных, используются как начальные значения для свойств среды выполнения, которые задаются позже в мастере. Они включают в себя свойства адаптера ресурса, фабрики управляемых соединений и спецификации активации.

Свойства соединения для ESW (мастер внешних служб) описаны в следующей таблице. Более подробное описание каждого свойства представлено в разделе, который следует за таблицей. Информация о том, как читать таблицу свойств, находится в разделе "Руководство по описанию свойств" на стр. 165.

Таблица 61. Свойства соединения для ESW (мастер внешних служб)

Имя свойства в мастере	Описание
Дополнительные свойства соединения драйвера JDBC	Дополнительные свойства для подключения к базе данных с помощью драйвера JDBC, применяемые совместно со свойствами UserName и Password.
"База данных" на стр. 189	Имя и версия программного обеспечения управления базой данных, к которой адаптер будет получать доступ

Таблица 61. Свойства соединения для ESW (мастер внешних служб) (продолжение)

Имя свойства в мастере	Описание
URL базы данных	URL базы данных, применяемый для подключения к базе данных
“Имя хоста” на стр. 190	Имя хоста или IP-адрес сервера базы данных
Имя класса драйвера JDBC	Имя класса драйвера JDBC
“Тип драйвера JDBC” на стр. 191	Тип используемого драйвера JDBC
Пароль	Пароль для соответствующего имени пользователя
“Номер порта” на стр. 192	Номер порта для подключения к экземпляру базы данных
“Префикс имен бизнес-объектов” на стр. 192	Префикс, добавляемый к имени бизнес-объекта
ИД системы	Имя ИД системы (SID) базы данных
Имя пользователя	Имя пользователя базы данных для подключения к ней

ESW (мастер внешних служб) использует свойства двунаправленного соединения для применения правильного двунаправленного преобразования данных, передаваемых в информационную систему предприятия.

Дополнительные свойства соединения драйвера JDBC

Содержит дополнительную информацию для подключения к базе данных с помощью драйвера JDBC.

Таблица 62. Дополнительные свойства соединения драйвера JDBC

Обязательное	Нет
Допустимые значения	Свойства подключения к базе данных задаются на уровне конкретных баз данных.
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Применяются совместно со свойствами UserName и Password для настройки соединения адаптера с базой данных. Укажите свойства в следующем формате: <i>имя:значения</i> . В качестве разделителя применяется точка с запятой (;).
Пример	Следующее значение этого свойства задает тайм-аут входа в систему: loginTimeout:20; ConnectionRetryCount:5; ConnectionRetryDelay:5
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

База данных

Это свойство указывает программное обеспечение, которое управляет базой данных, к которой адаптер получает доступ.

Таблица 63. Сведения о программном обеспечении базы данных

Строка	Описание
Обязательное	Да

Таблица 63. Сведения о программном обеспечении базы данных (продолжение)

Строка	Описание
Допустимые значения	Программное обеспечение базы данных Oracle по имени и номеру версии.
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	ESW (мастер внешних служб) использует значение этого свойства для установки значений по умолчанию и генерации списков опций, специфичных для этой базы данных, для других свойств. Например, если выбрано Oracle 10, поле класса драйвера JDBC в мастере показывает только те драйверы JDBC, которые поддерживаются этой версией базы данных Oracle.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

URL базы данных

Это свойство задает URL, специфичный для драйвера JDBC, для создания соединения с базой данных.

Таблица 64. Сведения об URL базы данных

Обязательное	Да
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Это значение специфично для программного обеспечения базы данных и используемого драйвера JDBC. Если ваш сервер базы данных поддерживает IPv6, можно указать часть имени хоста URL базы данных в формате IPv6. Заключите IP-адрес в квадратные скобки ([]).
Примеры	Ниже приведены типичные значения серверов базы данных Oracle. jdbc:oracle:thin:@9.26.248.148:1521:dev
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Имя хоста

Это свойство задает имя хоста или IP-адрес сервера базы данных.

Таблица 65. Имя хоста - Сведения

Строка	Описание
Обязательное	Да
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка

Таблица 65. Имя хоста - Сведения (продолжение)

Строка	Описание
Формат	Это имя хоста или IP-адрес сервера базы данных. Если сервер базы данных поддерживает IPv6, можно указать имя хоста в формате IPv6.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Имя класса драйвера JDBC

Это свойство задает имя класса драйвера JDBC.

Таблица 66. Сведения об имени класса драйвера JDBC

Строка	Описание
Обязательное	Да
Допустимые значения	Возможные значения зависят от типа и версии базы данных. Мастер показывает список известных драйверов.
Значение по умолчанию	Значение по умолчанию зависит от типа и версии базы данных.
Тип свойства	Строка
Формат	Хотя мастер показывает имя класса по умолчанию для выбранного типа драйвера JDBC, можно при необходимости ввести имя другого класса. Если выбрано Другое для значения драйвера JDBC, то значение по умолчанию не предоставляется, и необходимо ввести имя класса. Имя класса должно содержаться в файлах драйвера JDBC, предоставленных при запуске мастера.
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Тип драйвера JDBC

Это свойство задает тип используемого драйвера JDBC.

Таблица 67. Сведения о типе драйвера JDBC

Строка	Описание
Обязательное	Да
Допустимые значения	Возможные значения зависят от типа и версии базы данных. Мастер показывает список известных драйверов.
Значение по умолчанию	Значение по умолчанию зависит от типа и версии базы данных.
Тип свойства	Строка
Формат	Хотя в действительности вы выбираете или тип 2, или тип 4 (универсальный) для драйвера, каждая система базы данных имеет свое собственное имя для драйвера. Мастер показывает список драйверов, известных для каждой системы базы данных. Если ваш драйвер в нем не указан, выберите Другой. Информация в этом поле должна быть согласована с файлами драйвера JDBC, предоставленными при запуске мастера.
Локализуемое	Да

Таблица 67. Сведения о типе драйвера JDBC (продолжение)

Строка	Описание
Поддерживает двунаправленный текст	Нет

Пароль (Пароль)

Это свойство задает пароль для имени пользователя базы данных.

Таблица 68. Пароль - Сведения

Строка	Описание
Обязательное	Да
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Пароль связан с именем пользователя, введенным для подключения к базе данных с целью поиска объектов.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Номер порта

Это свойство задает номер порта для экземпляра базы данных.

Таблица 69. Сведения о номере порта

Обязательное	Да
Значение по умолчанию	Значение по умолчанию специфично для базы данных и инициализируется мастером, если выбран именованный драйвер в поле Тип драйвера JDBC . Если для типа драйвера выбрано Другой, то значение по умолчанию не предоставляется.
Тип свойства	Строка
Формат	Это номер порта для подключения к экземпляру базы данных. Это свойство не доступно, если для типа драйвера JDBC выбрано значение Другой.
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Префикс имен бизнес-объектов

Префикс, добавляемый к имени бизнес-объекта.

Таблица 70. Сведения о префиксе

Обязательное	Нет
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка

Таблица 70. Сведения о префиксе (продолжение)

Формат	Используйте префикс для удобства различия между типами бизнес-объектов.
Пример	Можно указать префикс <code>Org1e</code> для общих бизнес-объектов и <code>%AppName%</code> для бизнес-объектов, специфичных для приложения.
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

ИД системы

Это свойство задает имя ИД системы (SID) базы данных.

Таблица 71. Сведения об ИД системы

Обязательное	Да
Значение по умолчанию	Значение по умолчанию специфично для базы данных.
Тип свойства	Строка
Формат	ИД системы (SID) - это имя базы данных.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Имя пользователя (UserName)

Имя пользователя используется для подключения к базе данных.

Таблица 72. Имя пользователя - Сведения

Обязательное	Да
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Имя пользователя вводится для подключения к базе данных с целью поиска объектов.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Свойства адаптера ресурсов

Свойства адаптера ресурсов управляют общей работой адаптера, например, указывая пространство имен для бизнес-объектов. Свойства адаптера ресурсов устанавливаются с помощью ESW (мастер внешних служб) при настройке адаптера. После развертывания адаптера используйте административную консоль для изменения этих свойств.

Следующая таблица перечисляет свойства адаптера ресурсов и их назначение. Полное описание каждого свойства предоставлено в разделе, который следует за

таблицей. Информация о том, как читать таблицу свойств, находится в разделе 1 “Руководство по описанию свойств” на стр. 165.

Таблица 73. Свойства адаптера ресурсов для Adapter for Oracle E-Business Suite

Имя		Описание
В мастере	В административной консоли	
ИД адаптера	AdapterID	Это свойство задает экземпляр адаптера для обработки событий PMI, а также для ведения протоколов и трассировки.
Производитель базы данных	DatabaseVendor	Тип базы данных, применяемой адаптеров для специальной обработки.
Скрыть пользовательские данные в виде "XXX" в файлах протокола и трассировки	HideConfidentialTrace	Позволяет вместо пользовательских данных заносить в файлы протоколов и трассировки символы X.
Тайм-аут запроса	QueryTimeout	Максимальное время выполнения запросов SQL (в секундах)
Возвращать бизнес-объект, даже когда набор результатов хранимой процедура пуст	ReturnDummyBOForSP	Применяется для возврата выходных параметров, даже если набор результатов пустой.
(Не доступно)	enableHASupport	Не используйте это свойство.
(Не доступно)	LogFileMaxSize	Устаревшее
(Не доступно)	LogFilename	Устаревшее
(Не доступно)	LogNumberOfFiles	Устаревшее
Запрос SQL для проверки соединения	PingQuery	Запрос SQL, используемый для проверки надежности соединения с базой данных.
(Не доступно)	TraceFileMaxSize	Устаревшее
(Не доступно)	TraceFileName	Устаревшее
(Не доступно)	TraceNumberOfFiles	Устаревшее

ИД адаптера (AdapterID)

Задает конкретный экземпляр адаптера.

Таблица 74. ИД адаптера - Сведения

Обязательное	Да
Значение по умолчанию	001
Тип свойства	String

Таблица 74. ИД адаптера - Сведения (продолжение)

Применение	<p>Задает экземпляр адаптера в файлах протоколов и трассировки, а также помогает идентифицировать экземпляр адаптера в ходе мониторинга адаптеров. ИД адаптера вместе с идентификатором уровня адаптера (OEBSRA) образуют имя компонента, применяемое инструментом Анализатор протоколов и данных трассировки. Например, если свойство ИД адаптера содержит значение 001, то ИД компонента равен OEBSRA001.</p> <p>В случае применения нескольких экземпляров адаптера первые семь символов ИД адаптера должны быть уникальными для каждого экземпляра, для того чтобы информацию из файлов протоколов и трассировки можно было сопоставить с конкретным экземпляром адаптера. Такой подход также обеспечивает уникальность ИД компонентов экземпляров адаптера.</p> <p>Для того чтобы показать, каким образом длина ИД адаптера влияет на фильтрацию файлов протоколов и трассировки, предположим, что для двух экземпляров WebSphere Adapter for Oracle E-Business Suite в свойстве ИД адаптера указаны значения 001 и 002. ИД компонентов этих экземпляров (OEBSRA001 и OEBSRA002) достаточно коротки, чтобы оставаться уникальными, т.е. они позволяют отличать экземпляры друг от друга. Однако экземпляры с более длинными ИД адаптера нельзя отличить друг от друга. Например, предположим, что для двух экземпляров в свойстве ИД адаптера указаны значения Instance01 и Instance02. Пользователь не сможет отдельно проверить информацию из файлов протоколов и трассировки для каждого экземпляра адаптера, поскольку ИД компонента усекается до OEBSRAInstanc.</p> <p>В случае обработки входящих событий значение этого свойства указывается на уровне адаптера ресурса. В случае обработки исходящих запросов оно может быть указано как на уровне адаптера ресурса, так и на уровне фабрики управляемых соединений. После настройки адаптера для обработки исходящих запросов с помощью ESW (мастер внешних служб) можно задать свойства адаптера ресурса и фабрики управляемых соединений независимо друг от друга. В ходе настройки этих свойств с помощью редактора сборки WebSphere Integration Developer или административной консоли следует обратить внимание на согласованность значений свойств во избежание несогласованной маркировки записей протоколов и трассировки.</p>
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Производитель базы данных (DatabaseVendor)

Задает тип базы данных. Тип определяется в соответствии с производителем базы данных.

Таблица 75. Производитель базы данных - Сведения

Обязательное	Да
Допустимые значения	Oracle
Значение по умолчанию	ORACLE
Тип свойства	Строка
Формат	<p>Отдельные операторы SQL требуют специальной обработки, которая зависит от типа базы данных. Например, типы данных Struct и Array в Oracle требуют специальной обработки. Это свойство задает RDBMS, которая определяет тип базы данных.</p> <p>Укажите Oracle в качестве значения, соответствующего производителю базы данных.</p>
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Маскировать пользовательские данные как "XXX" в файлах протоколов и трассировки (HideConfidentialTrace) - Свойство

Позволяет заменять пользовательские данные в файлах протоколов и трассировки на символы X во избежание несанкционированного доступа к конфиденциальной информации.

Таблица 76. Маскировать пользовательские данные как "XXX" в файлах протоколов и трассировки - Сведения

Обязательное	Нет
Допустимые значения	True False
Значение по умолчанию	False
Тип свойства	Булевское
Применение	Если указано значение True, то вместо пользовательских данных адаптер заносит в файлы протоколов и трассировки символы X. В случае обработки входящих событий значение этого свойства указывается на уровне адаптера ресурса. В случае обработки исходящих запросов оно может быть указано как на уровне адаптера ресурса, так и на уровне фабрики управляемых соединений. После настройки адаптера для обработки исходящих запросов с помощью ESW (мастер внешних служб) можно задать свойства адаптера ресурса и фабрики управляемых соединений независимо друг от друга. В ходе настройки этих свойств с помощью редактора сборки WebSphere Integration Developer или административной консоли следует обратить внимание на согласованность значений свойств во избежание несогласованной маркировки записей протоколов и трассировки.
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Включить поддержку высокой готовности (enableHASupport)

Не используйте это свойство. Его значение должно быть равно true.

Тайм-аут запроса (QueryTimeOut)

Максимальное время выполнения запросов SQL (в секундах).

Таблица 77. Тайм-аут запроса - Сведения

Обязательное	Нет
Значение по умолчанию	Значение по умолчанию отсутствует
Единицы измерения	Секунды
Тип свойства	Integer
Формат	Если время выполнения запроса превышает значение тайм-аута, то база данных выбрасывает исключение SQL. Связанное сообщение заносится в файл протокола. Если значение не указано, то тайм-аут не отслеживается.
Локализуемое	Да

Таблица 77. Тайм-аут запроса - Сведения (продолжение)

Поддерживает двунаправленный текст	Нет
------------------------------------	-----

Возвращать бизнес-объект при получении пустого набора результатов хранимой процедуры (ReturnDummyBOForSP)

Применяется для возврата выходных параметров, даже если набор результатов пустой.

Таблица 78. Возвращать бизнес-объект при получении пустого набора результатов хранимой процедуры - Сведения

Обязательное	Нет
Допустимые значения	True False
Значение по умолчанию	False
Тип свойства	Булевское
Формат	Операция RetrieveSP возвращает набор результатов. Если набор результатов пуст и в свойстве ReturnDummyBOForSP указано значение False, то бизнес-объекты не создаются и выходные параметры вызова процедуры будут недоступны для загрузки. Если в свойстве ReturnDummyBOForSP указано значение True, то создается фиктивный бизнес-объект, содержащий значения соответствующих атрибутов.
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Запрос SQL для проверки соединения (PingQuery)

Запрос SQL, используемый для проверки надежности соединения с базой данных.

Таблица 79. Запрос проверки связи - Сведения

Обязательное	Нет
Тип свойства	Строка
Значение по умолчанию	Значение по умолчанию отсутствует
Формат	Содержит запрос SQL, применяемый для проверки соединения с базой данных. Адаптер выполняет проверку связи каждый раз при получении исключительной ситуации SQLExeption в ходе выполнения исходящих операций. Адаптер не выполняет попыток восстановления соединения. Если база данных недоступна, то адаптер отправляет уведомление контейнеру. За восстановление соединений с базой данных отвечает администратор пула соединений.
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Свойства спецификации активации

Свойства спецификации активации - это свойства, в которых хранится информация о конфигурации обработки входящего события для экспорта.

Установить свойства спецификации активации можно с помощью ESW (мастер внешних служб) во время конфигурации адаптера, а затем их можно изменить с помощью редактора сборки WebSphere Integration Developer или после развертывания с помощью административной консоли WebSphere Process Server или WebSphere Enterprise Service Bus.

В следующей таблице перечислены и описаны свойства спецификации активации. Полное описание каждого свойства предоставлено в разделе, который следует за таблицей. Информация о том, как читать таблицу свойств, находится в разделе “Руководство по описанию свойств” на стр. 165.

Таблица 80. Свойства спецификации активации для Adapter for Oracle E-Business Suite

Имя свойства		Описание
В мастере	В административной консоли	
Экземпляр адаптера для фильтрации событий	AdapterInstanceEventFilter	Идентификатор, управляющий обработкой событий в хранилище событий.
Дополнительные свойства соединения драйвера JDBC	DriverConnectionProperties	Дополнительные свойства для подключения к базе данных с помощью драйвера JDBC, применяемые совместно со свойствами UserName и Password.
Пространство имен бизнес-объекта	BusinessObjectNameSpace	Пространство имен для определений бизнес-объектов
Пользовательский запрос удаления	CustomDeleteQuery	Имя запроса, хранимой процедуры или хранимой функции, выполняемой после обработки каждого события с целью удаления ненужных записей.
Пользовательский запрос событий	CustomEventQuery	Имя запроса, хранимой процедуры или хранимой функции, выполняющей опрос событий.
Пользовательский запрос обновления	CustomUpdateQuery	Имя запроса, хранимой процедуры или хранимой функции, выполняемой после обработки каждого события с целью предотвращения повторной загрузки события в следующем цикле обработки событий.
Имя JNDI источника данных	DataSourceJNDIName	Имя источника данных JNDI для установления соединения с базой данных.
URL базы данных	DatabaseURL	URL базы данных, применяемый для подключения к базе данных
Производитель базы данных	DatabaseVendor	Тип базы данных, применяемой адаптеров для специальной обработки.
Не обрабатывать события, у которых установлено будущее системное время	FilterFutureEvents	Позволяет разрешить или запретить фильтрацию будущих событий путем сравнения системного времени каждого события с временем системы.
Обеспечить доставку события только один раз	AssuredOnceDelivery	Указывает, обеспечивает ли адаптер гарантированную однократную доставку событий.
Упорядочить события по	EventOrderBy	Порядок, в котором события извлекаются и обрабатываются
Тип запроса события	EventQueryType	Определяет, должно ли быть использовано стандартное хранилище событий или пользовательский запрос
Имя таблицы событий	EventTableName	Имя таблицы базы данных, которая содержит события, сгенерированные базой данных для входящей обработки

Таблица 80. Свойства спецификации активации для Adapter for Oracle E-Business Suite (продолжение)

Имя свойства		Описание
В мастере	В административной консоли	
Типы обрабатываемых событий	EventTypeFilter	Список с ограничителями, в котором указаны доставляемые адаптером события.
Количество повторов для недоставленных событий (FailedEventRetryLimit)	FailedEventRetryLimit	Указывает, сколько раз адаптер будет повторять попытку доставить событие, если это сделать не удалось.
Интервал между периодами опроса	PollPeriod	Время, в течение которого адаптер ожидает между периодами опроса.
Класс драйвера JDBC	JDBCDriverClass	Имя класса драйвера JDBC, применяемого для подключения к базе данных.
Максимальное число соединений	MaximumConnections	Ограничение числа соединений, доступных адаптеру для доставки входящих событий.
Минимальное число соединений	MinimumConnections	Минимальное число соединений, доступных адаптеру для доставки входящих событий.
Число попыток восстановить соединение с системой	RetryLimit	Число попыток восстановления входящего соединения после ошибки.
Пароль	Пароль	Пароль, разрешающий пользователю получать события из базы данных.
Количество опросов	PollQuantity	Число событий, доставляемых адаптером в ходе каждого периода опроса.
Тайм-аут запроса	QueryTimeOut	Максимальное время выполнения запросов SQL (в секундах)
Восстанавливать соединение EIS при запуске	RetryConnectionOnStartup	Позволяет разрешить или запретить восстановление соединения с Oracle DB, если оно не было установлено в ходе запуска.
Интервал повтора в случае сбоя соединения	RetryInterval	Время ожидания перед попыткой повторного подключения после ошибки входящей операции.
Фиктивный бизнес-объект возврата для RetrieveSP	ReturnDummyBOForSP	Применяется для возврата выходных параметров, даже если набор результатов пустой.
Запрос SQL для проверки соединения	PingQuery	Запрос SQL, используемый для проверки надежности соединения с базой данных.
Остановить адаптер при возникновении ошибки во время опроса	StopPollingOnError	Позволяет остановить опрос событий после ошибки в ходе опроса.
Хранимая процедура для выполнения после опроса	SPAAfterPoll	Имя хранимой процедуры, которая должна выполняться после каждого цикла опроса
Хранимая процедура для выполнения перед опросом	SPBeforePoll	Имя хранимой процедуры, которая должна выполняться перед фактическим вызовом опроса
Тип доставки	DeliveryType	Задаёт порядок доставки событий в ходе экспорта.
Имя пользователя	Имя пользователя	Имя пользователя базы данных, используемое для входящих событий

Экземпляр адаптера для фильтрации событий (AdapterInstanceEventFilter)

Позволяет разрешить экземпляру адаптера обрабатывать конкретные события в хранилище событий.

Таблица 81. Экземпляр адаптера для фильтрации событий - Сведения

Обязательное	Нет
Значение по умолчанию	null
Тип свойства	String
Применение	<p>Это свойство помогает выполнить миграцию WebSphere Business Integration Adapter for Oracle Applications до WebSphere Adapter for Oracle E-Business Suite. WebSphere Business Integration Adapter for Oracle Applications поддерживает распределение нагрузки за счет выделения нескольких экземпляров адаптера для обработки событий одного и того же типа. Если распределение нагрузки не требуется, то все события конкретного типа обрабатываются одним адаптером. Это свойство призвано обеспечить незаметную миграцию между WBIA и JCA в случае применения фильтрации connectorID.</p> <p>Как правило, WebSphere Adapter for Oracle E-Business Suite не использует этот способ распределения нагрузки, однако поддерживает его для миграции без изменения триггеров базы данных и других механизмов, отвечающих за регистрацию событий в хранилище событий.</p> <p>Свойство AdapterInstanceEventFilter property соответствует свойству ConnectorID WebSphere Business Integration Adapter for Oracle Applications.</p> <p>Для применения этой функции триггеры или другие механизмы, отвечающие за создание событий в хранилище событий, должны указывать подходящее значение в столбце connectorId.</p> <p>В разделе Табл. 82 рассмотрено взаимодействие между свойством AdapterInstanceEventFilter и значением столбца connectorId в хранилище событий.</p> <p>Если заданы оба свойства EventTypeFilter и AdapterInstanceEventFilter, то адаптер обрабатывает только те события, которые соответствуют обоим критериям. Таким образом, обрабатываются только те события, тип которых совпадает с указанным в свойстве EventTypeFilter, и столбец connectorId которых совпадает с указанным в свойстве AdapterInstanceEventFilter.</p>
Пример	См. раздел Табл. 82.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Таблица 82. Взаимодействие свойства AdapterInstanceEventFilter со столбцом connectorId в хранилище событий

Свойство AdapterInstanceEventFilter	Столбец connectorId события	Результат
null	null	Адаптер обрабатывает событие.
null	Instance1	Адаптер обрабатывает событие, поскольку столбец connectorId не выбран.
Instance1	Instance1	Адаптер обрабатывает событие.
Instance1	Instance2	Адаптер не обрабатывает событие, поскольку ИД экземпляров не совпадают.
Instance1	null	Адаптер не обрабатывает событие, поскольку ИД экземпляров не совпадают.

Дополнительные свойства драйвера JDBC (DriverConnectionProperties)

Содержит дополнительную информацию для подключения к базе данных с помощью драйвера JDBC.

Таблица 83. Дополнительные свойства соединения драйвера JDBC

Обязательное	Нет
Допустимые значения	Свойства подключения к базе данных задаются на уровне конкретных баз данных.
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Применяются совместно со свойствами UserName и Password для настройки соединения адаптера с базой данных. Укажите свойства в следующем формате: <i>имя:значения</i> . В качестве разделителя применяется точка с запятой (;).
Пример	Следующее значение этого свойства задает тайм-аут входа в систему: loginTimeout:20; ConnectionRetryCount:5; ConnectionRetryDelay:5
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Пространство имен бизнес-объекта (BusinessObjectNameSpace)

Это свойство указывает пространство имен для определений бизнес-объектов.

Таблица 84. Параметры свойств пространства имен бизнес-объекта

Обязательное	Нет
Значение по умолчанию	http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle
Тип свойства	Строка
Формат	Это значение добавляется как префикс к имени бизнес-объекта для хранения логически разделенных имен бизнес-объектов.
Пример	Следующий пример показывает бизнес-объект Schema1Customer с пространством имен по умолчанию: http://www.ibm.com/xmlns/prod/websphere/j2ca/oracle/Schema1Customer
Поддерживает двунаправленный текст	Нет

Пользовательский запрос удаления (CustomDeleteQuery)

Используйте это свойство, чтобы указать оператор SQL, хранимую процедуру или функцию, которые должны выполняться после обработки каждого события для удаления записей, которые могут быть удалены после доставки события.

Таблица 85. Сведения о пользовательском запросе на удаление

Обязательное	Нет
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Используйте это свойство, чтобы указать оператор SQL, хранимую процедуру или функцию, которые должны выполняться, когда свойство EventQueryType имеет значение Dynamic.
Локализуемое	Да

Таблица 85. Сведения о пользовательском запросе на удаление (продолжение)

Поддерживает двунаправленный текст	Да
------------------------------------	----

Пользовательский запрос событий (CustomEventQuery)

Используйте это свойство, чтобы указать оператор SQL, хранимую процедуру или функцию, которые должны выполняться для опроса событий в пользовательской обработке события.

Таблица 86. Сведения о пользовательском запросе события

Обязательное	Нет
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Используйте это свойство, чтобы указать оператор SQL, хранимую процедуру или функцию, которые должны выполняться в процессе каждого цикла опроса, когда свойство EventQueryType имеет значение Dynamic.
Примеры	<p>В следующем примере пользовательский запрос события выполняется как оператор SQL, который возвращает ID события, ключ объекта и имя объекта каждой записи в хранилище событий MY_EVENT_TABLE, столбец состояния которой имеет значение 0:</p> <pre>select event_id, object_key, object_name from MY_EVENT_TABLE where status = 0</pre> <p>В следующем примере возвращаемые записи событий ограничены значением свойства PollQuantity:</p> <pre>select event_id, object_key, object_name from MY_EVENT_TABLE where status = 0 and rownum < POLL QUANTITY</pre> <p>В следующем примере выполняется хранимая процедура с двумя параметрами:</p> <pre>CALL MY_EVENT_STORED_PROC (?,?)</pre> <p>В следующем примере выполняется хранимая функция с одним параметром и одним значением возврата:</p> <pre>? = CALL MY_EVENT_FUNCTION(?)</pre>
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Пользовательский запрос обновления (CustomUpdateQuery)

Используйте это свойство, чтобы указать оператор SQL, хранимую процедуру или функцию, которые должны выполняться после обработки каждого события, чтобы одинаковые события не собирались для обработки в последующем цикле события.

Таблица 87. Сведения о пользовательском запросе на изменение

Обязательное	Нет
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка

Таблица 87. Сведения о пользовательском запросе на изменение (продолжение)

Формат	Используйте это свойство, чтобы указать оператор SQL, хранимую процедуру или функцию, которые должны выполняться, когда свойство EventQueryType имеет значение Dynamic.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Имя JNDI источника данных (DataSourceJNDIName)

Имя источника данных JNDI для установления соединения с базой данных.

Таблица 88. Имя JNDI источника данных - Сведения

Обязательное	Нет
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка

Таблица 88. Имя JNDI источника данных - Сведения (продолжение)

Формат	<p>Позволяет указать имя JNDI источника данных в WebSphere Process Server или WebSphere Enterprise Service Bus с информацией о подключении к целевой базе данных.</p> <p>Для повышения производительности входящих и исходящих операций укажите имя источника данных, поддерживающего кэширование подготовленных операторов.</p> <p>Если также заданы свойства UserName и Password, то они переопределяют имя пользователя и пароль, указанные на уровне источника данных.</p> <p>Свойства, отвечающие за подключение к базе данных, применяются в следующем порядке:</p> <ol style="list-style-type: none"> 1. Если задано свойство DataSourceJNDIName, то адаптер использует его для подключения к базе данных. Если также заданы свойства UserName и Password, то они переопределяют имя пользователя и пароль, указанные на уровне источника данных. 2. Если свойство DataSourceJNDIName не задано, то адаптер подключается с помощью свойства XADataSourceName (если оно задано). Свойство DataSourceJNDIName представляет источник данных XA или пула соединений. Если источник данных JNDI, указанный на сервере с поддержкой транзакций XA, задан в ходе настройки адаптера, то адаптер принимает участие в транзакциях XA. Кроме того, для поддержки транзакций XA можно указать свойство XADataSourceName. 3. Если свойства DataSourceJNDIName и XADataSourceName не указаны, то адаптер использует свойства DatabaseURL, JDBCDriverClass, UserName и Password для установления соединения. <p>Не следует путать свойство Имя JNDI источника данных с именем JNDI фабрики управляемых соединений или спецификации активации на сервере. Ниже рассмотрены важные различия между типами имен JNDI:</p> <ul style="list-style-type: none"> • Имя JNDI источника данных <ul style="list-style-type: none"> – Задаст соединение с базой данных – Применяется вместо сохранения идентификационных данных пользователя в свойствах адаптера – Сохраняется в свойствах адаптера • Имя JNDI фабрики управляемых соединений или спецификации активации <ul style="list-style-type: none"> – Соединение с фабрикой управляемых соединений или спецификацией активации на сервере – Применяется вместо указания значения каждого свойства фабрики управляемых соединений или спецификации активации в мастере – Сохраняется в качестве целевого сервера в файле импорта
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

URL базы данных (DatabaseURL)

Это свойство задает URL, специфичный для драйвера JDBC, для создания соединения с базой данных.

Таблица 89. URL базы данных - Сведения

Обязательное	<p>Да, если заданы следующие свойства:</p> <ul style="list-style-type: none"> • DataSourceJNDIName • XADataSourceName
--------------	---

Таблица 89. URL базы данных - Сведения (продолжение)

Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	<p>В окне ESW (мастер внешних служб) составьте URL базы данных, заполнив соответствующие поля. Например, URL базы данных Oracle состоит из ИД системы (SID), имени хоста, и номера порта базы данных. В административной консоли введите полный URL базы данных.</p> <p>Если сервер базы данных поддерживает IPv6, то имя хоста можно указать в формате IPv6.</p> <p>Свойства, отвечающие за подключение к базе данных, применяются в следующем порядке:</p> <ol style="list-style-type: none"> 1. Если задано свойство DataSourceJNDIName, то адаптер использует его для подключения к базе данных. Если также заданы свойства UserName и Password, то они переопределяют имя пользователя и пароль, указанные на уровне источника данных. 2. Если свойство DataSourceJNDIName не задано, то адаптер подключается с помощью свойства XADataSourceName (если оно задано). Свойство DataSourceJNDIName представляет источник данных XA или пула соединений. Если источник данных JNDI, указанный на сервере с поддержкой транзакций XA, задан в ходе настройки адаптера, то адаптер принимает участие в транзакциях XA. Кроме того, для поддержки транзакций XA можно указать свойство XADataSourceName. 3. Если свойства DataSourceJNDIName и XADataSourceName не указаны, то адаптер использует свойства DatabaseURL, JDBCDriverClass, UserName и Password для установления соединения. <p>Если в качестве имени хоста указывается адрес IPv6, то его следует заключить в квадратные скобки ([]).</p>
Примеры	<p>В следующем примере показаны типичные URL баз данных:</p> <pre>jdbc:oracle:thin:@9.26.248.148:1521:dev</pre>
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Производитель базы данных (DatabaseVendor)

Задает тип базы данных. Тип определяется в соответствии с производителем базы данных.

Таблица 90. Производитель базы данных - Сведения

Обязательное	Да
Допустимые значения	Oracle
Значение по умолчанию	ORACLE
Тип свойства	Строка
Формат	<p>Отдельные операторы SQL требуют специальной обработки, которая зависит от типа базы данных. Например, типы данных Struct и Array в Oracle требуют специальной обработки. Это свойство задает RDBMS, которая определяет тип базы данных.</p> <p>Укажите Oracle в качестве значения, соответствующего производителю базы данных.</p>
Локализуемое	Нет

Таблица 90. Производитель базы данных - Сведения (продолжение)

Поддерживает двунаправленный текст	Нет
------------------------------------	-----

Тип доставки (DeliveryType)

Задаёт порядок доставки событий в ходе экспорта.

Таблица 91. Тип доставки - Сведения

Обязательное	Нет
Допустимые значения	ORDERED UNORDERED
Значение по умолчанию	ORDERED
Тип свойства	String
Применение	Поддерживаются следующие значения: <ul style="list-style-type: none"> • ORDERED: Адаптер последовательно доставляет события в точку экспорта. • UNORDERED: Адаптер одновременно доставляет все события в точку экспорта.
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Не обрабатывать события с системным временем в будущем (FilterFutureEvents)

Позволяет разрешить или запретить фильтрацию будущих событий путем сравнения системного времени каждого события с временем системы.

Таблица 92. Не обрабатывать события с системным временем в будущем - Сведения

Обязательное	Да
Допустимые значения	True False
Значение по умолчанию	False
Тип свойства	Булевское
Применение	Если указано значение True, то адаптер сравнивает время каждого события с временем системы. Если время события опережает время системы, то событие не доставляется. Если указано значение False, то адаптер доставляет все события.
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Гарантированная доставка событий (AssuredOnceDelivery)

Указывает, следует ли обеспечивать гарантированную однократную доставку входящих событий.

Таблица 93. Гарантированная доставка событий - Сведения

Обязательное	Да
Допустимые значения	True False
Значение по умолчанию	True
Тип свойства	Булевское
Применение	<p>Если для этого свойства указано значение True, то адаптер обеспечивает гарантированную однократную доставку. Это означает, что каждое событие доставляется только один раз. Значение False не обеспечивает гарантированную однократную доставку события, однако повышает быстродействие.</p> <p>Если для этого свойства указано значение True, то адаптер пытается сохранить информацию XID в хранилище событий. Если указано значение False, то адаптер не пытается сохранить эту информацию.</p> <p>Это свойство используется только в том случае, если компонент экспорта поддерживает транзакции. В противном случае транзакции недоступны независимо от значения этого свойства.</p>
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Упорядочить события по (EventOrderBy)

Порядок, в котором события извлекаются и обрабатываются.

Таблица 94. Упорядочить события по сведениям

Обязательное	Нет
Допустимые значения	Список через запятую (,) имен столбцов в хранилище событий и атрибуты упорядочивания asc и desc
Значение по умолчанию	event_time, event_priority
Тип свойства	Строка
Формат	Список через запятую имен столбцов из хранилища событий с необязательными атрибутами упорядочивания по возрастанию или убыванию.
Примеры	<p>Для того чтобы показать события, упорядоченные сначала по времени, а затем по приоритету, укажите:</p> <pre>event_time, event_priority</pre> <p>Для того чтобы показать события, упорядоченные сначала по имени объекта в возрастающем порядке, а затем по времени в убывающем порядке, укажите:</p> <pre>object_name asc, event_time desc</pre>
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Тип запроса события (EventQueryType)

Это свойство указывает, нужно ли использовать стандартную или пользовательскую обработку запроса.

Таблица 95. Сведения о типе запроса события

Обязательное	Да
Допустимые значения	Стандартная Динамическая
Значение по умолчанию	Стандартная
Тип свойства	Строка
Формат	Допустимые значения: Стандартная для стандартной обработки события, и Динамическая для пользовательской обработки события. Если это свойство имеет значение Динамическая, то используются свойства CustomEventQuery, CustomUpdateQuery и CustomDeleteQuery. Если это свойство имеет значение Стандартная, то эти свойства игнорируются.
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Имя таблицы событий (EventTableName)

Это свойство указывает имя таблицы в целевой базе данных, которая хранит события, используемые для входящей обработки.

Таблица 96. Сведения об имени таблицы событий

Обязательное	Да
Значение по умолчанию	WBIA_Oracle_EventStore
Тип свойства	Строка
Формат	Создайте хранилище событий перед началом настройки адаптера. Для стандартной обработки событий событие генерируется базой данных посредством триггера или другого механизма. Для пользовательской обработки запроса адаптер сохраняет события в хранилище событий при получении результата пользовательского запроса.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Типы обрабатываемых событий (EventTypeFilter)

Содержит список с ограничителями, в котором указаны доставляемые адаптером события.

Таблица 97. Типы обрабатываемых событий - Сведения

Обязательное	Нет
Допустимые значения	Типы бизнес-объектов, перечисленные через запятую (,).
Значение по умолчанию	null
Тип свойства	String

Таблица 97. Типы обрабатываемых событий - Сведения (продолжение)

Применение	События фильтруются по типу бизнес-объектов. Если это свойство задано, то адаптер доставляет только те события, которые указаны в списке. Значение null указывает, то фильтр не применяется и в ходе экспорта доставляются все события.
Пример	Для того чтобы получать только события, относящиеся к бизнес-объектам Customer и Order, укажите следующее значение: Customer,Order Если заданы оба свойства EventTypeFilter и AdapterInstanceEventFilter, то адаптер обрабатывает только те события, которые соответствуют обоим критериям. Таким образом, обрабатываются только те события, тип которых совпадает с указанным в свойстве EventTypeFilter, и столбец connectorId которых совпадает с указанным в свойстве AdapterInstanceEventFilter.
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Количество повторов для недоставленных событий (FailedEventRetryLimit)

Указывает, сколько раз адаптер будет повторять попытку доставить событие, если это сделать не удалось.

Таблица 98. Количество повторов для недоставленных событий - Сведения

Обязательное	Нет
Допустимые значения	Целые числа
Значение по умолчанию	5
Тип свойства	Целое число
Применение	Указывает, сколько раз адаптер должен выполнять повторные попытки отправки события. Допустимы следующие значения: Значение по умолчанию Если свойство не указано, то адаптер выполняет пять дополнительных попыток отправки события. 0 Число повторных попыток отправки события не ограничено. Если указано значение 0, то событие остается в хранилище событий. Оно никогда не добавляется в число необработанных событий. > 0 Выполняется указанное число повторных попыток отправки события. < 0 Повторные попытки отправки события не выполняются.
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Имя класса драйвера JDBC (JDBCDriverClass)

Имя класса драйвера JDBC, применяемого для подключения к базе данных.

Таблица 99. Имя класса драйвера JDBC - Сведения

Обязательное	Да, если не задано свойство DataSourceJNDIName
--------------	--

Таблица 99. Имя класса драйвера JDBC - Сведения (продолжение)

Допустимые значения	Значение, указанное в ходе настройки подключения к экземпляру базы данных.
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	<p>В ходе работы с ESW (мастер внешних служб) класс драйвера JDBC указан, если выбрана общая комбинация базы данных и драйвера (например, драйверы Type 4 для последних версий Oracle). Для драйверов Type 2 необходимо указать имя класса базы данных вручную.</p> <p>Если выбран драйвер Type 2 или базовый драйвер, то необходимо ввести имя класса драйвера JDBC.</p> <p>В административной консоли введите имя драйвера.</p> <p>Если задано свойство DataSource.JNDIName, то это свойство игнорируется.</p>
Примеры	<p>Имя класса драйвера JDBC отображается в ESW (мастер внешних служб) и административной консоли. Ниже приведены примеры свойств класса драйвера JDBC в ESW (мастер внешних служб) и административной консоли. ESW (мастер внешних служб):</p> <ul style="list-style-type: none"> Для подключения к базе данных Oracle 10 с помощью драйвера Type 4 выберите Oracle Thin Driver. <p>Административная консоль:</p> <p>Драйвер Oracle Thin JDBC oracle.jdbc.driver.OracleDriver</p>
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Максимальное число соединений (MaximumConnections)

Задаёт ограничение числа соединений, доступных адаптеру для доставки входящих событий.

Таблица 100. Максимальное число соединений - Сведения

Обязательное	Нет
Значение по умолчанию	1
Тип свойства	Целое число
Применение	Допустимы только положительные значения. Все положительные значения меньше 1 рассматриваются как 1. Ввод отрицательного значения или 1 может привести к ошибкам времени выполнения.
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Минимальное число соединений (MinimumConnections)

Задаёт минимальное число соединений, доступных адаптеру для доставки входящих событий.

Таблица 101. Минимальное число соединений - Сведения

Обязательное	Нет
Значение по умолчанию	1
Тип свойства	Целое число
Применение	Допустимы только положительные значения. Любое значение меньше 1 рассматривается как 1. Отрицательное значение или 1 для этого свойства могут привести к ошибкам времени выполнения.
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Пароль (Password)

Пароль для имени пользователя базы данных.

Таблица 102. Пароль - Сведения

Обязательное	Нет. Если задан псевдоним идентификации или свойство DataSourceJNDIName, то пароль указывать необязательно. Если заданы псевдоним идентификации, свойство DataSourceJNDIName и свойство Пароль, то преимуществом обладает свойство Пароль.
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Значение этого свойства переопределяет пароль, указанный для источника данных на сервере с помощью псевдонима идентификации или свойства DataSourceJNDIName.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Запрос SQL для проверки соединения (PingQuery)

Запрос SQL, используемый для проверки надежности соединения с базой данных.

Таблица 103. Запрос проверки связи - Сведения

Обязательное	Нет
Тип свойства	Строка
Значение по умолчанию	Значение по умолчанию отсутствует
Формат	Содержит запрос SQL, применяемый для проверки соединения с базой данных. Адаптер выполняет проверку связи каждый раз при получении исключительной ситуации SQLException в ходе выполнения исходящих операций. Адаптер не выполняет попыток восстановления соединения. Если база данных недоступна, то адаптер отправляет уведомление контейнеру. За восстановление соединений с базой данных отвечает администратор пула соединений.
Локализуемое	Нет

Таблица 103. Запрос проверки связи - Сведения (продолжение)

Поддерживает двунаправленный текст	Нет
------------------------------------	-----

Интервал между периодами опроса (PollPeriod)

Задаёт время, в течение которого адаптер ожидает между периодами опроса.

Таблица 104. Интервал между периодами опроса - Сведения

Обязательное	Да
Допустимые значения	Целые числа, большие или равные 0.
Значение по умолчанию	2000
Единицы измерения	Миллисекунды
Тип свойства	Целое число
Применение	Опросы выполняются с фиксированной частотой. Если по какой-либо причине цикл опроса задерживается (например, если он не успел завершиться), то следующий цикл опроса начинается немедленно.
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Максимальное число событий в периоде опроса (PollQuantity)

Задаёт число событий, доставляемых адаптером в ходе каждого периода опроса.

Таблица 105. Максимальное число событий в периоде опроса - Сведения

Обязательное	Да
Значение по умолчанию	10
Тип свойства	Целое число
Применение	Значение должно быть больше 0. В результате увеличения этого значения в течение периода опроса будет обрабатываться большее число событий, что может привести к снижению производительности адаптера. Уменьшение этого значения приведет к уменьшению числа обрабатываемых событий и незначительному повышению производительности адаптера.
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Тайм-аут запроса (QueryTimeout)

Максимальное время выполнения запросов SQL (в секундах).

Таблица 106. Тайм-аут запроса - Сведения

Обязательное	Нет
--------------	-----

Таблица 106. Тайм-аут запроса - Сведения (продолжение)

Значение по умолчанию	Значение по умолчанию отсутствует
Единицы измерения	Секунды
Тип свойства	Integer
Формат	Если время выполнения запроса превышает значение тайм-аута, то база данных выбрасывает исключение SQL. Связанное сообщение заносится в файл протокола. Если значение не указано, то тайм-аут не отслеживается.
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Интервал повтора в случае сбоя соединения (RetryInterval)

Это свойство задает время ожидания перед попыткой повторного подключения в случае ошибки, связанной со входящим соединением.

Таблица 107. Интервал повтора - Сведения

Обязательное	Да
Значение по умолчанию	2000
Единицы измерения	Миллисекунды
Тип свойства	Целое число
Применение	Допустимы только положительные значения. Это свойство задает время ожидания перед попыткой повторного подключения в случае ошибки, связанной со входящим соединением.
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Число попыток восстановить соединение с системой (RetryLimit)

Задает число попыток восстановления входящего соединения.

Таблица 108. Число попыток восстановить соединение с системой - Сведения

Обязательное	Нет
Допустимые значения	0 и положительные целые числа
Значение по умолчанию	0
Тип свойства	Целое число

Таблица 108. Число попыток восстановить соединение с системой - Сведения (продолжение)

Применение	Это свойство указывает число повторных попыток подключения адаптера к Oracle DB для обработки входящих событий. Значение 0 соответствует бесконечному числу повторов. Свойство RetryConnectionOnStartup позволяет разрешить повторное подключение адаптера к Oracle DB в случае ошибки.
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Восстанавливать соединение EIS при запуске (RetryConnectionOnStartup)

Позволяет разрешить или запретить повторные попытки подключения к Oracle DB, если соединение не было установлено в ходе запуска.

Таблица 109. Восстанавливать соединение EIS при запуске - Сведения

Обязательное	Нет
Допустимые значения	True False
Значение по умолчанию	False
Тип свойства	Булевское
Применение	Указывает, должен ли адаптер повторять попытки подключения к Oracle DB, если в ходе запуска адаптера соединение установить не удалось: <ul style="list-style-type: none"> Укажите значение False, если адаптер должен немедленно выдавать сведения о подключении к Oracle DB. Пример: разработка и тестирование приложения, получающего события от адаптера. Если соединение не установлено, то адаптер записывает информацию об ошибке в файлы протокола и трассировки и завершает работу. На административной консоли для приложения будет указано состояние Остановлено. После восстановления соединения запустите адаптер вручную. Укажите значение True, если немедленная реакция адаптера не требуется. Если соединение установить не удалось, то адаптер заносит информацию об ошибке в файлы протокола и трассировки и выполняет повторные попытки подключения. Свойство RetryInterval указывает частоту повторных попыток, а свойство RetryLimit - максимальное число повторных попыток. На административной консоли для приложения будет указано состояние Запущено.
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Возвращать бизнес-объект при получении пустого набора результатов хранимой процедуры (ReturnDummyBOForSP)

Применяется для возврата выходных параметров, даже если набор результатов пустой.

Таблица 110. Возвращать бизнес-объект при получении пустого набора результатов хранимой процедуры - Сведения

Обязательное	Нет
--------------	-----

Таблица 110. Возвращать бизнес-объект при получении пустого набора результатов хранимой процедуры - Сведения (продолжение)

Допустимые значения	True False
Значение по умолчанию	False
Тип свойства	Булевское
Формат	Операция RetrieveSP возвращает набор результатов. Если набор результатов пуст и в свойстве ReturnDummyBOForSP указано значение False, то бизнес-объекты не создаются и выходные параметры вызова процедуры будут недоступны для загрузки. Если в свойстве ReturnDummyBOForSP указано значение True, то создается фиктивный бизнес-объект, содержащий значения соответствующих атрибутов.
Локализуемое	Да
Поддерживает двунаправленный текст	Нет

Остановить адаптер при возникновении ошибки во время опроса (StopPollingOnError)

Позволяет разрешить или запретить продолжение опроса событий после ошибки в ходе опроса.

Таблица 111. Остановить адаптер при возникновении ошибки во время опроса - Сведения

Обязательное	Нет
Допустимые значения	True False
Значение по умолчанию	False
Тип свойства	Булевское
Применение	Если для этого свойства указано значение True, то адаптер останавливает опрос после ошибки. Если для этого свойства указано значение False, то адаптер регистрирует исключительную ситуацию и продолжает опрос событий.
Локализуемое	Нет
Поддерживает двунаправленный текст	Нет

Хранимая процедура, выполняемая после опроса (SPAfterPoll)

Это свойство задает имя хранимой процедуры или хранимой функции, выполняемой после каждого цикла опроса.

Таблица 112. Сведения о хранимой процедуре, выполняемой после опроса

Обязательное	Нет
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Хранимая процедура получает один параметр для количества опросов.

Таблица 112. Сведения о хранимой процедуре, выполняемой после опроса (продолжение)

Локализуемое	Да
Поддерживает двунаправленный текст	Да

Хранимая процедура, выполняемая перед опросом (SPBeforePoll)

Это свойство задает имя хранимой процедуры или хранимой функции, выполняемой перед фактическим вызовом опроса.

Таблица 113. Сведения о хранимой процедуре, выполняемой перед опросом

Обязательное	Нет
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Хранимая процедура получает один параметр для количества опросов.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Имя пользователя (UserName)

Имя пользователя базы данных.

Таблица 114. Имя пользователя - Сведения

Обязательное	Нет. Если задан псевдоним идентификации или свойство DataSourceJNDIName, то имя пользователя указывать необязательно. Если заданы псевдоним идентификации, свойство DataSourceJNDIName и свойство Имя пользователя, то преимуществом обладает свойство Имя пользователя.
Значение по умолчанию	Значение по умолчанию отсутствует
Тип свойства	Строка
Формат	Значение этого свойства переопределяет имя пользователя, указанное для источника данных на сервере с помощью псевдонима идентификации или свойства DataSourceJNDIName.
Локализуемое	Да
Поддерживает двунаправленный текст	Да

Глобализация

WebSphere Adapter for Oracle E-Business Suite - это глобализованное приложение, подходящее для применения в разных языковых и культурных средах. Адаптер доставляет текст сообщения с учетом поддерживаемого набора символов и локали сервера хоста. Кроме того, адаптер поддерживает преобразование данных двунаправленных языков между компонентами интеграции.

Глобализация и преобразование двунаправленного текста

Адаптер поддерживает наборы одно- и многобайтовых символов и доставку текста сообщений на указанном языке. Кроме того, он выполняет преобразование двунаправленного текста, т.е. позволяет обрабатывать файлы, в которых одновременно может содержаться текст, написанный слева направо (например, иврит, арабский язык) и справа налево (например, URL или имя каталога).

Глобализация

Глобализованные программные приложения спроектированы и разработаны для использования не в одной, в нескольких языковых и национальных средах. WebSphere Adapters, WebSphere Integration Developer, WebSphere Process Server и WebSphere Enterprise Service Bus написаны на Java. Данные в среде выполнения Java в виртуальной машине Java (JVM) представляются в формате Unicode. Формат Unicode содержит кодировки для символов из наиболее распространенных кодовых наборов (как одно-, так и многобайтовых). Следовательно, при передаче данных между компонентами системы интеграции выполнять их преобразование не требуется.

Для регистрации ошибок и информационных сообщения в протоколах адаптер использует локаль той системы, в которой он работает.

Преобразование двунаправленного текста

Двунаправленный текст - это текст (например, на арабском языке или иврите), содержащий как элементы с записью справа налево, так и элементы с записью слева направо. Приложение может отображать и обрабатывать двунаправленный текст несколькими способами. В WebSphere Process Server и WebSphere Enterprise Service Bus применяется стандартный формат Windows, однако, информационная система предприятия, обменивающаяся данными с WebSphere Process Server или WebSphere Enterprise Service Bus, может использовать другой формат. WebSphere Adapters преобразуют двунаправленные данные, которыми обмениваются две системы, так, что данные правильно обрабатываются и отображаются в обеих этих системах.

Формат двунаправленного текста

В WebSphere Process Server и WebSphere Enterprise Service Bus применяется формат двунаправленного текста ILYNN (implicit, left-to-right, on, off, nominal). Это формат, применяемый Windows. Если информационная система предприятия использует другой формат, то адаптер преобразует его перед передачей данных в WebSphere Process Server или WebSphere Enterprise Service Bus.

Формат двунаправленного текста описывается пятью атрибутами. Настройка свойств двунаправленного текста предусматривает задание каждого из этих атрибутов. Атрибуты и их значения перечислены в следующей таблице.

Таблица 115. Атрибуты формата двунаправленного текста

Положение символа	Назначение	Значения	Описание	Значение по умолчанию
1	Схема порядка	I	Неявный (логический)	I
		V	Визуальный	

Таблица 115. Атрибуты формата двунаправленного текста (продолжение)

Положение символа	Назначение	Значения	Описание	Значение по умолчанию
2	Направление	L	Слева направо	L
		R	Справа налево	
		C	Контекстный слева направо	
		D	По контексту, справа налево	
3	Симметричная перестановка	Y	Симметричная перестановка включена	Y
		N	Симметричная перестановка выключена	
4	Форма текста	S	Форма задается	N
		N	Форма не изменяется	
		I	Начальная форма	
		M	Промежуточная форма	
		F	Конечная подгонка	
		B	Изолированная форма	
5	Форма числа	H	Национальный (Хинди)	N
		C	Изменение формы по контексту	
		N	Форма чисел не изменяется	

Перед отправкой данных в WebSphere Process Server или WebSphere Enterprise Service Bus адаптер преобразует данные в логический формат с направлением слева направо.

Работа со свойствами двунаправленного текста

Свойства двунаправленного текста позволяют управлять преобразованием как содержимого, так и метаданных. Специальные свойства двунаправленного текста позволяют исключить из преобразования содержимое или метаданные, а также определить данные, требующие специальной обработки.

Типы свойств двунаправленного текста описываются в следующей таблице.

Таблица 116. Типы свойств двунаправленного текста

Тип свойства	Преобразование данных
EIS	Управляет форматом содержимого, а также данных, поступающих из информационной системы предприятия (базы данных).
Метаданные	Управляет форматом метаданных, а также данных, предоставляющих информацию о содержимом.
Пропустить	Определяет содержимое или метаданные для исключения из преобразования.
Специальный формат	Задаёт конкретный текст, например путь к файлам или URL, требующий отдельной обработки в ходе преобразования. Задаётся для содержимого и метаданных.

Свойства, управляющие преобразованием двунаправленного текста, можно задать в следующих областях:

- **Свойства адаптера ресурсов:** Применяются для хранения параметров конфигурации по умолчанию, в том числе свойства TurnBiDiOff, которое разрешает или запрещает преобразование двунаправленного текста для конкретного экземпляра адаптера. Для настройки этих свойств используйте административную консоль сервера.
- **Свойства фабрики управляемых соединений:** Применяются во время выполнения для создания экземпляра исходящего соединения с информационной системой предприятия. Созданные свойства фабрики управляемого соединения сохраняются в файле описания.
- **Свойства спецификации активации:** содержат информацию о конфигурации обработки входящих событий для конечной точки сообщений. Их можно задать с помощью ESW (мастер внешних служб) или административной консоли сервера.

Область действия свойств и механизм поиска

Адаптер выполняет преобразование двунаправленного текста после задания свойств двунаправленного текста. Применяемая при этом логика реализована на основе иерархического наследования значений свойств и механизма поиска.

Свойства, заданные для адаптера ресурсов, принадлежат верхнему уровню иерархической структуры; свойства, заданные в других областях и добавленные в бизнес-объект в качестве аннотаций, принадлежат более низким уровням. Например, если указать значения свойств двунаправленного текста EIS только для адаптера ресурсов, то эти значения будут наследоваться и использоваться преобразованиями, которым требуется заданное свойство двунаправленного текста EIS, как во входящих (спецификация активации), так и в исходящих (фабрика управляемых соединений) транзакциях.

Однако, если задать значения двунаправленных свойств типа EIS как для адаптера ресурсов, так и для активации спецификации, то преобразование, инициализируемое во входящей транзакции, будет использовать значения активации спецификации.

Логика обработки выполняет поиск значений свойств двунаправленного текста, необходимых для преобразования, с помощью механизма поиска. Механизм поиска начинает поиск заданных свойств с уровня, на котором вызывается преобразование, и продолжает его, перемещаясь вверх по иерархической структуре. Применяется первое допустимое найденное значение. Иерархическая структура просматривается в направлении от дочерних уровней к родительским; сестринские уровни не учитываются.

Свойства, доступные для преобразования двунаправленных данных

WebSphere Adapter for Oracle E-Business Suite имеет несколько свойств конфигурации, которые доступны для преобразования двунаправленных данных.

Адаптер позволяет обмен двунаправленными данными между клиентским приложением и базой данных, даже если данные в базе данных имеют другой двунаправленный формат, чем используется в среде выполнения. Можно использовать двунаправленные символы при настройке адаптера и в специфической для приложения информации бизнес-объектов. Следующие наборы свойств и специфической для приложения информации доступны для поддержки двунаправленных данных:

- Свойства конфигурации
 - Свойства спецификации активации
 - Свойства соединения для ESW (мастер внешних служб)
 - Свойства фабрики управляемых соединений
- Специфическая для приложения информация
 - ASI уровня бизнес-объекта
 - ASI уровня операции
 - ASI уровня атрибута

В следующих разделах перечислены специфические свойства конфигурации и информация, специфичная для приложения, которые применяются для преобразования двунаправленных данных.

Свойства спецификации активации

Следующие свойства спецификации активации для преобразования двунаправленных данных сценария:

- Пользовательский запрос на удаление
- Пользовательский запрос события
- Пользовательский запрос на обновление
- Дополнительные свойства соединения драйвера JDBC
- URL базы данных
- Упорядочить события по
- Имя таблицы событий
- Пароль
- Хранимая процедура для выполнения после опроса
- Хранимая процедура для выполнения после опроса
- Имя пользователя

Свойства соединения, используемые в мастере

Следующие свойства соединения для ESW (мастер внешних служб) используются для преобразования двунаправленных данных сценария:

- Имя пользователя
- Пароль

Свойства фабрики управляемых соединений

Следующие свойства управляемого соединения используются для преобразования двунаправленных данных сценария:

- Дополнительные свойства соединения драйвера JDBC
- URL базы данных
- Пароль
- Имя пользователя

Информация, специфичная для приложения бизнес-объекта

Следующая информация, специфичная для приложения бизнес-объекта, используется для преобразования двунаправленных данных сценария:

- TableName

- StatusColumnName
- SPName
- SelectStatement

Информация, специфичная для приложения операции

Следующая информация, специфичная для приложения операции, используется для преобразования двунаправленных данных сценария:

- StoredProcedureName
- PropertyName в Параметрах

Информация, специфичная для приложения атрибута

Следующая информация, специфичная для приложения атрибута, используется для преобразования двунаправленных данных сценария:

- ColumnName

Сообщения адаптера

Просмотр сообщений, посланных WebSphere Adapter for Oracle E-Business Suite в следующее расположение.

Ссылка на сообщения: <http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r2mx/topic/com.ibm.wbit.620.help.messages.doc/messages.html>

Показанная Web-страница содержит список префиксов сообщений. Щелкните на префиксе сообщения для просмотра всех сообщений с префиксами:

- Сообщения с префиксом CWYOE посланы WebSphere Adapter for Oracle E-Business Suite
- Сообщения с префиксом CWYDB выдаются общим компонентом WebSphere Adapter for JDBC и WebSphere Adapter for Oracle E-Business Suite
- Сообщения с префиксом CWYBS выдаются базовыми классами адаптера, используемыми всеми адаптерами.

Связанная информация

Приведенные ниже ссылки на справочные системы Information Center, IBM Redbooks, а также Web-страницы содержат связанную информацию по продукту WebSphere Adapter for Oracle E-Business Suite.

Примеры и руководства

На Web-сайте примеров и учебников управления бизнес-процессами доступны различные примеры и учебники, призванные облегчить работу с WebSphere Adapters. Просмотреть примеры и учебники можно следующими способами:

- Со страницы приветствия, которая открывается при запуске WebSphere Integration Developer. Для просмотра примеров и руководств для WebSphere Adapter for Oracle E-Business Suite нажмите **Извлечь**. Затем совершите обзор показанных для выбора категорий.
- В этом расположении в Web: <http://publib.boulder.ibm.com/bpcsamp/index.html>.

Информационные ресурсы

- Web-страница информационных ресурсов WebSphere Business Process Management содержит ссылки на статьи, Redbooks, документацию и предложения по образованию для изучения WebSphere Adapters: <http://www14.software.ibm.com/webapp/wsbroker/redirect?version=pix&product=wps-dist&topic=bpmroadmaps>
- Страница библиотеки WebSphere Adapters включает в себя ссылки на все версии документации: <http://www.ibm.com/software/integration/wbiadapters/library/infocenter/>

Информация о связанных продуктах

- WebSphere Business Process Management, 6.2, information center, который включает в себя информацию о WebSphere Process Server, WebSphere Enterprise Service Bus, и WebSphere Integration Developer: <http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r2mx/index.jsp>
- WebSphere Adapters, 6.1.x, information center: <http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/topic/com.ibm.wbit.612.help.adapter.emd.ui.doc/topics/tcreatecmps.html>
- WebSphere Business Integration Adapters Information Center: http://publib.boulder.ibm.com/infocenter/wbihelp/v6rxmx/index.jsp?topic=/com.ibm.wbi_adapters.doc/welcome_adapters.htm

Ресурсы developerWorks

- WebSphere Adapter Toolkit
- WebSphere business integration zone

Справка и поддержка

- Служба технической поддержки WebSphere Adapters: <http://www.ibm.com/software/integration/wbiadapters/support/>
- Технические замечания по WebSphere Adapters: <http://www.ibm.com/support/search.wss?tc=SSMKUK&rs=695&rank=8&dc=DB520+D800+D900+DA900+DA800+DB560&dtm>. В списке **Категория продукта** выберите имя адаптера и нажмите **Перейти**.

Примечания

Эта информация относится к продуктам и услугам, предлагаемым в США.

IBM может не предлагать эти продукты или услуги в других странах. За информацией о продуктах и услугах, доступных в вашей стране, обратитесь в представительство IBM. Упоминания продуктов, программ или услуг IBM не подразумевают, что можно использовать только упомянутые продукты, программы или услуги IBM. Вместо них можно использовать любой аналогичный продукт, программу или услугу, если это не нарушает авторских прав IBM на интеллектуальную собственность. Однако обязанность оценки работоспособности стороннего продукта лежит целиком на пользователе.

IBM может владеть патентами или заявками на патенты, относящимися к описанным в данном документе вопросам. Предоставление этого документа никоим образом не связано с предоставлением лицензии на эти патенты. Запросы относительно лицензий отправляйте в письменном виде по адресу:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Запросы относительно лицензий, касающихся информации в национальной кодировке (DBCS), отправляйте в письменном виде в IBM Intellectual Property Department или по адресу:

IBM World Trade Asia Corporation Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

Следующий абзац не относится к Великобритании, а также к другим странам, в которых это заявление противоречит местному законодательству: INTERNATIONAL BUSINESS MACHINES CORPORATION предоставляет настоящую публикацию на условиях “как есть”, без каких-либо явных или подразумеваемых гарантий, включая неявные гарантии соблюдения прав, коммерческой ценности и пригодности для какой-либо цели. В некоторых странах отказ от явных или подразумеваемых гарантий запрещен для определенных ситуаций, поэтому это заявление может не иметь силы для вас.

Эта информация может содержать технические и типографические неточности и опечатки. Эта информация изменяется на регулярной основе, и изменения включаются в будущие издания. IBM может вносить изменения и/или улучшения в продукты и программы, описанные в этой публикации, в любое время без дополнительного уведомления.

Все ссылки на сторонние по отношению IBM Web-сайты приведены для удобства и ни в коей мере не способствуют продвижению этих Web-сайтов. Материалы этих Web-сайтов не являются частью продукта IBM, поэтому их использование целиком зависит от вас.

IBM может использовать или распространять любую предоставленную вами информацию по своему усмотрению, не будучи связанной перед вами никакими обязательствами.

Обладатели лицензии на эту программу, желающие получить о ней информацию в целях: (i) обмена информацией между независимо созданными программами и прочими программами (включая и эту) (ii) взаимного применения переданной информации, должны обратиться по адресу:

IBM Corporation
Department 2Z4A/SOM1
294 Route 100
Somers, NY 10589-0100
U.S.A.

Такая информация может быть доступна на основании определенных условий и соглашений, которые могут также включать оплату.

Описанная в этом документе лицензионная программа и все поставляемые для нее лицензионные материалы предоставляются IBM на условиях IBM Customer Agreement, IBM International Program License Agreement или аналогичного соглашения.

Все данные о производительности были получены в управляемой среде. Поэтому результаты для других операционных сред могут существенно отличаться. Некоторые измерения были выполнены в разрабатываемых системах, поэтому их результаты могут отличаться от измерений в рабочих системах. Некоторые показатели могли быть получены экстраполяцией. Фактические результаты могут отличаться. Пользователи документа должны проверить применимость данных в своей среде.

Информация о продуктах сторонних производителей получена от поставщиков соответствующих продуктов, из их опубликованных объявлений или прочих общедоступных источников. IBM не тестировала эти продукты и не может подтвердить никакие заявления относительно производительности, совместимости или прочих возможностей сторонних продуктов. Вопросы по возможностям сторонних продуктов направляйте производителям этих продуктов.

Все заявления, касающиеся намерений и планов IBM, могут изменяться и отзываться без предварительного уведомления, и отражают только текущие цели и задачи.

Эта информация содержит примеры данных и отчетов из фактических деловых операций. Для того чтобы иллюстрация была полной, примеры содержат имена отдельных лиц, компаний и продуктов. Все такие имена являются вымышленными, и любое совпадение с именами и адресами физических существей является чисто случайным.

Лицензия:

Эта информация содержит примеры прикладных программ и их исходный текст, иллюстрирующий приемы программирования в различных операционных средах. Разрешается копировать, изменять и распространять эти примеры программ в любой форме, без необходимости выплачивать что-либо IBM, для целей разработки, использования, маркетинга или распространения прикладных программ, соответствующих интерфейсу прикладных программ операционной платформы, для которой они были написаны. Эти примеры не прошли исчерпывающего

тестирования. Тем самым IBM не гарантирует надежности, пригодности или работоспособности этих программ, даже подразумеваемой.

Каждая копия или часть этих примеров программ или любые производные материалы должны содержать заявление об авторских правах следующего вида: © (имя компании) (год). Часть этого исходного кода предоставлена IBM Corp. Sample Programs. © Copyright IBM Corp. _введите год или годы_. All rights reserved.

В электронных копиях этой информации могут не присутствовать фотографии и цветные изображения.

Информация о программном интерфейсе

Информация об интерфейсе программирования, если она предоставлена, предназначена для упрощения создания приложений с помощью этой программы.

Программные интерфейсы общего назначения позволяют создавать программное обеспечение, которое может использовать возможности этих программных средств.

Однако эта информация может также содержать сведения для диагностики, изменения и настройки. Сведения для диагностики, изменения и настройки предназначены для отладки программного обеспечения.

Предупреждение:

Не используйте сведения для диагностики, изменения и настройки как интерфейс программирования, потому что они могут меняться.

Товарные и сервисные знаки

IBM, эмблема IBM и `ibm.com` являются товарными знаками или зарегистрированными товарными знаками International Corporation в США и/или других странах. Если первое вхождение товарного знака IBM отмечено символом товарного знака (® или ™), то он является зарегистрированным или общим товарным знаком IBM в США на момент публикации этой информации. Кроме того, он может быть зарегистрированным или обычным товарным знаком в других странах. Текущий список товарных знаков IBM опубликован по следующему адресу: <http://www.ibm.com/legal/copytrade.shtml>

Linux является зарегистрированным товарным знаком Линуса Торвальдса (Linus Torvalds) в США и/или других странах.

Microsoft и Windows NT являются товарными знаками корпорации Майкрософт в США, других странах или тех, и других.

Java, а также все товарные знаки и эмблемы на основе Java являются товарными знаками Sun Microsystems, Inc. в США и других странах.

UNIX является зарегистрированным товарным знаком Open Group в США и других странах.

Другие названия компаний, продуктов и услуг могут быть товарными или сервисными знаками других фирм.

В данном продукте применяется программное обеспечение, разработанное в рамках проекта Eclipse Project <http://www.eclipse.org>).

Индекс

A

Adapter for Oracle E-Business Suite
администрирование 121
Adapter for Oracle E-Business Suite, модуль
завершение работы 131
запуск 131
установка файла EAR на сервере 119
экспорт в файл EAR 118

C

CEI (Общая инфраструктура событий) 136

D

DataSourceJNDIName 6
developerWorks 222

E

ESW (мастер внешних служб)
запуск 59
идентификация в 44
свойства соединения 59

F

FFDC (сбор данных независимого сбоя) 141

I

IBM WebSphere Adapter Toolkit 222

P

PMI (Инфраструктура отслеживания производительности)
настройка 133
описание 132
просмотр статистики производительности 134

R

Redbooks, WebSphere Adapters 221

U

UDF, см. пользовательская функция 40

W

WebSphere Adapters, 6.0.2.x, информация 222
WebSphere Adapters, версия 6.0, информация 222
WebSphere Business Process Management, 6.1.x, информация 222
WebSphere Enterprise Service Bus
информация 222
WebSphere Extended Deployment 48

WebSphere Integration Developer

информация 222
среда тестирования 111

WebSphere Process Server

информация 222

WebSphere Process Server или WebSphere Enterprise Service Bus

развертывание в 116

X

XAResourceNotAvailableException 142

A

автономный адаптер

замечания об использовании 47
описание 45

свойства адаптера ресурсов, установка 127

свойства спецификации активации, установка 130

свойства фабрики управляемых соединений, установка 128

адаптер

проект, создание 57

адаптер, технические заметки 222

Анализатор протокола 138

Анализатор протоколов и данных трассировки, поддержка 137

B

бизнес-объект запроса

создать на основе оператора SELECT 40

структура 26

бизнес-объект хранимой процедуры 34

бизнес-объекты 22, 161

атрибуты 151

запрос 40

как просмотреть 63, 88

кардинальность 28

несколько родительских объектов 84, 106

соглашение об именах 164

составные ключи 84, 106

типы атрибутов 152

хранимая процедура 34

B

ведение протокола

настройка свойств с помощью административной
консоли 138

внешние зависимости, добавление 111, 112

внешний ключ 151

встроенное развертывание 112

встроенный адаптер

замечания об использовании 47

описание 45

свойства адаптера ресурсов, установка 121

свойства спецификации активации, установка 125

свойства фабрики управляемых соединений, установка 123

входящая обработка 3

Г

гарантированная однократная доставка 18

З

зависимости программного обеспечения 58

зависимости, добавление 111, 112

запись нового значения 5

запуск приложений адаптера 131

защита

скрытие конфиденциальных данных 43

защита Java 2 44, 45

защита, Java 2 45

И

идентификация

ESW (мастер внешних служб) 44

время выполнения 45

описание 44

иерархические бизнес-объекты 28

Информация о WebSphere Application Server 222

информация о WebSphere Integration Adapters 222

информация о бизнес-объекте 151

информация уровня приложения 161

для атрибутов дочернего бизнес-объекта типа 158

для простых атрибутов 153

Инфраструктура обработки событий общего формата (CEI) 136

исключительные ситуации

XAResourceNotAvailableException 142

исходящая обработка 2

исходящие операции

список 5

К

количество объектов 28

количество элементов 151

конфиденциальная, трассировка 43

конфиденциальные данные, скрытие 43

кэширование подготовленного оператора 49

Л

локальные транзакции 5

М

матрица совместимости 1

матрица, совместимость 1

Н

настройка

ведение протокола 138

Система сбора статистики (PMI) 133

трассировка 138

О

обзор настройки 54

обзор процедуры настройки модуля 53

обнаружение неполадок

исключительная ситуация

XAResourceNotAvailableException 142

ресурсы для самостоятельного изучения 143

устранение обычных неполадок 144

обработка событий

пользовательские 18

стандартные 18

обработка события

пользовательская 4

стандартная 4

образование, WebSphere Adapters 221

объекты NULL

извлечение 11

одноуровневые бизнес-объекты 28

операции

ApplyChanges 12

Create 6

Delete 13

Execute 14

Exists 15

Retrieve 7

RetrieveAll 8

Update 11

операция ApplyChanges 12

операция Create 6

операция Delete 13

операция Execute 14

операция Exists 15

для бизнес-объектов представлений базы данных 15

для бизнес-объектов псевдонимов 15

для бизнес-объектов синонимов 15

для бизнес-объектов таблицы базы данных 15

ограничения 15

Операция Retrieve 7

операция RetrieveAll

для бизнес-объектов пользовательских запросов 10

для бизнес-объектов таблиц базы данных 8

операция Update 11

опрос 20

остановка приложений адаптера 131

отладка

исключительная ситуация

XAResourceNotAvailableException 142

ресурсы для самостоятельного изучения 143

отслеживание производительности 132

П

пакетные файлы адаптеров 139

первичный ключ 151

поддержка

обзор 137

ресурсы для самостоятельного изучения 143

техническая 222

поиск внешней службы

информация атрибута 151

пользовательская функция 40

пользовательские запросы

стандартный SQL 19

храняемая процедура 19

храняемая функция 20

пользовательские свойства

адаптер ресурсов 121, 127

спецификация активации 125, 130

фабрика управляемых соединений 123, 128

поправка 5

- представление
 - структура бизнес-объекта 23
- приложение адаптера
 - завершение работы 131
 - запуск 131
- примеры 51
- проект коннектора 57
- производительность
 - кэширование подготовленного оператора 49
- производительность адаптера 132
- псевдоним идентификации 45, 56

Р

- развертывание
 - в рабочей среде 116
 - в среде тестирования 111
 - опции 45
 - среды 111
- распределенные транзакции, см. транзакции XA 185
- Реализация Java 113
- реализация адаптера
 - защита 44
- реализация, Java 113
- редактор сборки, изменение информации, специфичной для приложения адаптера 84, 106
- ресурсы developerWorks, WebSphere Adapters 221
- ресурсы для самостоятельного изучения 143
- руководства 51

С

- сбой
 - описание 141
- сбой бизнеса 141
- сбор данных независимого сбоя (FFDC) 141
- свойства
 - адаптер ресурсов 121, 127
 - конфигурация входящего соединения 187
 - конфигурация исходящего соединения 165
 - свойства конфигурации
 - входящие 187
 - исходящий 165
 - соединение с внешней службой 167, 188
 - спецификация активации 125, 130
 - список 198
 - фабрика управляемых (J2C) соединений 123, 128
- свойства адаптера ресурсов
 - настройка в административной консоли 121, 127
 - сведения 172, 193
- свойства атрибута 151
- Свойства выбора метаданных
 - как указать (входящие) 96
 - как указать (исходящие) 76
- свойства конфигурации входящего соединения 187
- свойства конфигурации исходящего соединения 165
- свойства соединения с внешней службой 167, 188
- свойства спецификации активации
 - настройка в административной консоли 125, 130
 - список 198
- свойства спецификации взаимодействия
 - изменение 109
- свойства фабрики управляемых (J2C) соединений
 - настройка в административной консоли 123, 128
- свойства фабрики управляемых соединений
 - XADataSourceName 185

свойства фабрики управляемых соединений (продолжение)

- Имя источника данных XA 185
- сведения 176
- свойство enableNASupport 48
- Свойство Ограничение числа попыток 213
- свойство спецификации взаимодействия 186
- Связанная информация 221
- связанные продукты, информация 221
- связывание компонентов 113
- Система сбора статистики (PMI)
 - настройка 133
 - описание 132
 - просмотр статистики производительности 134
- сложные типы данных 34
- соглашение об именах для бизнес-объектов 164
- сообщения адаптера 221
- сообщения, адаптер 221
- специальная информация о приложении
 - добавление в объект 62, 87
- среда выполнения
 - идентификация в 45
 - развертывание файла EAR в 116
- среда высокого коэффициента готовности
 - входящие процессы 48
 - исходящие процессы 49
 - описание 48
 - развертывание в 48
- среда с кластерами
 - входящие процессы 48
 - исходящие процессы 49
 - описание 48
 - развертывание в 48
- среда тестирования
 - добавление модуля к 115
 - развертывание в 111, 115
 - тестирование модулей 116
- статистические данные производительности 134
- структура бизнес-объекта
 - для бизнес-объектов запросов 26
 - для бизнес-объектов таблиц или представлений 23
 - для бизнес-объектов хранимых процедур 25
- структура бизнес-объектов 23
- схема бизнеса 4

Т

- таблица
 - структура бизнес-объекта 23
- техническая поддержка 222
- технические заметки, WebSphere Adapters 221
- технические комментарии 1, 143, 222
- тип атрибута, бизнес-объект 152
- типы данных
 - сложный 34
- транзакции 5
 - с DataSourceJNDIName 6
- транзакции XA 5
 - базы данных Oracle 6
- Транзакции XA
 - Имя источника данных XA 185
- транзакции, см. также транзакции XA и локальные транзакции 5
- трассировка
 - настройка свойств с помощью административной консоли 138
- трассировка, файлы
 - расположение 140

- требования к аппаратному и программному обеспечению 1
- требования к аппаратному обеспечению 1
- требования к программному обеспечению 1
- требования, аппаратное и программное обеспечение 1
- триггеры для пользовательских таблиц 55

У

- установка файла EAR 119
- установка хранилища событий 55
- устранение неполадок
 - исключительная ситуация
 - XAResourceNotAvailableException 142
 - обзор 137
 - ресурсы для самостоятельного изучения 143

Ф

- фабрика соединений J2C
 - фабрика управляемых соединений *see* 176
- Файл EAR
 - установка на сервере 119
 - экспорт 118
- файл JAR, добавление внешних 111, 112
- файл RAR (архив адаптера ресурса)
 - описание 117
 - установка на сервере 117
- файл SystemOut.log 140
- файл trace.log 140
- файл архива адаптера ресурса (RAR)
 - описание 117
 - установка на сервере 117
- файлы
 - файл протокола SystemOut.log 140
 - файл трассировки trace.log 140
- Файлы драйвера JDBC 58
- файлы протокола
 - активация 138
 - изменение имени файла 140
 - отключение 138
 - расположение 140
 - уровень детализации 138
- файлы протоколов и трассировки 137
- файлы трассировки
 - активация 138
 - изменение имени файла 140
 - отключение 138
 - уровень детализации 138
- функции защиты
 - адаптер 44
 - защита Java 2 44

Х

- хранилище событий 20, 21
- храняемая процедура 14
 - изображение определения 39
 - определение 33
 - структура бизнес-объекта 25
- храняемые процедуры
 - обзор 33
 - операторы SQL 33
- храняемые функции
 - обзор 40

Ц

- целевой компонент 113

Э

- экспорт модуля в файл EAR 118



Напечатано в Дании