

SiTex

SiTex™

Пакеты
обновлений

Copyright© 2008, ООО «Систематика». Все права защищены.

Никакая часть этого документа не может быть воспроизведена или обработана в системах обработки данных, скопирована или использована в других документах без письменного уведомления ООО «Систематика».

Информация, содержащаяся в этом документе, может быть изменена ООО «Систематика» без предварительного уведомления. ООО «Систематика» не гарантирует отсутствия ошибок или опечаток в этом документе.

SiTex™ является зарегистрированной торговой маркой ООО «Систематика». Остальные торговые марки являются зарегистрированными торговыми марками их владельцев.

Аннотация

Данный документ представляет собой описание и основные правила работы с пакетами обновлений.

Рекомендован как инструкция по сборке пакетов обновлений, их корректного переноса и установки.

Используемые соглашения

В данном документе используются следующие текстуальные соглашения:

Жирный текст и *текст курсивом*

Для выделения важной информации используется жирный текст, используется также *курсив* для выделения определений терминов, отдельных значащих слов или фраз.

| |
|--|
| <p>Вид окон и структура папок, приведенная в данном документе, может незначительно отличаться на разных контекстах из-за различных настроек, но общий функционал остается неизменным.</p> |
|--|

Содержание

| | |
|---|-----------|
| 1. НАЗНАЧЕНИЕ | 5 |
| 2. СОЗДАНИЕ ПАКЕТА ОБНОВЛЕНИЯ | 6 |
| 3. СБОР ПАКЕТА ОБНОВЛЕНИЯ | 9 |
| 4. УСТАНОВКА ПАКЕТА ОБНОВЛЕНИЯ | 10 |
| 5. ШАГИ ОБНОВЛЕНИЙ | 11 |
| 5.1. Шаг «Пакет обновлений»..... | 12 |
| 5.2. Шаг «SQL»..... | 13 |
| 5.3. Шаг «Объект»..... | 14 |
| 5.4. Шаг «Файл» | 17 |
| 7. УТИЛИТА «ДОБАВИТЬ ОБЪЕКТ В ПО» | 18 |
| 8. ПРИМЕРЫ НАСТРОЕК ПАКЕТОВ ОБНОВЛЕНИЙ | 21 |
| 8.1. Упаковка мета-описания атрибута | 22 |
| 8.2. Упаковка мета-описания атрибутов | 23 |
| 8.3. Упаковка мета-описания класса | 25 |
| 8.4. Упаковка всех объектов класса | 26 |
| 8.5. Упаковка мета-описания класса и всех его объектов..... | 27 |
| 8.6. Упаковка папки с объектами | 28 |
| 8.7. Создание пакета на удаление атрибута..... | 30 |

1. НАЗНАЧЕНИЕ

Пакеты обновлений (патчи) предназначены для расширения функциональности системы, запущенной в эксплуатацию, устранения выявленных в ее процессе ошибок.

Пакеты обновление не предназначены для переноса большого объема данных. Для этого используйте пакет репликации с признаком «перенос только данных»

Пакеты обновлений по-умолчанию отображаются в папке *Администрирование / Пакеты обновлений*.

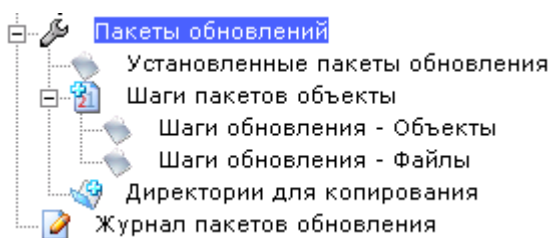


Рис. 1. Папки, относящиеся к пакетам обновлений в консоли управления

При этом:

- в папке *Пакеты обновлений* находятся описания пакетов обновлений, их шаги и настройки.
- в папке *Установленные пакеты обновлений* содержится информация об установленных на контекст пакетах обновлений: дата, логин пользователя, установившего пакет, а также лог последней установки
- В подпапках *Шаги пакетов обновлений* расположены шаги пакетов обновлений, сгруппированные по типам
- *Журнал пакетов обновлений* – содержит записи, фиксирующие факт установки пакета обновления

2. СОЗДАНИЕ ПАКЕТА ОБНОВЛЕНИЯ

Создать пакет обновлений можно из папки *Администрирование / Пакеты обновлений*.

Окно создание пакета обновления приведено ниже:

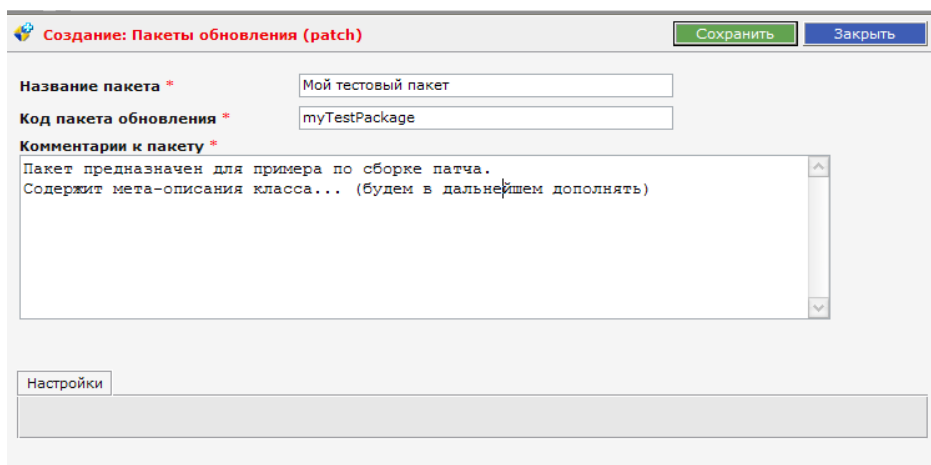


Рис. 2. Форма создания пакета обновления

Для создания патча необходимо заполнить три обязательных поля:

- *Название пакета* – имя патча, кратко описывающее функционал или данные, переносимые пакетом
- *Код пакета обновления* – уникальный код пакета обновления. В случае нахождения двух пакетов с одним именем система считает их идентичными.
- *Комментарии к пакету* – максимально подробное и полное описание всех данных, собранных в пакет

В процессе сборки пакета эти поля разрешается изменять/дополнять

После нажатия кнопки «Сохранить» пакет обновлений принимает следующий вид:

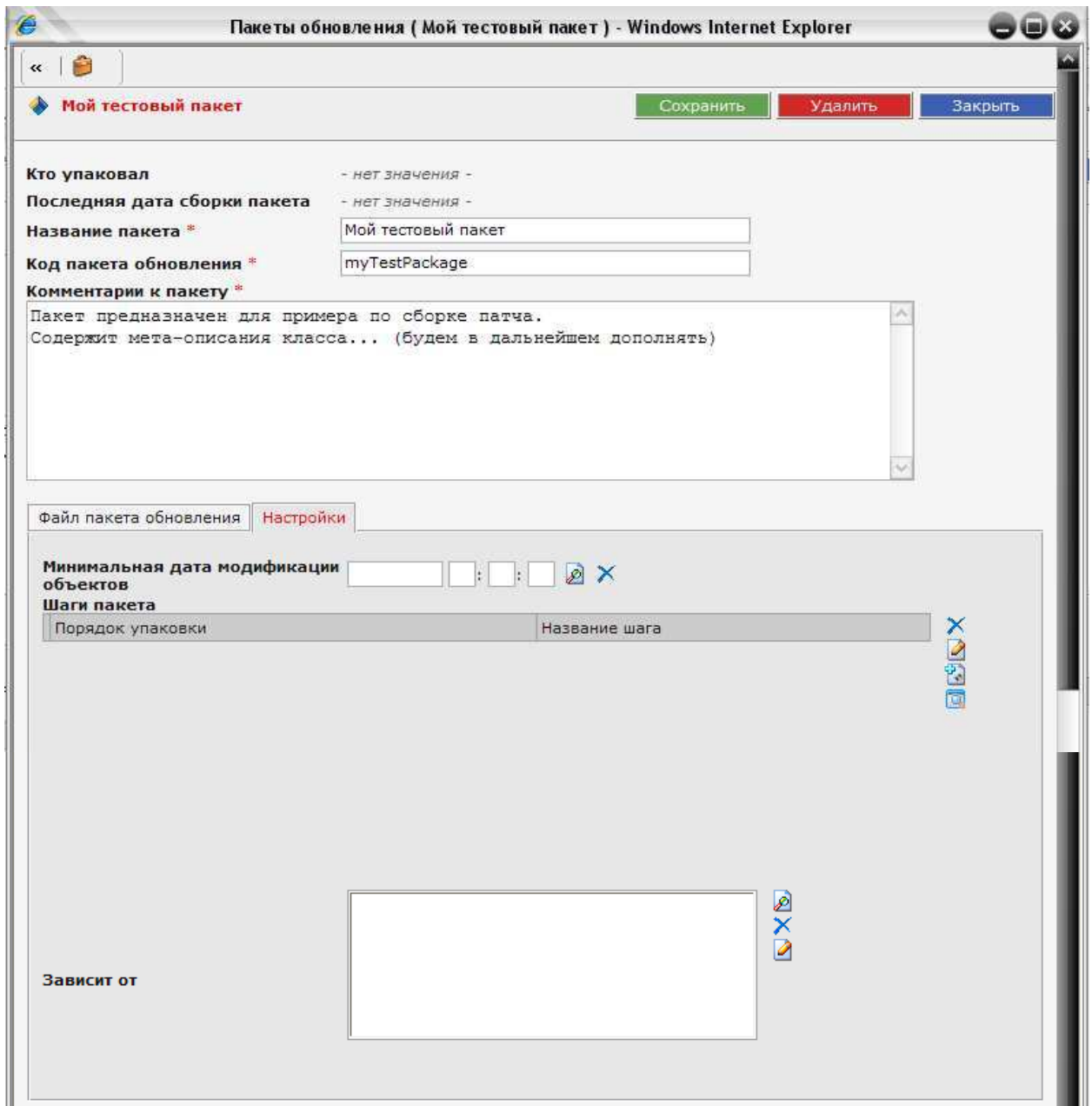


Рис. 3. Форма редактирования пакета обновления

Появляется вкладка *Настройки*, на которой содержатся настройки пакета обновления и описание включаемых в пакет данных:

- *Минимальная дата модификации объектов* – время, с которым сравнивается время изменения объектов. В случае указания, в пакет упаковываются лишь те объекты, значение атрибута типа «время изменения» которых больше указанной даты
- *Шаги пакета* – перечень шагов, которые входят в данный пакет обновления. Каждый пакет обновления может состоять из любого количества шагов, отличных от нуля. Упаковка шагов, а также распаковка, производится согласно **Порядку упаковки**

- *Зависит от* – список пакетов обновлений, от которых зависит текущий. При установке данного пакета, будет произведена проверка того, установлены или нет зависимые пакеты; если нет, то пакет обновления не будет установлен. Например, для обновления конкретного функционала требуется установить несколько пакетов обновлений. Для каждого из них можно задать наличие каких установленных пакетов необходимо, чтобы его можно было распаковать.

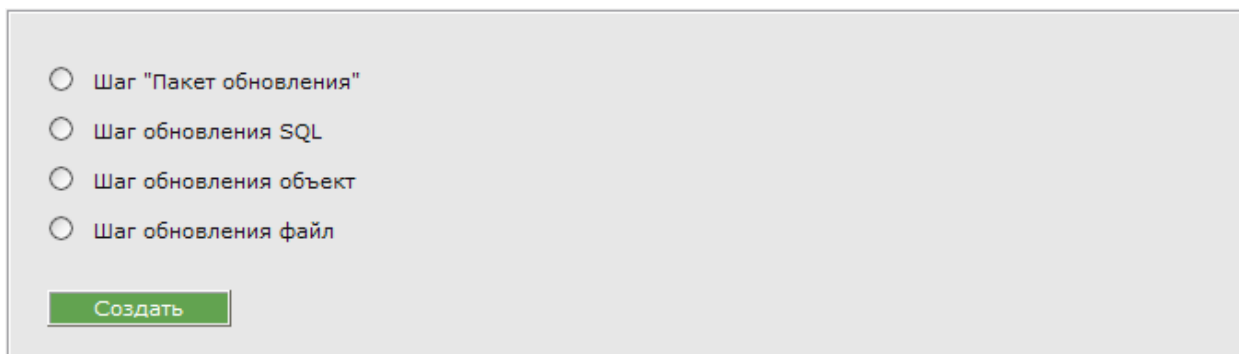
Основной описательной единицей пакета является шаг обновления. Каждый шаг содержит набор независимых объектов, устанавливаемых одновременно.

В случае переноса взаимосвязанных объектов, их необходимо разместить в разных шагах пакета обновления. Причем, зависимый объект необходимо упаковать в шаг с большим порядком упаковки.

При создании шага необходимо указать порядок упаковки и название шага.

В случае наличия двух шагов с одинаковыми порядками упаковки, последовательность их упаковки и распаковки случайна

Шаги добавляются в список нажатием на кнопку создания объекта справа от списка, после чего на экране отображается окно выбора типа шага



Шаг "Пакет обновления"

Шаг обновления SQL

Шаг обновления объект


Шаг обновления файл

Рис. 4. Выбор типа шага пакета обновления

Шаги бывают 4 типов, каждый из которых мы детально рассмотрим далее.

3. СБОР ПАКЕТА ОБНОВЛЕНИЯ

После создания патча или внесения изменений в его структуру, формируется файл, содержащий все данные, описанные в шагах пакета обновления. Сформированный файл может быть передан для обновления других контекстов или помещен в репозиторий на случай восстановления системы.

Формирование пакетов обновления запускается нажатием на кнопку  в панели инструментов формы редактирования пакета.

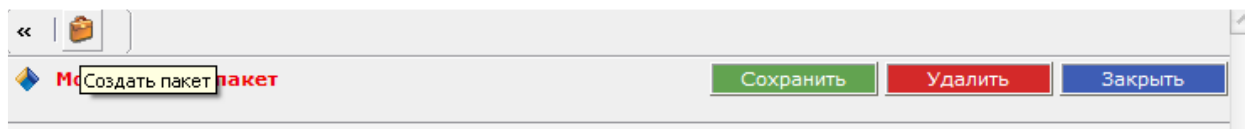


Рис. 51. Утилита формирования файла пакета обновления

В результате формирования пакета формируется лог создания.

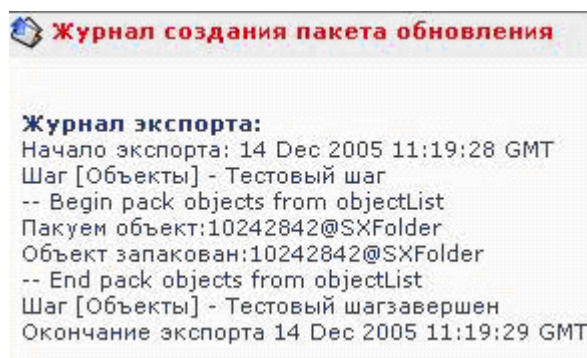


Рис. 62. Журнал формирования файла пакета обновления

После чего на закладке «*Файл пакета обновления*» появляется информация о дате сбора, размер и имя файла.

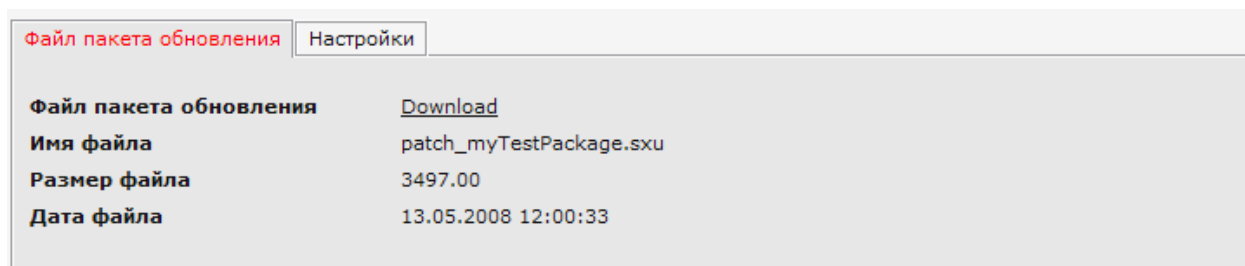



Рис. 73. Закладка «Файл пакета обновления»

Чтобы скачать сформированный пакет, достаточно нажать на ссылку [Download](#) .

После запуска утилиты формирования файла пакета обновления нажмите на кнопку «Сохранить» на форме редактирования пакета для обновления информации на закладке

4. УСТАНОВКА ПАКЕТА ОБНОВЛЕНИЯ

После того, как сформированный пакет обновления может быть установлен администратором на любой другой контекст.

Для чего администратор выбирает папку Администрирование / Пакеты обновлений. В панели инструментов отображается кнопка , нажатием на которую запускается утилита установки пакета обновления. В открывшейся форме необходимо выбрать соответствующий файл с помощью кнопки Browse.

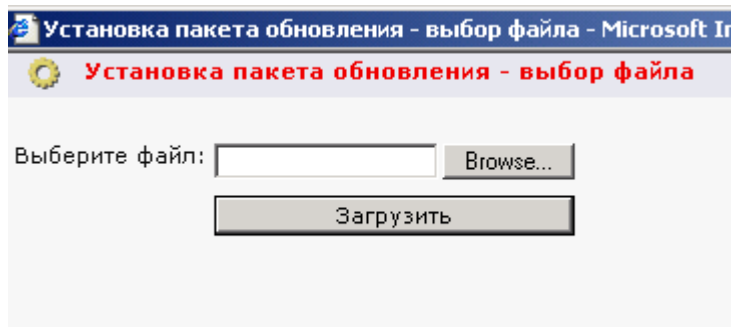


Рис. 8. Загрузка пакета обновления

После выбора файла и нажатия кнопки «Загрузить» появляется окно с полным описанием устанавливаемого пакета: комментарий к пакету, кол-во шагов и порядок их установки.

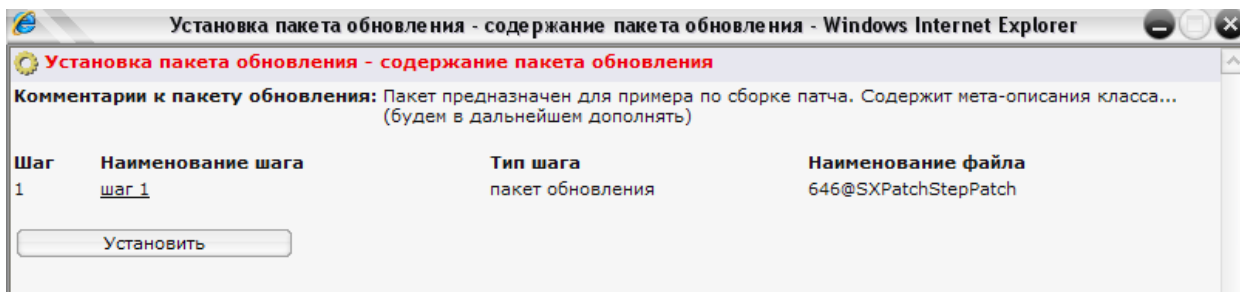


Рис. 9. Информация об устанавливаемом пакете

После ознакомления с изменениями, вносимыми пакетом обновлений, для установки пати необходимо нажать на кнопку «Установить»

После того, как выбранный файл загружен, он автоматически отобразится в папке «Установленные пакеты обновления».

| Установленные пакеты обновления (Найдено записей: 871) | | | | |
|--|-------------------------|-----------------------|-------------------|---|
| « < [1-50] [51- 100] [101- 150] [151- 200] [201- 250] [251- 300] [301- 350] [351- 400] [401- 450] [451- 500] > » | | | | |
| | ▲ Дата установки пакета | Код пакета обновления | Кто установил | Комментарии к пакету |
| <input type="checkbox"/> | 13.05.2008 12:14:26 | myTestPackage | Ульянов Сергей | Пакет предназначен для примера по сборке патча. мета-описания класса... (будем в дальнейшем доп |
| <input type="checkbox"/> | 22.04.2008 | | Ульянов | Пакет предназначен для примера по сборке патча. мета-описания класса... (будем в дальнейшем доп |

Рис. 10. Информация об установленных пакетах

5. ШАГИ ОБНОВЛЕНИЙ

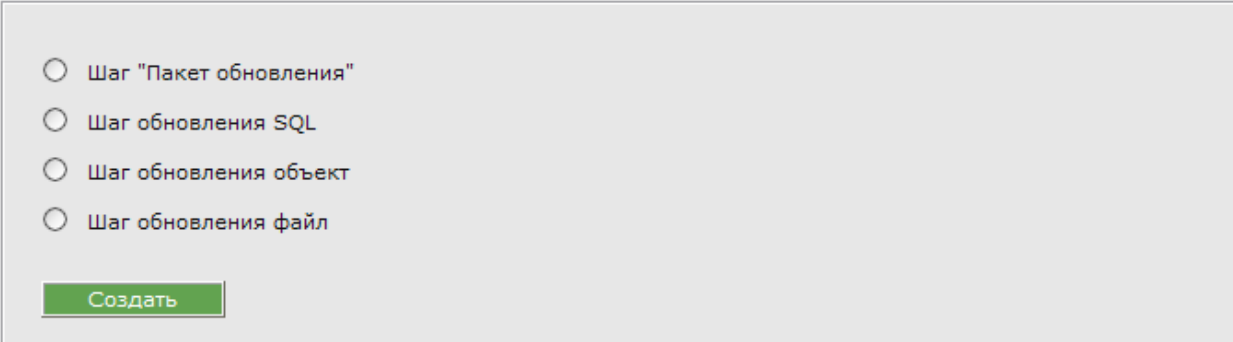
На данный момент существует 4-ре типа шагов обновлений . Каждый шаг содержит набор независимых объектов, устанавливаемых одновременно.

В случае переноса взаимосвязанных объектов, их необходимо разместить в разных шагах пакета обновления. Причем, зависимый объект необходимо упаковать в шаг с большим порядком упаковки.

При создании шага необходимо указать порядок упаковки и название шага.

В случае наличия двух шагов с одинаковыми порядками упаковки, последовательность их упаковки и распаковки случайна

Шаги добавляются в список нажатием на кнопку создания объекта справа от списка, после чего на экране отображается окно выбора типа шага



Шаг "Пакет обновления"

Шаг обновления SQL

Шаг обновления объект

Шаг обновления файл

Рис. 11. Выбор типа шага пакета обновления

5.1. Шаг «Пакет обновлений»

Создание: Шаг "Пакет обновления" Сохранить Закрыть

Пакет *

Порядок упаковки *

Пакет обновления

Файл пакета обновления

Название шага *

Рис. 12. Форма создания шага типа «Пакет обновлений»

Шаг «пакет обновлений» предназначен для переноса других, собранных ранее, пакетов обновлений.

При этом существует два способа добавления имеющихся патчей:

- В поле «*Файл пакета обновления*» можно добавить собранный файл пакета обновления, сохраненный в файловой системы локальной машины
- В поле «*Пакет обновления*» можно добавить описания другого патча, хранящегося на текущем контексте

Таким образом, с помощью шага «Пакет обновлений» можно создать пакет, агрегирующий все имеющиеся ранее пакеты.

5.2. Шаг «SQL»

Создание: Шаг обновления SQL

Сохранить Закреть

Пакет * Мой тестовый пакет

Порядок упаковки * 1

Название шага *

Запрос для БД MSSQL

Запрос для БД Oracle

Запрос для БД DB2

Запрос для БД MySQL

Запрос

Рис. 13. Форма создания шага типа «SQL»

С помощью шага «SQL» на контексте, на который устанавливается пакет, можно выполнять различные SQL-запросы.

При установке система автоматически определяет тип БД и использует SQL-запрос с соответствующей вкладки. Если последняя не заполнена, то, по-умолчанию, используется вкладка «Запрос для БД MSSQL». Поэтому, если вы используете синтаксис или функции, характерные для одного типа БД, то сформируйте аналогичный запрос и для других типов БД

На сам запрос для MSSQL, накладываются некоторые ограничения: нельзя использовать запрос, в котором используется оператор GO

GO не относится ни к SQL, ни к T-SQL. Это прикладная команда, не используемая в JDBC.

5.3. Шаг «Объект»

Создание: Шаг обновления объект

Сохранить Закрыть

Пакет * Мой тестовый пакет

Порядок упаковки * 1

Название шага *

Папка для распаковки

Упаковывать папку

Настройки

Если объект не существует Создать

Если объект существует Обновить

Только те, у которых стоит галочка "Для репликации"

Не упаковывать корневые объекты

Использовать каскадный экспорт

Проверять безопасность для пользователя

Пользователь для проверки безопасности

Инициализировать хранилище

Отключать handler'a при установке пакета обновления

Упаковывать безопасность

Не упаковывать объекты

Не упаковывать содержимое папки

Рис. 14. Форма создания шага типа «Объект»

После ввода необходимых полей и нажатия кнопки «Сохранить» форма примет следующий вид:

Рис. 15. Форма редактирования шага типа «Объект»

Данный шаг имеет больше всего настроек и является самым частоприменяемым:

- *Папка для распаковки* – все объекты, выбранные в списке, будут привязаны к данной папке (если это будет возможным);
- *Упаковывать папку* – если установлена «Папка для распаковки» и установлен данный атрибут, тогда это папка будет тоже упакована;
- *Если объект не существует* – выбор из выпадающего списка (по умолчанию установлена опция «Создать»):
 - Создать;
 - Пропустить;
- *Если объект существует* – выбор из выпадающего списка (по умолчанию установлена опция «Обновить»):
 - Пропустить;
 - Создать (Дублировать);
 - Обновить;
 - Удалить;
- *Список объектов* – выбор объектов для обновления;
- *Список классов* – выбор объекты каких классов буду упакованы. В процессе установки пакета обновлений будут перенесены все экземпляры выбранных классов.

Переносятся только объекты указанных классов. Мета-описание классов перенесено не будет

- Упакуются значения только тех атрибутов, у которых стоит галочка «*Для репликации*»;
- *Не упаковывать корневые объекты* – упакованы будут только объекты, хранящиеся в папках, сами папки не будут упакованы, в противном случае упаковываются и объекты, и папки;
- *Использовать каскадный экспорт* – при выборе папок с объектами упакованы будут не только объекты выбранных папок, но и всех вложенных в них папок. В случае же выбора любого объекта – будут упакованы все объекты, на которые ссылаются атрибуты с признаком «каскадно упаковывать»
- *Проверить безопасность для пользователя* – признак того, что в процессе создания пакета обновления будут проверяться права доступа пользователя, выбранного в следующем поле к упакованным объектам;
- *Пользователь для проверки безопасности* – пользователь, чьи права будут проверяться при создании пакета;
- *Инициализировать хранилище* – если в одном пакете переносится и метаинформация и сами экземпляры класса, тогда на шаге добавления метаинформации, этому атрибуту следует установить положительное значения. После обработки данного шага хранилище с метаинформацией будет инициализировано.

5.4. Шаг «Файл»

Создание: Шаг обновления файл

Сохранить Закрыть

Пакет * Мой тестовый пакет

Порядок упаковки * 1

Название шага *

Настройки

Директории для упаковки

Рис. 16. Форма создания шага типа «Файл»

На данный момент находится в состоянии разработки

7. УТИЛИТА «ДОБАВИТЬ ОБЪЕКТ В ПО»

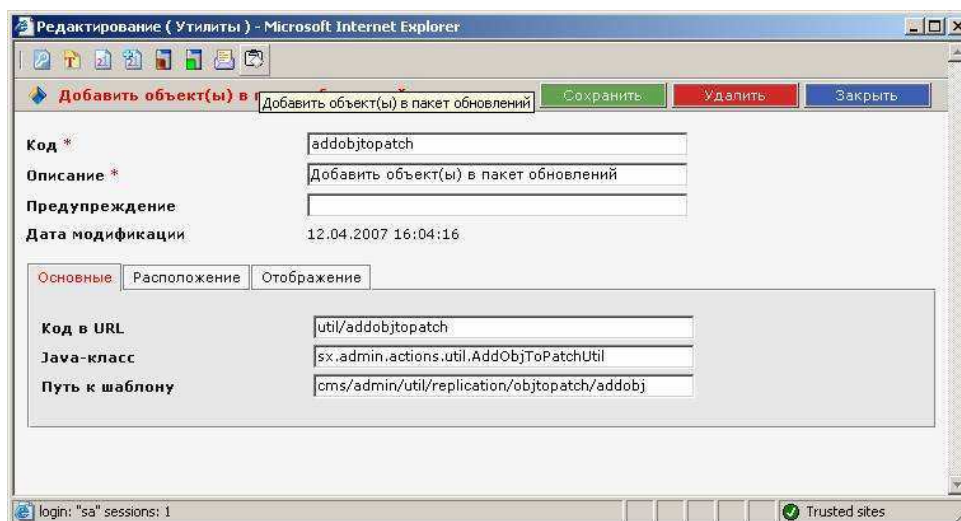


Рис. 17. Утилита добавления объектов в пакет обновлений

В системе существует утилита «Добавления объектов в пакет обновлений», упрощающая работу с пакетами обновлений, но не меняющая логику работы.

Сама утилита располагается на тех объектах, которые указаны на закладке «Расположение» (по умолчанию, на всех папках, атрибутах и мета-описании классов)

Утилита позволяет по факту изменения объекта добавлять его в существующий патч, или создать для него новый. В интерфейс вынесены самые необходимые настройки, более тонкую настройку можно произвести, если зайти в сам патч/шаг.

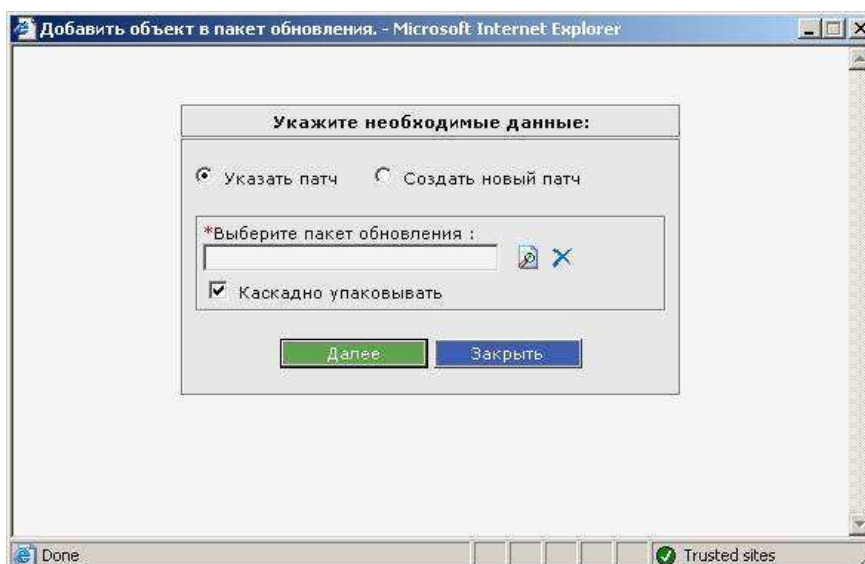
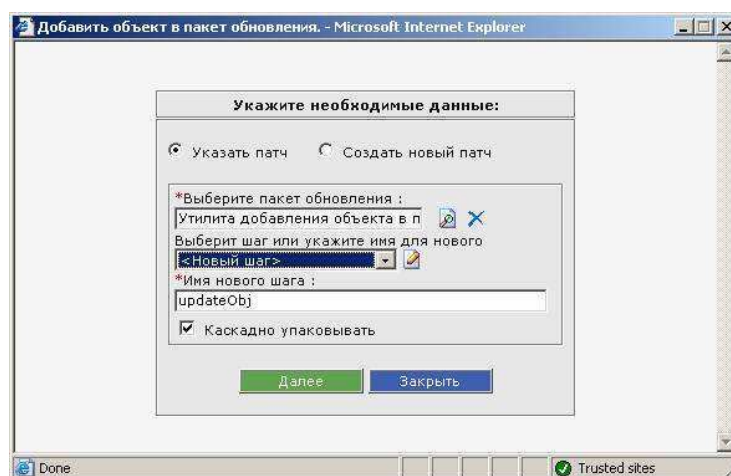
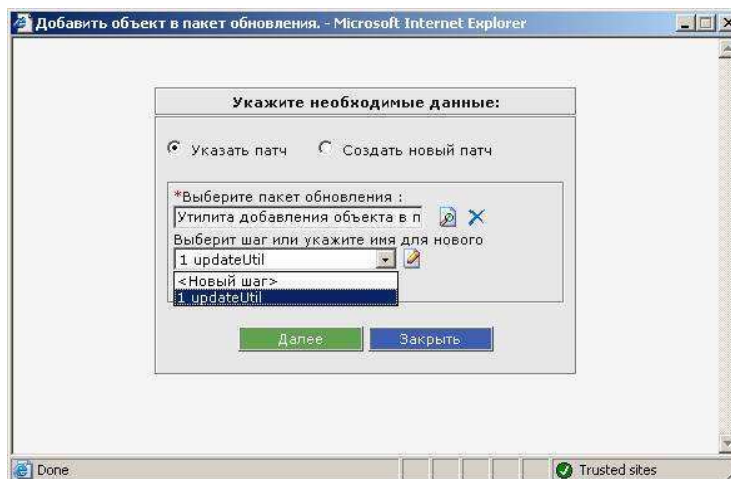


Рис. 18. Ввод параметров в утилите добавления объектов в пакет обновлений

Раздел Указать патч:

- Можно выбрать один из уже имеющихся пакетов обновления в системе
- По факту выбора появляется список имеющихся шагов в патче. Можно добавить в существующий, а можно создать новый шаг.

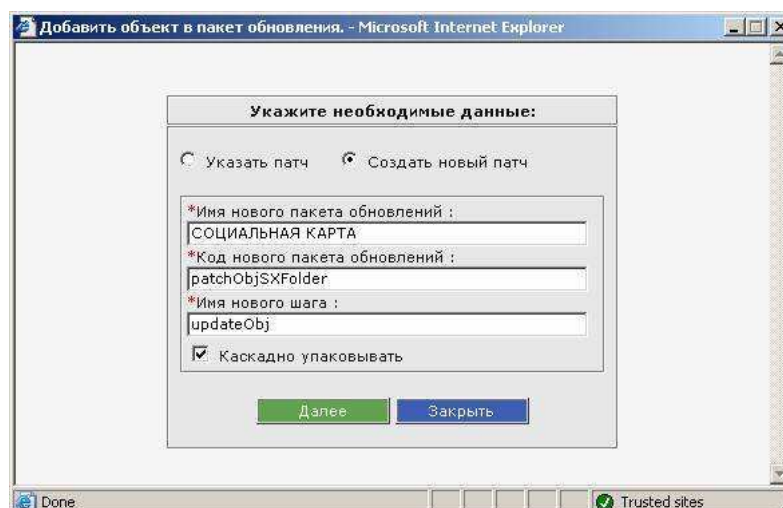


- Каскадность проставляется только для создаваемых шагов, для имеющихся остается как есть.

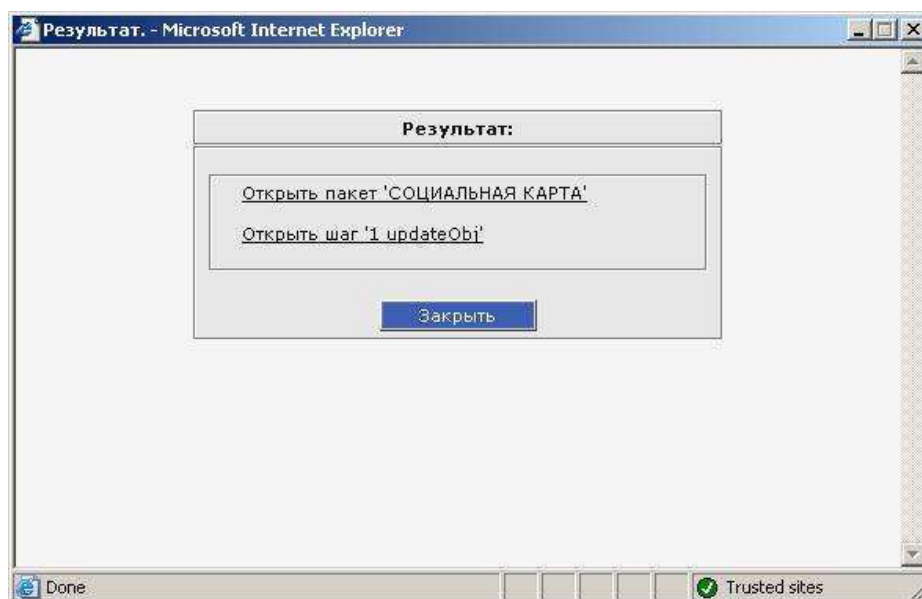
Раздел *Создать новый патч:*

- Система автоматически предлагает имя и код пакета, а также имя нового шага.

Во избежание проблем настоятельно рекомендуется кодовое имя пакета и имена, сгенерированные системой, изменить на другие



После нажатия на кнопку «Далее», системой откроется следующее окно, содержащее ссылки на созданный пакет обновлений и шаг, в который был добавлен выбранный объект.



Далее, если это необходимо можно внести правки в шаг пакета или в сам пакет.

Утилита добавления объекта в пакет обновлений лишь упрощает процесс создания пакета, ответственность за содержимое остается на пользователе.

Из особенностей утилиты также можно отметить:

- Если вы упаковываете атрибут или класс, то появится настройка для создаваемого шага – инициализировать хранилище или нет.
- Системой запоминается последний пакет и шаг, куда были добавлены данные, и автоматически проставляется, при следующем вызове утилиты.
- Для вновь создаваемого шага, его очередность увеличивается на 1.

После завершения работы утилиты не забудьте открыть пакет и собрать его

8. ПРИМЕРЫ НАСТРОЕК ПАКЕТОВ ОБНОВЛЕНИЙ

Эта часть руководства поможет вам максимально быстро и без ошибок собрать пакет обновлений, упаковав все необходимые данные.

Итак, подготовка пакета обновления (патча) всегда состоит из следующих шагов:

- Создание непосредственно пакета. Заполнение его названия, кода и добавление комментариев
- Наполнение пакета шагами различных типов с различными порядками упаковки.
- Сбор пакета с помощью утилиты «Собрать пакет»
- Скачивание итогового файла

\ Независимо от того, какие данные мы упаковываем, эти шаги всегда обязательны.

Особо много вопросов возникает при создании шагов типа «Объект». Шаги этого типа не только самые частоиспользуемые, но и самые сложные в настройке. Следующие примеры призваны помочь при создании шагов этого типа

| |
|--|
| <p>Приведенные примеры не описывают все возможные ситуации, но повторяют многие из них, демонстрируя наиболее частые случаи</p> |
|--|

8.1. Упаковка мета-описания атрибута

В случае упаковки мета-описания одного атрибута шаг типа «Объект» должен выглядеть подобным образом

The screenshot shows a configuration window with the following fields and options:

- Порядок упаковки ***: 1
- Название шага ***: атрибут
- Папка для распаковки**: объект не выбран
- Упаковывать папку**:
- Настройки** (tab):
 - Если объект не существует**: Создать
 - Если объект существует**: Обновить
 - Список объектов**: Код
 - Список классов**: (empty)
 - Только те, у которых стоит галочка "Для репликации"**:
 - Не упаковывать корневые объекты**:
 - Использовать каскадный экспорт**:
 - Проверять безопасность для пользователя**:
 - Пользователь для проверки безопасности**: объект не выбран
 - Инициализировать хранилище**:

Комментарии:

- Атрибут указывается в поле «Список объектов»
- Так как атрибут является частью мета-описания, то при его обновлении необходимо проинициализировать хранилище. Поэтому и стоит признак «Инициализировать хранилище»
- При переносе мета-описания атрибута признак «Каскадно реплицировать» имеет смысл ставить только тогда, если у атрибута указан «Критерий выборки объектов» или атрибут имеет «Перечислимые значения»

8.2. Упаковка мета-описания атрибутов

В случае упаковки мета-описания нескольких атрибута есть два варианта:

Если атрибуты **независимы** друг от друга, то их можно упаковать в один шаг, который будет выглядеть приблизительно так:

Комментарии:

- Атрибуты указываются в поле «Список объектов»
- Так как атрибуты являются частью мета-описания, то при их обновлении необходимо проинициализировать хранилище. Поэтому и стоит признак «Инициализировать хранилище»
- При переносе мета-описания атрибута признак «Каскадно реплицировать» имеет смысл ставить только тогда, если у атрибута указан «Критерий выборки объектов» или атрибут имеет «Перечислимые значения»

Если же атрибуты **зависимы**, то необходимо упаковать в разные шаги. Причем вводимый атрибут необходимо упаковать в шаге с наименьшим порядком упаковки:

| Шаги пакета | |
|------------------|---------------|
| Порядок упаковки | Название шага |
| 1 | атрибут1 |
| 2 | атрибут2 |

порядок упаковки

Название шага * атрибут1

Папка для распаковки объект не выбран

Упаковывать папку

Настройки

Если объект не существует Создать

Если объект существует Обновить

Код

Список объектов

Список классов

Только те, у который стоит галочка "Для репликации"

Не упаковывать корневые объекты

Использовать каскадный экспорт

Проверять безопасность для пользователя

Пользователь для проверки безопасности объект не выбран

Инициализировать хранилище

Название шага * атрибут2

Папка для распаковки объект не выбран

Упаковывать папку

Настройки

Если объект не существует Создать

Если объект существует Обновить

Объектный запрос

Список объектов

Список классов

Только те, у который стоит галочка "Для репликации"

Не упаковывать корневые объекты

Использовать каскадный экспорт

Проверять безопасность для пользователя

Пользователь для проверки безопасности объект не выбран















Инициализировать хранилище

Примеры зависимых атрибутов:

- Если у атрибута1 на закладке «Зависимости» в поле «Редактировать после» указать атрибут2, то сначала:
 - в шаге1 упаковываем атрибут2
 - а в шаге2 упаковываем атрибут1
- Если атрибут1 является «Обратной ссылкой» на класс2 на атрибут2, то
 - в шаге1 упаковываем атрибут2
 - а в шаге2 упаковываем атрибут1

8.3. Упаковка мета-описания класса

В случае упаковки мета-описания одного класса шаг типа «Объект» должен выглядеть подобным образом

| | | |
|---|---|---|
| Название шага * | <input type="text" value="класс"/> | |
| Папка для распаковки | <input type="text" value="объект не выбран"/> |     |
| Упаковывать папку | <input type="checkbox"/> | |
| Настройки | | |
| Если объект не существует | <input type="text" value="Создать"/> |  |
| Если объект существует | <input type="text" value="Обновить"/> |  |
| Список объектов | <input type="text" value="Архивное дело"/> |   |
| Список классов | <input type="text"/> |   |
| Только те, у которых стоит галочка "Для репликации" | <input type="checkbox"/> | |
| Не упаковывать корневые объекты | <input type="checkbox"/> | |
| Использовать каскадный экспорт | <input checked="" type="checkbox"/> | |
| Проверять безопасность для пользователя | <input type="checkbox"/> | |
| Пользователь для проверки безопасности | <input type="text" value="объект не выбран"/> |     |
| Инициализировать хранилище | <input checked="" type="checkbox"/> | |

Комментарии:

- Класс указывается в поле «Список объектов»
- Так как класс является частью мета-описания, то при его обновлении необходимо проинициализировать хранилище. Поэтому и стоит признак «Инициализировать хранилище»
- При переносе мета-описания класса признак «Каскадно реплицировать» ставить нужно практически всегда, т.к. при переносе класса должны упаковаться и атрибуты. Исключение: если мы хотим просто обновить описание класса – название или поле в БД

8.4. Упаковка всех объектов класса

В случае упаковки всех объектов одного класса шаг типа «Объект» должен выглядеть подобным образом

Название шага *

Папка для распаковки

Упаковывать папку

Настройки

Если объект не существует

Если объект существует

Список объектов

Список классов

Только те, у которых стоит галочка "Для репликации"

Не упаковывать корневые объекты

Использовать каскадный экспорт

Проверять безопасность для пользователя

Пользователь для проверки безопасности

Инициализировать хранилище

Комментарии:

- Класс указывается в поле «Список классов».

При этом упаковываются только объекты класса, а само мета-описание класса не упаковывается

- Так как объекты класса не являются мета-информацией, то ставь признак «Инициализировать хранилище» не надо
- Если надо перенести только объекты указанного класса, то признак «Каскадно реплицировать» также ставить не надо. Однако, если необходимо упаковать и объекты, на которые ссылаются атрибуты с признаком «каскадно реплицировать», то этот признак надо указать

8.5. Упаковка мета-описания класса и всех его объектов

В случае упаковки в одном пакете и мета-информации класса и его объектов, необходимо создать два шага, в первый включить только мета-описание класса, а во второй – его объекты

| Шаги пакета | | |
|-------------|------------------|----------------|
| | Порядок упаковки | Название шага |
| » | 1 | класс |
| » | 2 | объекты класса |

Название шага * класс

Папка для распаковки объект не выбран

Упаковывать папку

Настройки

Если объект не существует Создать

Если объект существует Обновить

Архивное дело

Список объектов

Список классов

Только те, у который стоит галочка "Для репликации"

Не упаковывать корневые объекты

Использовать каскадный экспорт

Проверять безопасность для пользователя

Пользователь для проверки безопасности объект не выбран

Инициализировать хранилище

Название шага * объекты класса

Папка для распаковки объект не выбран

Упаковывать папку

Настройки

Если объект не существует Создать

Если объект существует Обновить

Архивное дело

Список объектов

Список классов

Только те, у который стоит галочка "Для репликации"

Не упаковывать корневые объекты

Использовать каскадный экспорт

Проверять безопасность для пользователя

Пользователь для проверки безопасности объект не выбран

Инициализировать хранилище

Комментарии:

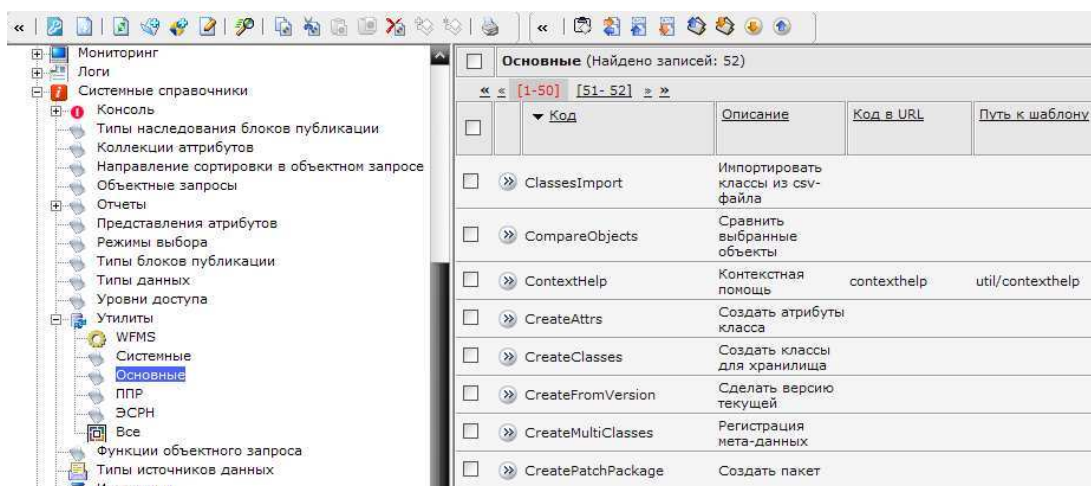
- Мета-информация всегда должна распаковываться раньше объектов, причем после изменения меты необходимо *инициализировать хранилище*.

Упаковывать в один шаг и мета-описание, и объекты класса категорически неправильно

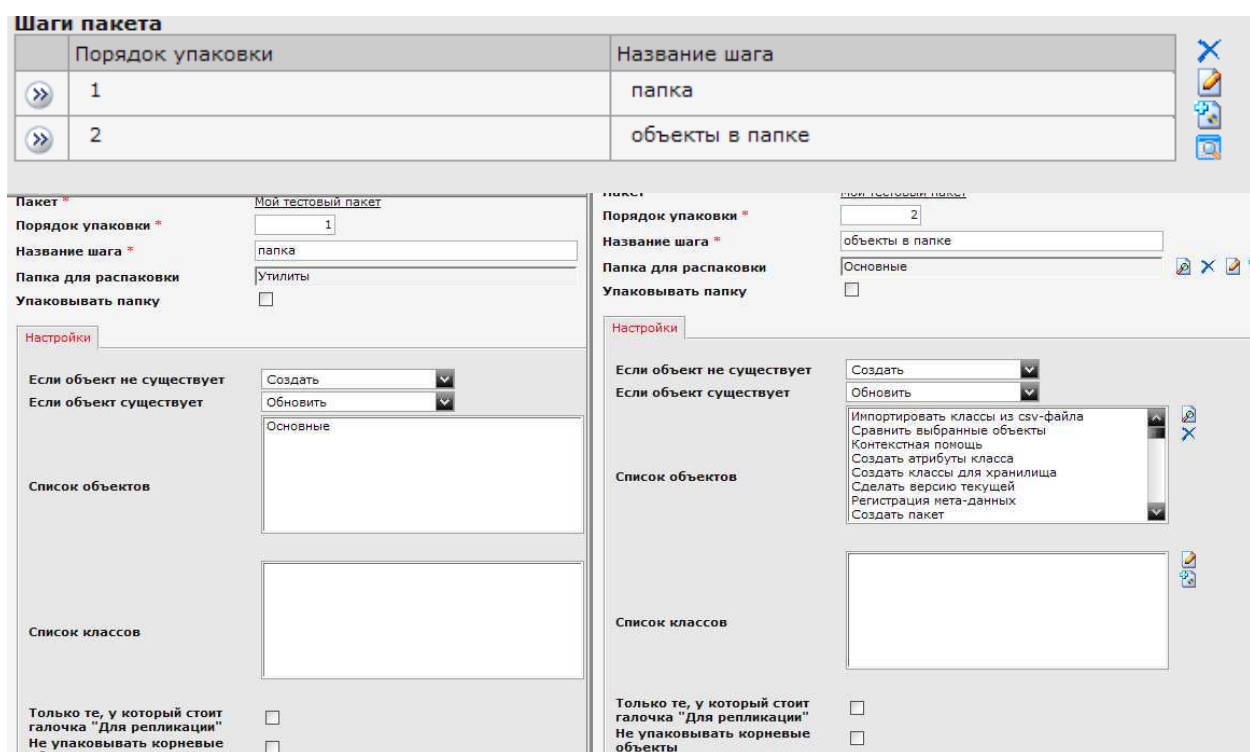
8.6. Упаковка папки с объектами

При упаковке папки со всем содержимым (объектами, входящими в папку) рекомендуется разбить пакет на два шага: в первом упаковать папку, привязав ее к вышестоящей, во втором – упаковать объекты, привязав их к папке

Пример: упакуем папку со всеми основными утилитами



Итоговый пакет содержит два шага



Комментарии:

- Обратите внимание, что в первом шаге папка «Основные» привязывается к вышестоящей «Утилиты». Подразумевается, что на контексте куда будет установлен пакет папке «Утилиты» уже есть – иначе папка «подвиснет» без родителя.

- Второй шаг содержит все объекты в исходной папке.

Если вы не уверены, есть ли на другом контексте необходимая иерархия папок, то упакуйте ее максимально полно

- Оба шага упаковываются без инициализации меты, т.к. они не содержат элементов мета-информации
- Ни в одном шаге не стоит «каскадность». В шаге с папкой она не нужна, в случае же с объектами – мы предполагаем, что по каскадности другие объекты упаковывать не надо

Признак «каскадно упаковывать» сигнализирует о том, что упаковываются все объекты, на которые ссылаются атрибуты типа «каскадно упаковывать»

8.7. Создание пакета на удаление атрибута

Редкое, но очень полезное свойство пакета обновлений – удалять указанные объекты с системы.

Для того, чтобы создать шаг пакета, удаляющий какой-либо объект, необходимо:

- в поле «*список объектов*» указать объект, который будем удалять
- в настройках шага в поле «*если объект не существует*» указать **пропустить**, а в поле «*если объект существует*» - **удалить**
- собранный таким образом пакет позволит удалить указанный объект на других контекстах
- в случае шага на удаление признак «*каскадно реплицировать*» использовать не рекомендуется

Внимание!!! Если объект на контексте, где собирался пакет, тоже удален, то пересобирать такой пакет категорически нельзя

Пример: при создании пакета на удаление атрибута шаг выглядит подобным образом:

The screenshot shows a configuration window for creating a package step. The main configuration area includes:

- Порядок упаковки ***: Input field with value "1".
- Название шага ***: Input field with value "удаление атрибута".
- Папка для распаковки**: Input field with value "объект не выбран".
- Упаковывать папку**: Unchecked checkbox.

The **Настройки** (Settings) section includes:

- Если объект не существует**: Dropdown menu with "Пропустить" selected.
- Если объект существует**: Dropdown menu with "Удалить" selected.
- Список объектов**: A large empty text area with a "Номер" label above it.
- Список классов**: A large empty text area.

At the bottom, there are three unchecked checkboxes:

- Только те, у которых стоит галочка "Для репликации"
- Не упаковывать корневые объекты
- Использовать каскадный экспорт