

Введение

Задачи по выполнению обменов данными между различными базами встают перед 1С программистами довольно часто. Обычно требуется настроить регулярный обмен между несколькими программами, существующими на предприятии, или планируется переход на новую программу и перенос остатков. Эти задачи бывают сложны, с множеством нюансов и скрытых проблем, и разработчики, знакомые с этой областью, всегда в цене.

Сложность задач по переносу данных со временем растет. Появляются новые области, требующие настройки обменов, как, например, решения на мобильной платформе. Поэтому появляются все новые средства, предлагаемые фирмой «1С» для этих целей.

Весной 2015 года фирма «1С» предложила новый программный продукт **«Конвертация данных 3.0»** и технологию обменов через **XDTO-сериализацию**. Эта технология разработана для более удобной реализации обменов между типовыми решениями 1С, а также и для многих других важных задач. Поэтому можно ожидать, что в скором времени она будет использоваться весьма широко.

Занятие 1. Описание технологии и подготовка к переносу данных

Краткий обзор существующих технологий обмена между базами 1С

Существует несколько видов обменов данными между программами 1С. Некоторые из них постепенно устаревают, но большинство существует одновременно, решая свой определенный круг задач.

Технологии обмена традиционно делятся на две группы:

- Обмены между базами с идентичными конфигурациями
- Обмены по правилам

Обмены между базами с идентичными конфигурациями включают в себя:

- Распределенные информационные базы
- Обработка ВыгрузкаЗагрузкаДанныхXML, которая входит в комплект поставки Конвертации данных 2.0.

Распределенные информационные базы – это базы с идентичными конфигурациями, организованные в структуру с четко установленной иерархией. Это облегчает обмен данными, которые, во-первых, не нужно трансформировать, а во-вторых, имеют точно установленный приоритет при загрузке

Обмены по правилам – это такие обмены, которые используют xml-файл правил. В этом файле содержится вся информация о том, как должен изменяться каждый объект при выгрузке и загрузке. Разработчик обмена настраивает файл правил с помощью программы «Конвертация данных, ред. 2.0»

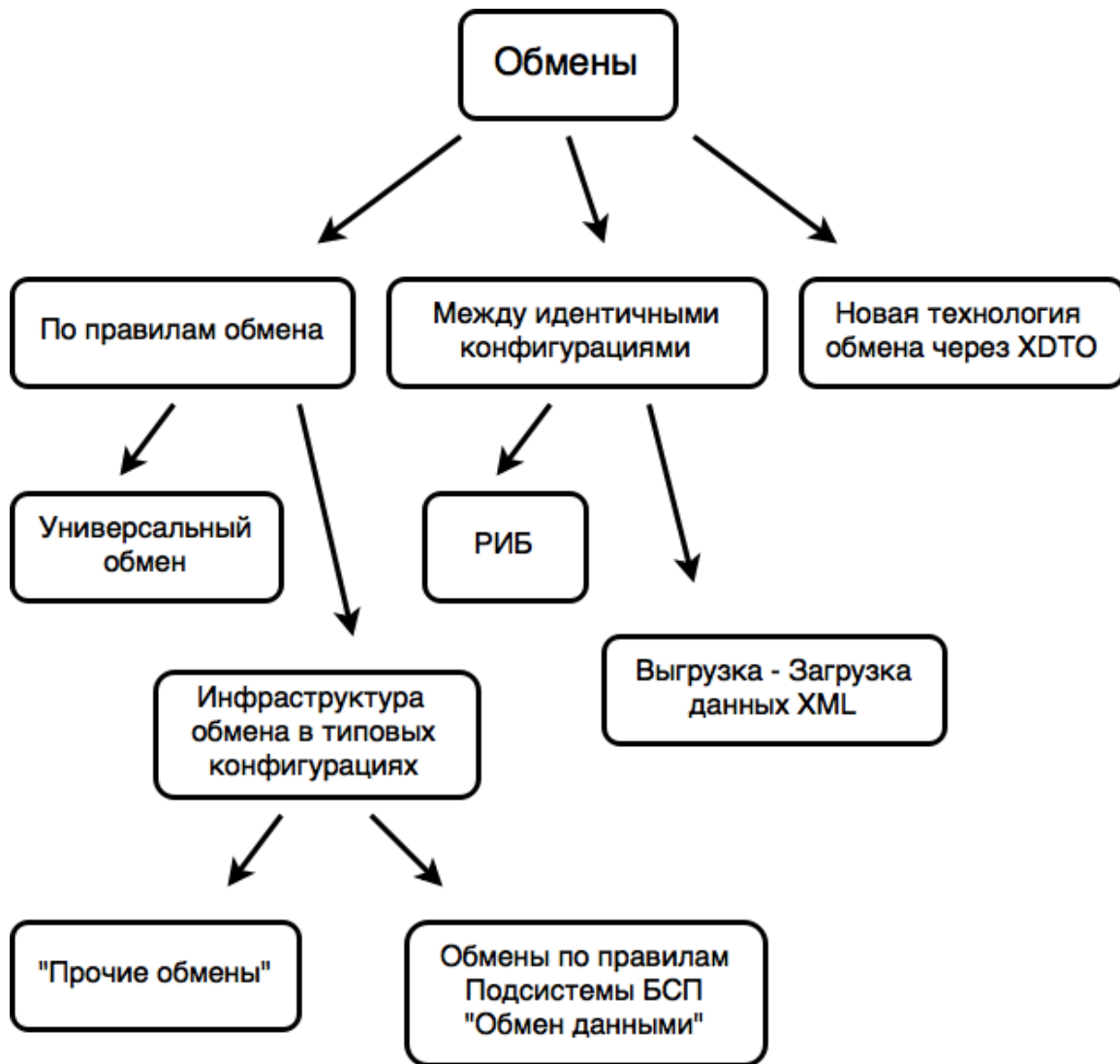
Среди обменов по правилам исторически первым появился Универсальный обмен. Он представляет из себя только одну внешнюю обработку, однако имеет ряд существенных преимуществ перед другими, более сложными обменами.

Универсальный обмен – это очень гибкий и прозрачный механизм. Он, с одной стороны, позволяет решить практически любую задачу по трансформации переносимых данных, включая перенос объектов «один в несколько» и «несколько в один». С другой стороны, механизм его работы довольно прозрачен за счет небольшого количества кода, благодаря чему обмен практически всегда приводит к ожидаемым результатам.

Обмены по правилам, существующие сейчас в типовых конфигурациях, имеют много общего с универсальным обменом. Они интегрированы в типовые решения, включают в себя обработки, формы и общие модули. Их применение более узкое, они были созданы для регулярного обмена объектами между

типовыми конфигурациями, для поддержания в них актуальной информации.

Для удобства пользователя в этих обменах реализовано множество интерфейсных решений, которые позволяют удобно настраивать различные параметры обмена, устанавливать вручную соответствие объектов, и другие.



Новая технология обмена через XDTO стоит в этом ряду несколько особняком. Типовые конфигурации, выпускаемые фирмой «1С», развиваются в направлении унификации объектов и подсистем.

Одинаковые по смыслу объекты в разных конфигурациях приобретают одинаковые названия и аналогичную структуру. Поэтому для новой технологии был разработан формат данных, созданный на основе этих стандартизованных объектов. Данные из типовых конфигураций выгружаются в объекты нового формата и загружаются из них, подвергаясь самым минимальным изменениям.

Преимуществом этой новой технологии является то, что она может применяться не только для обмена между базами на платформе 1С. Ее может использовать любая программа, которая готова работать с объектами заданного формата. Поэтому она может решать широкий круг задач, связанных с миграцией объектов типовых конфигураций 1С.

В данном курсе рассматривается эта новая технология, ее компоненты и методика ее применения.

Создатели новой технологии предполагают два основных варианта ее использования для обмена между базами 1С:

- Регулярный обмен измененными объектами. Этот вариант предполагает использование плана обмена, на узлах которого регистрируются для выгрузки измененные объекты. Этот план обмена тесно интегрирован в механизм обмена и играет в нем важную роль.
- Перенос начальных остатков. Реализация этого варианта сложнее, требует написания большого количества кода и имеет очень мало работающих примеров.

Компоненты обмена

В отличие от обменов по правилам, в обмене через универсальный формат не используется никакого файла правил. Но некоторая трансформация объектов при выгрузке и при загрузке все же возможна. Вместо xml-файла за нее отвечает программный код, который должен находиться в отдельном общем модуле в составе конфигурации. Разработчик обмена сначала настраивает соответствие объектов конфигурации и объектов формата и все возможные трансформации данных с помощью программы «**1С Конвертация данных**» редакции **3.0**, и на основании этой настройки программа сама формирует текст общего модуля.

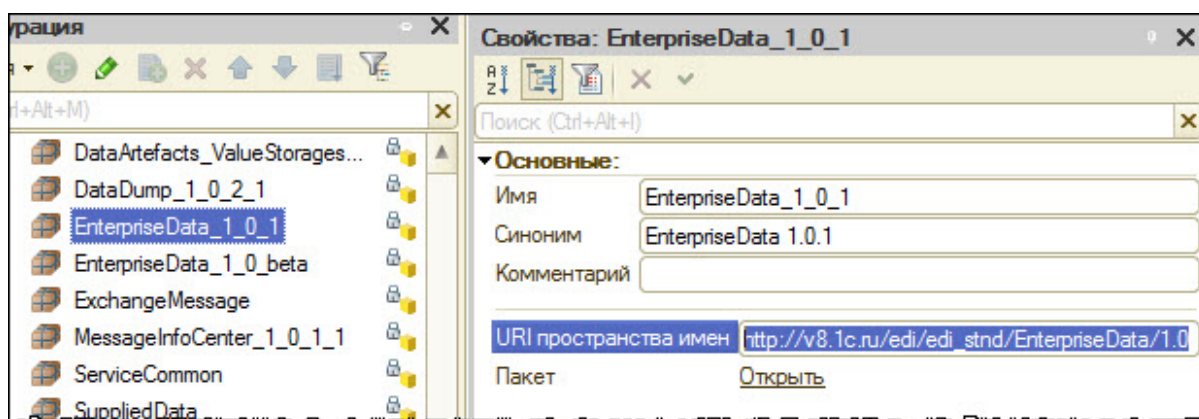
Однако это не все, что необходимо для обмена данными по новой технологии. В конфигурацию базы, участвующей в обмене, должна быть встроена **стандартная подсистема «Обмен данными» библиотеки стандартных подсистем (БСП)**, начиная с версии 2.2.5. В этой подсистеме содержится ряд общих модулей, обработок и форм, образующих инфраструктуру обмена, то есть позволяющих удобно настроить и использовать технологию.

В курсе будет использоваться не только сам файл конфигурации БСП, но и установленная демонстрационная база, которая будет использоваться в качестве примера настройки обмена.

Ну и, наконец, необходима сама **схема формата данных**, на основании которой будет происходить обмен. Она поставляется в виде двух xsd – схем:

- **ExchangeMessage**, в которой описаны базовые типы
- **EnterpriseData**, в которой находится непосредственно список объектов формата.

Эти схемы можно найти в конфигурации Библиотеки стандартных подсистем и в типовых конфигурациях в ветке *XDTO-пакеты*.



Также в каталоге поставки «Конвертации данных» находится обработка MD83Exp.erf, с помощью которой структура метаданных баз, участвующих в обмене, может быть сохранена в файл, чтобы потом загрузить ее в базу Конвертации данных и произвести настройку обмена.

В курсе будут использоваться модельные базы Исходная ИБ и Конечная ИБ, между которыми будет постепенно настроен обмен данными через универсальный формат.

Принятые обозначения

БСП – 1С: Библиотека стандартных подсистем

КД – 1С: Конвертация данных

Конвертация – это набор правил по преобразованию объектов из одной информационной базы в объекты универсального формата и обратно, из объектов формата в объекты этой базы

«Обмен данными XDTO» = «Обмен через универсальный формат» – Новая технология обменов данными с использованием Конвертации данных, ред. 3.0.

ПКО – Правила конвертации объектов

ПКС – Правила конвертации свойств

ПОД – Правила обработки данных

Версии программного обеспечения

При разработке курса использовались следующие версии программных продуктов:

Технологическая платформа 1С версии 8.3.6.2152

1С: Библиотека стандартных подсистем версии 2.3.2.43

1С: Конвертация данных версии 3.0.2.1

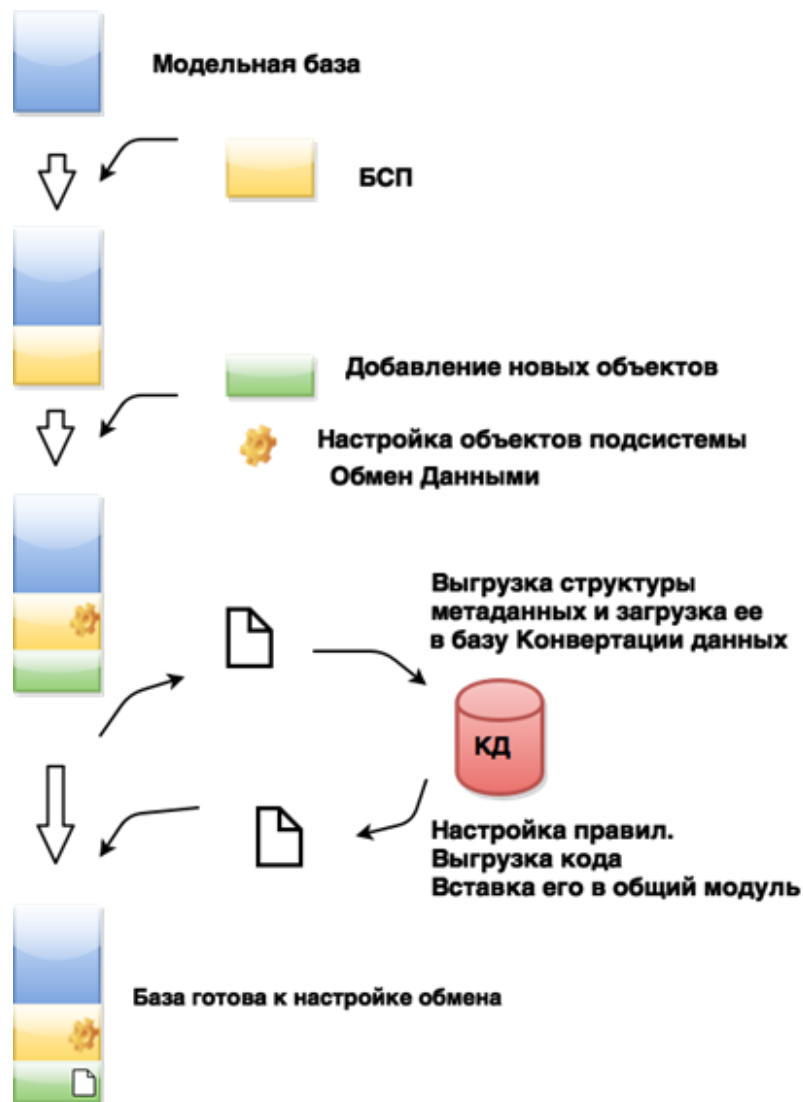
Задача первого модуля

Задача, решаемая в первом модуле курса – это создание одностороннего регулярного переноса измененных объектов из базы Исходная ИБ в базу Конечная ИБ. А также формирование и перенос в том же направлении начальных остатков ТМЦ на заданную дату.

Исходная ИБ и Конечная ИБ – это модельные базы. Они были созданы специально для данного курса и содержат ряд объектов метаданных для демонстрации различных задач по обменам данными.

Выгрузки этих баз можно скачать с сайта: <http://кypсы-по-1c.рф/1c-v8/dataconv-3/startpage/>. Но сразу после скачивания и установки проводить в них обмен через универсальный формат будет невозможно. Для этого обмена необходимо, чтобы конфигурация содержала подсистему БСП *Обмен данными*. Поэтому прежде, чем настраивать правила переноса объектов, необходимо выполнить ряд предварительных действий:

- Объединение конфигураций модельных баз с БСП
- Настройка всех компонент подсистемы Обмен данными
 - План обмена
 - Общие модули
 - Общие команды
 - Подписки на события
- Настройка обновления программы
- Выгрузка структуры метаданных модельных баз и загрузка ее в Конвертацию данных
- Сохранение файлов формата и загрузка их в Конвертацию данных
- Создание в КД логики преобразования объектов и формирование текста общего модуля
- Настройка синхронизации данных



Интеграция с Библиотекой стандартных подсистем

Поскольку модельные базы Исходная ИБ и Конечная ИБ не содержат стандартной подсистемы *Обмен данными*, то перед настройкой обмена необходимо произвести тесную выборочную интеграцию этих баз с Библиотекой стандартных подсистем.

Перед настройкой обмена необходимо:

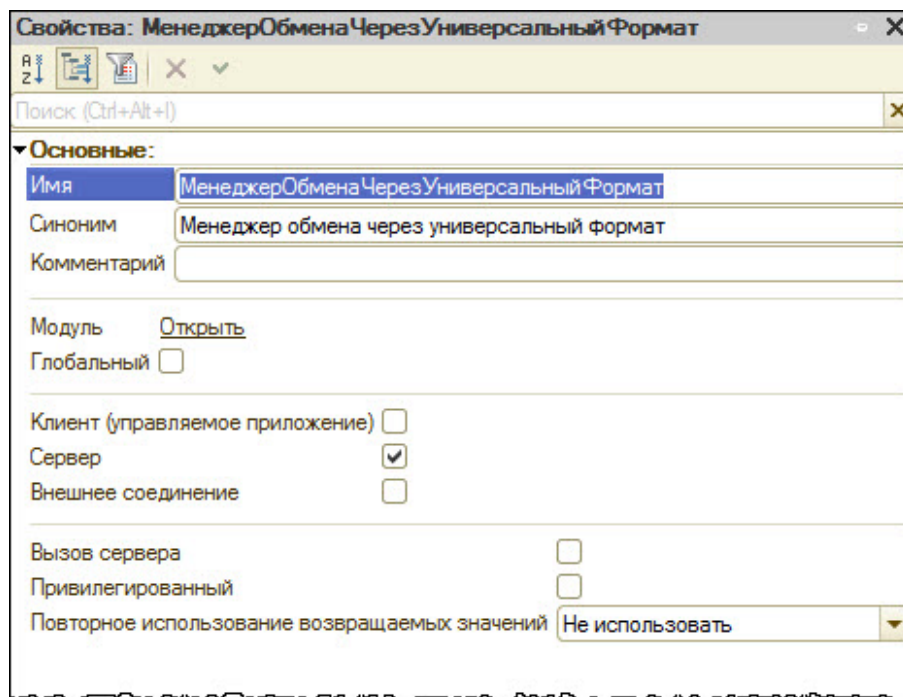
- Скачать с сайта и установить модельные базы Исходная ИБ и Конечная ИБ
- Установить дистрибутивы БСП и КД.
- Создать из шаблона пустую базу Конвертации данных, в которой будет настраиваться логика переноса данных.
- Кроме того, чтобы иметь пример настроенного обмена, нужно создать из шаблона Демонстрационную базу БСП.

В процессе настройки нужно будет выполнить следующие шаги:

1. Создание нового общего модуля для хранения логики преобразований объектов
2. Создание плана обмена и его настройка
3. Отключение режима совместимости
4. Объединение новых объектов в отдельную подсистему
5. Определение состава подсистем, с которыми необходимо произвести интеграцию
6. Объединение конфигурации модельной базы с конфигурацией БСП
7. Установка номера версии конфигурации

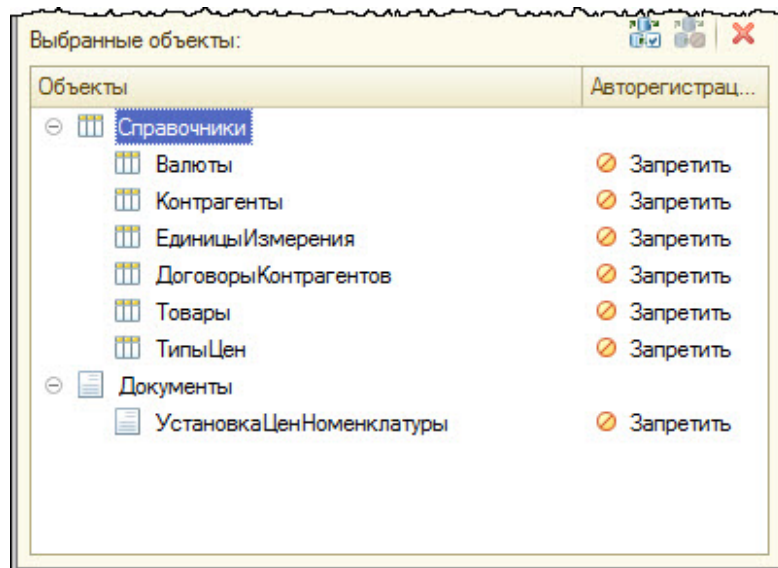
Выполним последовательно все эти действия.

Шаг 1. В конфигураторе Исходной базе создадим новый **общий модуль**, который будет называться **МенеджерОбменаЧерезУниверсальныйФормат**. В него будет помещен код, сформированный в Конвертации данных после настройки логики обмена.

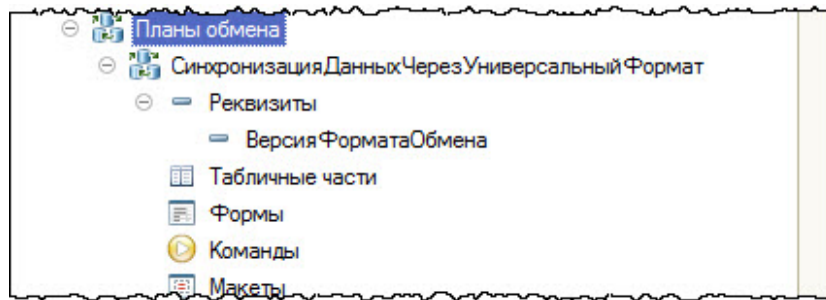


Шаг 2. По условиям задачи обмен данными должен быть регулярным, то есть основанным на плане обмена. Поэтому прежде, чем производить интеграцию с БСП, создадим в конфигурации базы Исходная ИБ новый план обмена.

- Дадим ему имя **СинхронизацияДанныхЧерезУниверсальныйФормат**. Это стандартное имя, которое лучше не изменять.
- Включим в его **состав** ряд объектов нашей конфигурации (авторегистрацию для всех объектов необходимо запретить):
 - Справочники: *Контрагенты, Договоры контрагентов, Единицы измерения, Валюты, Типы цен, Товары*
 - Документ *Установка цен номенклатуры*

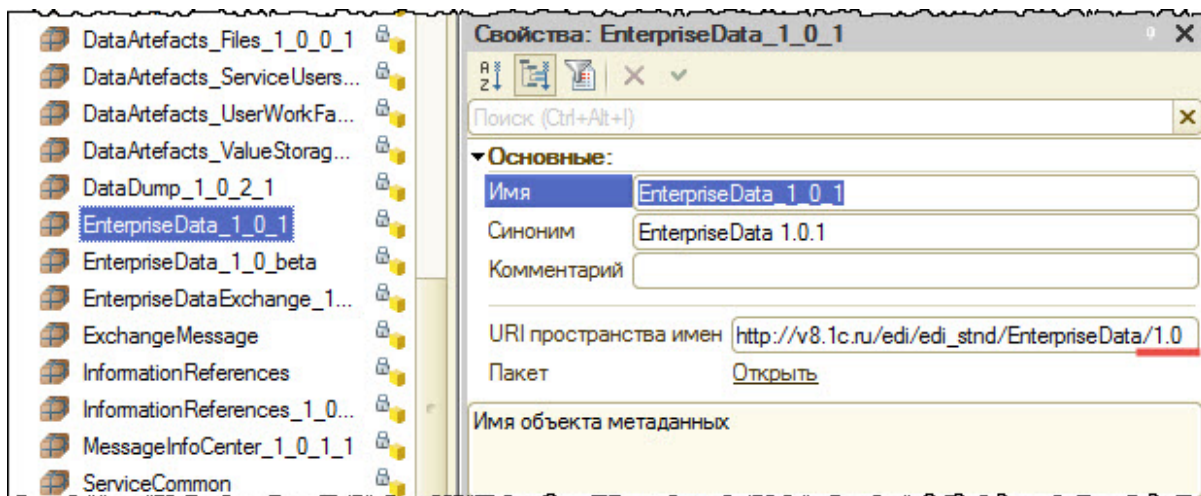


- Откроем Демонстрационную базу БСП в режиме *Конфигуратор* и найдем в ветке *Общие - Планы обмена* нужный нам план обмена, который называется «Демо: Синхронизация данных через универсальный формат». У этого плана обмена есть **реквизит Версия формата обмена**. Это обязательный реквизит, его нужно скопировать в новый план обмена.



- В процессе выполнения обменов используются некоторые процедуры и функции, которые должны располагаться в модуле менеджера плана обмена. Поэтому у нового плана обмена необходимо заполнить **модуль менеджера**. Используем для этого аналогичный план обмена из демонстрационной базы БСП, откроем его модуль менеджера и скопируем его содержимое целиком в модуль менеджера нового плана обмена. Далее необходимо настроить его согласно решаемой задаче.

- Функция *НастройкаОтборовНаУзле()* – необходимо оставить в структуре настроек элемент *ВерсияФорматаОбмена*. Здесь нужно указать ту версию формата EnterpriseData, которую мы планируем использовать. Чтобы посмотреть ее, нужно в конфигураторе БСП в ветке *Общие - XDTO-Пакеты* найти пакет EnterpriseData. Откроем его свойства. В пространстве имен после последнего слеша указан номер версии.



Если таких пакетов несколько, будем использовать тот, у которого номер версии больше. В данном случае это версия «1.0»

функция *НастройкаОтборовНаУзле* (*ВерсияКорреспондента*, *ИмяФормы*, *ИдентификаторНастройки* = "") **Экспорт**

```
СтруктураНастроек = Новый Структура;
СтруктураНастроек.Вставить ("ВерсияФорматаОбмена", "1.0");
Возврат СтруктураНастроек;
```

Конецфункции

- Функции *НастройкаОтборовНаУзлеБазыКорреспондента()*, *ЗначенияПоУмолчаниюНаУзле()*, *ЗначенияПоУмолчаниюНаУзлеБазыКорреспондента()* должны возвращать пустую структуру *СтруктураНастроек*.

функция *НастройкаОтборовНаУзлеБазыКорреспондента* (*ВерсияКорреспондента*, *ИмяФормы*, *ИдентификаторНастройки*) **Экспорт**

```
СтруктураНастроек = Новый Структура;
Возврат СтруктураНастроек;
```

Конецфункции

функция *ЗначенияПоУмолчаниюНаУзле* (*ВерсияКорреспондента*, *ИмяФормы*, *ИдентификаторНастройки*) **Экспорт**

```
СтруктураНастроек = Новый Структура;
Возврат СтруктураНастроек;
```

Конецфункции

функция ЗначенияПоУмолчаниюНаУзлеБазыКорреспондента (ВерсияКорреспондента, ИмяФормы, ИдентификаторНастройки) Экспорт

СтруктураНастроек = Новый Структура;
Возврат СтруктураНастроек;

Конецфункции

- Функции *ОпределитьВариантСинхронизацииДокументов()*, *ОпределитьВариантСинхронизацииСправочников()* – оставить без изменений
- Функция *КраткаяИнформацияПоОбмену()* – Настроим поясняющий текст для отображения при настройке обмена.

функция КраткаяИнформацияПоОбмену (ИдентификаторНастройки = "") Экспорт

ПоясняющийТекст = "";

Если ИдентификаторНастройки = "ТолькоОтправка" **Тогда**

ПоясняющийТекст = НСтр("ru = 'Позволяет отправлять данные в программу | Конечная ИБ'");

ИначеЕсли ИдентификаторНастройки = "ТолькоПолучение" **Тогда**

ПоясняющийТекст = НСтр("ru = 'Позволяет получать данные из программы | Конечная ИБ'");

ИначеЕсли ИдентификаторНастройки = "Двухсторонний" **Тогда**

ПоясняющийТекст = НСтр("ru = 'Позволяет синхронизировать данные между | программами Исходная ИБ и Конечная ИБ'");

Иначе

ПоясняющийТекст = НСтр("ru = 'Позволяет синхронизировать данные с любой | программой, поддерживающей универсальный формат обмена'");

КонецЕсли;

Возврат ПоясняющийТекст;

Конецфункции

- Функции *ПодробнаяИнформацияПоОбмену()*, *ОписаниеОграниченийПередачиДанных()*, *ОписаниеЗначенийПоУмолчанию()*, *ОписаниеОграниченийПередачиДанныхБазыКорреспондента()*, *ОписаниеЗначенийПоУмолчаниюБазыКорреспондента()*, *ПояснениеДляНастройкиПараметровУчета()*, *ПояснениеДляНастройкиПараметровУчетаБазыКорреспондента()* должны возвращать пустую строку
- Процедуры *ОбработчикПроверкиПараметровУчета()*, *НастроитьИнтерактивнуюВыгрузку()*, Функция *ПредставлениеОтбораИнтерактивнойВыгрузки()* – очистить содержимое, это «заглушки».
- Функции *ПараметрыВыгрузкиПоУмолчанию()*, *ОрганизацииОтбораИнтерактивнойВыгрузки()* – должны возвращать пустую коллекцию *Результат*

функция ПараметрыВыгрузкиПоУмолчанию (Получатель)

Результат = Новый Структура;

Возврат Результат;

Конецфункции

функция ОрганизацииОтбораИнтерактивнойВыгрузки (Знач ТаблицаОтбора) Экспорт

Результат = Новый СписокЗначений;

Возврат Результат;

Конецфункции

- Процедура *ДобавитьСписокОрганизаций()* – можно удалить
- Процедуры *ПриПодключенииККорреспонденту()*, *ПриОтправкеДанныхОтправителя()*, *ПриПолученииДанныхОтправителя()* – очистить содержимое
- Процедура *ОпределитьНастройку()*. В ней задаются настройки, используемые при создании нового обмена данными. Нужно описать все заголовки пунктов меню выпадающего списка и установить параметры *ОтображатьНастройкуОтборовНаУзле* и *ОтображатьЗначенияПоУмолчанию* в значение *Ложь*

Настройки. ПредупреждатьОНесоответствииВерсийПравилОбмена = Ложь;

```

Настройки.Вставить ("ЭтоПланОбменаХДТО", Истина);

Если ИдентификаторНастройки = "ТолькоОтправка" Тогда
    Настройки.Вставить ("ЗаголовокКомандыДляСозданияНовогоОбменаДанными",
        НСтр("ru = 'Отправка данных'"));
    Настройки.Вставить ("ЗаголовокПомощникаСозданияОбмена",
        НСтр("ru = 'Отправка данных в Конечную ИБ'"));
    Настройки.Вставить ("ЗаголовокУзлаПланаОбмена",
        НСтр("ru = 'Отправка данных в Конечную ИБ'"));

ИначеЕсли ИдентификаторНастройки = "ТолькоПолучение" Тогда

    Настройки.Вставить ("ЗаголовокКомандыДляСозданияНовогоОбменаДанными",
        НСтр("ru = 'Получение данных'"));
    Настройки.Вставить ("ЗаголовокПомощникаСозданияОбмена",
        НСтр("ru = 'Получение данных из Конечной ИБ'"));
    Настройки.Вставить ("ЗаголовокУзлаПланаОбмена",
        НСтр("ru = 'Получение данных из Конечной ИБ'"));

ИначеЕсли ИдентификаторНастройки = "Двухсторонний" Тогда

    Настройки.Вставить ("ЗаголовокКомандыДляСозданияНовогоОбменаДанными",
        НСтр("ru = 'Полная синхронизация'"));
    Настройки.Вставить ("ЗаголовокПомощникаСозданияОбмена",
        НСтр("ru = 'Синхронизация данных с Конечной ИБ'"));
    Настройки.Вставить ("ЗаголовокУзлаПланаОбмена",
        НСтр("ru = 'Синхронизация данных с Конечной ИБ'"));

Иначе
    Настройки.Вставить ("НаименованиеКонфигурацииКорреспондента",
        НСтр("ru = 'Через универсальный формат обмена'"));
КонецЕсли;

Настройки.ОтображатьЗначенияПоУмолчаниюНаУзле = Ложь;
Настройки.ОтображатьНастройкуОтборовНаУзле = Ложь;
Настройки.ВариантыНастроекОбмена = МассивДоступныхВариантовНастроекОбмена();

```

- Функции *ИмяФайлаНастроекДляПриемника()*, *ИмяФормыСозданияНачальногоОбраза()* оставить без изменений
- Функция *ИспользоватьПомощникСозданияОбменаДанными()* должна возвращать значение *Истина*.
- Функции *ПланОбменаИспользуетсяВМоделиСервиса()*, *КорреспондентВМоделиСервиса()* должны возвращать значение *Ложь*
- Функции *ИспользуемыеТранспортыСообщенийОбмена()*, *ОбщиеДанныеУзлов()* – оставить без изменения
- Функция *ИмяКонфигурацииИсточника()* – оставить вызов процедуры *УстановитьПривилегированныйРежим()*. Функция должна возвращать значение *Метаданные.Имя*

```

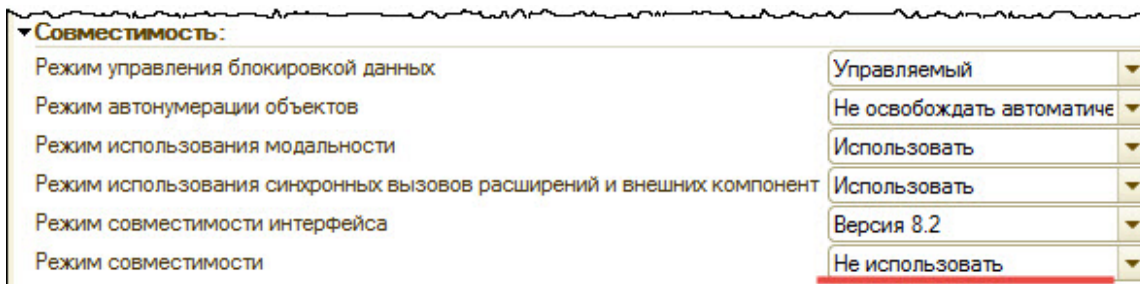
УстановитьПривилегированныйРежим (Истина);
Возврат Метаданные.Имя;

```

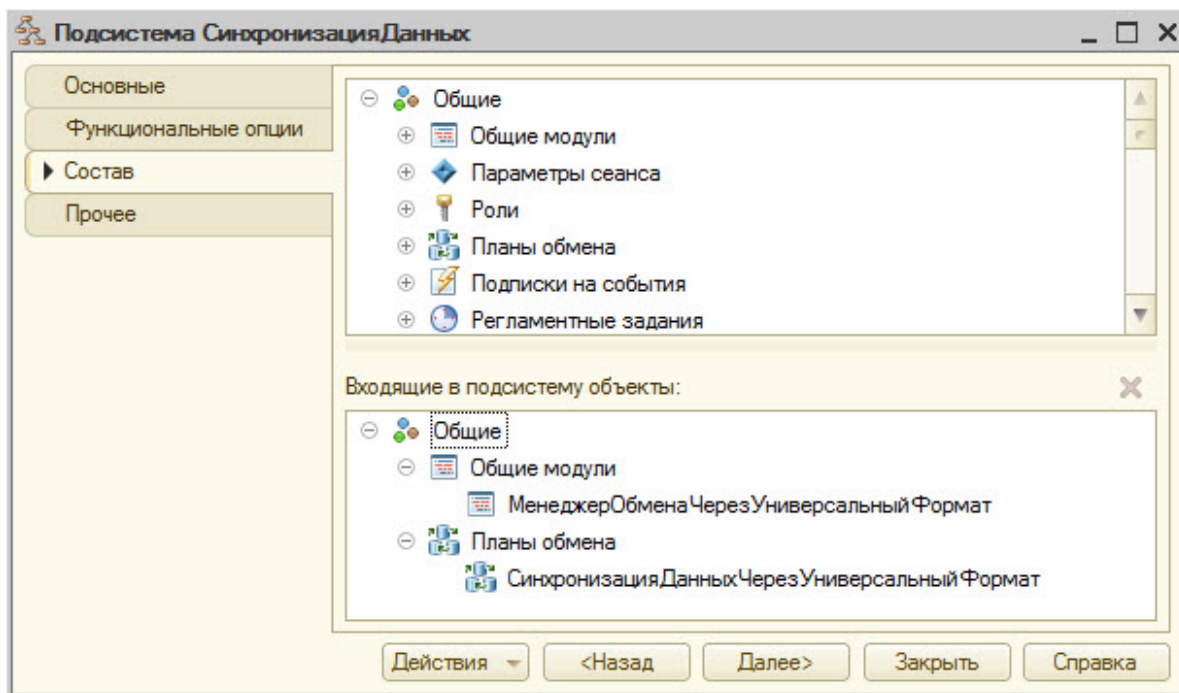
- Функция *МассивДоступныхВариантовНастроекОбмена()* – оставить без изменений
- Функция *ФорматОбмена()* должна возвращать строку пространства имен формата без указания версии (то есть до последнего слеша). Пространство имен указано в свойствах ХДТО-пакета *EnterpriseData*. Его можно найти в демонстрационной базе БСП.
- Процедура *ПолучитьВерсииФорматаОбмена()* – здесь требуется указать, какой общий модуль будет содержать логику обмена через универсальный формат определенной версии. Дело в том, что база может обмениваться данными, используя разные версии формата, и каждой версии может соответствовать отдельный общий модуль, в котором реализована отдельная логика. По умолчанию в процедуре добавляются две версии обмена, «1.0.beta» и «1.0». В курсе будет использоваться версия 1.0, но в данном случае можно оставить их обе. Общий модуль нужно указать тот, что уже был создан на шаге 1 – *МенеджерОбменаЧерезУниверсальныйФормат*.
- Проверим модуль на синтаксические ошибки и выполним сохранение. Система может выдать предупреждение о том, что план обмена не включен ни в одну подсистему. Чуть позже мы это исправим.

Шаг 3. В настройках модельных баз включен режим совместимости с версией 8.2. Стандартные

подсистемы версии 2.3 для работы требуют отключенного режима совместимости. Для этого нужно двойным щелчком по корневому узлу конфигурации открыть окно *Свойства* и установить свойство *Режим совместимости* в значение *Не использовать*.



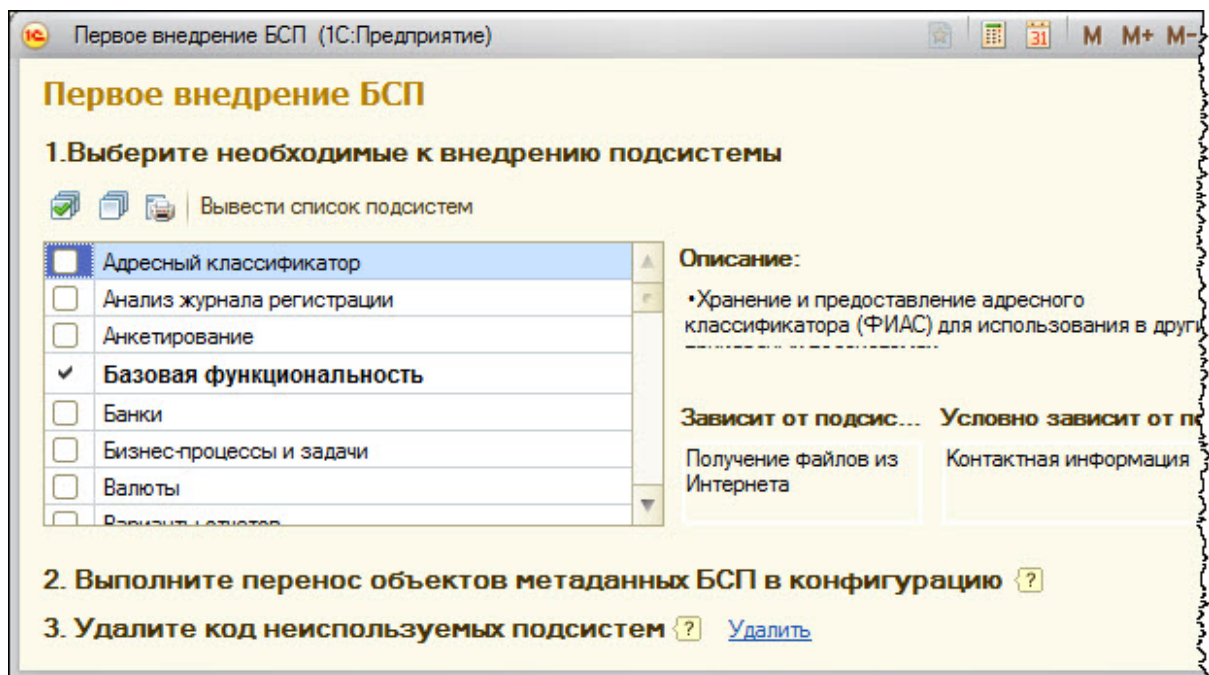
Шаг 4. Объединим новые объекты в отдельную подсистему. Назовем ее **СинхронизацияДанных**. Настроим ее состав, пока в нем будет только план обмена и общий модуль *Менеджер обмена через универсальный формат*. Откроем командный интерфейс и снимем флажок *Видимость* напротив плана обмена.



Шаг 5. Далее откроем базу Исходная ИБ в пользовательском режиме. В каталоге шаблонов БСП есть инструмент, который может упростить процесс интеграции подсистем БСП. Это внешняя обработка, она называется «Первое внедрение БСП».

Но для ее работы необходим общий модуль *СтроковыеФункцииКлиентСервер*. Его нужно скопировать из Демонстрационной базы БСП в Исходную конфигурацию. И после этого сохранить и перезапустить ее.

Найдем обработку в каталоге шаблонов и откроем. Для этого выберем пункт *Главного меню Файл – Открыть*, найдем путь к каталогу шаблонов БСП, откроем каталог ExtFiles и откроем файл *ПервоеВнедрениеБСП.erf*.

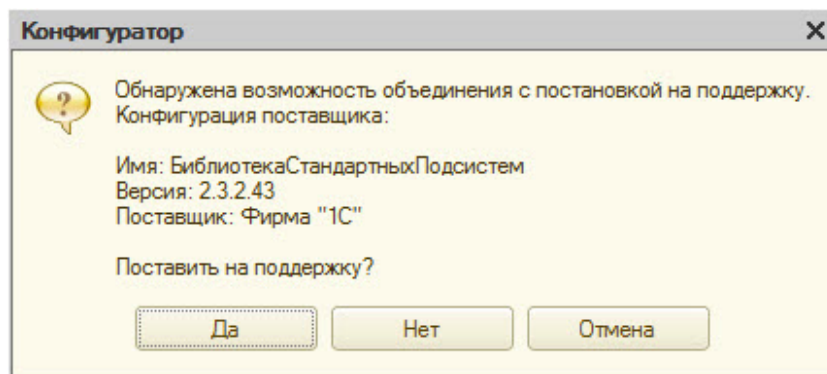


На первом этапе работы этой обработки предлагается выбрать те подсистемы, с которыми будет происходить интеграция. При этом обязательные подсистемы «**Базовая функциональность**» «**Обновление версии ИБ**» и «**Пользователи**» уже отмечены флажками.

Отмечаем подсистему «**Обмен данными**». Одновременно флажок установился и напротив подсистемы «**Работа с почтовыми сообщениями**». Это означает, что эту подсистему тоже необходимо интегрировать для корректной работы обменов.

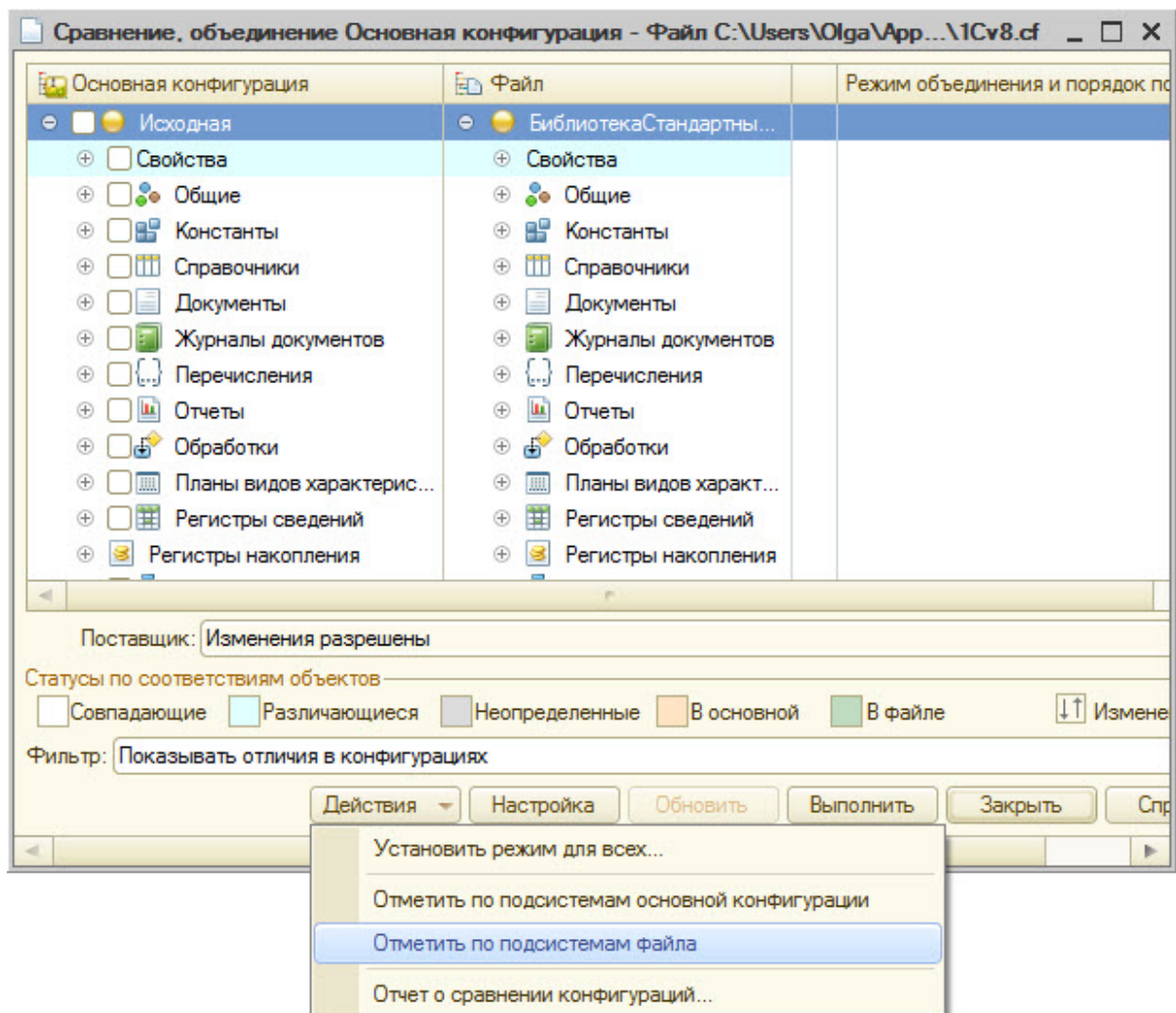
Шаг 6. Закроем пользовательский режим и перейдем к объединению с конфигурацией БСП. В Исходной ИБ в Конфигураторе выберем в *Главном меню* пункт *Конфигурация – Сравнить, объединить с конфигурацией из файла...* и снова укажем путь к каталогу шаблонов БСП. Выберем для объединения файл конфигурации 1Cv8.cf.

Программа предлагает встать на поддержку.

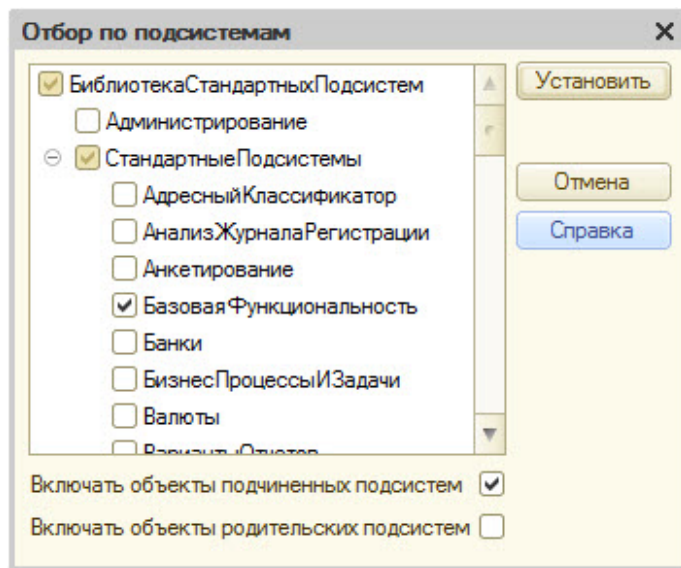


Следует выбрать вариант *Да*, чтобы впоследствии иметь возможность обновлять стандартные подсистемы на новые версии.

В открывшемся окне снимем флажок с корневого узла. Нажмем внизу окна кнопку *Действия* и в выпадающем меню выберем пункт *Отметить по подсистемам файла*.



В появившемся окне также снимем флажок с корневого элемента и выберем те подсистемы, которые необходимо интегрировать:



- Базовая функциональность
- Обмен данными
- Обновление версии ИБ
- Пользователи
- Работа с почтовыми сообщениями

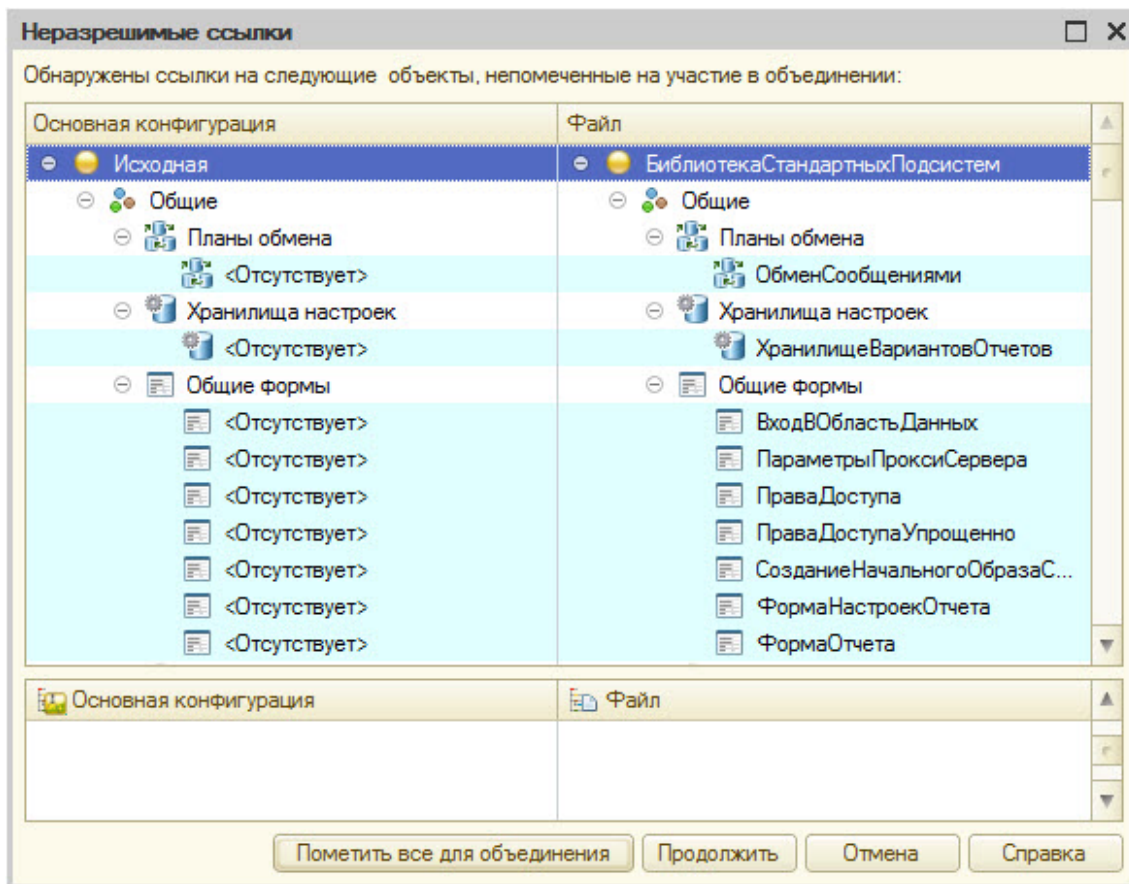
Нажмем кнопку *Установить*. Программа автоматически расставила флажки у объектов, входящих в эти 5 подсистем.

Откроем ветку *Общие – Подсистемы – Стандартные подсистемы* и проверяем, установлены ли там флажки напротив нужных подсистем. Если нет, устанавливаем.

Откроем ветку *Свойства* и отметим флажками *Основные роли*, *Модуль управляемого приложения* и *Модуль сеанса*.

Нажмем кнопку *Выполнить*.

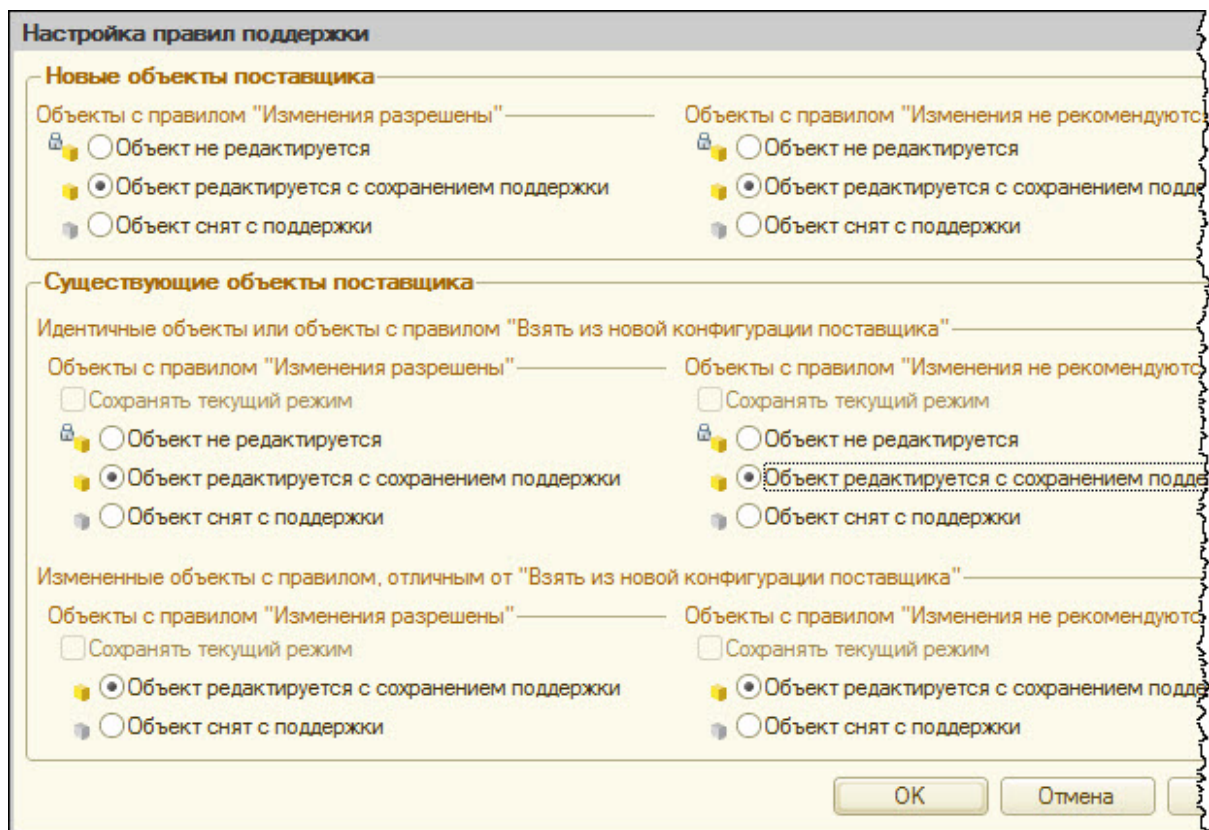
В появившемся окне программа сообщает, что в выбранных для интеграции объектах существуют ссылки на другие объекты, которые не были выбраны.



Эта ситуация не приведет к ошибкам. Нажмем кнопку *Продолжить*.

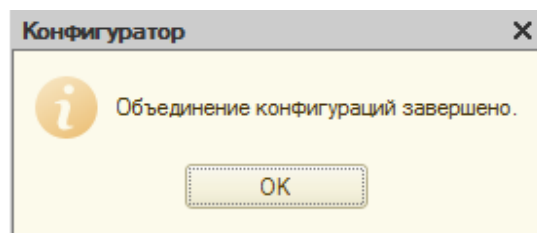
Следующее окно касается настройки правил поддержки.

При выполнении реальных задач необходимо очень внимательно подходить к установке правил поддержки объектов. Скорее всего, необходимо будет оставить все переключатели в режиме *Объект не редактируется*, и включать для них возможность изменения при необходимости точно. Но, имея в виду, что решается учебная задача, и для экономии времени установим все переключатели в положение *Объект редактируется с сохранением поддержки*.



Нажмем кнопку *OK*.

Система подтверждает, что объединение успешно завершено.



Шаг 7. Установим **версию конфигурации**. Снова двойным кликом по корневому элементу дерева конфигурации откроем окно *Свойства*, найдем свойство *Версия* и впишем туда значение «1.0.0.0».

На этом интеграция Стандартных подсистем в Исходную ИБ успешно завершена. Теперь необходимо выполнить настройку всех компонентов подсистем.

Настройка компонент подсистемы Обмен данными

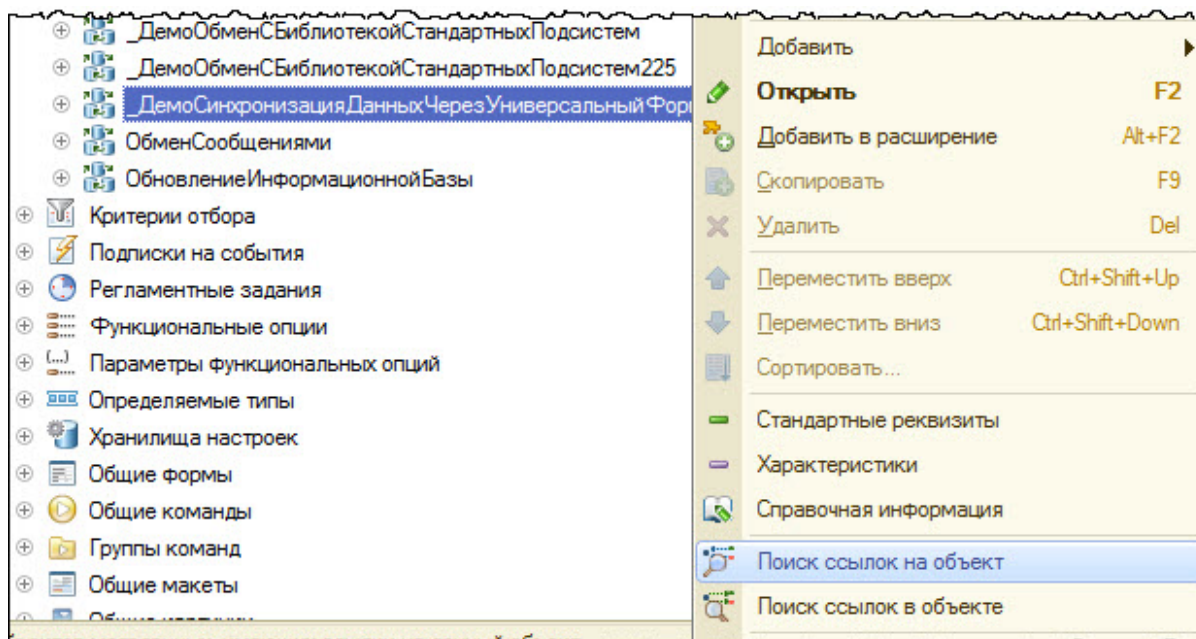
В этом уроке нам предстоит настроить следующие компоненты:

- Общие команды
- Подписки на события
- Общий модуль ОбменДаннымиПереопределяемый
- Подсистема Синхронизация данных

Выполним эти настройки.

Шаг 1. Первым делом настроим **общие команды**. При объединении с конфигурацией БСП ряд общих команд был добавлен в Исходную ИБ. У части из них необходимо установить тип параметра команды, чтобы они были доступны из формы узла плана обмена.

Чтобы узнать, какие именно команды надо изменить, откроем Демонстрационную базу БСП и найдем в ней план обмена для синхронизации через универсальный формат. Кликнем по нему правой кнопкой мыши и из *Контекстного меню* выберем пункт *Поиск ссылок на объект*.



В служебном окне появился список объектов. Из него необходимо выбрать все общие команды, скопировать их в буфер обмена и вставить в новый текстовый документ в базе «Исходная ИБ».

- Загрузить Правила Регистрации Объектов
- Настроить Параметры Транспорта Сообщений Обмена
- Получить Настройки Синхронизации Для Другой Программы
- Синхронизировать
- Синхронизировать С Дополнительными Параметрами
- События Отправки
- События Получения
- Состав Отправляемых Данных
- Сценарии Синхронизации
- Удалить Настройку Синхронизации

Каждую команду из этого списка нужно найти в ветке конфигурации *Общие – Общие команды* и в ее свойствах указать **тип параметра команды** – *План обмена. Синхронизация Данных Через Универсальный Формат* (Для этого удобно закрепить окно дерева конфигурации и окно свойств).

Шаг 2. Чтобы в Исходной ИБ происходил регулярный обмен данными, необходимо, чтобы измененные объекты регистрировались к выгрузке. Для этого нужно настроить **подписки на события**. Обратимся к Демонстрационной базе БСП и найдем в ветке конфигурации *Общие – Подписки на события* те подписки, которые относятся к плану обмена для синхронизации через универсальный формат:

- *Демо Синхронизация Данных Через Универсальный Формат Зарегистрировать Изменение*
- *Демо Синхронизация Данных Через Универсальный Формат Зарегистрировать Изменение Документа*
- *Демо Синхронизация Данных Через Универсальный Формат Зарегистрировать Удаление*

Скопируем каждую из них в Исходную ИБ, изменим имя, убрав префикс *Демо*. Укажем источник событий – это должны быть те объекты, которые входят в состав плана обмена.

У подписки *Зарегистрировать Изменение* – справочники: *Контрагенты, Договоры контрагентов, Единицы измерения, Валюты, Типы цен, Товары*.

У подписки *Зарегистрировать Изменение Документа* – документ *Установка цен номенклатуры*

У подписки *Зарегистрировать Удаление* – все эти объекты.

Создадим общий модуль *Обработчик Подписок На События*, и выберем его в свойстве *Обработчик* каждой из подписок. Тексты обработчиков необходимо также скопировать из Демонстрационной базы БСП, изменив имя плана обмена в коде обработчиков.

В Конечной ИБ подписки на события можно не настраивать, так как из нее по условиям задачи выгрузка производится не будет.

Шаг 3. Осталось настроить общий модуль *Обмен Данными Переопределяемый*. Найдем его в Исходной ИБ и откроем. В нем есть две весьма важные процедуры:

Процедура *ПриОпределенииПрефиксаИнформационнойБазыПоУмолчанию()* – в ней необходимо задать значение параметра Префикс:

```
Префикс = "ИБ" ; // в Исходной базе.
```

```
Префикс = "КБ" ; // в Конечной базе.
```

Процедура *ПолучитьПланыОбмена()* – в ней в коллекцию *ПланыОбменаПодсистемы* необходимо добавить нужный план обмена:

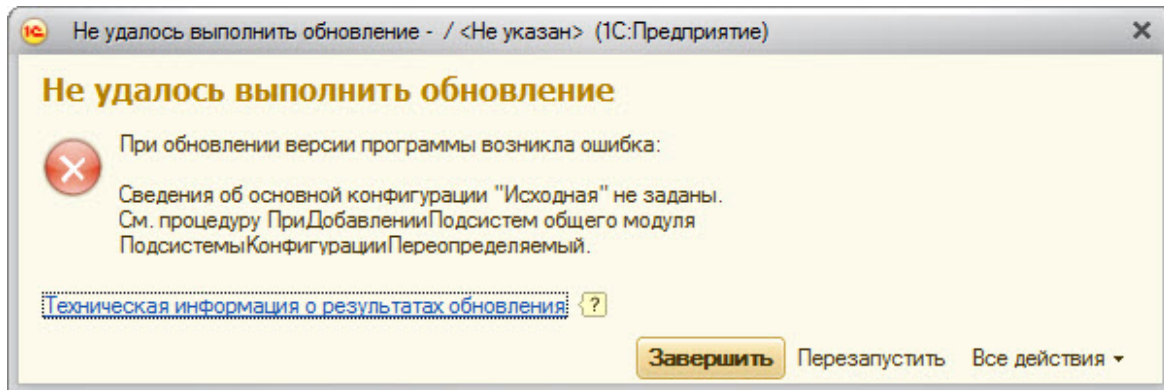
```
ПланыОбменаПодсистемы.Добавить (  
Метаданные.ПланыОбмена.СинхронизацияДанныхЧерезУниверсальныйФормат) ;
```

Остальные процедуры оставим без изменения.

Шаг 4. Включим общую команду *СинхронизацияДанных* в подсистему *Синхронизация данных* для того, чтобы она отображалась в интерфейсе.

Настройка корректного обновления системы

Система готова к запуску. Но если выполнить его сейчас, при попытке обновления будет выдана ошибка.



Чтобы она не возникала, настроим корректное обновление основной конфигурации нашей базы. В качестве примера используем Демонстрационную базу БСП. Откроем в ней модуль *ПодсистемыКонфигурацииПереопределяемый*. В нем одна процедура *ПриДобавленииПодсистем()*. Здесь в коллекцию *МодулиПодсистем* нужно добавить модуль, в котором будут процедуры, необходимые для обновления основной конфигурации. Пример этих процедур можно посмотреть в одном из модулей, созданных для этой цели в БСП. Откроем модуль *ДемоОбновлениеИнформационнойБазыБСП*. Скопируем из него целиком область *ПрограммныйИнтерфейс*, затем откроем Исходную ИБ, создадим общий модуль *ОбновлениеИсходнойИБ* и вставим в него скопированный код. Этот код нужно донастроить следующим образом:

Процедура *ПриДобавленииПодсистемы()* должна содержать строки:

```
Описание.Имя = Метаданные.Имя ;  
Описание.Версия = Метаданные.Версия ;  
  
// Требуется библиотека стандартных подсистем.  
Описание.ТребуемыеПодсистемы.Добавить ("СтандартныеПодсистемы") ;
```

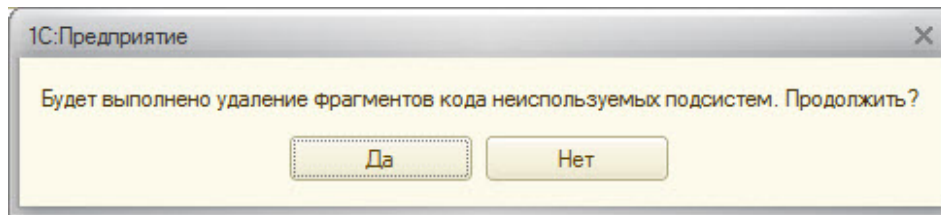
Содержимое остальных процедур нужно очистить, но удалять их совсем нельзя, поскольку иначе будут возникать ошибки.

Теперь откроем в Исходной ИБ модуль *ПодсистемыКонфигурацииПереопределяемый* и укажем созданный нами модуль.

```
МодулиПодсистем.Добавить ("ОбновлениеИсходнойИБ") ;
```

Запустим базу Исходная ИБ в пользовательском режиме. В окне со списком новых и измененных объектов нажмем кнопку *Принять*.

Откроем снова обработку *Первое внедрение БСП*. На этом этапе осталось выполнить пункт 3 *Удалить код неиспользуемых подсистем*. Для этого закроем Конфигуратор и нажмем на ссылку *Удалить*.



В открывшемся окне нажмем *Да* и дождемся сообщения о количестве произведенных замен.

Теперь можно закрыть окно обработки *Первое внедрение БСП*.

Для того чтобы убедиться, что внедрение БСП было проведено успешно, можно использовать отчет ПроверкаВнедренияБСП.erf, который также находится в подкаталоге ExtFiles каталога шаблонов БСП.

Откроем в пользовательском режиме этот отчет и нажмем на кнопку *Сформировать*.

Подсистема конфигурации			Подробное описание ошибки
Объект метаданных	Подсистема БСП	Краткое описание ошибки	
Без подсистемы			1
Подсистема.СтандартныеПодсистемы.Подсистема.ОбновлениеВерсииИБ	Обновление версии ИБ	Отсутствует макет ОписаниеИзменений Системы	В конфигурации не создан общий макет ОписаниеИзмененийСистемы.
Синхронизация данных			9
ПланОбмена.СинхронизацияДанныхЧерезУниверсальныйФормат	Обмен данными	Отсутствует обязательная вставка кода	В модуле МодульМенеджера объекта ПланОбмена.СинхронизацияДанныхЧерезУниверсальныйФормат отсутствует обязательная вставка кода "Процедура ПолучитьДополнительныеДанныеДляКорреспондента(ДополнительныеДанные) Экспорт"
ПланОбмена.СинхронизацияДанныхЧерезУниверсальныйФормат	Обмен данными	Отсутствует обязательная вставка кода	В модуле МодульМенеджера объекта ПланОбмена.СинхронизацияДанныхЧерезУниверсальныйФормат отсутствует обязательная вставка кода "Функция НастройкаОтборовНаУзле(ВерсияКорреспондента, ИмяФормы, ИдентификаторНастройки) Экспорт"

Всего было найдено 10 ошибок. Пройдем по строкам этого отчета и проанализируем их все.

Первая ошибка касается общего макета *ОписаниеИзмененийСистемы*. Его можно скопировать из демонстрационной базы БСП, и эта ошибка возникать не будет. Можно также изменить этот макет так, что информация в нем соответствовала действительности. Но на работу программы влияет только его наличие.

Все остальные ошибки касаются модуля менеджера плана обмена *Синхронизация данных через универсальный формат*. Здесь проверяется наличие обязательных процедур и функций, а также их правильного объявления. Поэтому несмотря на то, что функция *НастройкаОтборовНаУзле()* уже есть в нашем модуле менеджера, она, тем не менее, фигурирует в отчете.

Если сравнить объявление этой функции, предложенное в отчете с тем, которое находится в модуле менеджера, можно увидеть, в чем разница. Для переменной *ИдентификаторНастройки* указано значение по умолчанию - пустая строка. Если удалить его, эта строка перестанет появляться в отчете.

Таким образом, изменяя объявления существующих процедур и функций, и добавляя те, которых не хватает, можно добиться того, что отчет будет формироваться пустым.

На этом внедрение БСП успешно завершено.

Все вышеописанные действия необходимо повторить с модельной базой Конечная ИБ!

Из Исходной в Конечную ИБ можно скопировать:

- План обмена

- Синхронизация данных через универсальный формат
- Общие модули
 - Менеджер обмена через универсальный формат
 - Обновление Исходной ИБ (переименовать в Обновление Конечной ИБ)

Все остальные действия выполнить также, как выше описано для Исходной ИБ, кроме подписок на события. Ни они, ни их обработчики в Конечной ИБ не нужны.

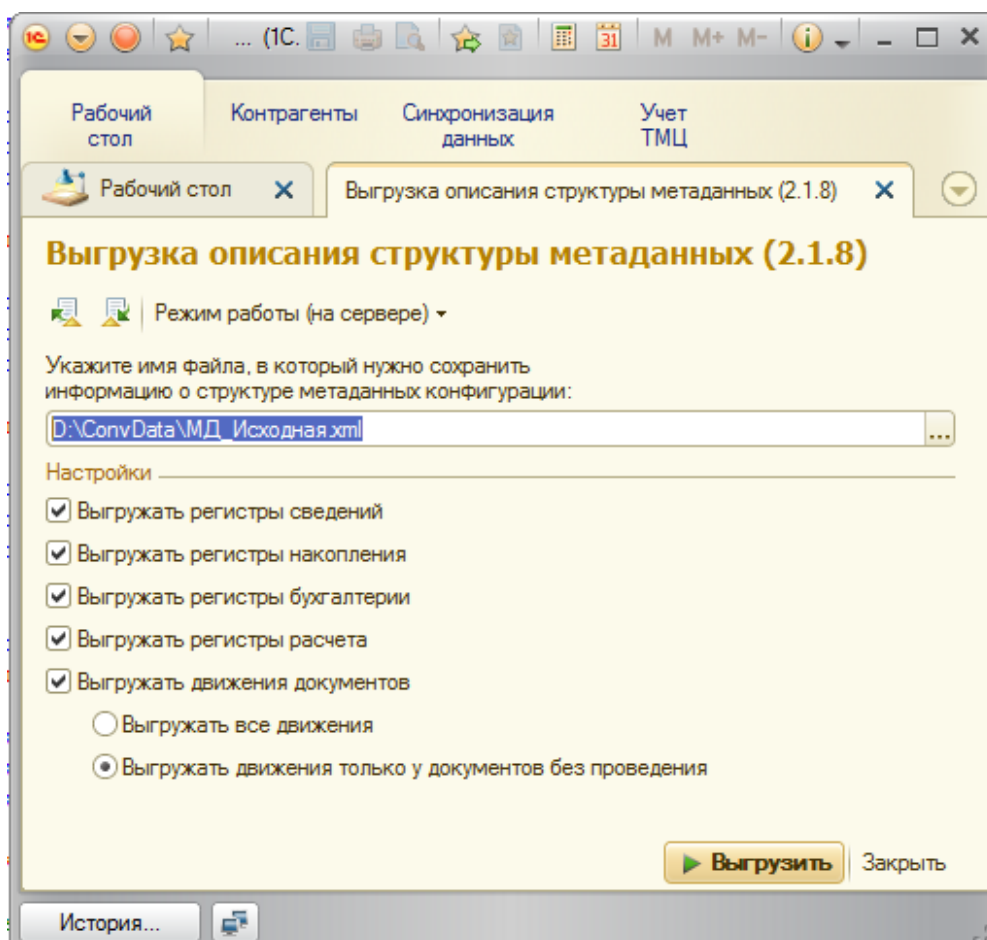
Подготовка к переносу данных

Теперь прежде, чем создавать новый обмен данными, необходимо выполнить настройку правил преобразования объектов. Для этого нужно

- Выгрузить информацию о структуре объектов метаданных конфигурации Исходная ИБ
- Загрузить в программу *Конвертация данных 3.0* эту информацию
- Выгрузить из Исходной ИБ xsd-схемы используемого формата
- Загрузить эти схемы в программу *Конвертация данных 3.0*

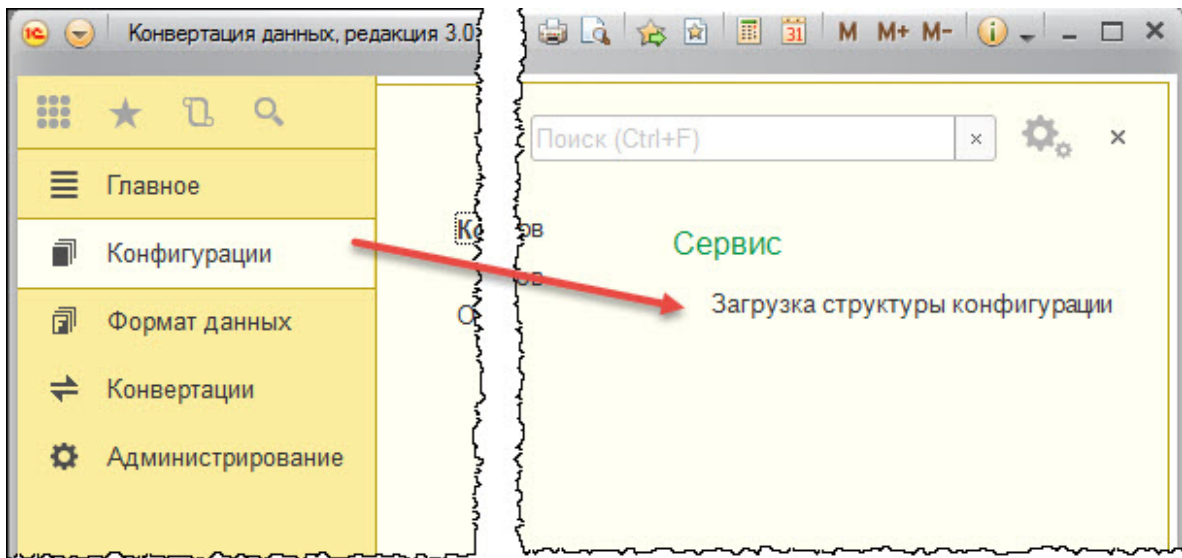
Выполним эти действия.

Откроем базу Исходная ИБ в пользовательском режиме. Используя пункт *Выпадающего меню – Файл – Открыть* вызовем диалог выбора файла. Необходимо найти в каталоге шаблонов каталог *Conversion*, а в нем каталог, соответствующий используемой версии Конвертации данных – *3_0_2_1*. Откроем его, найдем и выберем обработку MD83Exp.epf.



Необходимо указать в поле ввода путь к файлу, в который будет сохранена информация о структуре метаданных конфигурации Исходная ИБ.

Следует снять все флажки в окне обработки и нажать кнопку *Выгрузить*. Программа сообщает о том, что выгрузка прошла успешно. Теперь откроем программу *Конвертация данных 3.0*. Выберем пункт главного меню *Конфигурации* и в разделе сервис выберем команду *Загрузка структуры конфигурации*.

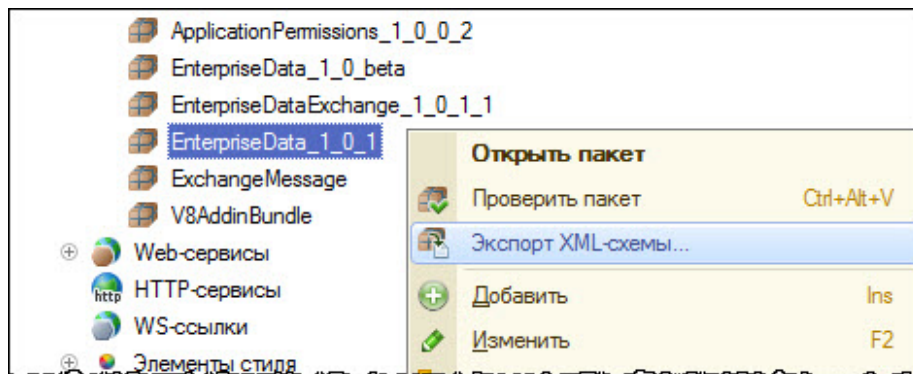


Укажем путь к файлу со структурой конфигурации и нажмем кнопку *Выполнить загрузку*.

После того, как система сообщит об успешной загрузке, выберем пункт Главного меню *Конфигурации* – команда *Конфигурации*. В появившемся списке находится один элемент – *Исходная*.

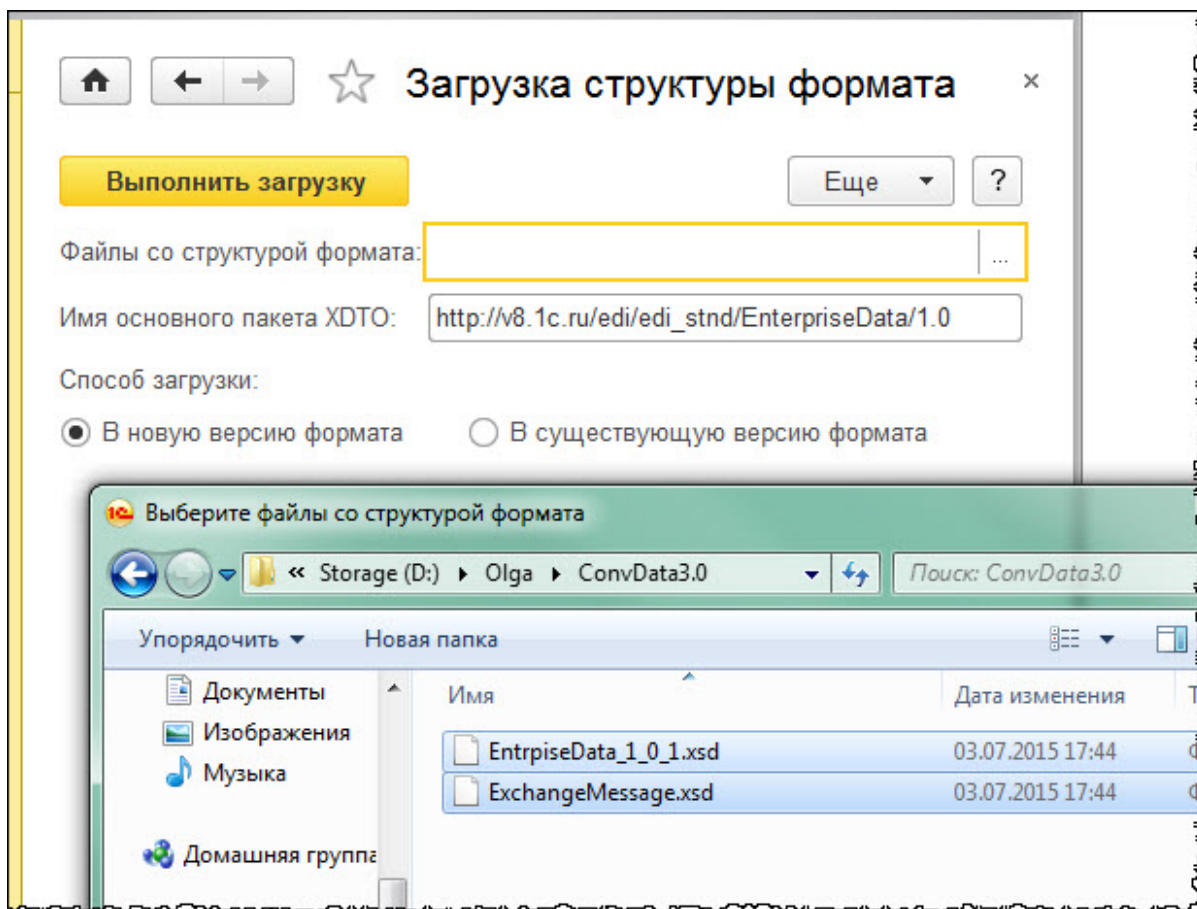
Точно так же выполняется выгрузка информации о структуре данных из базы Конечная ИБ.

Теперь выполним загрузку формата. Найдем в дереве конфигурации *Исходная ИБ* в ветке *Общие* – *XDTO-пакеты* пакет, который называется *EnterpriseData_1_0_1*. Кликнем по нему правой кнопкой мыши и из контекстного меню выберем пункт *Экспорт XML – схемы*.



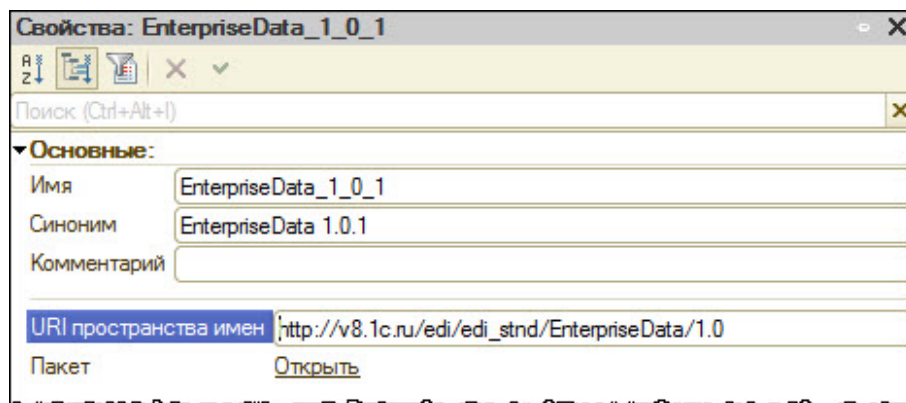
Укажем путь к файлу, в который будет сохранена эта схема. Так же сохраним схему *ExchangeMessage*.

В КД выберем пункт главного меню *Формат данных* и в разделе *Сервис* – *Загрузка структуры формата*



В открывшемся окне есть поле, в котором нужно указать одновременно оба файла формата. Для этого нужно нажать на кнопку выбора, в открывшемся окне с помощью множественного выбора отметить сразу оба файла xsd и нажать кнопку *Открыть*.

Имя основного пакета XDTO можно скопировать из свойств XDTO-пакета EnterpriseData_1_0_1 в конфигураторе Исходной ИБ, свойство *URI пространства имен*.



Нажать на кнопку *Выполнить загрузку*. Когда загрузка формата будет успешно завершена, система выдаст об этом сообщение.

Далее можно переходить непосредственно к настройке правил.