

Web-сервисы

Механизм Web-сервисов в системе «1С:Предприятие» является средством поддержки сервисно-ориентированной архитектуры (Service-Oriented Architecture, SOA).

Сервисно-ориентированная архитектура представляет собой прикладную архитектуру, в которой все функции определены как независимые сервисы с вызываемыми интерфейсами. Обращение к этим сервисам в определенной последовательности позволяет реализовать тот или иной бизнес-процесс.

Сервисно-ориентированная архитектура предлагает новый подход к созданию распределенных информационных систем, в которых программные ресурсы рассматриваются как сервисы, предоставляемые по сети. Такой подход позволяет обеспечить быструю консолидацию распределенных компонентов (сервисов) в единое решение для поддержки определенных бизнес-процессов.

Механизм Web-сервисов позволяет использовать систему «1С:Предприятие» как набор сервисов в сложных распределенных и гетерогенных системах, а также позволяет интегрировать ее с другими промышленными системами с использованием сервисно-ориентированной архитектуры.

Конфигурация системы «1С:Предприятие» может экспортировать свою функциональность через Web-сервисы. Определения Web-сервисов задаются в дереве конфигурации и становятся доступны произвольным информационным системам благодаря публикации их на веб-сервере.

Кроме этого, система «1С:Предприятие» может обращаться к Web-сервисам сторонних производителей как через статические ссылки, определенные в дереве конфигурации, так и с помощью динамических ссылок, создаваемых средствами встроенного языка.

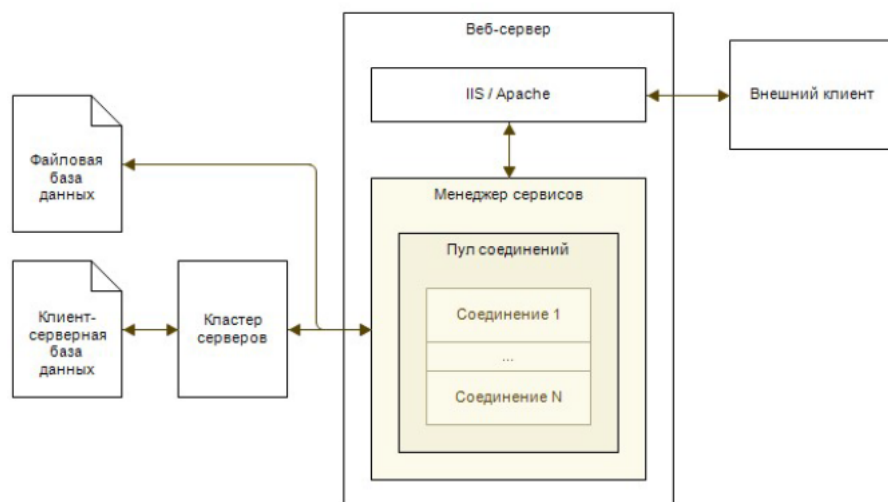


Рис. 1. Web-сервисы

В основе сервисной архитектуры системы «1С:Предприятие 8» находится менеджер сервисов. Менеджер сервисов выполняет следующие функции:

- управление пулом соединений с информационными базами;
- поддержка WSDL описания сервиса;
- реализация протокола SOAP, сериализация сообщений, вызов соответствующего сервиса.

Менеджер сервисов выполняется в процессе сервисного хоста, который выполняет функцию приема/передачи сообщений из/в менеджер сервисов. В качестве сервисного хоста может использоваться веб-сервер IIS или Apache.

Менеджер сервисов содержит в себе пул соединений, через которые идет взаимодействие с базами данных системы «1С:Предприятие».

Механизм Web-сервисов, реализованный в системе «1С:Предприятие», поддерживает следующие стандарты:

- SOAP 1.1,
- SOAP 1.2,
- WSDL 1.1,
- WS-I Basic Profile 1.1,
- HTTP 1.1,
- TLS 1.x (TLS 1.1 и 1.2 поддерживаются, если не требуется передача клиентского сертификата на сервер), включая криптографические алгоритмы, соответствующие ГОСТ Р 34.10-2001, Р 34.10-94, Р 34.11-94 и 28147-89;
- MTOM;
- Аутентификация: Basic, NTLM/Negotiate.

Для получения доступа к Web-сервису необходимо использовать адрес, который формируется следующим образом:

`http://host/base/ws/ИмяWebСервиса`

или

`http://host/base/ws/АдресWebСервиса.`

Более подробно рассмотрим составные части адреса:

`http://host/base` – обычный URL, по которому выполняется доступ, например, к информационной базе с помощью веб-клиента. При наличии разделителей, не поддерживается указание значений разделителей с помощью параметра `Z` командной строки запуска клиентского приложения.

`ws` – признак того, что выполняется обращение к Web-сервису (в отличие от `hs`, который определяет доступ к HTTP-сервису, см. здесь).

`ИмяWebСервиса` – имя Web-сервиса. Задается в свойстве объекта Web-сервис.

`АдресWebСервиса` – описывает альтернативное имя для доступа к Web-сервису. Задается в свойстве `Имя файла публикации` объекта Web-сервис. Может быть изменено при публикации Web-сервиса.

Обращения по имени `ИмяWebСервиса` и адресу `АдресWebСервиса` Web-сервиса являются равносильными.

Предоставление функциональности через Web-сервисы

Для того чтобы функциональность системы «1С:Предприятие» стала доступна внешним потребителям Web-сервисов, нужно выполнить следующие действия:

- создать в конфигурации необходимое количество Web-сервисов,
- опубликовать Web-сервисы с помощью специального инструмента конфигурирования.

Описание процедуры публикации Web-сервисов и описание использования Reverse Proxy для доступа к «1С:Предприятию» написано в главе 7 «Руководство администратора» на диске ИТС.

Создание Web-сервиса заключается:

- в добавлении в дерево метаданных объекта конфигурации Web-сервис,
- описании операций, которые может выполнять создаваемый Web-сервис,
- описании параметров операций Web-сервиса.

Объект конфигурации Web-сервис содержит модуль, в котором создаются процедуры на встроенном языке, выполняемые при вызове тех или иных операций Web-сервиса. Типы параметров операций Web-сервиса описываются с помощью типов XDTO и могут представлять собой либо значения XDTO, либо объекты XDTO.

Вызов Web-сервиса происходит следующим образом:

- из пула соединений выбирается подходящее соединение с информационной базой; при отсутствии необходимого соединения соединение создается;
- создается новый сеанс и для созданного сеанса вызывается событие УстановкаПараметровСеанса (в модуле сеанса);
- выполняется вызов затребованного метода Web-сервиса, при этом происходит вызов обработчика УстановкаПараметровСеанса() (в модуле сеанса) каждый раз, когда происходит обращение к не инициализированному параметру сеанса.

СОВЕТ. Не рекомендуется выполнять ресурсоемкие операции в обработчике события УстановкаПараметровСеанса.

Событие УстановкаПараметровСеанса модуля сеанса вызывается на сервере в привилегированном режиме. Модуль вызванного сервиса исполняется на сервере в обычном режиме.

Модуль сеанса служит для инициализации параметров сеанса и выполнения некоторого набора команд при вызове любого Web-сервиса системы «1С:Предприятие».

Пример реализации Web-сервиса

Например, требуется создать Web-сервис системы «1С:Предприятие», который должен по переданному номеру расходной накладной возвращать состав ее табличной части. Аналогичный пример с помощью HTTP-сервисов будет рассмотрено в другой статье.

Для описания возвращаемого значения создадим пакет XDTO ДанныеРасходнойНакладной с пространством имен

`http://www.MyCompany.ru/shipment`, содержащий три типа объектов XDTO:

- Номенклатура – для передачи данных элемента справочника Номенклатура. Этот тип объекта XDTO будет содержать следующие свойства:
- Наименование – тип string из пространства имен `http://www.w3.org/2001/XMLSchema`;
- ПолноеНаименование – тип string из пространства имен `http://www.w3.org/2001/XMLSchema`;
- ШтрихКод – тип string из пространства имен `http://www.w3.org/2001/XMLSchema`;
- ЗакупочнаяЦена – тип int из пространства имен `http://www.w3.org/2001/XMLSchema`.
- СтрокаРасходнойНакладной – для передачи данных одной строки расходной накладной. Этот тип объекта XDTO будет содержать следующие свойства:
- Номенклатура – тип Номенклатура из пространства имен `http://www.MyCompany.ru/shipment`; представляет собой ссылку на объект XDTO, который мы определили выше;
- Количество – тип int из пространства имен `http://www.w3.org/2001/XMLSchema`;
- Цена – тип int из пространства имен `http://www.w3.org/2001/XMLSchema`;
- Сумма – тип int из пространства имен `http://www.w3.org/2001/XMLSchema`.
- РасходнаяНакладная – для передачи данных всех строк расходной накладной. Этот тип объекта XDTO будет содержать единственное свойство:

- Состав – тип СтрокаРасходнойНакладной из пространства имен `http://www.MyCompany.ru/shipment`. Представляет собой ссылку на объект XDTO, который мы определили выше. Для того чтобы это свойство могло содержать неограниченное множество значений, необходимо установить его свойство Верхняя граница в значение -1.

После того как необходимые типы XDTO созданы, следует добавить в конфигурацию новый Web-сервис ДанныеРасходнойНакладной со следующими значениями свойств:

- URI Пространства имен – `http://www.MyCompany.ru/shipment`;
 - Пакеты XDTO – ДанныеРасходнойНакладной;
 - Имя файла публикации – `shipment.1cws`.
- У созданного Web-сервиса следует определить операцию Получить со следующими значениями свойств:
- Тип возвращаемого значения – РасходнаяНакладная из пространства имен `http://www.MyCompany.ru/shipment`;
 - Возможно пустое значение – установлен;
 - Имя процедуры – Получить.

У операции Получить следует определить параметр НомерДокумента со следующими значениями свойств:

- Тип значения – тип string из пространства имен `http://www.w3.org/2001/XMLSchema`;
- Направление передачи – Входной.

После этого следует открыть модуль созданного Web-сервиса и разместить в этом модуле функцию Получить(), которая будет выполняться при вызове данного Web-сервиса.

Функция Получить(НомерДокумента) Экспорт

// Получить объект расходной накладной по переданному номеру

ДокументСсылка = Документы.РасходнаяНакладная.НайтиПоНомеру(НомерДокумента, ТекущаяДата());

Если ДокументСсылка.Пустая() Тогда

Возврат Неопределено;

КонецЕсли;

Документ = ДокументСсылка.ПолучитьОбъект();

// Получить типы объектов XDTO

НоменклатураТип = ФабрикаXDTO.Тип("http://www.MyCompany.ru/shipment", "Номенклатура");

РасходнаяНакладнаяТип = ФабрикаXDTO.Тип("http://www.MyCompany.ru/shipment", "РасходнаяНакладная");

СтрокаРасходнойНакладнойТип = ФабрикаXDTO.Тип("http://www.MyCompany.ru/shipment", "СтрокаРасходнойНакладной");

// Создать объект XDTO расходной накладной

РасходнаяНакладная = ФабрикаXDTO.Создать(РасходнаяНакладнаяТип);

Для Каждого СтрокаДокумента Из Документ.Состав Цикл

// Создать объекты XDTO строки расходной накладной

// и номенклатуры

```

СтрокаРасходнойНакладной = ФабрикаXDTO.Создать(СтрокаРасходнойНакладнойТип);
Номенклатура = ФабрикаXDTO.Создать(НоменклатураТип);
// Заполнить свойства номенклатуры
Номенклатура.Наименование = СтрокаДокумента.Номенклатура.Наименование;
Номенклатура.ПолноеНаименование = СтрокаДокумента.Номенклатура.ПолноеНаименование;
Номенклатура.ШтрихКод = СтрокаДокумента.Номенклатура.ШтрихКод;
Номенклатура.ЗакупочнаяЦена = СтрокаДокумента.Номенклатура.ЗакупочнаяЦена;
// Заполнить свойства строки расходной накладной
СтрокаРасходнойНакладной.Номенклатура = Номенклатура;
СтрокаРасходнойНакладной.Количество = СтрокаДокумента.Количество;
СтрокаРасходнойНакладной.Цена = СтрокаДокумента.Цена;
СтрокаРасходнойНакладной.Сумма = СтрокаДокумента.Сумма;
// Добавить строку расходной накладной
РасходнаяНакладная.Состав.Добавить(СтрокаРасходнойНакладной);
КонецЦикла;
// Вернуть расходную накладную
Возврат РасходнаяНакладная;
КонецФункции

```

В заключение следует опубликовать созданный Web-сервис на веб-сервере, например <http://www.MyCompany.ru>, в каталоге shipment.

Работа с веб-сервисами сторонних поставщиков

Система «1С:Предприятие» может использовать веб-сервисы, предоставляемые другими поставщиками, несколькими способами:

- с помощью статических ссылок, создаваемых в дереве конфигурации;
- с помощью динамических ссылок, создаваемых средствами встроенного языка;
- комбинацией предыдущих способов.

Преимущество использования статических ссылок заключается в большей скорости работы, т. к. описание веб-сервиса поставщика получается один раз, при создании ссылки. В дальнейшем при обращении к данному веб-сервису используется существующее описание веб-сервиса.

При использовании динамических ссылок описание веб-сервиса поставщика будет получаться системой «1С:Предприятие» каждый раз при вызове веб-сервиса, что, естественно, будет замедлять работу с данным веб-сервисом. Однако преимуществом такого подхода является возможность получения актуального описания веб-сервиса поставщика. При использовании же статических ссылок для получения актуального описания веб-сервиса следует выполнить повторный импорт WSDL-описания средствами конфигуратора и сохранение измененной конфигурации.

При эксплуатации прикладных решений может возникать ситуация, когда один и тот же веб-сервис предоставляется по разным адресам (URL), однако имеет абсолютно одинаковое описание (WSDL). В этом случае возникает потребность загрузить описание веб-сервиса в конфигурацию (создать объект в дереве объектов конфигурации), но во время использования указать конкретный адрес, по которому расположен веб-сервис. Комбинированный способ позволяет работать таким образом. В качестве примера можно рассмотреть следующую ситуацию: есть тиражируемый веб-сервис, выполняющий некоторую функцию. Прикладное решение, написанное на «1С:Предприятии», пользуется услугами данного сервиса, при этом адрес сервиса может быть различным (сервис тиражируемый), а описание – фиксированное. Тогда в прикладное решение может загрузить описание веб-сервиса, а в настройках прикладного решения предусмотреть ввод адреса конкретного экземпляра сервиса, который (адрес) и будет использоваться при работе.

Также доступно и другой способ: используется динамическая ссылка, но адрес расположения веб-сервиса получается не из файла описания (WSDL), а непосредственно указывается при создании объекта.

При попытке загрузить описание Web-сервиса в конфигураторе (создание статической ссылки) или при использовании динамической ссылки (при помощи объекта WSOпределения), система выполняет проверку загружаемого описания Web-сервиса (WSDL). Если в описании Web-сервиса присутствует ошибка (с «точки зрения» системы «1С:Предприятия»), то описание не будет загружено и будет сгенерировано исключение. В тексте исключения будет находиться подробная диагностика причин отказа в загрузке. Ошибки WSDL располагаются в порядке их обнаружения. Каждая ошибка WSDL содержит запись с детальным описанием следующего вида:

```

<Тип элемента с ошибкой>.<Имя>
[<Тип элемента с ошибкой>.<Имя>[...]]
<Описание ошибки>
<Тип элемента с ошибкой>
<Имя>

```

В этом описании:

- Тип элемента с ошибкой – тип элемента WSDL. Чаще всего соответствует имени тега в xml-файле, описывающем Web-сервис.
- Описание ошибки – описание конкретной ошибки.
- Имя – имя объекта WSDL, в котором произошла ошибка.

Пример использования статической WS-ссылки

В качестве примера использования веб-сервисов стороннего поставщика рассмотрим обращение к Web-сервису, который ранее был создан в примере выше

Прежде всего, следует добавить в дерево конфигурации новый объект конфигурации WS-ссылка с именем ДанныеРасходнойНакладной, сылающийся на опубликованный сервис. Для этого следует выполнить импорт WSDL-описания опубликованного сервиса и в качестве URL указать <http://www.MyCompany.ru/shipment/ws/Shipment.1cws?wsdl>. Описание импорта WSDL-описания написано в другой статье. После этого, например, в модуле приходной накладной, можно создать процедуру, приведенную ниже. Она заполняет табличную часть документа данными расходной накладной поставщика, полученными с помощью веб-сервиса поставщика.

```

Процедура ПолучитьДанныеРасходнойНакладной(НомерНакладнойПоставщика)
// Создать WS-прокси на основании ссылки
Прокси = WSCсылки.ДанныеРасходнойНакладной.СоздатьWSПрокси("http://www.MyCompany.ru/shipment", "ДанныеРасходнойНакладной", "ДанныеРасходнойНакладной" = Неопределено Тогда
Возврат;
КонецЕсли;
// Заполнить приходную накладную полученными данными
Для Каждого СтрокаНакл Из ДанныеНакладной.Состав Цикл
НоваяСтрока = ДокументОбъект.Состав.Добавить();
НоваяСтрока.Количество = СтрокаНакл.Количество;
НоваяСтрока.Цена = СтрокаНакл.Цена;
НоваяСтрока.Сумма = СтрокаНакл.Сумма;
// Найти элемент номенклатуры по переданным данным

```

// (например, по штрихкоду)

НоваяСтрока.Номенклатура = Справочники.Номенклатура. НайтиПоРеквизиту("ШтрихКод", СтрокаНакл.Номенклатура.ШтрихКод);

КонецЦикла;

КонецПроцедуры

Если адрес реального расположения сервиса отличается от адреса, который использовался во время загрузки описания веб-сервиса в конфигурацию, то новый адрес необходимо явно указать при создании объекта WSPрокси:

// Создать WS-прокси на основании ссылки

Прокси = WSCсылки.ДанныеРасходнойНакладной. СоздатьWSPрокси("http://www.MyCompany.ru/shipment", "ДанныеРасходнойНакладной", "ДанныеРасходнойНакладной");

Пример использования динамической WS-ссылки

Использование динамической ссылки отличается от использования статической ссылки только способом создания WS-прокси и отсутствием необходимости создавать WS-ссылку в дереве конфигурации.

Если провести сравнение с примером, представленным в предыдущем разделе, то, в отличие от создания прокси на основе статической ссылки, при использовании динамической ссылки WS-прокси создается с помощью конструктора следующим образом:

// Создать WS-прокси на основании WS-определения

Определение = Новый WSOпределения("http://www.MyCompany.ru/shipment/ws/Shipment.1cws?wsdl");

Прокси = Новый WSPрокси(Определение, "http://www.MyCompany.ru/shipment", "ДанныеРасходнойНакладной", "ДанныеРасходнойНакладнойSoap");

В то время как создание WS-прокси на основе статической ссылки выглядит следующим образом:

// Создать WS-прокси на основании ссылки

Прокси = WSCсылки.ДанныеРасходнойНакладной. СоздатьWSPрокси("http://www.MyCompany.ru/shipment", "ДанныеРасходнойНакладной", "ДанныеРасходнойНакладной");

Если адрес реального расположения сервиса отличается от адреса, который указан в WSDL-файле, используемом при создании определения веб-сервисов, новый адрес необходимо явно указать при создании объекта WSPрокси на основании определения веб-сервисов:

// Создать WS-прокси на основании WS-определения

Определение = Новый SOпределения("http://www.MyCompany.ru/shipment/ws/Shipment.1cws?wsdl");

Прокси = Новый WSPрокси(Определение, "http://www.MyCompany.ru/shipment", "ДанныеРасходнойНакладной", "ДанныеРасходнойНакладнойSc

Редактирование свойств Web-сервиса

На закладке Основные вводится имя, синоним и комментарий объекта.

На закладке Операции создаются подчиненные объекты Операции, которые, в свою очередь, могут иметь подчиненные объекты Параметры, необходимые для работы с объектами данного типа. Описание подчиненных объектов выполняется в палитре свойств.

Свойства «Операции»

Помимо общих свойств объектов конфигурации операция Web-сервиса содержит следующие свойства:

- Тип возвращаемого значения – тип значения, которое возвращает операция Web-сервиса. Может являться типом значения XDTO или типом объекта XDTO.
 - Возможно пустое значение – показывает, может ли возвращаемое значение принимать неопределенное значение.
 - В транзакции – показывает, будет ли выполняться код модуля Web-сервиса в транзакции или нет. Если свойство установлено, то при вызове Web-сервиса автоматически будет начата транзакция, а при завершении работы Web-сервиса транзакция будет либо зафиксирована, либо произойдет откат транзакции (в зависимости от результатов выполнения). Если свойство не установлено, при начале исполнения модуля Web-сервиса транзакция не будет начата.
 - Имя метода – имя экспортируемой процедуры модуля Web-сервиса, которая будет выполнена при вызове данного свойства.
- На закладке Подсистемы указывается, к каким подсистемам относятся объекты данного типа.
- На закладке Прочее определяются следующие свойства:
- URI пространства имен – содержит URI пространства имен Web-сервиса. Каждый Web-сервис может быть однозначно идентифицирован по своему имени и URI пространству имен, которому он принадлежит. Пространство имен сервиса не должна совпадать с известными пространствами имен, которые уже используются или зарезервированы другими организациями. Рекомендуется в пространство имен сервиса включать фрагмент, уникальный для организации, которая ведет разработку Web-сервиса. Например, для организации с названием Промресурс имеет смысл начинать все пространства имен с префикса, например, http://promresurs.com. Тогда URI пространства имен Web-сервиса будет иметь вид http://promresurs.com/public/services/OurService.
 - Пакеты XDTO – перечень пакетов XDTO, типы которых могут использоваться в качестве типов возвращаемого значения операций и типов параметров операций Web-сервиса.
 - Имя файла публикации – имя файла описания Web-сервиса, который расположен на веб-сервере.
- По кнопке Модуль открывается редактор модуля Web-сервиса.

Свойства «Параметр»

Помимо общих свойств объектов конфигурации параметр операции Web-сервиса содержит следующие свойства:

- Тип значения – тип значения параметра операции Web-сервиса. Может являться типом значения XDTO или типом объекта XDTO.
- Возможно пустое значение – показывает, может ли значение параметра операции принимать неопределенное значение.
- Направление передачи – определяет направление передачи данных с помощью данного параметра. Возможные значения:
- Входной – означает, что параметр используется для передачи данных Web-сервису;
- Выходной – означает, что параметр используется для получения данных от Web-сервиса;
- Входной-Выходной – означает, что параметр может использоваться как для передачи данных, так и для их получения от Web-сервиса.