



ОРГАНИЗАЦИЯ ИМПОРТА ДАННЫХ НА ОСНОВЕ WEB-СЛУЖБ

И.С. Берендаков (ФГУП "ИТМ и ВТ им. С.А. Лебедева РАН)

Рассматривается подход к организации импорта данных, основанный на механизме Web-служб. В качестве источников информации выступают корпоративные системы внешних организаций, в качестве приемника – единая система отчетности. Приведено описание процесса проектирования спецификаций для методов Web-служб. Описан пример получения необходимой информации из БД внешнего источника и передачи ее по протоколу SOAP клиентскому приложению системы отчетности.

Механизм Web-служб является хорошей альтернативой при выборе подхода к реализации импорта данных. Иногда информационные системы внешних коммерческих организаций уже содержат Web-службы, которые позволяют программно получить информацию прямо из хранилищ данных этих систем. Нужно лишь вызвать удаленные методы, которые предоставляют имеющиеся Web-службы, и получить запрашиваемые данные. Рассмотрим ситуацию, когда внешний информационный источник не содержит Web-служб, специально разработанных для проведения импорта данных.

Предположим, имеется несколько связанных определенным образом организаций, предоставляющих свои данные в некий центр сбора отчетной информации. Каждая организация имеет некоторую корпоративную информационную систему, которая позволяет управлять бизнес-процессами компании, а также вносить, редактировать и просматривать данные, специфичные для рода деятельности этой компании. Для хранения информации все организации используют БД под управлением различных СУБД (например, Oracle, MySQL или MS SQL Server). Таким образом, информационная система каждой организации – это внешний источник данных для разрабатываемого центра отчетности. А значит встает задача создания единой системы отчетности, данные в которую должны импортироваться из нескольких разнородных источников информации.

Механизм Web-служб предоставляет возможность организации удобного интерфейса взаимодействия между БД внешних источников и системой отчетности. Для этого необходимо создать соответствующие Web-службы на серверах коммерческих организаций, предоставляющих свои данные, и сообщить разработчикам системы отчетности URL-адреса этих Web-служб. Если внешние информационные источники имеют одинаковую структуру хранилищ данных, то задача создания Web-служб упрощается, иначе необходимо спроектировать и реализовать индивидуальную Web-службу.

При этом важно определить, какие именно данные необходимы для построения отчетности. Проведение анализа поможет спроектировать Web-службу таким образом, что она будет содержать набор только тех методов, которые возвращают действительно требуемые данные. Если информационными источниками явля-

ются корпоративные системы сторонних производителей, и эти системы имеют готовые Web-службы, то в большинстве случаев требуется серьезная доработка этих служб, так как редко разработанная третьим лицом Web-служба сможет предоставить все необходимые данные для конкретной системы отчетности. Обычно требуется добавление к этой Web-службе нескольких новых методов. Таким образом, этап проектирования Web-служб является важным звеном в цепочке работ по организации импорта данных для системы отчетности.

Определение набора методов, необходимых для передачи информации в систему отчетности, подразумевает и разработку их спецификаций. Каждый метод Web-службы может иметь несколько входных параметров и возвращаемое значение. В данном случае входные параметры – это критерии запроса к таблицам БД внешнего источника, а возвращаемое значение – это объект или массив объектов, содержащих данные из запрашиваемых таблиц. Каждый возвращаемый объект должен хранить поля, которые присутствуют в строках, выбираемых запросом. Ответственность за формирование и выполнение SQL-запроса ложится на метод Web-службы.

Рассмотрим пример. Пусть в БД внешнего источника существует небольшая справочная таблица, и для построения отчетов необходимо получить ее содержимое. Следовательно, Web-служба должна содержать метод, возвращающий все строки данной таблицы. Этот метод не нуждается во входных параметрах, так как возвращает все содержимое справочной таблицы, что говорит об отсутствии критериев запроса. Но следует включить параметр, идентифицирующий пользователя, выполняющего данную операцию для разграничения прав доступа к таблицам БД внешнего информационного источника.

Разграничение прав доступа через дополнительный параметр может быть запрограммировано следующим образом. Клиентское приложение, интегрированное в систему отчетности, вызывает удаленный метод Web-службы, выполняющий инициализацию пользователя. При удачной инициализации этот метод возвращает некий идентификатор, который передается в описанный выше метод получения содержимого справочной таблицы, который также вызывается в клиентском приложении, но уже после метода инициализации. Web-служба внешнего источника анализирует полученный иден-

тификатор. Если прав недостаточно, то может быть возвращено сообщение, информирующее пользователя о невозможности получения данных. В противном случае, в рассматриваемом методе Web-службы формируется простой SQL-запрос к справочной таблице.

Взаимодействие с БД организуется посредством соответствующих технологий, предоставляемых библиотеками классов используемого высокоуровневого языка программирования. Например, для Web-службы, основанной на платформе .NET Framework, этими технологиями могут быть ADO.NET или Hibernate (www.hibernate.org). Полученные в результате запроса данные необходимо поместить в контейнер, который также является возвращаемым значением для описываемого метода. Этим контейнером может быть массив объектов, в котором каждый объект содержит данные только одной строки. Если заведомо известно, что SQL-запрос вернет лишь одну строку или значение, то вместо массива лучше использовать отдельный программный объект. Естественно, тип (класс) этих объектов должен соответствовать структуре справочной таблицы. Класс объектов должен быть определен внутри Web-службы или одной из используемых ею библиотек.

Взаимодействие клиентского приложения системы отчетности и удаленной Web-службы происходит по протоколу SOAP (Simple Object Access Protocol, простой протокол доступа к объектам), основанному на языке XML и использующему стандартные Internet-протоколы передачи данных.

Итак, набор методов описываемой Web-службы — это интерфейс взаимодействия системы отчетности с БД внешнего источника. Правильно спроектированный интерфейс упростит задачу создания продуманной и полноценной Web-службы, а значит и задачу организации самого импорта данных в целом. Определение методов интерфейса невозможно без определения их спецификаций. Спецификация метода включает его наименование, типы данных и названия его входных параметров, а также тип возвращаемого значения. Важно, чтобы методы и их параметры имели информативные наименования. Это облегчит использование методов Web-службы для разработчиков системы отчетности. По названию метода должно быть понятно, какую именно информацию он возвращает. Желательно создание документации, описывающей доступный функционал и возможности Web-службы. Входные параметры методов следует определять с использованием простых типов данных. Тип возвращаемого значения напрямую зависит от того, что именно возвращает метод.

Рассмотрим более детально метод Web-службы, возвращающий строки справочной таблицы. Предположим, что эта таблица хранит должности сотрудников внешней организации и называется duties. Ее структура включает два обязательных для заполнения поля: идентификатор (`duty_id`) и наименование (`duty_name`) должности. Метод Web-службы, возвращающий содержимое этой таблицы,

должен иметь информативное наименование, поэтому логичнее всего назвать его, например, `GetDuties`. Данному методу следует передавать один входной параметр, который идентифицирует пользователя, выполняющего операцию `userKey`. Метод `GetDuties` должен вернуть набор строк, состоящих из двух полей, поэтому необходимо создать пользовательский класс, объекты которого соответствуют строкам таблицы `duties`. Массив этих объектов и будет возвращаться методом.

Дальнейшее проектирование интерфейса Web-службы необходимо проводить по аналогичному сценарию. В результате будут получены спецификации всех методов, требуемых для организации импорта данных из внешнего источника. А это значит, что задача проектирования Web-службы будет завершена.

Для реализации метода Web-службы может быть применен любой высокоуровневый язык программирования, поддерживающий механизм Web-служб. Помимо C#.NET к таким языкам относятся также VB.NET, Java и Python. Все они ориентированы на создание современных приложений глобальной сети Internet. Подобная свобода выбора средств реализации Web-служб удобна для разработчиков корпоративной системы отчетности. Ведь склонившись к использованию какого-то конкретного языка программирования для написания Web-службы, ее создатели не накладывают никаких ограничений на природу технологий и средств, применяемых для разработки множества клиентских программ, вызывающих удаленные методы данной службы. Это возможно только благодаря применению платформу-независимого протокола SOAP, являющегося основным механизмом Web-служб. Таким образом, процесс создания системы отчетности никак не зависит от фактора языка программирования, используемого для написания Web-службы, данные которой запрашиваются.

Итак, спроектировав и реализовав Web-службы всех внешних информационных источников, можно сказать, что подготовительная работа по созданию API (Application Programming Interface) для доступа к данным завершена. Теперь не составит труда провести импорт всей необходимой информации. Но для этого должно быть подготовлено хранилище данных системы отчетности, т. е. создана сама БД и соответствующие таблицы в ней. На основе содержимого этих таблиц и будут строиться статистические или какие-либо другие отчеты. Следовательно, процесс загрузки данных из внешних источников заключается в программном опросе Web-служб (а точнее в вызове их удаленных методов), получении запрашиваемой информации и записи ее в таблицы БД системы отчетности. Оформив это решение в виде единого приложения, которое необходимо запускать каждый раз при выполнении загрузки, и интегрировав приложение с системой отчетности, можно получить удобный механизм импорта данных, основанный на технологии Web-служб.

Берендаков Илья Сергеевич — ведущий разработчик ФГУП "ИТМ и ВТ им. С.А. Лебедева РАН"

Контактный телефон (910)486-72-25. E-mail: ber83@mail.ru