

ФАКУЛЬТЕТ ТЕХНИЧЕСКОЙ КИБЕРНЕТИКИ

СЕКЦИЯ «ИНФОРМАЦИОННЫЕ И УПРАВЛЯЮЩИЕ СИСТЕМЫ»

УДК 681.3.06

М.М.Вершинин (асп., каф. ИУС)

РАЗРАБОТКА ОБРАЗЦОВ ДЛЯ АВТОМАТИЧЕСКОЙ ГЕНЕРАЦИИ ПРОСТЫХ КЛИЕНТОВ ДЛЯ КОМПОНЕНТОВ EJB И WEB СЕРВИСОВ

Компонентные технологии платформы Java 2, Enterprise Edition (J2EE), реализующие многоуровневую архитектуру серверной стороны, обладают универсальностью и рассчитаны на широкий круг пользователей, как опытных, так и начинающих. Технология Enterprise JavaBeans (EJB), являющаяся основной среди прочих технологий, входящих в состав J2EE, эффективно используется для создания распределенных Интернет-ориентированных приложений и реализации сложных программных систем. Производители контейнеров платформы J2EE берут на себя проблемы, связанные с установкой и поддержанием связи, передачей информации между удаленными компьютерами, поддержкой транзакций, обеспечением режима секретности и т.д. Пользователь, как из кирпичиков, строит свое приложение, используя стандартизованные компоненты EJB [1].

Существуют специальные программные средства разработки, позволяющие практически полностью свести процесс разработки J2EE-приложения к визуальному проектированию. К числу таких инструментов относится Borland@R Together@R ControlCenter@R. Таким образом, вести разработку J2EE-приложения может даже не очень квалифицированный программист.

Если компонент EJB уже создан и размещен на сервере приложений (самим программистом или предоставлен ему третьей стороной), возникает резонный вопрос, как вызвать этот компонент и как к нему обратиться. Эта проблема усугубляется тем, что у каждого сервера есть свои особенности. Клиенты для компонентов EJB - достаточно простые классы, однако написание кода клиента требует некоторого времени и может вызвать затруднения у не очень опытных пользователей.

Таким образом, автоматизация процесса создания EJB-клиентов является исключительно актуальной задачей.

Такие же проблемы возникают при использовании готовых и зарегистрированных Web сервисов [2, 3]. Поэтому в той же мере важна проблема автоматической генерации клиента для Web сервиса. В случае Web сервисов ситуация еще более усложняется в связи с тем, что это достаточно новая технология, использующая, кроме всего прочего, WSDL (Web Services Definition Language) - язык, который конечному пользователю знать совершенно не обязательно.

Часто удобнее использовать так называемый «прокси» (проху)-клиент. «Прокси»-клиент – это класс, содержащий все те же методы, что и компонент, из которых непосредственно и идет обращение к самому компоненту.

Постановка задачи. Автором решалась следующая задача: автоматическая генерация клиентов для компонентов EJB и Web сервисов.

Основной сценарий действий пользователя при автоматической генерации клиента состоит из следующих шагов:

- Пользователь щелкает мышью по кнопке «Class by Pattern» («Класс по образцу») линейки инструментов диаграммы классов и затем второй раз где-либо на свободном пространстве диаграммы. Появляется диалоговое окно «Choose Pattern» («Выбор образца»).
- Пользователь выбирает нужный образец из дерева поиска в левой части диалогового окна. В правой части диалогового окна при этом открываются поля для ввода параметров, отражающих специфику выбранного образца.
- Пользователь вводит или выбирает из списка значения параметров выбранного клиента и подтверждает свой выбор. После этого соответствующий код класса-клиента должен создаваться автоматически.

Реализация. К числу параметров образца EJB-клиента относятся, например, имя класса-клиента; интерфейсы, с которыми будет работать создаваемый EJB-клиент; JNDI- имя для поиска (lookup) в пространстве JNDI-имен; URL сервера приложений по умолчанию. Для основных образцов EJB-клиентов вводится дополнительный параметр, в котором можно указать имя диаграммы, на которой пользователь хочет поместить ярлык класса-клиента. Для клиента к Web сервису входными данными являются либо WSDL-файл, либо сами классы и интерфейсы.

Полученные результаты. Автором разработаны образцы (patterns) для создания EJB-клиентов, учитывающие специфику следующих серверов приложений:

- IBM WebSphere 3.x, 4.0;
- BEA WebLogic 5.1, 6.x, 7.0, 8.0;
- J2EE Reference Implementation, 1.3.1;
- Sun One 6.5;
- SAP J2EE 2.0.

Для Web сервисов созданы два образца для возможных типов клиентов:

- Простой Java-клиент;
- EJB Web сервис клиент на основе компонента session stateless bean.

Выводы. Создание клиентов к компонентам EJB и Web сервисам на основе предлагаемых образцов существенно экономит время программиста и делает процесс разработки приложения более технологичным.

ЛИТЕРАТУРА:

1. Enterprise JavaBeans™ 2.0 Specification. - <http://java.sun.com/products/ejb/2.0.html/>.
2. Apache Software Foundation. – <http://www.apache.org/>.
3. Web services. - <http://java.sun.com/webservices/>.