

ОСОБЕННОСТИ РАЗРАБОТКИ КЛИЕНТСКОГО МОДУЛЯ ДЛЯ АИС ПРОВЕРКИ И КОНТРОЛЯ ЗНАНИЙ С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИИ DATASNAP

С.В. Алешин, ст. преп. ГВУЗ «ПГТУ»

Клиентское программное обеспечение в распределенном многозвенном приложении имеет особенности архитектуры, продиктованные его ролью – большая часть бизнес-логики и функций обработки данных сосредоточены в сервере приложений. Такая схема призвана обеспечить более высокую эффективность обработки запросов многочисленных удаленных клиентов, а также упрощает обслуживание клиентского ПО. Клиенты, выполняющие лишь необходимый минимум операций, называются «тонкими». Клиенты многозвенных приложений должны выполнять следующие функции:

- соединение с сервером приложений, прием и передача данных;
- отображение средствами пользовательского интерфейса;
- простейшие операции редактирования;
- сохранение локальных копий данных.

Клиентское приложение в трехзвенной модели должно обладать лишь минимально необходимым набором функций, делегируя большинство операций по обработке данных серверу приложений.

В первую очередь удаленное клиентское приложение должно обеспечить соединение с сервером приложений. Для этого используется один из трех компонентов соединений DataSnap, использующих в качестве транспорта технологии DCOM, сокет Windows или HTTP. Компоненты соединения DataSnap предоставляют интерфейс IAppServer, используемый компонентами-провайдерами на стороне сервера и компонентами TClientDataSet на стороне клиента для передачи пакетов данных.

По своей структуре клиентское приложение подобно обычному приложению баз данных. Клиент многозвенного распределенного приложения должен содержать компоненты, инкапсулирующие набор данных, которые связаны с визуальными компонентами отображения данных.

Компонент-провайдер представляет собой мост между набором данных сервера приложений и клиентским набором данных. Он обеспечивает формирование и передачу пакетов данных клиентскому приложению и прием от него сделанных изменений. Очевидно, что набор данных сервера должен быть скопирован клиентским приложением в некий локальный буфер. При этом должен использоваться эффективный механизм загрузки данных сравнительно небольшими порциями,

что позволяет значительно разгрузить транспортный канал между клиентом и сервером приложений.

В клиентской части многозвенного распределенного приложения используется компонент TClientDataSet, который инкапсулирует набор данных, переданный при помощи компонента-провайдера из удаленного набора данных. Он обеспечивает выполнение следующих основных функций:

- получение данных от удаленного сервера и передача ему сделанных изменений с использованием удаленного компонента-провайдера;
- представление набора данных при помощи локального буфера и поддержка основных операций, унаследованных от класса TDataSet;
- объединение записей набора данных при помощи агрегатных функций для получения суммарных данных;
- локальное сохранение набора данных в файле и последующее восстановление набора данных из файла;
- представление набора данных в формате XML.

Компонент TClientDataSet обладает таким же набором функций, что и обычный компонент, инкапсулирующий набор данных, поскольку является предком класса TDataSet. Основное же отличие заключается в том, что источник данных для него доступен только через удаленный компонент-провайдер. Это означает, что сохранение изменений и обновление набора данных осуществляется локально, без обращения к источнику данных.

При разработке презентационной логики клиентского приложения (интерфейса пользователя) для АИС проверки и контроля знаний используются компоненты отображения данных. Большинство компонентов предназначены для работы с отдельным полем, т. е. при перемещении по записям набора данных такие компоненты показывают текущие значения только одного поля. Эти компоненты соединяются с набором данных через специальный компонент TDataSource. Отдельную группу составляют компоненты синхронного просмотра данных. Они обеспечивают показ значений поля из одной таблицы в соответствии со значениями поля из другой таблицы.

Таким образом, многозвенные распределенные приложения обеспечивают эффективное взаимодействие большого числа удаленных «тонких» клиентов с сервером БД при помощи ПО промежуточного слоя. Наиболее распространенной моделью является трехзвенная модель, где ПО промежуточного слоя состоит только из сервера приложений. В многозвенных распределенных приложениях в основном используются «тонкие» клиенты, делегирующие большинство функ-

ций ПО промежуточного слоя. Основой клиентского приложения является компонент `TClientDataSet`, который инкапсулирует набор данных и обеспечивает его использование при помощи локального буфера. Соединение с удаленным сервером приложений осуществляется при помощи компонентов `DataSnap`.