

Создание информационной системы, ориентированной на дистанционное обучение, с помощью технологии J2EE

Еременко А.С., Персиков А.В.

Харьковский национальный университет радиоэлектроники,
Харьков, Украина,

E-mail: alexere@ukr.net, w_seal@mail.ru

Abstract. The system of remote training is compound distributed system. Its creating is system-depending task with next common aspects: multi-speed supporting, great number of network connections, need for fixed QoS, multi-platform supporting. Actual problem is penetration of market of mobile abonent. Thus we have multi-sided problem. For its solving is need universal software, which will decide all problems.

Введение

Сегодня все больше и больше разработчиков хотят создавать распределенные транзакционные корпоративные приложения и использовать преимущества в скорости, защищенности и надежности, обеспечиваемые серверными технологиями [1]. Если вы уже работаете в области создания систем дистанционного обучения (СДО), то вам известно, что в современном, быстро меняющемся и выдвигающем все новые требования, мире электронной коммерции и информационных технологий приложения должны проектироваться, создаваться и внедряться за меньшие деньги, с большей скоростью и меньшими затратами ресурсов, чем это было ранее.

СДО есть многофункциональная система, разработка которой является сложной задачей [2]. Разработчики такой системы могут встретиться со следующими проблемами:

- возможность создания системы как таковая;
- скорость разработки;
- возможность реинжиниринга;
- управление частями системы, работающими в режиме реального времени;
- необходимость поддержки множества технологий.

Все это обуславливает необходимость в выборе универсального средства, которое позволит реализовать унифицированным способом, как бизнес-логику, так и систему управления СДО. В данной работе автор делает попытку привлечь внимание разработчиков Web-систем к архитектуре Java 2 Enterprise Edition, призванной решить вышеназванные проблемы [3].

Реализация мультимедийных порталов с помощью J2EE

Система дистанционного обучения (ДО) является примером многозвенной распределенной системы (рис. 1), реализацией которой занимаются множество групп разработчиков.

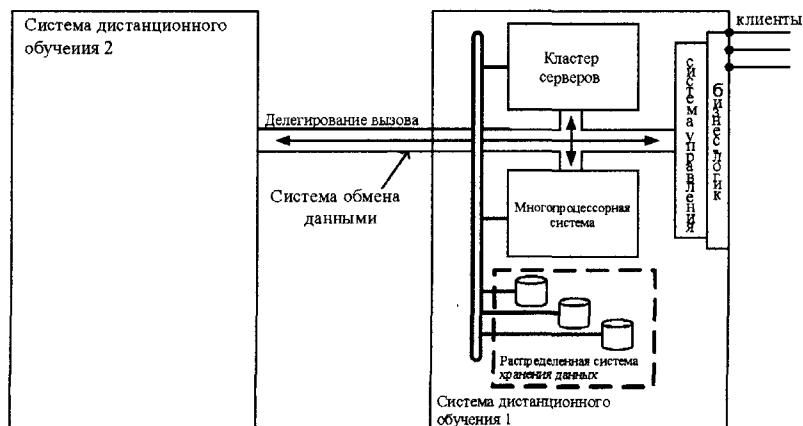


Рис. 1. Схема реализации системы дистанционного обучения на платформе операторского класса

Из рисунка видно, что вся СДО делится на две части: бизнес-логика и система управления (обработки) контентом. Задачей, которая ставится перед оператором услуг дистанционного обучения, является создание мультимедийного портала с возможностью предоставления услуг различного уровня и стоимости. Для обеспечения максимального маркетингового запаса мультимедиа-порталы операторского класса должны быть готовы интегрировать в себя любые услуги, которые могут возникнуть в будущем. При этом важно, чтобы абонентский трафик в процессе предоставления той или иной услуги не проходил через одну точку сети. Поэтому основными действиями портала является выполнение функций менеджера доступа к различным ресурсам сети и целенаправленного распределения трафика между узлами сети. Все ресурсы при этом контролируются соответствующими серверами приложений.

Решения такого класса должны быть готовы обслужить клиента из любого сектора рынка с любой спецификой информационной активности. Поэтому, необходима более глубокая адаптивность к нуждам абонентов. Она может быть достигнута за счет использования мощной библиотеки формализованных WEB-объектов, с которыми сталкиваются абоненты, и развитой системой их параметризации для структурирования контента портала.

Как правило, организация, которая занимается предоставлением услуг дистанционного обучения, не создает собственную инфраструктуру для хранения и обменом данных, а предпочитает арендовать ресурсы оператора телекоммуникационных услуг и/или услуг хранения данных. Это вызвано

высокой стоимостью инсталляции и обслуживания инфраструктуры: не каждая организация способна закупить множество лицензий на ПО и организовывать собственные системы хранения данных и управления; кроме того, высококвалифицированные специалисты с опытом работы в этой области являются редкостью. Поэтому выходом из ситуации является аренда хранилищ данных и серверов для обработки запросов.

Наиболее распространенными подходами при реализации распределенных систем являются компонентно-ориентированный подход (J2EE) и обмен данными на основе форматированных документов (XML + SOAP).

Основной особенностью (и преимуществом) J2EE является компонентно-объектный подход к организации доступа к функциональным возможностям тех или иных частей системы. Основой информационного обмена в J2EE является CORBA – наиболее популярная архитектура среди middleware-решений. Существует множество стандартных (платных и бесплатных) служб, функционирующих на основе этой архитектуры. Решения CORBA для систем реального времени используются при создании систем управления телекоммуникационными сетями, что расширяет возможности разработчиков (при реализации стыков с TMN-системами) для доступа к низкоуровневым функциям управления качеством обслуживания потоков данных и позволяет оптимизировать процесс транспортирования (одновременно с промежуточной обработкой) контента. Возможность регулирования уровня QoS непосредственным воздействием на параметры ORB также является ощутимым преимуществом данного программного обеспечения промежуточного уровня при организации системы управления. Основным недостатком CORBA является необходимость в покупке множества лицензий на использование ORB, что для некоторых организаций является проблемой из-за высокой стоимости последних.

Рассматривая второй подход, можно выделить две отличительные особенности: универсальность при низкой стоимости лицензий и сложность наращивания возможностей системы. Создание высокопроизводительных SOAP/Web служб требует написания значительного количества кода, причем уровень абстракции кода получается низким; сети под управлением SOAP/Web служб, реализованных различными компаниями, обладают недостаточными возможностями взаимодействия, т.к. на сегодняшний день нет общепринятых стандартов и протоколов взаимодействия операторов служб. Одним из наиболее серьезных недостатков служб, построенных на основе обмена XML-документами, является низкая скорость обработки потока данных. В противовес простоте использования XML, существует необходимость разбора документов, что приводит к значительным затратам оперативной памяти и процессорного времени. Кроме того, при передаче базы данных по сети в формате XML часто возникают проблемы с перегруженностью каналов. Эти особенности обуславливают использование решений на основе XML в основном для мобильных сетей, где возможности абонентского оборудования еще не позволяют внедрять широкополосные мультимедийные службы и где скорость доступа к информации является не самым критичным параметром (тем не

менее, существуют широкополосные решения и для мобильных сетей, но доступ к ним никак нельзя назвать массовым). Среди решений для операторов мобильных сетей можно выделить решения финской компании Celtius (<http://www.celtius.com>).

Рассмотрим реальное решение для оператора СДО на основе технологии J2EE (рис. 2).

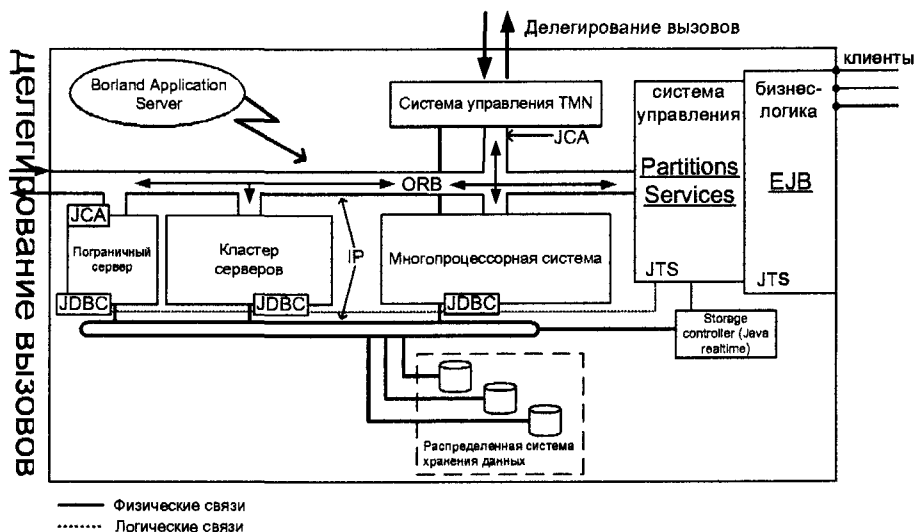


Рис. 2. Реализация мультимедийного портала операторского класса с помощью технологии J2EE

Пограничный сервер используется для осуществления защиты сети и служит шлюзом для транспортировки данных напрямую от/к другим операторам. Storage Controller является устройством реального времени, позволяющим контролировать потоки данных в сетях хранения данных (как правило, Fibre Channel, Fibre Channel over IP или Storage over IP).

Использование объектного (и как его развитие, компонентного) подхода позволяет организовать весь портал в виде объектов и отношений между объектами. Доступ к устройствам осуществляется посредством интерфейсов, при обращении к которым вызываются те или иные функции системы. Таким образом происходит инкапсуляция логики и ресурсов системы в объект. Дополнение возможностей осуществляется путем наследования логики родительских объектов.

Посредством объектов возможно решать различные задачи: кластеризации, миграции кода от оператора к оператору, эффективного распределения ресурсов на логическом уровне системы и т.д.

Одним из эффективных инструментов J2EE является Borland Enterprise Server (BES) – набор сервисов и утилит, позволяющий разрабатывать, устанавливать и управлять промышленными приложениями в корпоративной среде. Настройки BES позволяют создать распределенную среду управления

как пользовательского, так и системного уровня для управления группами объектов (рис. 3).

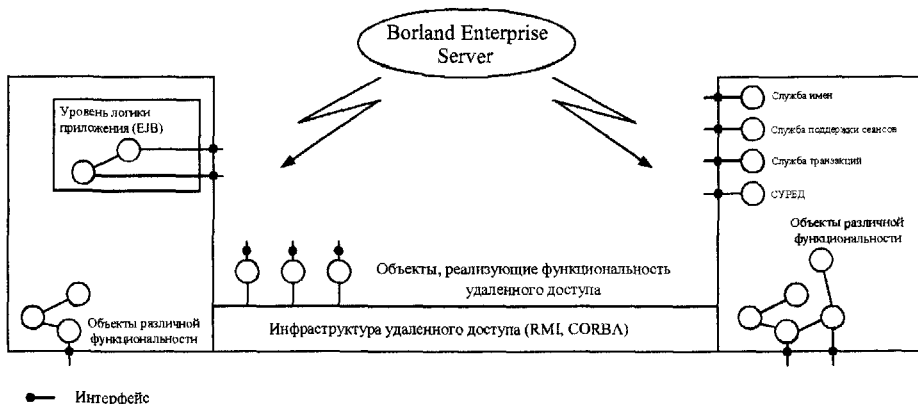


Рис. 3. Управление всеми частями распределенной системы посредством BES

Возможности управления всеми объектами системы из одного центра унифицированным способом изначально заложены в J2EE, поскольку данная архитектура придерживается стратегии расширяемости возможностей путем наследования базовых классов. Для других реализаций, например, служб на основе обмена документами XML, не существует единого подхода, позволяющего эффективно управлять всеми частями системы.

Свойство объектов быть управляемыми и наблюдаемыми (реализуется на уровне базовых классов Java) может широко использоваться для улучшения (и упрощения) функции управления при делегировании вызовов. Дело в том, что способы реализации систем управления тщательно скрываются операторами (в целях повышения безопасности и сохранения в тайне know how функциональности системы), что не позволяет повысить эффективность межоператорного взаимодействия. Судить о деталях реализации системы возможно при получении объектов (перенос кода) и при воздействии на базовые функции объектов. Особую роль при получении сведений о внутренней структуре сети играет ORB: воздействуя на его методы и свойства и наблюдая за изменением параметров системы, становится возможным принимать решения, позволяющие увеличить скорость обмена данными, тем самым уменьшив время отклика на запрос.

При наличии возможности обмена кодом и наблюдения за реакцией системы управления на первый план выходят вопросы безопасности системы. В системе должны быть реализованы механизмы идентификации, аутентификации и целостности данных. Все они изначально присутствуют в J2EE; кроме того, каждый программный код (объект, компонент, агент) сопровождается цифровой подписью источника.

При межсистемном обмене в качестве основы используется система безопасности CORBA, основанная на протоколе Kerberos, шифровании данных, общем средстве доступа к службам безопасности и прикладном интерфейсе API

общей службы безопасности. Служба безопасности разделяется на два уровня: первый уровень обеспечивает необходимый уровень безопасности приложений, для которых безопасность особой роли не играет и сводится к контролю доступа и аудиту; второй уровень поддерживает больше средств безопасности и позволяет приложениям управлять безопасностью при вызовах объектов. Служба безопасности также включает администрирование стратегии безопасности приложений, что делает стратегию переносимой.

Отдельной проблемой является создание контента. Как правило, созданием учебных материалов занимаются сотрудники (преподаватели), не обладающие специальными знаниями в области информационных систем. Данная проблема решается на уровне бизнес-логики, когда данные (обычно поступающие в виде XML-документов) анализируются и ранжируются. Система делает предсказание, насколько будет популярна информация и направляет данные в то хранилище, где скорость извлечения будет оптимальной. Как правило, для «чернового» ранжирования используется закон Ципфа, затем, в целях увеличения производительности, информация перемещается с одних носителей на другие во время отсутствия к ней запросов.

Выводы

Преимуществом использования одной технологии (в данном случае Java) является возможность максимальной интеграции частей архитектуры, которые позволяет создавать высокоэффективные приложения реального времени при минимальных временных затратах (как на освоение программистами новой технологии, так и на написание и развертывание системы). Унифицированный подход к выбору средств разработки позволяет уменьшить количество используемых протоколов и шлюзов для обеспечения взаимодействия в гетерогенной среде. Наличие библиотек Java Development Kit и тесная интеграция с серверами Apache Tomcat, OrbixWeb, iPlanet, WebSphere и многими другими позволяет дополнить среду разработки множеством функций для эффективного создания служб.

Одним из общепризнанных достоинств Java является функционирование в гетерогенной среде. Это позволяет легко реализовывать программных агентов, которые позволяют рассредоточить логику приложений по множеству серверов. Создание универсальных агентов без использования байт-кода проблематично.

Особенностью продуктов, реализованных на основе J2EE-платформы, является возможность экономной эксплуатации и наращивания возможностей системы управления ресурсами. Достаточно высокие начальные затраты компенсируются низким показателем общей стоимости владения системой (Total Cost Ownership, TCO), что позволяет при правильном ведении бизнеса через год получать прибыль (только от эксплуатации инфраструктуры; прибыль от продаж контента не учитывается).

Для создания СДО, как правило, не строится собственная высокоскоростная и высоконадежная инфраструктура: современные системы интегрируются в инфраструктуры, обладающие вышеперечисленными свойствами

(например, сети телекоммуникационных провайдеров). Поэтому необходимо специализированное программное обеспечение для подключения серверов и получения доступа к необходимым функциям. Практически все производители аппаратных средств телекоммуникационной направленности поддерживают Java путем создания библиотек со специализированным API. Поддержка CORBA позволяет достаточно просто интегрировать СДО в инфраструктуры TMN, что позволяет системе взаимодействовать с множеством стандартизированных телекоммуникационных устройств. Кроме того, при реализации Parlay-шлюзов, становится возможным использования контента мобильными пользователями. Поэтому, если необходимо универсальное решение для программирования как систем управления, так и бизнес-логики, J2EE является наиболее подходящим кандидатом. Быстрая окупаемость лицензий, множество библиотек для работы в распределенной среде, согласованность с унифицированным процессом разработки – все это позволит создать качественное надежное решение в кратчайшие сроки.

Литература

1. Цимбал А.А., Аншина М.Л. Технологии создания распределенных систем. СПб.: Питер, 2003. – 576 с.
2. Якобсон А., Буг Г., Рамбо Дж. Унифицированный процесс разработки программного обеспечения. – СПб.: Питер, 2002. – 496 с.
3. Ленди М., Сиддикви С., Свишер Д. Borland JBuilder. Руководство разработчика. – М.: Издательский дом «Вильямс», 2004. – 864 с.



Представление математических объектов средствами языка MathML в системе дистанционного обучения

Келеберда И.Н.¹, Лесная Н.С.², Маковецкий С.Д.³, Сокол В.В.⁴

Харьковский национальный университет радиозлектроники,
Харьков, Украина,

E-mail: ¹kin@kture.kharkov.ua, ²lesna@kture.kharkov.ua,

³sdmakovetskiy@ukr.net, ⁴vitalysokol@ukr.net

Abstract. The MathML-based system of high quality presentation of mathematical objects is elaborated and implemented in a course of distance learning. The inclusion of mathematical formulae in distance learning documents is realized according to the guidelines of the W3C consortium. The visualization of documents is fulfilled through the XSL-conversion which defines the type of the used browser and carries out the binding of the objects to an available viewer.