

Интеграция автоматизированных систем в единое информационное пространство

Дмитрий Волков, к.ф.-м.н., директор по развитию, "Стек Софт"

В настоящее время большинство операторов связи заинтересовано в комплексной автоматизации технологических и бизнес-процессов или, по крайней мере, в достаточно сильной интеграции различных автоматизированных систем. Большое количество используемых для решения разных задач систем с определенного момента начинает негативно влиять на развитие бизнеса в целом. Связано это, прежде всего, с необходимостью поддержания множества интерфейсов, связывающих различные системы между собой. Более того, иногда создание связующих компонентов представляется весьма сложной задачей из-за отсутствия изначально заложенных в систему механизмов интеграции или проблем, связанных с использованием так называемых унаследованных систем. Даже один компонент, не вписывающийся в общую концепцию автоматизации, но отвечающий за важную часть бизнес-процесса, может настолько сильно понизить эффективность системы в целом, что сама идея автоматизации будет представляться весьма сомнительной. Поэтому при проектировании информационного пространства особое внимание следует уделять механизмам интеграции систем, неважно, идет ли речь о создании и внедрении новых систем или реконструкции унаследованных.

Принципиально можно выделить три основных типа интеграции систем:

- интеграция на уровне данных;
- интеграция на уровне методов работы с данными;
- смешанная интеграция.

В первом варианте разные системы работают над общими данными, но каждая при этом использует собственные методы. В этом случае необходимо регламентировать работу с общим

хранилищем данных или обеспечить механизмы передачи данных от одной системы к другой. В качестве примера этого типа интеграции можно привести взаимодействие биллинговой и бухгалтерской системы в части передачи информации о начислениях за услуги из биллинговой системы в бухгалтерскую для формирования соответствующей отчетности.

Во втором варианте разные системы используют для работы над собственными данными одинаковые методы. Примером опять же может послужить интеграция биллинговой и бухгалтерской систем в части вычисления значения налогов по тем или иным схемам, законодательно утвержденным для всех систем. Логичнее выделить сложный и часто модифицируемый алгоритм в общий метод и пользоваться им из всех заинтересованных систем, чем поддерживать его в каждой отдельно. Помимо алгоритмов вычисления к общим методам относят механизмы авторизации, компоненты пользовательского интерфейса, работу с периферийными устройствами и т.п.

Наконец, третий тип подразумевает, что в процессе взаимодействия системы могут использовать как общие данные, так и общие методы. И чем сложнее технологические и бизнес-процессы компании, а также варианты их взаимодействия, тем предпочтительнее выглядит данный тип интеграции.

Действительно, в довольно большом разнообразии задач всегда можно выделить как подмножество общих данных, так и набор общих методов. С точки зрения построения единого информационного пространства, формализация этих подмножеств и наборов - наиболее важная задача на этапе первоначального проектирования новой, или на этапе реконструкции унаследо-

ванной, системы. Другой важной задачей является выбор методологии работы с общими данными и методами. В этой статье мы не будем рассматривать совместную работу приложений над общими данными - эта проблематика к настоящему моменту хорошо проработана, обратим внимание на форматы представления данных, предназначенных для обмена информацией, и на технологии разработки и использования общих методов.

В настоящее время наиболее бурно развиваются два направления интеграции различных приложений и систем на их основе:

- разработка промышленных стандартов для описания сущностей той или иной предметной области на языках типа SGML/XML [1];
- разработка инструментов для реализации интероперабельных (взаимодействующих и совместно использующих различные ресурсы) систем на основе объектной модели.

К сожалению, это бурное развитие к настоящему моменту не принесло ожидаемых результатов: промышленных стандартов в областях, наиболее интересных операторам связи, до сих пор нет, да и перспективы их появления весьма туманны, а имеющиеся в настоящий момент инструменты разработки интероперабельных систем не позволяют в общем случае создавать высокоэффективные приложения. Но теория обоим направлениям на сегодняшний день уже вполне состоялась и может быть применена.

Основное достоинство описания сущностей на языке XML состоит в простоте преобразования имеющегося представления в какое-либо другое, будь то XML документ другого формата, "родной" формат другого приложения или даже выходной доку-



мент. Таким образом, если изначально предусмотреть наличие XML представлений для внутренних сущностей системы, то в дальнейшем вопросы интеграции в области обмена данными будет решать гораздо проще. Даже если впоследствии появится промышленный стандарт для сущностей данного типа, то преобразование уже имеющегося представления к стандартному виду не должно вызвать серьезных проблем. Больше того, подобное преобразование может быть осуществлено без вмешательства во внутреннюю логику системы путем установки соответствующих конверторов перед входными и выходными интерфейсами. Несмотря на то, что XML - прежде всего, язык описания данных, весьма перспективным представляется его использование для описания действий, необходимых для обеспечения интеграции различных систем. Действительно, запросы одной системы к другой на выполнение тех или иных операций удобно описывать в виде универсальных XML уведомлений.

В основе интероперабельных систем лежит объектная модель, причем объект - это самостоятельная часть программного кода, реализующая какие-либо сервисы. Для доступа к этим сервисам объект предоставляет потенциальным потребителям набор методов, причем во главу угла ставится одно из основных свойств объектной технологии - инкапсуляция, то есть декларируется лишь интерфейс методов объекта, вся реализация

этих методов скрыта от внешнего мира. Взаимодействие интероперабельных систем происходит благодаря обмену интерфейсами объектов (одна система знает о сервисах, предоставляемых другой) и наличию возможностей использования одной системой сервисов другой. Таким образом, процесс проектирования нового приложения (реализации нового технологического или бизнес-процесса) сводится к анализу применимости уже существующих объектов для реализации требуемой функциональности и интеграции их в законченный. Недостающие компоненты могут быть разработаны и включены в проект как локальные или могут быть оформлены как объекты, доступные для использования в других приложениях. В настоящее время существует несколько идеологий подобного "объектного мира", наиболее

известной и проработанной в данный момент является идеология, предложенная консорциумом Object Management Group (OMG) - стандарт Common Object Request Broker Architecture (CORBA) [2]. Для получения более полной информации об интероперабельных системах следует обратить внимание и на более практическую концепцию - Enterprise Java Beans (EJB) [3], предложенную компанией Sun Microsystems. Даже если использование данных технологий в "чистом" виде представляется неэффективным, при разработке собственной платформы весьма полезно учесть этот опыт и обеспечить совместимость на уровне протоколов с системами, созданными на их основе.

Концепция комплексной автоматизации, предложенная компанией "Стек Софт", сочетает оба направления и подразумевает разработку на основе единой объектной платформы линейки программных продуктов для

- управления ресурсами и конфигурации оборудования;
- авторизации и сбора статистики;
- оценки показателей качества сервиса.

Интегрирующим ядром АСР "Опума" является система управления ресурсами, ответственная за описание, управление и конфигурацию всех имеющихся у оператора ресурсов, как физических, так и логических: оборудования, канальной инфраструктуры, адресного пространства, преysкурантов, клиентской базы и т.п. Система способна самостоятельно осуществлять управление оборудованием путем формирования конфигурационных файлов и рассылки управляющих воздействий, а также опознавать внешнее изменение инфраструктуры, получая соответствующие XML уведомления. Таким образом, система в любой момент времени обладает полной информацией о том, какие ресурсы кем используются, что позволяет организовать эффективное управление инфраструктурой в целом и является хорошей базой для построения CRM-систем и систем мониторинга (некоторые функции этих систем реализованы непосредственно в АСР "Опума").

Ядро системы поддерживает набор методов работы с данными, которым может воспользоваться любая из работающих с ним систем. В итоге стало возможным использование единой системы авторизации (тем не менее, с жестким контролем полномочий), стандартизованного пользовательского интерфейса, аналитического блока и т.п. Благодаря тому, что различные системы работают в одном информационном пространстве, появляется возможность реализации сложных алгоритмов взаимодействия, например, таких как отслеживание с помощью системы мониторинга перерывов в предоставлении сервиса, вычисление по схеме взаимосвязей ресурсов пострадавших клиентов и, при необходимости, автоматическая компенсация в соответствии с условиями регламента предоставления услуг. Так как описание форматов и управление данными является исключительной прерогативой ядра системы, значительно упрощается процесс взаимодействия с внешними по отношению к продуктам "Опума" системами. Импорт и экспорт данных осуществляется единичными методами и может быть выполнен ядром по запросу одной из работающих с ним "род-

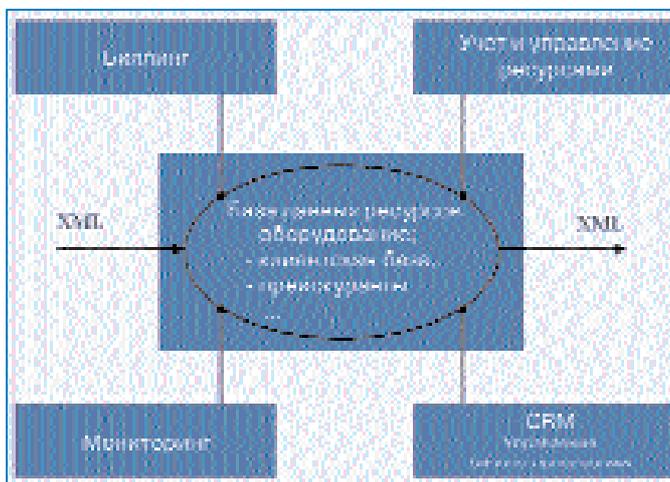


Рис. 1. Взаимодействие различных систем через единое ядро

решения отдельных задач и тесную интеграцию систем разных производителей на основе стандартизованного обмена данными. Основная идея этого подхода состоит в создании единого интеграционного ядра, на базе которого строятся продукты для автоматизации различных аспектов деятельности оператора, при этом обмен данными с внешними системами осуществляется при помощи передачи и приема XML представлений различных сущностей.

Первым продуктом компании "Стек Софт", построенным по данной технологии, стала Автоматизированная Система Расчетов (АСР) "Опума" [4]. Помимо отвечающего всем современным требованиям расчетного модуля, данная система позволяет решать задачи, не имеющие, с общепринятой точки зрения, непосредственного отношения к биллингу:

ных" систем.

Описанный подход дает значительную экономию финансовых и людских ресурсов, необходимых на внедрение и поддержку систем, и значительно повышает эффективность работу в информационном пространстве оператора в целом.

Литература.

1. Applying XML and Web services standards in industry. <http://www.xml.org>
2. Object Management Group. <http://www.omg.org>
3. Enterprise JavaBeans. - Sun

Microsystems <http://java.sun.com>

4. Волков Д.А. "Онума - биллинг и управление ресурсами в распределенных сетях", Компьютерная телефония. Биллинг, N5(11) 2001 г.

5. Онума. <http://www.onyma.ru>