

ИНТЕГРАЦИЯ ИТ: ОСНОВНЫЕ ПОНЯТИЯ И ТЕХНОЛОГИИ

Интеграция – задача очень объемная. Она тянет за собой огромную цепочку проблем, которые приходится решать компаниям, вставшим на путь обновления ИТ-инфраструктуры. Это и адаптация программных модулей под специфику деятельности, и корпоративные закупки новых аппаратных средств, и обучение сотрудников новым технологиям.

Интеграция может рассматриваться как возможность обмениваться данными между приложениями или как полная реструктуризация информационной среды предприятия для решения каких-то сверхзадач. Если только первое не вызывает особенных сложностей, то соответствующие технологии приняты и внедрены уже давно. Во втором случае за изменением тянется целая цепочка других проблем: как обеспечить «диалог» между программами, какие решения предпочесть для хранения данных, для разработки и взаимодействия программ и т. д. Задача усложняется еще и тем, что новые модули нужно объединять с существующими так, чтобы работа не останавливалась надолго. Иногда это находится на грани невозможного, поскольку методы интеграции приложений и данных порой требуют полного изменения структуры и формата БД, из-за чего перестают работать старые программы.

Сейчас есть три технологии, которые именуют «три "И"» (или «три "Е"» в английском варианте). Это интеграция корпоративных приложений (enterprise application integration, сокр. EAI), интеграция корпоративной информации (enterprise information integration, сокр. EII) и программное обеспечение для извлечения, преобразования и загрузки данных (extract, transform and load, сокр. ETL).

Эти технологии могут быть использованы для широкого круга задач: от интеграции в режиме реального времени до пакетной интеграции и от интеграции данных до интеграции приложений.

Краткие определения каждой из функций:

EAI – это технология, с помощью которой организация добивается централизации и оптимизации интеграции корпоративных приложений, обычно используя те или иные формы технологии оперативной доставки информации (push technology), которая управляется внешними событиями (event-driven);

ETL – это технология, которая преобразует данные (обычно с помощью их пакетной обработки) из операционной среды, включающей гетерогенные технологии, в интегрированные, согласующиеся между собой данные, пригодные для использования в процессе поддержки принятия

решений. ETL-технология ориентирована на базы данных, например, хранилище, витрину или операционный склад данных;

EII – это технология для интеграции в режиме реального времени несопоставимых типов данных из многочисленных источников как внутри, так и за пределами корпорации. Инструменты EII обеспечивают универсальный уровень доступа к данным и используют технологию поиска информации (pull technology) или возможности работы по запросам. Технология EII ориентирована на конкретных сотрудников, которые получают информацию через инструментальную панель или отчет [3].

Далее необходимо рассмотреть место этих технологий в уже существующей архитектуре. Технология EAI интегрирует транзакции двух или более приложений, технология ETL интегрирует данные операционных систем и компонентов поддержки принятия решений, а технология EII осуществляет виртуальную интеграцию данных из различных источников.

Технология EAI наиболее функциональна тогда, когда необходимо связать приложения в реальном времени для автоматизации бизнес-процессов. Второй случай применения EAI – это ситуация, когда необходимо, чтобы изменения, внесенные в одно приложение (обычно это небольшой набор записей), были отражены во всех других. Эта технология очень хорошо справляется с задачей фиксации изменений и их переноса в соответствующие приложения или системы. В бизнес-среде под «интеграцией приложений» понимается адаптация существующих настольных приложений, ERP-компонентов, данных в БД-форматах или открытых XML-стандартов, а также Java-апплетов к созданию новых бизнес-возможностей за счет совместного использования распределенных ресурсов.

Технология ETL оказывается наиболее полезной в тех случаях, когда необходимо создать хранилище данных, содержащее хорошо документированные и надежные данные для исторического анализа, например для анализа временных рядов или многомерных запросов. Эта технология также используется для интеграции ключевых справочных данных. Технология ETL незаменима для таких задач, как удаление дублирующихся данных, осуществление процессов проверки качества данных и т. п. Эти инструменты также используются для создания отдельных витрин данных, обслуживающих конкретный отдел или бизнес-процесс или предназначенных для каких-либо долгосрочных целей. Инструменты ETL дают пользователю возможность запустить повторяющиеся процессы для большей слаженности действий и возможности их многократного использования. Такие процессы включают создание точных технических метаданных, поддерживающих общую целостность среды business intelligence (BI) [3].

Технология EII лучше всего подходит в тех случаях, когда необходимо создать общий шлюз (gateway) с единым языком и точкой доступа

к несогласованным источникам данных. Такие инструменты предоставляют приложениям и конечным пользователям возможности более гибкого, а также незапланированного доступа к данным, при этом не требуя постоянного использования данных или долговременных целей для получения этого доступа. Помимо традиционных реляционных баз данных, инструменты ЕИ могут работать с XML- и LDAP-файлами, плоскими файлами и другими нереляционными данными. Эти инструменты также способны представлять реляционные данные в формате XML или формате web-сервисов. Особенно полезны инструменты ЕИ, если есть необходимость добавить к справочным данным хранилища дополнительные детали, в частности детальную информацию в реальном времени (например, сопоставление исторической данных с текущей ситуацией).

Кроме понимания того, когда необходимо использовать эти технологии, нужно также знать и проблемы, которые им присущи. Во-первых, внедрение этих технологий требует от ИТ-персонала глубокого понимания тех требований, которые предъявляются к данным для принятия как тактических, так и стратегических решений. Применительно к технологии ETL это означает, что необходимые данные извлекаются, преобразуются и загружаются в виде, пригодном для использования непосредственно аналитиками или ЕИ-сервером. В случае ЕИ-технологии, способы представления данных должны удовлетворять отчетным требованиям аналитиков, т. е. данные должны быть пригодны для использования в аналитических отчетах. Во всех случаях понимание источников данных и требований, предъявляемых к данным, является необходимым шагом при внедрении этих технологий и безусловно оправдывает то время, которое приходится тратить, чтобы достичь этого понимания.

Кроме того, необходимо понимать, что внедрение этих инструментов в уже сложившуюся архитектуру требует от бизнес- и ИТ-персонала разработки такой стратегии управления данными и приложениями, которая будет постоянно поддерживать этот процесс в активном состоянии. Обязательной составляющей такой стратегии должно быть осознание того, что повышается важность механизмов архивирования, а также того, что с самого начала должны быть созданы контрольные журналы. Это необходимо для обеспечения слаженности и надежности интегрированных данных и приложений.

И наконец, очень важен постоянный мониторинг производительности и эффективности этих технологий в условиях конкретной инфраструктуры. Их производительность в значительной степени будет зависеть от скорости архивирования данных, размеров и детальности данных, а также от эффективности функционирования системы в условиях полной нагрузки. При определении производительности также следует оценить влияние, которое эти инструменты могут оказывать на операцион-

ные приложения и системы. Поэтому необходим постоянный мониторинг и этого влияния [4].

Интеграция корпоративных приложений (ЕАИ) актуальна и по сей день, и особенно для корпоративных пользователей. Дело в том, что для большинства организаций поддержание в синхронизованном состоянии данных из множества гетерогенных систем по-прежнему является серьезной задачей. Именно поэтому ЕАИ и остается важнейшим типом интеграции. Рассмотрим эту технологию подробнее.

Методы интеграции

Для взаимодействия приложений обычно используются такие методы, как обмен файлами, общая база данных, удаленный вызов и асинхронный обмен сообщениями. В этом списке нет прямого обмена данными между базами данных приложений: этот метод ближе не к интеграции приложений, а к перемещению данных. С точки зрения интеграции приложений важна возможность в процессе обмена данными выполнять какую-то содержательную обработку (например, при загрузке накладных пересчитывать товарные остатки). Прямой обмен данными, который обычно выполняется средствами класса ETL (extract, transfer, load) или самодельными утилитами, обычно такой возможности не предоставляет.

- **Обмен файлами**

Обмен файлами пожалуй, самый распространенный подход к организации взаимодействия. Это связано с относительной простотой реализации, а также существованием стандартных (или «почти» стандартных) форматов обмена. Например, большая часть корпоративных информационных систем позволяет загружать и выгружать файлы, например в формате CSV (Comma-Separated Values – «поля, разделенные запятыми»). Но у этого подхода есть и недостатки; если необходимо оперировать сложными структурами, то простые форматы обмена уже не пригодны. Возникающие в таких случаях специализированные форматы файлов должны «понимать» взаимодействующие системы, что ведет к жесткой зависимости систем друг от друга. Этот недостаток обычно преодолевают всевозможными утилитами конвертации данных. Кроме того, обычно обмен файлами подразумевает участие человека – кто-то должен выгрузить файл, скопировать его на другой компьютер, загрузить. Однако если интегрируемые методом обмена файлами системы имеют возможность автоматической загрузки/выгрузки (например, по расписанию), то данный подход позволяет построить полностью автоматизированное решение, которое вследствие своей простоты обладает высокой надежностью и пропускной способностью.

- Общая база данных

Данный подход концептуально очень прост: несколько информационных систем или приложений используют одну базу данных. Главный его недостаток – связь между интегрированными приложениями настолько тесная, что иногда невозможно заметить границу между ними (обычно так интегрируются продукты одного производителя). Примером такого подхода могут служить большинство ERP-систем, где различные модули системы используют одну базу. Однако слишком тесная связь превращает конгломерат интегрированных приложений в монолит, в «суперсистему», отдельные части которой с трудом поддаются самостоятельной модернизации и замене. С этим борются, используя механизмы серверов баз данных (представления данных, промежуточные таблицы и т. п.), но далеко не всегда эффективно.

- Удаленный вызов

Стандарты на удаленный вызов процедур возникли два десятка лет назад, позволяя программному коду, который выполняется на одном компьютере, вызывать код на другом. Стандарты: RPC, CORBA, DCOM, SOAP. Основной недостаток удаленного вызова – требование работоспособности всех задействованных приложений в момент взаимодействия.

- Асинхронный обмен сообщениями

Это, пожалуй, единственный из перечисленных выше подходов, который создавался специально для интеграции информационных систем. Идея концептуально проста и напоминает работу электронной почты. Когда приложению А необходимо вызвать какое-то действие в приложении Б, оно формирует соответствующее сообщение с данными и инструкциями и отправляет его посредством системы доставки сообщений. Слово «асинхронный» означает, что приложение А не должно ждать, пока сообщение дойдет до Б, будет обработано, сформирован ответ и т. п. Сообщение гарантированно доставляется благодаря механизму очередей сообщений, которые снимают с взаимодействующих систем заботу о надежности сети передачи данных, работоспособности взаимодействующих систем в конкретные моменты времени и т. д.

Недостаток данного подхода – высокая цена. Система гарантированной доставки на основе очередей сообщений обычно сама по себе недешева; единственным известным мне исключением является Microsoft Message Queue (MSMQ), компонент серверных операционных систем семейства Windows. Правда, есть и свободно распространяемые бесплатные (например, ActiveMQ), которые, тем не менее, нужно развернуть, обучить специалистов, поддерживать, написать адаптеры между системой доставки и приложениями и т. д. [5].

Библиографический список

1. Интеграция корпоративных приложений: основные понятия [Электронный ресурс].– Режим доступа: <http://www.citcity.ru/11132/>.
2. Имхофф Клодия (Claudia Imhoff). Аналитические решения: понимание трех составляющих [Электронный ресурс].– Режим доступа: <http://www.citcity.ru/11156/>.
3. John Taylor. Thoughts from the Integration Consortium: Enterprise Information Integration: A New Definition (Вести из Консорциума по интеграции. Интеграция корпоративной информации – новое определение) [Электронный ресурс].– Режим доступа: http://www.dmreview.com/article_sub.cfm?articleId=1009669.
4. Радаев Александр. Как сделать интеграцию бизнес-приложений эффективной? [Электронный ресурс].– Режим доступа: <http://www.insapov.ru/integration-business-application.html>.
5. Боркус Владислав. Методы и инструменты интеграции корпоративных приложений: Отчет/ RC Group.– М.: RC Group, 2006.– 13 с.
6. Дэвид Оринштейн. Интеграция корпоративных приложений.– М.: Microsoft Press, 2002.– 454 с.
7. Советов Б.Я. Информационные технологии: Учебник для студентов вузов / Б.Я. Советов, В.В. Цехановский.– М.: Высшая школа, 2006.– 260 с.
8. Проектирование и реализация баз данных Microsoft SQL Server 2000 // Официальный учебный курс Microsoft.– 3-е издание.– М.: Питер, 2006.– 650 с.
9. Alexander C., Ishikawa S., Silverstein M. A Pattern Language – Towns Buildings Construction.– Oxford University Press, 1977.
10. Alpert S., Brown K., Woolf B. The Design Patterns Smalltalk Companion. Addison-Wesley, 1998.