Уфа : УГАТУ, 2009 Вестник УГАТУ • Управление, ВТ и И Т.12, № 1(30). С. 127–132

УДК 004.45

Д. В. ТОРШИН, Н. И. ЮСУПОВА

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ ЗАДАЧИ ИНТЕГРАЦИИ РАЗРОЗНЕННЫХ КОМПЬЮТЕРНЫХ СИСТЕМ

В статье представлены основные результаты проведенного диссертационного исследования, посвященного интеграции компьютерных систем обработки данных (информационных систем) в единое интегрированное информационное пространство. Отражены основные положения предложенного подхода и описаны результаты внедрения прототипа программного обеспечения, реализующего подход. *Компьютерная система* ; *интеграция* ; *обработка данных*

Проблема интеграции информационных систем изучается с момента, когда получили распространение первые базы данных – с 70-х гг. XX века. До настоящего времени задача интеграции остается недостаточно проработанной, поскольку наряду с комплексностью и высокой сложностью данной задачи, в связи со стремительным развитием информационных технологий меняются и требования к самим информационным системам, а как следствие – и к их интеграции.

Проблема интеграции информационных систем и ресурсов затронута в публикациях таких авторов, как А. Кудинова [2], Н. В. Ермакова и др. Корпорации IBM, Microsoft, Oracle и др. ведут собственные закрытые исследования данной проблематики и предлагают готовые программные решения.

В данной статье рассматриваются основные подходы к организации информационной системы, интегрирующей данные нескольких независимых информационных систем с произвольной структурой данных, и приводится пример реализации такой системы для анализа эффективности.

ПОСТАНОВКА ЗАДАЧИ

Рассмотрим задачу интеграции информационных ресурсов различных информационных систем для получения возможности проведения комплексного анализа данных и связывания информационных систем в рамках межсистемных бизнес-процессов.

Наиболее удачным решением для такой задачи будет построение централизованной системы интеграции данных [1], которая возьмет на себя функции связи данных из нескольких ИС.

Рассмотрим основные требования к такой централизованной системе интеграции данных для обеспечения запрашиваемой функциональности.

Пусть имеются N информационных систем, работающих в одной предметной области. Задача и данные каждой информационной системы могут пересекаться с задачами и данными других информационных систем. Кроме того, пусть задано ограничение – невозможность внесения корректировок в уже существующие информационные системы-источники.

Требуется разработать информационную систему, позволяющую интегрировать данные N систем в единое представление для оперативного доступа к консолидированной информации при требовании уменьшения нагрузки на каналы связи и вычислительную мощность.

Основываясь на практическом опыте авторов статьи и результатах исследовательских работ [2], [3], [4], [5], [6] формулируются требования к функциональной возможности интегрирующей информационной системы (далее – ИИС). ИИС должна поддерживать следующие функции:

Контактная информация: (347)273-77-17

● консолидация данных [2] различных информационных систем с описанием структуры данных в центральном узле;

● функция импорта/экспорта данных;

● работа в автономном и оперативном режимах: данные могут быть добавлены в ИИС и считаны из ИИС как в реальном времени, так и обработкой отложенных запросов;

● возможность подключения новых ИС к ИИС с помощью настраиваемых расширений;

● функция системного историзма для предотвращения изменения данных без возможности восстановления;

● функция автоматического контроля качества данных, добавляемых в ИИС;

● функция выверки дубликатов записей и объектов;

● расширенная система прав, ограничивающих работу с данными на самом низком уровне к хранилищу данных;

● универсальный инструмент визуального представления данных для просмотра информации, хранимой в ИИС (интерфейс пользователя);

● функция сложного поиска по данным в ИИС (язык запросов данных).

АЛГОРИТМИЗАЦИЯ МЕТОДОВ РАБОТЫ ИИС

Алгоритм добавления/изменения данных в ИИС включает в себя несколько этапов (рис. 1).

Данные в виде пакета поступают на один из входов, где проверяются на верность формата данных, воспринимаемого ИИС. Затем данные трансформируются в формат ИИС, используя при этом настройки конкретной ИС (так называемые шаблоны преобразования данных).

После успешной трансформации данные валидируются (проводятся автоматические проверки на соответствие форматам данных, их полноту и т. п.).

Выверка дубликатов определяет действие (добавление нового объекта или изменение уже существующего в базе данных ИИС), а также конкретизирует, какие именно поля нужно изменить.

После выверки дубликатов происходит сохранение изменений в базе данных ИИС. В этот момент срабатывает системный историзм, который не только меняет данные в базе данных, но и запоминает предыдущее состояние, а также кто и когда производит изменения.

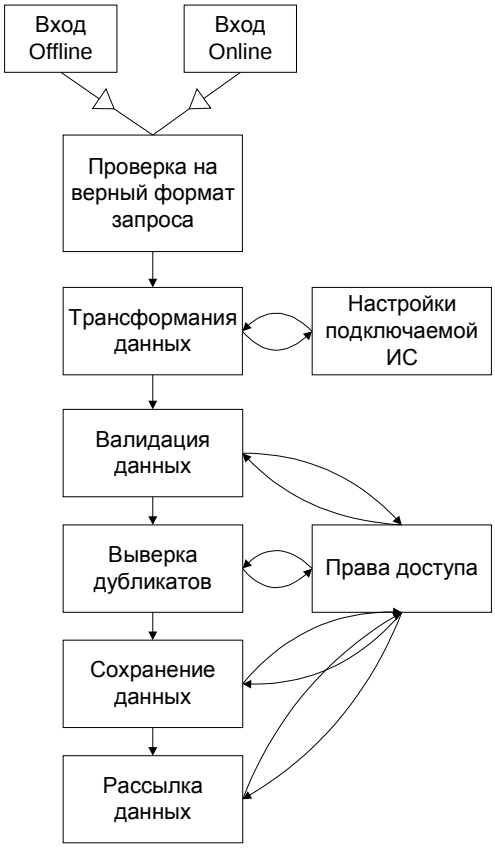

Рис. 1

Три последних шага алгоритма обращаются к информации о правах доступа, чтобы не возникло ситуации, когда меняются данные, на изменение которых у ИС нет прав.

После сохранения данных сразу или через некоторый промежуток времени производится рассылка изменений подписанным ИС.

Получение данных из ИИС возможно двумя способами: либо ИИС самостоятельно рассылает измененные данные с помощью системы рассылки, либо ИС посылает запрос, а ИИС отправляет ответ с данными.

В случае рассылки алгоритм прост: сама рассылка – это такой же запрос, отправляемый ИИС самой себе от имени ИС, которая должна получить рассылку.

Таким образом, данные из ИИС выгружаются всегда по поисковому запросу (рис. 2).

Шаги проверки на верный формат запроса и трансформации данных остаются без изменений. Валидация данных в данном алгоритме проверяет сами критерии поиска, а также корректность запрашиваемой информации на предмет доступности по системе прав доступа.

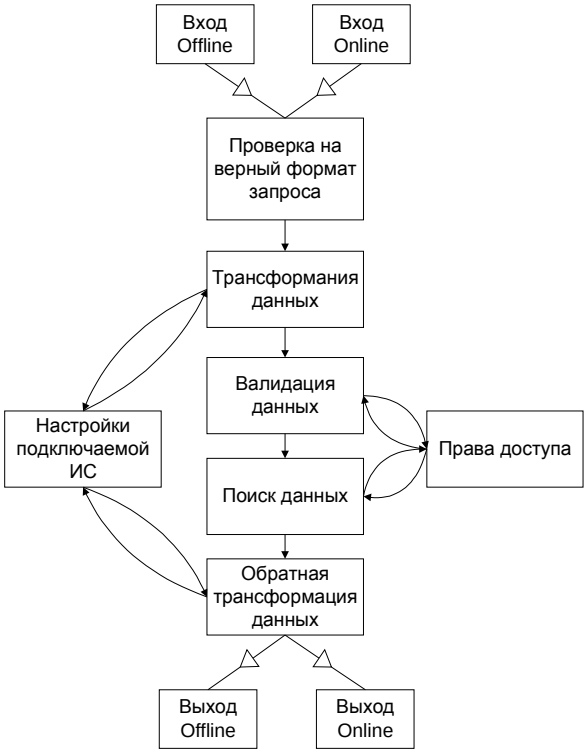

Рис. 2

Если валидация пройдена успешно, запускается непосредственно механизм поиска данных, который формирует обратный пакет данных для запрашивающей ИС.

Сформированный пакет проходит стадию обратной трансформации структуры данных и самих данных: запрашивающая система получает данные в том формате, в котором и отправляла запрос, независимо от структуры базы данных ИС.

После завершения трансформации информация отправляется запрашивающей стороне.

ПРОГРАММНОЕ СРЕДСТВО – РЕАЛИЗАЦИЯ

Описанные алгоритмы реализованы в программной среде, названной Data Manipulation Core (DMC) – программном решении, удовлетворяющем всем требованиям, предъявленным ИИС.

DMC – это серверный программный компонент, устанавливаемый в системе в виде сервиса. Он выполняет роль платформы для разработки конкретной ИИС и берет на себя реализацию большинства заявленных функциональных требований.

DMC обладает возможностью Object-Relational Mapping (ORM-преобразований) [7]. Данная возможность позволяет программисту, разрабатывающему ИИС, использовать понятие «класс», «объект», «поле объекта» и т.п. вместо прямого обращения к базе данных. Программист не заботится о физическом хранении объектов в базе данных и доступе к ним посредством SQL-запросов – эту функциональность берет на себя модуль ORM.

Поскольку структура данных в ИИС должна быть такой, чтобы данные, передаваемые в ИИС из всех ИС, имеющих на это доступ, можно было разместить в базе данных, структура базы должна быть достаточно гибкой, простой, но объединяющей все прочие структуры.

В DMC программист описывает классы, например, «Человек» и задает параметры, а также их типы. DMC автоматически создает в базе данных соответствующие сущности, связи и т.п. и следит за целостностью. Более того, прочие модули DMC также работают с классами и объектами, описанными программистом, и не зависят от базы данных.

DMC позволяет импортировать и экспортировать данные из своего хранилища, преобразовывая объекты любых описанных классов в XML-нотацию или наоборот, из XML-записи в объекты в оперативной памяти.

Универсальный конвертер XML-файлов, являющийся внешним модулем для DMC, позволяет изменить формат структуры данных XML.

В DMC в качестве внешнего модуля реализован Web-сервис, который берет на себя функции взаимодействия в режиме реального времени. Второй внешний модуль – модуль ftp – позволяет реализовать асинхронный отложенный режим. При этом он работает по таймеру, опрашивая через заданные промежутки времени входящую папку на ftp-сервере и проверяя наличие новых пакетов данных.

Оба модуля передают полученные XML-запросы сервису DMC, который берет на себя их обработку (преобразование в нужный формат, превращение в объекты соответствующих классов, выверку и сохранение в базе данных или подготовку ответного пакета).

Модуль преобразования форматов, использующий DMC, но являющийся внешним по отношению к сервису приложением, использует шаблоны преобразования XML-файлов для конвертации форматов и структур данных.

В хранилище DMC также содержится информация о каждой ИС, подключенной к ИИС (включая права доступа). Данная информация

используется модулем преобразования форматов при конвертации.

Одной из уникальных особенностей ORM-системы DMC является встроенный системный историзм всех операций изменения данных в хранилище. Любой объект, который добавляется, изменяется или удаляется, благодаря системе историзма DMC не меняется безвозвратно – в хранилище сохраняется его копия в «старом» виде. Модуль историзма затем позволяет восстанавливать данные из этих копий.

Модуль историзма настраивается таким образом, чтобы удалять самые старые копии (например, если объект меняется 50-й раз, то его первая версия удаляется). Однако при наличии достаточных ресурсов рекомендуется не использовать данную настройку.

В DMC функция преобразования объектов классов из XML-файлов в объекты в оперативной памяти проверяет каждое поле на соответствие формату, весь объект – на целостность и полноту данных и т. п. Проверяются необходимые связи и непротиворечивость данных. Все это производится автоматически модулем импорта пакетов.

Задача валидации и обеспечения совместимости данных выходит за рамки рассмотрения в данной статье, однако отметим, что для проведения данного анализа DMC предоставляет все необходимое. Во-первых, DMC проводит анализ структур данных на этапе преобразования объектов из формата в формат. Во-вторых, благодаря встроенному модулю приоритезации информационных систем-источников проводится проверка на возможность внесения (обновления) объекта в центральном хранилище. В-третьих, благодаря применению в шаблоне преобразования регулярных выражений, задающих ограничение на сами данные, у администратора системы появляется удобный простой инструмент задания правил валидации данных. Наконец, в-четвертых, благодаря объектно-ориентированному подходу к описанию структур данных при задании структур центрального хранилища программист может описать на любом Microsoft .Net Framework 2.0-совместимом языке программирования собственные функции-валидаторы любого уровня сложности. Данные функции DMC автоматически запускает при помещении данных в хранилище (такие функции работают по принципу триггеров-событий).

При назначении в объектах идентификаторов DMC следит за тем, чтобы в хранилище не оказалось дубликатов. Система выверки дубликатов является одной из самых сложных с алгоритмической и вычислительной точек зрения, поскольку позволяет учесть сложные комбинации групп ключевых полей (объектов), которые могут включать поля различных сущностей, а также принимает решения о слиянии и разделении объектов при обнаружении конфликтных ситуаций.

Так, система выдает объекту при сохранении новый идентификатор, если объект «не похож» ни на один существующий, или может объединить объект с другими, выдав единый идентификатор.

Для некоторых классов объектов такие функции объединения запрещаются – это определяется программистом.

В ИИС система прав доступа имеет огромное значение, поскольку система должна предотвращать затирание данных из корректных источников данными из некорректных источников.

Поэтому в DMC разработана такая система прав доступа, которая настраивается как на классы, так и на конкретные объекты классов, на отдельные поля объектов, на группы полей, а также на группы объектов или классов. При этом у программиста ИИС остается возможность расширить систему прав особыми программируемыми на языке C# условиями.

В DMC используется также система приоритетов источников данных для различных сущностей, их комбинаций, а также деталей: полей, групп полей. Система работает следующим образом: если объект был добавлен в хранилище из более приоритетного источника, его нельзя затереть или изменить менее приоритетным, а вот заполнить пустующие поля – можно.

В качестве клиентской программы, называемой DMC Client, используется Windows-приложение, взаимодействующее с любой системой, построенной на основе DMC.

Клиент умеет взаимодействовать с ORM, классами и объектами, а также отображать их визуально, линейно и т. п. Он умеет строить универсальные поисковые формы по классам, редактировать объекты и сохранять изменения в хранилище данных DMC. При этом клиент является мощным инструментом прямого изменения данных в хранилище, поэтому использовать его должны исключительно администраторы системы.

DMC Client позволяет задавать сложные условия поиска по объектам, их подобъектам и

связанными с ними объектам. Поиск позволяет задавать критерии по полям, сущностям (объектам), группам объектов и т. п. Сами критерии могут быть единичными (например, «1» или «master*»), групповыми (например, «1, 2, 4, 8» или «Me, you, they»), интервальными (например, «1–15») и смешанными (например, «1–4, 6, 8–1000, 2000»).

Поиск не обеспечивает задание критериев вычислимыми функциями (такими, как сумма, количество и т.п.).

DMC имеет компонент подписки ИС на изменения данных. По таймеру проверяются все измененные данные, которые могут интересовать каждую систему (поскольку все изменения в данных сохраняются с точной датой и временем, найти «свежие» несложно), готовится группа объектов, преобразовывается в XML и отсылается в автономном режиме ИС-получателю.

АНАЛИЗ ЭФФЕКТИВНОСТИ ПОДХОДА НА ОСНОВЕ ОЦЕНКИ РАБОТОСПОСОБНОСТИ РАЗРАБОТАННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Экспериментальный анализ эффективности предложенного подхода к интеграции и разработанного программного обеспечения – DMC – производился путем тестовой интеграции 3 информационных систем, наполненных данными в разных структурах:

1) информационная система 1 – 52544 объектов нежилого фонда. Адрес объекта описывается в произвольной форме;

2) информационная система 2 – 81057 адресов зданий и построек. Адреса имеют жесткую структуру с разбиением на город, улицу, район, номер дома, литеру, дробь;

3) информационная система 3 – 7488 договоров аренды нежилых помещений. Связка с объектами нежилого фонда – прописываем адреса объекта в произвольной форме.

Для тестовой интеграции на основе DMC были разработаны шаблоны преобразования, соответствующие структурам данных информационных систем, а также был разработан общий универсальный формат данных.

В ходе экспериментальной загрузки в систему объектов были получены следующие результаты на компьютере с процессором Intel Core 2 Duo 2,4 ГГц и 2 Гб ОЗУ:

1) общее время загрузки информации из 3-х систем – 24 минуты;

2) количество принятых системой данных из первой информационной системы – 35223 объекта, из второй – 57894, из третьей – 6102. Поскольку договор аренды ссылается на объект аренды с учетом адреса, то валидацию и выверку прошли 6102 договора, которые были привязаны к соответствующим объектам нежилого фонда по адресам.

Экспертный анализ результатов работы системы позволил сделать следующие выводы:

1) скорость работы системы является приемлемой (обработано свыше 140 тысяч объектов без учета связей между ними), поскольку общий объем обработанной входной информации превысил 200 Мбайт, а оперативный объем выделенной памяти превысил 100 Гбайт;

2) эффективность автоматического метода валидации и выверки не позволили сохранить в хранилище данных объекты с неверными адресами, ошибками и неполной информацией. Таким образом, в хранилище оказались только полные и выверенные данные;

3) все данные, которые не прошли фильтры по критериям полноты, непротиворечивости и точности, были выделены отдельным списком с указанием причины непринятия системой результатов.

Таким образом, экспериментальный анализ разработанного программного обеспечения показал эффективность предложенного подхода.

ЗАКЛЮЧЕНИЕ

Теоретические выкладки о требованиях к интегрирующей информационной системе были реализованы в программном средстве Data Manipulation Core – платформе для разработки интегрирующих информационных систем, связывающих разрозненные информационные системы с разными структурами данных. DMC берет на себя большинство технических вопросов и позволяет прикладным программистам интегрирующих системы быстро развертывать новые эталонные хранилища консолидированных данных, подключать или отключать различные информационные системы и производить синхронизацию данных.

Кроме того, отдельное средство – DMC Client – является универсальным средством визуализации структур данных и самих данных, находящихся в хранилище данных, организуемом DMC.

В ходе проведения эксперимента для определения применимости DMC было показано, что система быстро обработала большое количество данных и определила связи между данными из разных систем и в разных форматах, тем самым дав основания о возможности рекомендации ее для решения прикладных задач.

СПИСОК ЛИТЕРАТУРЫ

1. **Торшин, Д. В.** Анализ основных типов архитектур взаимодействия нескольких информационных систем / Д. В. Торшин, Н. И. Юсупова // Интеллектуальные системы обработки информации и управления : сб. ст. Регион. зимн. шк.-сем. аспирантов и молодых ученых. Т. 2. Уфа : Технология, 2006. 312 с.

2. **Кудинов, А.** Интеграция данных и Хранилища / А. Кудинов [Электронный ресурс] (http://www.iso.ru/journal/articles/441.html).

3. **Торшин, Д. В.** Конвертация и перенос данных в задачах интеграции информационных ресурсов / Д. В. Торшин, Н. И. Юсупова // Актуальные проблемы в науке и технике : сб. ст. 2-ой регион. зимн. шк.-сем. аспирантов и молодых ученых. Т. 2. Уфа : Технология, 2007. С. 50–55.

4. **Торшин, Д. В.**Пример реализации принципов разработки программ конвертации данных на практике / Д. В. Торшин, Н. И. Юсупова // Интеллектуальные системы обработки информации и управления : сб. ст. 2-ой регион. зимн. шк.-сем. аспирантов и молодых ученых. Т. 1. Уфа : Технология, 2007. С. 71–76.

5. **Torshin, D.** Triple solution of data integrity and recovery problem / D. Torshin // Proc. of the 9th Intern. Workshop on Computer Science and Information Technologies CSIT’2007, Ufa, Russia, 2007.

6. **Extract**, transform, load. From Wikipedia, the free encyclopedia [Электронный ресурс]. (http://en.wikipedia.org/wiki/Extract%2C_transform%2C_load).

7. **Object-relational** mapping. From Wikipedia, the free encyclopedia [Электронный ресурс]. (http://en.wikipedia.org/wiki/Object-relational_mapping).

ОБ АВТОРАХ

Юсупова Нафиса Исламовна, проф., зав. каф. выч. мат. и киб., декан ФИРТ. Дипл. радиофизик (Воронежск. гос. ун-т, 1975). Д-р техн. наук по упр-ю в техн. сист. (УГАТУ, 1998). Иссл. в обл. критич. сит. упр-я, информатики.

Торшин Дмитрий Вячеславович, асп. той же каф. Руководитель группы, ЗАО «Синтерра». Дипл. инж. по мат. обеспечению и администрированию инф. систем. Готовит дис. в обл. интеграции комп. систем обработки данных.