УДК 681.518

Н. А. МАРЧЕНКО, канд. техн. наук, доцент НТУ «ХПИ»,
В. В. САМАРСКИЙ, студент НТУ «ХПИ»

РАЗРАБОТКА РАСПРЕДЕЛЕННОЙ СИСТЕМЫ ДЛЯ АВТОМАТИЗАЦИИ ПРОЦЕССОВ ПРЕДСТАВЛЕНИЯ ДАННЫХ

В статті розглядується концепція, узагальнена архітектура та основні етапи розробки розподіленої системи для автоматизації процесів подання даних. Розглянуті основні можливості та завдання даної системи.

В статье рассматривается концепция, обобщенная архитектура и основные этапы разработки распределенной системы для автоматизации процессов представления данных. Описаны основные возможности и задачи данной системы.

This article discusses the concept, architecture and the main stages of the development of distributed systems to automate the processes of data visualization, main opportunities and challenges of the system.

Введение. К настоящему времени во многих организациях в Украине и других странах накоплены огромные объемы бизнес-информации о клиентах, поставщиках, партнерах, результатах финансовой деятельности и проч. Без этой информации не возможна деятельность организаций. Системы Business Intelligence наряду с хранилищами данных и приложениями бизнес-аналитики

представляют собой инструментарий, который позволяет извлечь максимум информации из имеющихся первичных данных, выявить скрытые закономерности и тренды, построить прогностические модели, т.е. в конечном счете, превратить имеющиеся у компании данные в источник дополнительной прибыли [1,2]. В свою очередь, разработка Business Intelligence систем довольно ресурсоемкий, дорогой и долгий по времени процесс.

Постановка задачи. В данной статье поставлена задача описания методики упрощения и автоматизации процесса разработки Business Intelligence систем при минимальном вмешательстве пользователя.

Описание структуры и технологий реализации. Обобщенная схема работы системы, которая бы автоматизировала процессы представления данных, видна на рис 1. При такой схеме пользователь только подготавливает данные и выполняет конфигурирование системы, т.е. управляет требованиями к своему проекту (указывает необходимые ему измерение, метрики, калькуляции), а вся программная часть выполняется автоматически.

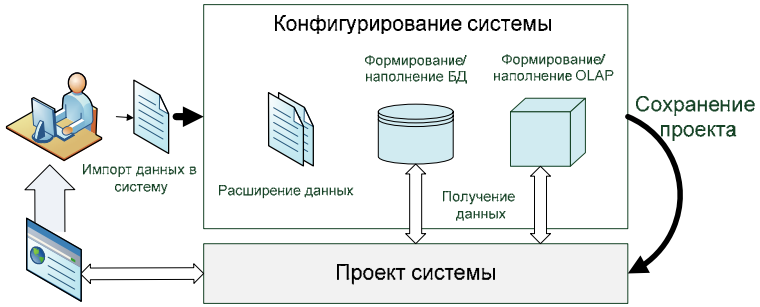

Рис. 1. Общая схема работы системы

Реализация системы представления данных в рамках поставленной задачи выполнена с помощью трехуровневой архитектуры (см. рис. 2).

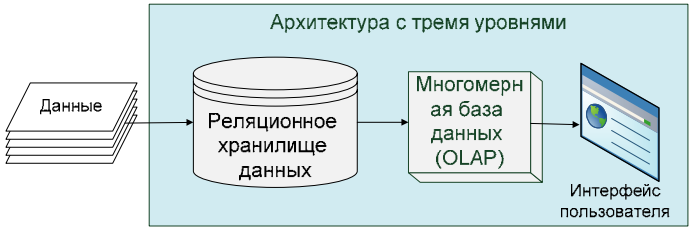

Рис. 2. Трехуровневая архитектура Business Intelligence проекта

Идеальными программными продуктами, на базе которых был реализован конечный продукт, являются Microsoft SQL Server и платформа .NET [1,3]. СУБД Microsoft SQL Server позволила реализовать обработку и хранение данных, а при помощи технологии .NET были разработаны модуль настройки и пользовательский интерфейс.

В рамках трехуровневой архитектуры реализации системы было выполнено:

- модуль импорта данных с поддержкой различных источников, таких как текстовые файлы, файлы Microsoft Office Excel и таблицы/представления внешних баз данных;
- реляционное хранилище с гибкой и расширяемой архитектурой, позволяющей избежать жесткой привязки к источнику и виду данных;
- модуль автоматической сборки многомерной базы данных, согласно требованиям пользователя и его действиям;
- универсальный пользовательский интерфейс, реализующий три уровня доступа к данным: графический (всевозможные графики), обобщенный (данные сгруппированы согласно имеющимся измерениям, пользователь имеет возможность анализировать зависимости, тенденции и т.д.), детальный уровень – самый низкий уровень, на котором можно увидеть данные, пришедшие на вход вместе с расширенными калькуляциями.

Проектирование хранилища данных. Наибольший интерес с точки зрения реализации вызывает хранилище данных (реляционное хранилище данных и многомерная база данных). Ниже представлен подход к архитектуре хранилищ данных, известный как хранилище с архитектурой шины или подход Ральфа Кимболла (см. рис. 3).

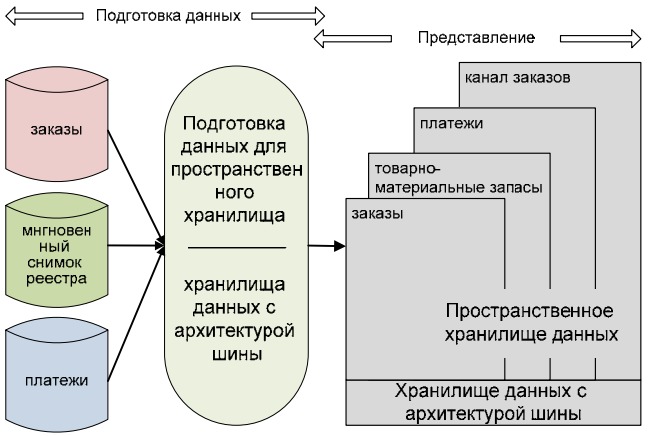

Рис. 3. Пространственное хранилище данных

В этой модели первичные данные преобразуются в информацию, пригодную для использования, на этапе подготовки данных. При этом обязательно принимаются во внимание требования к скорости обработки информации и качеству данных. Ряд операций совершается централизованно, например, поддержание и хранение общих справочных данных, другие действия могут быть распределенными.

Область представления пространственно структурирована, при этом она может быть централизованной или распределенной. Пространственная модель хранилища данных содержит ту же атомарную информацию, что и нормализованная модель (подход Билла Инмона), но информация структурирована по-другому, чтобы облегчить ее использование и выполнение запросов. Эта модель включает как атомарные данные, так и обобщающую информацию (агрегаты в связанных таблицах или многомерных кубах) в соответствии с требованиями производительности или пространственного распределения данных. Запросы в процессе выполнения обращаются к все более низкому уровню детализации без дополнительного перепрограммирования со стороны пользователей или разработчиков приложения.

Типичными чертами подхода Ральфа Кимболла являются [5]:

1. Использование пространственной модели организации данных с архитектурой «звезда» (star scheme).
2. Хранилище данных с архитектурой шины обладает следующими характеристиками:

- оно пространственное;
- оно включает витрины данных, посвященные только одной предметной области или имеющие только одну таблицу фактов (fact table);
- оно может содержать множество витрин данных в пределах одной базы данных.

3. Хранилище данных не является единым физическим репозиторием (в отличие от подхода Билла Инмона). Это «виртуальное» хранилище. Это коллекция витрин данных, каждая из которых имеет архитектуру типа «звезда».

Кроме того, логическую структуру реляционной базы данных можно представить как совокупность подмножеств объектов, обобщенных по их назначению или функциям, которые они выполняют (см. рис. 4).

Объекты, обеспечивающие работу модуля конфигурирования – набор процедур и таблиц, которые участвую исключительно в построении системы.

Объекты ETL процесса – набор процедур, таблиц, представлений, функций, которые представляют собой последовательный процесс загрузки данных в систему, наполнения справочников, расстановку ключей в таблицах фактов и т. д.

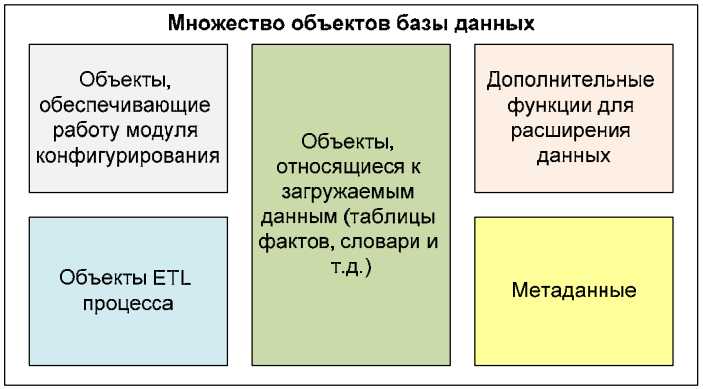

Рис. 4. Обобщенная логическая схема базы данных

К дополнительным функциям для расширения данных относятся специальные процедуры и функции, которые позволяют создавать вычисляемые поля (калькуляции) на основе имеющихся данных. Сюда можно отнести функцию нечеткого сравнения строк, основанную на методах Левенштейна и Хэмминга (при равной длине двух строк).

Расстоянием Левенштейна $d(u,v)$ между строками u и v называется наименьшее количество операций редактирования, необходимое, чтобы перевести u в v. Из соображений обратимости операций редактирования, имеем $d(v,u)=d(u,v)$ В общем виде расстояние Хэмминга d_H для объектов u и v размерности p задаётся функцией [4]:

$$d_H(X_i,X_j)=\sum_{s=1}^{p}\left|x_i^{(s)}-x_j^{(s)}\right|. \qquad (1)$$

Данная функция может быть полезна для устранения ошибок ввода человеком информации в базу данных или для сравнения адресов/терминов/названий, написание которых может в какой-то мере отличаться, но передавать единый смысл.

Выводы. Таким образом, разработанная система в состоянии дать ее пользователю возможность быстрого создания несложной Business Intelligence системы без привлечения сторонних специалистов, при помощи которой становится возможным представить большой массив данных в ином, более понятном и наглядном виде, увидеть тенденции и скрытые закономерности в них. Также данное решение позволит подсказать, нужно ли в дальнейшем разрабатывать полноценную подобную систему.

Список литературы: ***1.*** *Brian Larson*. Delivering Business Intelligence with Microsoft SQL Server 2005. McGraw-Hill/Osborne, 2006. – 792 с. ***2.*** *Троелсен Э.* Язык программирования C# 2005 и платформа.NET 2.0. 3-е издание. – Пер с англ. М. ООО “И.Д. Вильямс”, 2007. – 1168 с. ***3.*** *Robert Wrembel, Christian Koncilia*. Data Warehouses and OLAP: Concepts, Architectures and

Solutions. Idea Group Inc, 2006. – 332 с. ***4.*** *Richard W. Hamming.* Error-detecting and error-correcting codes, Bell System Technical Journal 29(2):147-160, 1950. ***5.*** *Ralph Kimball; Margy Ross.* The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling (Second Edition ed.). New York: Wiley, 2002. – 464 с. ***6.*** *W. H. Inmon.* Building the Data Warehouse (Fourth Edition). New York: Wiley, 2005. – 543 с.

Поступила в редколлегию 12.01.09