

# ИНТЕРАКТИВНАЯ АНАЛИТИЧЕСКАЯ ОБРАБОТКА ДАННЫХ В СОВРЕМЕННЫХ OLAP-СИСТЕМАХ

**И.Ю. Каширин,**

доктор технических наук, профессор кафедры «Вычислительная и прикладная математика» Рязанского государственного радиотехнического университета, Адрес: 390005, г. Рязань, ул. Гагарина, 59/1, кафедра «Вычислительная и прикладная математика», Каширин И.Ю. Тел: 4-912-214-667. E-mail: igor\_kashirin@mail.ru

**С.Ю. Семченков,**

ассистент кафедры «Теоретическая и прикладная механика» Рязанского государственного радиотехнического университета, E-mail: amfibius@mail.ru

*В статье рассмотрены аспекты обработки данных с использованием многомерного анализа. Приведены основные понятия многомерной модели, рассмотрена классификация OLAP систем. Освещены вопросы совместного использования OLAP и OLTP систем, проведен сравнительный анализ существующих систем.*

**Ключевые слова:** OLAP-системы. Многомерная модель обработки данных. Интерактивный анализ данных.

## Введение

В области технологий обработки данных существует два взаимодополняющих друг друга направления развития:

- ♦ системы, ориентированные на операционную обработку данных – системы обработки данных (СОД);
- ♦ системы, ориентированные на анализ данных – системы поддержки принятия решений (СППР).

В недавнем прошлом, когда говорилось о стремительном росте числа реализаций информационных систем, прежде всего, имелись в виду системы, предназначенные для оперативной обработки данных. На первых этапах автоматизации требуется структурировать процессы повседневной рутинной обработки данных. На это ориентированы традиционные СОД. Для построения таких систем исполь-

зуются реляционные системы управления базами данных (РСУБД), такие как Oracle [1], Microsoft SQL Server [2], MySQL [3], PostgreSQL [4] и другие.

Однако у такого подхода есть и недостатки, связанные с наличием нескольких источников данных. Основная проблема – необходимость согласования данных в разных источниках информации. При этом в лучшем случае проводится репликация баз данных (БД) как процесс копирования информации из одной базы в несколько других.

Необходимо учитывать следующие особенности репликации:

1) репликация – ресурсоемкий процесс, при котором большая нагрузка приходится на центральный процессор, оперативную память, сетевые каналы;

2) при большом количестве информации временные затраты на репликацию зачастую не оправдывают полученных результатов даже при достаточном количестве системных ресурсов.

С другой стороны, отсутствие согласованности информации приводит к разрозненности данных при их дальнейшей обработке. В любом случае, одни и те же данные многократно повторяются в различных СОД.

Таким образом, с ростом количества данных возникает противоречие: информация есть и её слишком много, в то же время она не структурирована, не согласована, разрознена, не всегда достоверна, её практически невозможно найти в СОД и получить. На разрешение этого противоречия нацелена технология OLAP.

OLAP (On-Line Analytical Processing – интерактивная аналитическая обработка данных) [5] – один из способов представления и анализа данных. При этом информация представляется в виде многомерного куба с возможностью произвольного манипулирования ею. Многомерные модели рассматривают данные либо как факты с соответствующими численными параметрами, либо как текстовые измерения, которые характеризуют эти факты.

Актуальность применения OLAP систем заключается в том, что они позволяют работать с данными в терминах предметной области без знания архитектуры хранения информации [6]. Кроме того, OLAP технология позволяет поддерживать различные типы бизнес-правил, использующиеся на предприятиях.

### Многомерная модель для интерактивной аналитической обработки данных

Концепция OLAP-систем основана на следующих понятиях [7] (рис. 1).

1) Многомерный куб ( $n$  – число измерений многомерного куба) – это многомерная структура, состоящая из множества ячеек и хранящая взаимосвязанные данные, описывающие предметную область или её составляющую. Ячейка является атомарной структурой куба и при отображении располагается внутри него. Многомерный куб может создаваться и храниться как на физическом уровне представления данных (многомерные OLAP системы), так и на концептуальном уровне (виртуально) с помощью схем «звезда» или «снежинка» (реляционные OLAP системы).

2) Показатель  $q_j$  ( $j = 1...p$ ,  $p$  – количество показателей) – это типизированная величина, которая является предметом анализа (например, показателями являются величина закупок, величина продаж, посещаемость сайта и т.д.). Один многомерный куб может содержать несколько показателей, причём

каждому конкретному значению показателя соответствует единственная ячейка многомерного куба.

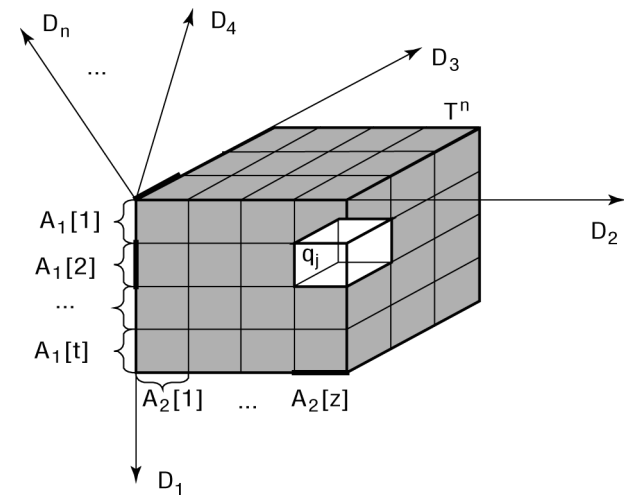


Рис. 1. Многомерная модель данных

3) Измерение  $D_i$  ( $i = 1...n$ ) – это множество объектов (называемых членами измерения или значениями измерения) одного или нескольких типов, организованных в виде иерархической структуры и обеспечивающих информационный контекст типизированного показателя. Например, измерение «товар» может состоять из следующих значений: «бытовая техника», «компьютеры», «мобильные телефоны» и т.д. Значения, связанные с измерением, характеризуют какое-либо классификационное свойство сущностей предметной области. Совокупность измерений образует множество измерений  $D = \{D_1, D_2, \dots, D_n\}$ . На рис. 1.1  $\{A_1[1], A_1[2], \dots, A_1[t]\}$  – множество значений измерения  $A_1$  ( $t$  – количество элементов измерения  $A_1$ ),  $\{A_2[1], \dots, A_2[z]\}$  – множество значений измерения  $A_2$  ( $z$  – количество элементов измерения  $A_1$ ). Измерение принято отображать в виде ребра многомерного куба, а члены измерения – в виде точек или участков, откладываемых на ребре.

Множество всех измерений куба образует систему координат представляемого пространства данных. Значения измерений играют роль индексов, находящихся в ячейках куба. Особенностью измерений является их иерархическая структура, которая используется для агрегации и детализации значений показателей. Наиболее часто используются следующие два вида иерархий [8]:

◆ *Сбалансированные* – иерархии, в которых число уровней определено её структурой неизменно, и все ветви имеют одинаковую глубину. Для формирования сбалансированной иерархии необходимо наличие связи «один-ко-многим» между объектами

менее детального уровня по отношению к объектам более детального уровня. Например, сбалансированной является трехуровневая иерархия «направления деятельности – категории товаров – производители товаров» (рис. 2).

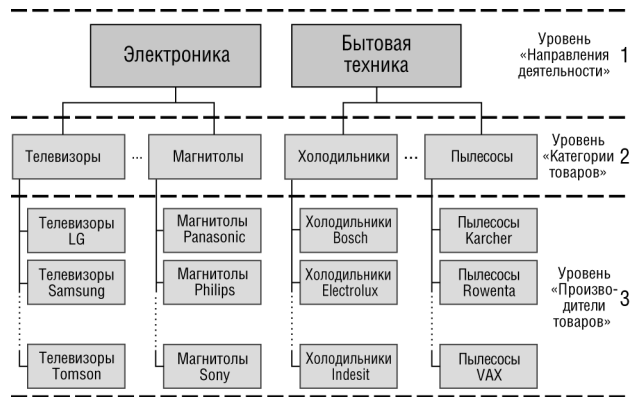


Рис. 2. Пример сбалансированной иерархии

◆ **Несбалансированные** – иерархии, в которых число уровней может быть изменено, некоторые уровни логически не одинаковы, и глубина ветвей иерархии может быть разной. Все объекты несбалансированной иерархии принадлежат одному типу. Например, несбалансированной является иерархия компаний-поставщиков (рис. 3). Все элементы этой иерархии имеют тип «поставщик».

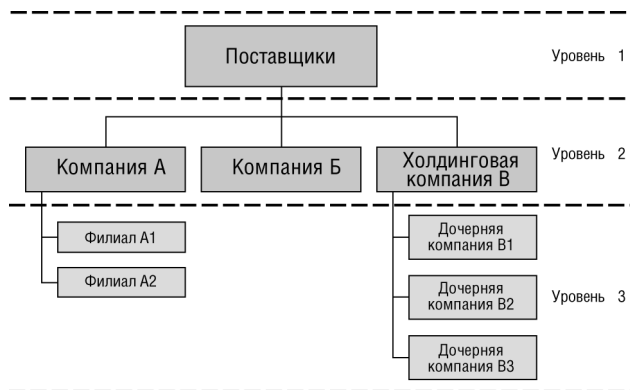


Рис. 3. Пример несбалансированной иерархии

4) Элемент измерения – уровень в иерархии значений координат измерения многомерного куба. В приведённом примере измерение содержит 3 элемента («направления деятельности», «категории товаров», «производители товаров»), каждый из них характеризуется номером уровня в иерархии.

5) Атрибут измерения. Атрибуты измерения описывают его элементы, но не образуют иерархии. Например, для каждого поставщика можно хранить адрес его электронной почты. Различные уровни

измерения могут образовывать различные атрибуты. Одни атрибуты могут относиться ко всем уровням измерений, другие – лишь к определенным.

В отличие от традиционных СОД, OLAP системы должны иметь возможность работы с тремя типами данных [9]: ретроспективными, оперативными и прогнозируемыми.

### Классификация OLAP систем

OLAP системы [10] можно классифицировать по следующим параметрам:

- 1) по способу организации данных в многомерных кубах;
- 2) по способу хранения на физическом уровне;
- 3) по месту нахождения OLAP машины, рассчитывающей многомерные кубы;
- 4) по степени готовности к применению для конечного пользователя.

В различных OLAP системах используются два основных варианта организации данных [11]: гиперкубическая и поликубическая модели. В гиперкубической модели все показатели должны определяться одним и тем же набором измерений. Поликубическая модель предполагает, что в БД определяется несколько гиперкубов с различной размерностью и с различными измерениями в качестве их граней.

Как исходные, так и агрегированные данные могут храниться либо в реляционных, либо в многомерных структурах. Применяются три способа хранения данных на физическом уровне [12].

1) MOLAP (Multidimensional OLAP – многомерный OLAP) – исходные и агрегированные данные хранятся в многомерных структурах. MOLAP напрямую поддерживает многомерные представления данных с помощью многомерного механизма хранения. Скорость вычисления агрегатных значений одинакова для любого из измерений. Однако в этом случае многомерная база данных оказывается избыточной, так как многомерные данные полностью содержат исходные реляционные данные. Достоинства и недостатки MOLAP архитектуры приведены в табл. 1.

2) ROLAP (Relational OLAP – реляционный OLAP) – исходные данные остаются в той же реляционной базе данных, где они изначально и находились. Агрегированные данные помещают в специально созданные для их хранения служебные таблицы в той же базе данных. Серверы ROLAP наследуют возможности масштабирования и работы с транзакциями реляционных систем, однако существенные

Таблица 1

**Достоинства и недостатки MOLAP систем**

Достоинства	Недостатки
Напрямую поддерживается многомерное представление данных.	Низкий коэффициент использования дискового пространства, особенно в случае разреженных данных.
Многомерные запросы путём непосредственного доступа к ячейкам гипер-куба (без SQL).	Отсутствуют единые стандарты на интерфейс, языки описания и манипулирования данными.

различия между запросами в стиле OLAP и SQL являются причиной низкой производительности. Достоинства и недостатки ROLAP архитектуры приведены в табл. 2.

Таблица 2

**Достоинства и недостатки ROLAP систем**

Достоинства	Недостатки
Развитые средства администрирования; хорошая масштабируемость.	Для сложных запросов SQL не является оптимальным.
Обеспечивают высокий уровень защиты данных и разграничения прав доступа.	Неодинаковое время выполнения запроса для различных измерений.

3) HОLAP (Hybrid OLAP – гибридный OLAP) – гибридная архитектура, которая объединяет технологии ROLAP и MOLAP. HОLAP могут работать в двух режимах: вертикальной декомпозиции и горизонтальной декомпозиции [13]. В первом случае HОLAP системы применяют ROLAP подход для разреженных областей многомерного пространства (исходные данные) и MOLAP – подход для плотных областей (агрегированные данные). При горизонтальной декомпозиции применяют MOLAP-подход для оперативных данных и ROLAP-подход – для исторических.

По месту нахождения OLAP машины выделяют клиентский OLAP (DOLAP – Desktop OLAP) и серверный OLAP. В случае DOLAP сервер отправляет клиенту исходные данные, клиент выполняет расчет многомерных кубов и выдает результаты пользователю. В случае серверного OLAP сервер выполняет расчет многомерных кубов и отправляет клиенту конечный результат, клиент выдает принятый результат пользователю.

По степени готовности к применению OLAP продукты делятся на две категории [14]:

1) OLAP компонента – это инструмент разработчика. С её помощью разрабатываются клиентские OLAP программы. MOLAP компоненты являются инструментами генерации запросов к серверу. Одна из наиболее доступных компонент – Decision Cube в составе Borland C++ Builder [15].

2) Инструментальные OLAP системы – это программные продукты, предназначенные для создания аналитических приложений. Различают две категории инструментальных систем: системы для программирования и системы для быстрой настройки. Представителем первой категории является аналитическая платформа ProClarity [16]. OLAP-системы для быстрой настройки – это средства, предоставляющие визуальный интерфейс для создания приложений без программирования. В такой технологии реализованы инструменты пакета BusinessObjects [17] и аналитической платформы «Контур» [18].

**Совместное использование OLAP и OLTP систем**

Многомерный подход возник практически одновременно и параллельно с реляционным, но только начиная с 1993 г. к нему был проявлен всеобщий интерес. В 1993 г. появилась статья [19] одного из основоположников реляционного подхода Кодда, в которой он сформулировал 12 требований к средствам реализации OLAP и произвел анализ недостатков реляционного подхода.

Позднее эти требования были переработаны в тест FASMI (Fast Analysis of Shared Multidimensional Information – Быстрый Анализ Разделяемой Многомерной Информации), который определяет требования к продуктам OLAP [20]. Это определение состоит из пяти требований:

1) Быстрый – означает, что система должна обеспечивать выдачу большинства ответов пользователям в пределах 5 секунд;

2) Анализ – означает, что система может справляться с любым логическим и статистическим анализом, характерным для данного приложения, и обеспечивает его сохранение в виде, доступном для конечного пользователя;

3) Разделяемой – означает, что система осуществляет все требования защиты конфиденциальности и обеспечивает блокировку множественных модификаций на соответствующем уровне своевременным, безопасным способом;

4) Многомерной – ключевое требование. Система должна обеспечить многомерное концептуальное представление данных, включая полную поддержку для иерархий. Под концептуальным представлением

понимается описание данных в форме системы понятий и отношений предметной области. При этом на минимальное количество измерений и используемую технологию БД не накладывается никаких ограничений;

5) Информации – означает, что приложение должно иметь возможность обращаться к любой нужной информации, независимо от ее формата, объема и места хранения.

OLAP системы, как правило, не являются первичными источниками информации, они получают данные из внешних источников. Наиболее часто таким внешним источником является OLTP (On-Line Transactional Processing – оперативная обработка транзакций) система – система операционной обработки данных (СОД), построенная на базе реляционной СУБД. OLAP системы и OLTP системы существенно различаются. Сравнительная характеристика этих технологий по основным параметрам приведена в *табл. 3*.

Таблица 3

Сравнение OLAP и OLTP систем

Характеристика	OLTP	OLAP
Взаимодействие с пользователем	На уровне транзакции	На уровне всей БД
Время отклика	Не регламентируется	Секунды
Данные, используемые при обращении пользователя к системе	Отдельные записи	Группы записей
Характер данных (детализация)	В основном детализированные (самый низкий уровень детализации)	В основном агрегированные
Изменчивость данных	Высокая (данные обновляются с каждой транзакцией), данные обновляются маленькими порциями	Низкая (во время запроса данные обновляются редко), данные обновляются большими порциями
Источники данных	В основном внутренние (по отношению к системе)	В основном внешние (по отношению к системе)
Возраст данных	Текущие (за период от нескольких месяцев до одного года)	Исторические, текущие и прогнозируемые
Характер запросов	В основном предсказуемые	Непредсказуемые

Следует отметить, что в OLAP системах время является одним из основных критериев, по которому данные упорядочиваются в процессе обработки и представления пользователю. Вследствие этого в OLAP системе должны быть предусмотрены необходимые возможности для работы с временным измерением, такие как поддержка нестандартных календарных циклов.

Не менее важно и то, что многие критически необходимые для OLTP-систем функциональные возможности, реализуемые в РСУБД, являются избыточными для OLAP систем. Например, в OLAP системах данные обычно загружаются достаточно большими порциями из различных внешних источников (РСУБД, электронных таблиц и т.д.) [21].

Непосредственная реализация аналитической системы на основе OLTP-базы сложна. Это объясняется несколькими причинами:

1) OLAP запросы к базам данных чаще всего бывают сложными и требуют много времени. Прямой доступ к OLTP базе существенно снижает общую производительность оперативной системы, так как аналитические запросы конкурируют с транзакциями;

2) разнообразные OLTP системы неоднородны по типу используемых синтаксических соглашений и концептуальных допущений, поэтому их интеграция затруднена;

3) данные в OLTP системах часто «зашумлены», неполны и несогласованны;

4) модели данных разных OLTP систем не взаимосвязаны между собой;

5) информация за прошлые периоды теряется при обновлении OLTP-базы;

6) в OLTP-базе не хранятся данные в агрегированном, ненормализованном виде, что необходимо для оперативной аналитической обработки.

Для реализации аналитической системы необходимо наличие отдельной многомерной СУБД, интегрирующей данные из внешних источников и обрабатывающей аналитические запросы пользователей системы. С учётом этого, архитектура системы отчетности имеет следующий вид (*рис. 4*).

OLAP технология не служит заменой существующей программной инфраструктуры, она лишь дополняет ее. Пользователь по-прежнему имеет возможность обращаться к существующим РСУБД, электронным таблицам и плоским файлам. Однако

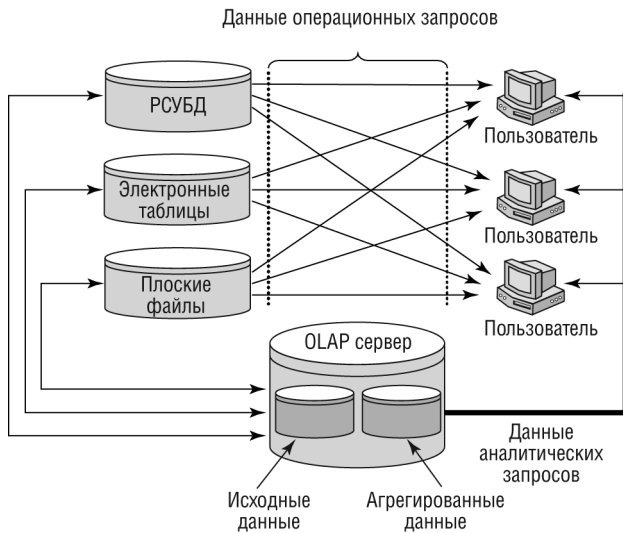


Рис. 4. Архитектура отчётной системы с применением OLAP-технологии

для выполнения аналитических запросов используется обращение к OLAP серверу.

### Современные системы аналитической обработки данных

Сейчас существует множество OLAP систем от различных производителей.

Microsoft Analysis Services [22] (является составной частью СУБД Microsoft SQL Server) разработан в 1996 г. Он использует архитектуру хранения данных с применением ROLAP/MOLAP/HOLAP. В Microsoft SQL Server присутствует единая модель OLAP и реляционной отчетности. Модели анализа данных можно строить по произвольным срезам куба. При выборе одного измерения другие, связанные с ним, можно рассматривать как вложенные таблицы, причём к каждому допускается применение независимых срезов. Новая версия продукта позволяет применять OLAP-модель к глобальному кубу, делить его на группы, создавать кластеры в режиме реального времени.

OLAP Option to Oracle Database [23] является составной частью Oracle Database Enterprise Edition. OLAP Option полностью интегрирован на уровне ядра СУБД в Oracle Database Enterprise Edition. Поддерживаются как MOLAP, так и ROLAP архитектуры. В качестве языка описания и манипулирования данными используется язык OLAP DML, который может быть встроен в синтаксис языка SQL (для обработки данных в многомерной БД может быть использован язык Java). Для администрирования многомерных СУБД используется приложе-

ние Analytic Workspace Manager. Для интеграции с Microsoft Excel служит OracleBI Spreadsheet Add-In. Oracle OLAP Option поддерживает параллельное выполнение запросов и разделение многомерного куба на разделы, позволяет контролировать распределение системных ресурсов и статус долговременных операций, содержит инструменты для создания резервных копий многомерной БД и восстановления.

Palo – MOLAP сервер, в котором значения записываются непосредственно в оперативную память. Palo рассчитан прежде всего на взаимодействие с Excel, большинство задач, таких как создание измерений и кубов, выполняются с помощью плагина для Excel. Однако Palo предоставляет SDK (Software Development Kit – комплект программ для разработчика ПО) для языков C, PHP, .NET и Java). ПО Palo Eclipse Client (на основе Eclipse Framework) позволяет проектировать и просматривать измерения и кубы в платформонезависимой среде Eclipse Framework.

Mondrian – ROLAP сервер с открытым исходным кодом, написанный на Java и поддерживающий язык запросов MDX (MultiDimensional eXpressions – многомерные выражения). Кубы и измерения определяются путем создания схемы XML [24]. Архитектура Mondrian состоит из четырех уровней: уровень представления, уровень измерений, главный уровень и физический уровень. Уровень представления определяет, как будет отображаться информация, и как пользователь будет взаимодействовать с системой для определения запросов. Уровень измерений анализирует, проверяет и выполняет MDX запросы. Главный уровень поддерживает кэш с агрегированными данными. Уровень хранения – реляционная СУБД.

Сравнительные характеристики OLAP-систем приведены в табл. 4.

### Заключение

Анализ современного состояния OLAP-технологий позволяет говорить о серьезных перспективах их развития. Многомерная обработка информации становится необходимым компонентом любого хранилища данных. В то же время, широкое разнообразие подходов к реализации таких систем и отсутствие должной унификации форматов хранения и алгоритмов обработки данных заставляет искать новые подходы к описанию и разработке OLAP-систем. Эти подходы должны использовать в своей основе математический формализм, доста-

Таблица 4

Сравнительные характеристики OLAP-систем

Характеристики	Microsoft Analysis Services	OLAP Option to Oracle Database	Palo	Mondrian
Сравниваемые версии	2008 Standart Edition	11g	2.5	3.0.4
Поддерживаемые платформы	Windows XP, Windows Server 2003	Windows 2000, Linux	Windows 2000, Linux	Java SDK 1.4.2
OLAP API	XML for Analysis (XMLA)	Java API (Oracle JDBC)	NET API, PHP API, Java API, C API	XMLA, Olap4j (Java API)
Язык запросов	MDX	SQL	нет	MDX
Язык манипулирования данными	T-SQL	OLAP DML	нет	нет
Создание отчетов на основе данных из OLAP-системы	Microsoft Reporting Services	Oracle BI Publisher	Excel	Pentaho Report Designer
Интеграция с Microsoft Excel	Microsoft Excel Pivot Tables	OracleBI Spreadsheet Addin	Excel Addin	Pentaho Spreadsheet Services
Инструменты резервирования данных	SQL Server Management Studio	Recovery Manager, Oracle Data Pump	нет	нет

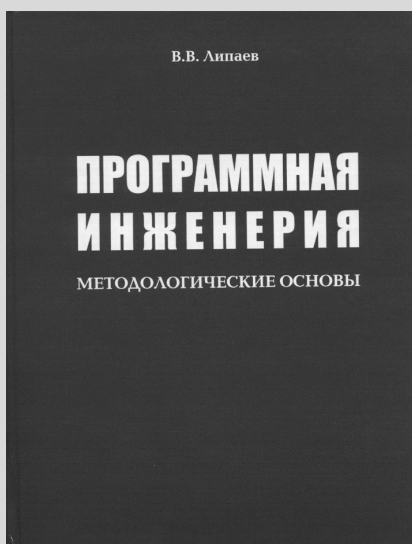
точный для описания и анализа большинства существующих систем. Новые исследования должны привести не только к выделению базовых принципов структурного OLAP-проектирования, но и дать возможность конструктивной оптимизации, а также автоматизированного проектирования систем многомерной обработки данных, обеспечивающего надежную и оперативную работу информационных хранилищ. ■

Литература

- Oracle Database [Электронный ресурс]—URL: <http://www.oracle.com/database/index.html> (дата обращения: 10.11.2008).
- SQL Server 2008 Overview, data platform, store data [Электронный ресурс] – URL: <http://www.microsoft.com/sql/default.mspx> (дата обращения: 10.11.2008).
- The world’s most popular open source database [Электронный ресурс] – URL: <http://www.mysql.com/> (дата обращения: 10.11.2008).
- The world’s most advanced open source database [Электронный ресурс] – URL: <http://www.postgresql.org> (дата обращения: 10.11.2008).
- Shoshani A. OLAP and statistical databases: similarities and differences // 16th ACM SIGACT SIGMOD SYGART Symp. On Principles of Database Systems, 1997. P. 185–196.
- Keith Laker «OLAP Workshop 1: Basic OLAP Concepts» [Электронный ресурс] – URL: <http://oracleolap.blogspot.com/2007/12/olap-workshop-1-basic-olap-concepts.html> (дата обращения: 10.11.2008).
- Барсегян А.А., Куприянов М.С., Степаненко В.В., Холлод И.И. Технологии анализа данных: Data Mining, Visual Mining, Text Mining, OLAP. – СПб.: БХВ-Петербург, 2007. – 384 с.
- Malinowski E., Zimanyi E. Hierarchies in a Multidimensional Model: From Conceptual modeling to Logical Representation// Data&Knowledge Engineering, 2006. Vol. 59. № 2. P. 348 – 377.
- Jurgens M. Index Structures for Data Warehouses. Dissertation Fachbereich fur Mathematik und Informatik an der Freien Universitat Berlin eingereicht am 22.12.1999.
- OLAP product reviews [Электронный ресурс]—URL: <http://www.olapreport.com/ProductsIndex.htm> (дата обращения: 10.11.2008).
- Nigel Pendse. Multidimensional data structures [Электронный ресурс] – URL: <http://www.olapreport.com/MDStructures.htm> (дата обращения: 10.11.2008).
- OLAP, MOLAP and ROLAP: What’s next? [Электронный ресурс] – URL: <http://www.itweb.co.za/office/bytes/0407060916t.htm> (дата обращения: 10.11.2008).
- O’Brien J. Next-Generation OLAP. The Future of Dimensional Analysis // The TDWI BI Executive Summit. San Diego, Aug. 18–20, 2008.

- Чаусов В., Амириди Ю. Классификация аналитических систем. Три года спустя // Журнал «Банки и технологии» – 2002. – №6.
- Послед Б.С. Borland C++ Builder 6. Разработка приложений баз данных. – СПб.: ДиасофтЮП, 2003. – 320 с.
- Microsoft Office PerformancePoint Server [Электронный ресурс] – URL: <http://www.microsoft.com/business/performancepoint/productinfo/previousversions.aspx>.

17. Business Intelligence from Business Objects [Электронный ресурс] – URL: <http://www.businessobjects.com/> (дата обращения: 10.11.2008).
18. Intersoft Lab [Электронный ресурс] – URL: <http://www.iso.ru/> (дата обращения: 10.11.2008).
19. Codd E.F. Providing OLAP (on-line analytical processing) to user-analysts: An IT mandate. E.F. Codd and Associates, 1993.
20. Nigel Pendse. What is OLAP? [Электронный ресурс] – URL: <http://www.olapreport.com/fasmi.htm> (дата обращения: 10.11.2008).
21. Thilo Maier. A Formal Model of the ETL Process for OLAP-Based Web Usage Analysis // Proceedings of the sixth WEBKDD workshop: Webmining and Web Usage Analysis (WEBKDD'04), in conjunction with the 10th ACM SIGKDD conference (KDD'04). Seattle, Washington, USA, August 22, 2004.
22. SQL Server 2005 Analysis Services [Электронный ресурс] – URL: <http://www.microsoft.com/sql/technologies/analysis/default.aspx> (дата обращения: 10.11.2008).
23. Oracle OLAP [Электронный ресурс] – URL: <http://www.oracle.com/technology/products/bi/olap/olap.html> (дата обращения: 10.11.2008)
24. Julian Hyde. Creating interactive OLAP applications with MySQL Enterprise and Mondrian [Электронный ресурс] – URL: <http://www.scribd.com/doc/2606603/Creating-Interactive-OLAP-Applications-with-MySQL-Enterprise-and-Mondrian-Presentation> (дата обращения: 10.11.2008).



*В рамках Инновационной образовательной программы вышла книга*  
**В.В.Липаева**  
**«Программная инженерия.**  
**Методологические основы»**  
*(Допущено УМО по образованию в области менеджмента в качестве учебника для студентов ВУЗов, обучающихся по направлению «Бизнес-информатика»).*

Владимир Васильевич Липаев – профессор кафедры управления разработкой программного обеспечения ГУ-ВШЭ, главный научный сотрудник Института системного программирования РАН. Около 40 лет занимается исследованиями и разработкой программного обеспечения, методов и инструментальных средств для создания управляющих программ реального времени высокого качества. Под его руководством разработаны крупные инструментальные системы программной инженерии, широко использовавшихся в оборонной промышленности и частично эксплуатируемые до настоящего времени.

Учебник содержит курс лекций, отражающий методологические основы современной программной инженерии, обеспечивающей жизненный цикл (ЖЦ) сложных программных средств (ПС).

Учебник целесообразно использовать при обучении студентов старших курсов, аспирантов и менеджеров проектов при создании сложных комплексов программ на всём их ЖЦ (64 часа лекций и 32 часа семинарских занятий). Курс ориентирован также на заказчиков, менеджеров крупных проектов, аналитиков и ведущих специалистов, обеспечивающих этапы ЖЦ сложных ПС и систем, к которым предъявляются высокие требования по качеству функционирования и ограничены доступные ресурсы разработки.