Технология OLAP

Информационные системы масштаба предприятия, как правило, содержат приложения, предназначенные для комплексного многомерного анализа данных, их динамики, тенденций и т.п. Такой анализ в конечном итоге призван содействовать принятию решений. Нередко эти системы так и называются — системы поддержки принятия решений.

Принять любое управленческое решение невозможно не обладая необходимой для этого информацией, обычно количественной. Для этого необходимо создание хранилищ данных (Data warehouses), то есть процесс сбора, отсеивания и предварительной обработки данных с целью предоставления результирующей информации пользователям для статистического анализа (а нередко и создания аналитических отчетов).

Ральф Кимбалл (Ralph Kimball), один из авторов концепции хранилищ данных, описывал хранилище данных как "место, где люди могут получить доступ к своим данным". Он же сформулировал и основные требования к хранилищам данных:

- поддержка высокой скорости получения данных из хранилища;
- поддержка внутренней непротиворечивости данных;
- возможность получения и сравнения так называемых срезов данных (slice and dice);
- наличие удобных утилит просмотра данных в хранилище;
- полнота и достоверность хранимых данных;
- поддержка качественного процесса пополнения данных.

Типичное хранилище данных, как правило, отличается от обычной реляционной базы данных. Во-первых, обычные базы данных предназначены для того, чтобы помочь пользователям выполнять повседневную работу, тогда как хранилища данных предназначены для принятия решений. Например, продажа товара и выписка счета производятся с использованием базы данных, предназначенной для обработки транзакций, а анализ динамики продаж за несколько лет, позволяющий спланировать работу с поставщиками, — с помощью хранилища данных.

Во-вторых, обычные базы данных подвержены постоянным изменениям в процессе работы пользователей, а хранилище данных относительно стабильно: данные в нем обычно обновляются согласно расписанию (например, еженедельно, ежедневно или ежечасно — в зависимости от потребностей). В идеале процесс пополнения

представляет собой просто добавление новых данных за определенный период времени без изменения прежней информации, уже находящейся в хранилище.

И в-третьих, обычные базы данных чаще всего являются источником данных, попадающих в хранилище. Кроме того, хранилище может пополняться за счет внешних источников, например статистических отчетов.

Системы поддержки принятия решений обычно обладают средствами предоставления пользователю агрегатных данных для различных выборок из исходного набора в удобном для восприятия и анализа виде. Как правило, такие агрегатные функции образуют многомерный (и, следовательно, нереляционный) набор данных (нередко называемый гиперкубом или метакубом), оси которого содержат параметры, а ячейки — зависящие от них агрегатные данные - причем храниться такие данные могут и в реляционных таблицах, но в данном случае говорится о логической организации данных, а не о физической реализации их хранения). Вдоль каждой оси данные могут быть организованы в виде иерархии, представляющей различные уровни их детализации. Благодаря такой модели данных пользователи могут формулировать сложные запросы, генерировать отчеты, получать подмножества данных.

Технология комплексного многомерного анализа данных получила название OLAP (On-Line Analytical Processing). OLAP — это ключевой компонент организации хранилищ данных. Концепция OLAP была описана в 1993 году Эдгаром Коддом, известным исследователем баз данных и автором реляционной модели данных. В 1995 году на основе требований, изложенных Коддом, был сформулирован так называемый тест FASMI (Fast Analysis of Shared Multidimensional Information — быстрый анализ разделяемой многомерной информации), включающий следующие требования к приложениям для многомерного анализа:

- предоставление пользователю результатов анализа за приемлемое время (обычно не более 5 с), пусть даже ценой менее детального анализа;
- возможность осуществления любого логического и статистического анализа, характерного для данного приложения, и его сохранения в доступном для конечного пользователя виде;

- многопользовательский доступ к данным с поддержкой соответствующих механизмов блокировок и средств авторизованного доступа;
- многомерное концептуальное представление данных, включая полную поддержку для иерархий и множественных иерархий (это — ключевое требование OLAP);
- возможность обращаться к любой нужной информации независимо от ее объема и места хранения.

Следует отметить, что OLAP-функциональность может быть реализована различными способами, начиная с простейших средств анализа данных в офисных приложениях и заканчивая распределенными аналитическими системами, основанными на серверных продуктах. Но прежде чем говорить о различных реализациях этой функциональности, давайте рассмотрим, что же представляют собой кубы OLAP с логической точки зрения.

OLAP-система состоит из множества компонент. На самом высоком уровне представления система включает в себя источник данных, OLAP-сервер и клиента. Источник данных представляет собой источник, из которого берутся данные для анализа. Данные из источника переносятся или копируются на OLAP-сервер, где они систематизируются и подготавливаются для более быстрого впоследствии формирования ответов на запросы. Клиент - это пользовательский интерфейс к OLAP-серверу.

Источником в OLAP-системах является сервер, поставляющий данные для анализа. В зависимости от области использования OLAP-продукта источником может служить Хранилище данных, наследуемая база данных, содержащая общие данные, набор таблиц, объединяющих финансовые данные или любая комбинация перечисленного. Способность OLAP-продукта работать с данными из различных источников очень важна. Требование единого формата или единой базы, в которых бы хранились все исходные данные, не подходит администраторам баз данных. Кроме того, такой подход уменьшает гибкость и мощность OLAP-продукта. Администраторы и пользователи полагают, что OLAP-продукты, обеспечивающие извлечение данных не только из различных, но и из множества источников, оказываются более гибкими и полезными, чем те, что имеют более жесткие требования.

Прикладной частью OLAP-системы является OLAP-сервер. Эта составляющая выполняет всю работу (в зависимости от модели системы), и хранит в себе всю информацию, к которой

обеспечивается активный доступ. Архитектурой сервера управляют различные концепции. В частности, основной функциональной характеристикой OLAP-продукта является использование для хранения данных многомерной (ММБД, MDDB) либо реляционной (РДБ, RDB) базы данных..

Клиент - это как раз то, что используется для представления и манипуляций с данными в базе данных. Клиент может быть и достаточно несложным - в виде таблицы, включающей в себя такие возможности OLAP, как, например, вращение данных (пивотинг) и углубление в данные (дриллинг), и представлять собой специализированное, но такое же простое средство просмотра отчетов или быть таким же мощным инструментом, как созданное на заказ приложение, спроектированное для сложных манипуляций с данными. Интернет является новой формой клиента. Кроме того, он несет на себе печать новых технологий; множество интернет-решений существенно отличаются по своим возможностям в целом и в качестве OLAP-решения - в частности. В данном разделе обсуждаются различные функциональные свойства каждого типа клиентов.

Новым членом семейства OLAP-клиентов является Интернет. Существует масса преимуществ в формировании OLAP-отчетов через Интернет. Наиболее существенным представляется отсутствие необходимости в специализированном программном обеспечении для доступа к информации.

Приложения - это тип клиента, использующий базы данных OLAP. Они идентичны инструментам запросов и генераторам отчетов, описанным выше, но, кроме того, они вносят в продукт более широкие функциональные возможности. Приложение, как правило, обладает большей мощностью, чем инструмент запроса.

Технологии хранилищ данных

Во всем мире организации накапливают или уже накопили в процессе своей деятельности большие объемы данных. Эти коллекции данных хранят в себе большие потенциальные возможности по извлечению новой, аналитической информации, на основе которой можно и необходимо строить стратегию фирмы, выявлять тенденции развития рынка, находить новые решения, обусловливающие успешное развитие в условиях конкурентной борьбы. Для некоторых фирм такой анализ является неотъемлемой

частью их повседневной деятельности, но большинство, очевидно, только начинает приступать к нему всерьез.

Попытки строить системы принятия решений, которые обращались бы непосредственно к базам данных **систем оперативной обработки транзакций** (OLTP-систем), оказываются в большинстве случаев неудачными. Во-первых, аналитические запросы «конкурируют» с оперативными транзакциями, блокируя данные и вызывая нехватку ресурсов. Во-вторых, структура оперативных данных предназначена для эффективной поддержки коротких и частых транзакций и в силу этого слишком сложна для понимания конечными пользователями и, кроме того, не обеспечивает необходимой скорости выполнения аналитических запросов. В-третьих, в организации, как правило, функционирует несколько оперативных систем; каждая со своей базой данных. В этих базах используются различные структуры данных, единицы измерения, способы кодирования и т.д. Для конечного пользователя (аналитика) задача построения какого-либо сводного запроса по нескольким подобным базам данных практически неразрешима.

Для того чтобы обеспечить возможность анализа накопленных данных, организации стали создавать **хранилища данных**, которые представляют собой интегрированные коллекции данных, которые собраны из различных систем оперативного доступа к данным. Хранилища данных становятся основой для построения систем принятия решений. Несмотря на различия в подходах и реализациях, всем хранилищам данных свойственны следующие общие черты:

- **Предметная ориентированность.** Информация в хранилище данных организована в соответствии с основными аспектами деятельности предприятия (заказчики, продажи, склад и т.п.); это отличает хранилище данных от оперативной БД, где данные организованы в соответствии с процессами (выписка счетов, отгрузка товара и т.п.). Предметная организация данных в хранилище способствует как значительному упрощению анализа, так и повышению скорости выполнения аналитических запросов. Выражается она, в частности, в использовании иных, чем в оперативных системах, схемах организации данных. В случае хранения данных в реляционной СУБД применяется схема «звезды» (star) или «снежинки» (snowflake). Кроме того, данные могут храниться в специальной многомерной СУБД в n-мерных кубах.

- **Интегрированность.** Исходные данные извлекаются из оперативных БД, проверяются, очищаются, приводятся к единому виду, в нужной степени агрегируются (то есть вычисляются суммарные показатели) и загружаются в хранилище. Такие интегрированные данные намного проще анализировать.
- **Привязка ко времени.** Данные в хранилище всегда напрямую связаны с определенным периодом времени. Данные, выбранные их оперативных БД, накапливаются в хранилище в виде «исторических слоев», каждый из которых относится к конкретному периоду времени. Это позволяет анализировать тенденции в развитии бизнеса.
- **Неизменяемость.** Попав в определенный «исторический слой» хранилища, данные уже никогда не будут изменены. Это также отличает хранилище от оперативной БД, в которой данные все время меняются, «дышат», и один и тот же запрос, выполненный дважды с интервалом в 10 минут, может дать разные результаты. Стабильность данных также облегчает их анализ.

Вышеприведенные особенности были впервые сформулированы в 1992 году «отцом-основателем» хранилищ данных Биллом Инмоном (Bill Inmon) в его книге «Building the Data Warehouse».

5.1 Архитектура и компоненты хранилища данных

Непрерывный процесс

Англоязычный термин «Data Warehousing», который сложно перевести лаконично на русский язык, означает «создание, поддержку, управление и использование хранилища данных» и хорошо подтверждает тот факт, что речь идет о процессе. Цель этого процесса — непрерывная поставка необходимой информации нужным сотрудникам организации. Этот процесс подразумевает постоянное развитие, совершенствование, решение все новых задач и практически никогда не кончается, поэтому его нельзя уместить в более или менее четкие временные рамки, как это можно сделать для разработки традиционных систем оперативного доступа к данным.

Хранилища и киоски данных

Хранилища данных могут быть разбиты на два типа: корпоративные хранилища данных (enterprise data warehouses) и киоски данных (data marts).

Корпоративные хранилища данных содержат информацию, относящуюся ко всей корпорации и собранную из множества оперативных источников для консолидированного анализа. Обычно

такие хранилища охватывают целый ряд аспектов деятельности корпорации и используются для принятия как тактических, так и стратегических решений. Корпоративное хранилище содержит детальную и обобщающую информацию; его объем может достигать от 50 Гбайт до одного или нескольких терабайт. Стоимость создания и поддержки корпоративных хранилищ может быть очень высокой. Обычно их созданием занимаются централизованные отделы информационных технологий, причем создаются они сверху вниз, то есть сначала проектируется общая схема, и только затем начинается заполнение данными. Такой процесс может занимать несколько лет.

Киоски данных содержат подмножество корпоративных данных и строятся для отделов или подразделений внутри организации. Киоски данных часто строятся силами самого отдела и охватывают конкретный аспект, интересующий сотрудников данного отдела. Киоск данных может получать данные из корпоративного хранилища (зависимый киоск) или, что более распространено, данные могут поступать непосредственно из оперативных источников (независимый киоск).

Киоски и хранилища данных строятся по сходным принципам и используют практически одни и те же технологии.

Основные компоненты

Основными компонентами хранилища данных являются следующие:

- оперативные источники данных;
- средства проектирования/разработки;
- средства переноса и трансформации данных;
- СУБД;
- средства доступа и анализа данных;
- средства администрирования.

Среда Microsoft Data Warehousing Framework

Процессы создания, поддержки и использования хранилищ данных традиционно требовали значительных затрат, что в первую очередь было вызвано высокой стоимостью доступных на рынке специализированных инструментов. Эти инструменты практически не интегрировались между собой, так как были основаны не на открытых и стандартных, а на частных и закрытых протоколах, интерфейсах и т.д. Сложность и дороговизна делали практически невозможным построение хранилищ данных в небольших и средних фирмах, в то время как потребность в анализе данных испытывает любая фирма, независимо от масштаба.

Корпорация Microsoft давно осознала важность направления, связанного с хранилищами данных, и необходимость принятия мер по созданию инструментальной и технологической среды, которая позволила бы минимизировать затраты на создание хранилищ данных и сделала бы этот процесс доступным для массового пользователя. Это привело к созданию **Microsoft Data Warehousing Framework** (рис. 4) — спецификации среды создания и использования хранилищ данных. Данная спецификация определяет развитие не только новой линии продуктов Microsoft (например, Microsoft SQL Server 7.0), но и технологий, обеспечивающих интеграцию продуктов различных производителей. Открытость среды Microsoft Data Warehousing Framework обеспечила ее поддержку многими производителями ПО, что, в свою очередь, дает возможность конечным пользователям выбирать наиболее понравившиеся им инструменты для построения своих решений.

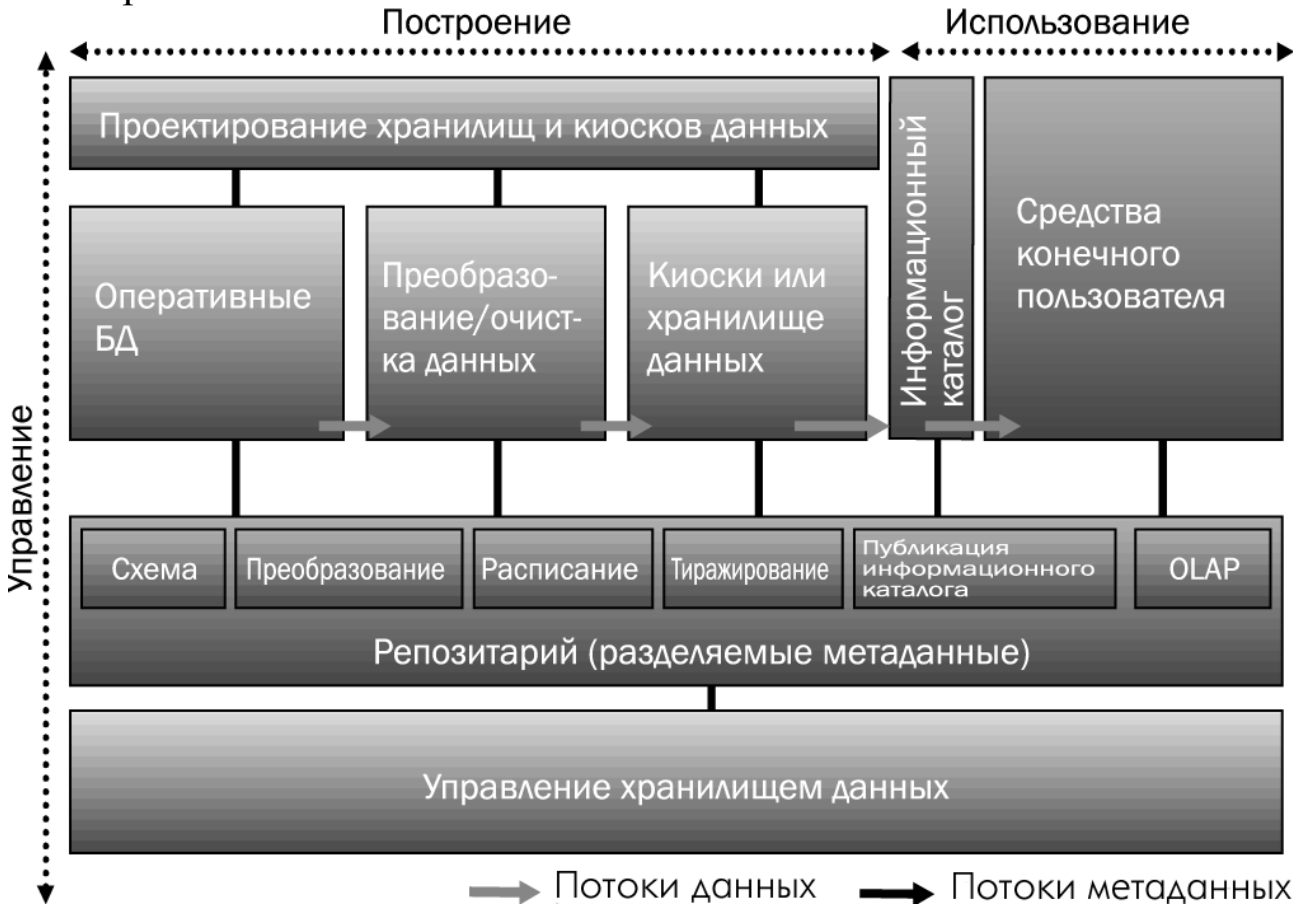

Рис. 4. Microsoft Data Warehousing Framework

Цель Microsoft Data Warehousing Framework — упростить разработку, внедрение и администрирование решений на основе хранилищ данных. Эта спецификация призвана обеспечить:

- открытую архитектуру, которая легко интегрируется и расширяется третьими фирмами;
- экспорт и импорт гетерогенных данных наряду с их проверкой, очисткой и возможным ведением истории накопления;
- доступ к разделяемым метаданным со стороны процессов разработки хранилища, извлечения и трансформации данных, управления сервером и анализа данных конечными пользователями;
- встроенные службы планирования задач, управления дисковой памятью, мониторинга производительности, оповещения и реакции на события.

Хранилище данных

2.1. Концепция хранилища данных

Стремление объединить в одной архитектуре системы поддержки принятия решений (СППР) возможностей OLTP-систем и систем анализа, требования к которым во многом противоречивы, привело к появлению концепции хранилищ дан ных (ХД). Первые статьи, посвященные ХД, появились в 1988 г., их авторами были Б. Девлин и П. Мэрфи. В 1992 г. У. Инмон подробно описал данную концепцию в своей монографии "По строение хранилищ данных".

В основе концепции ХД лежит идея разделения данных, используемых для оперативной обработки и для решения задач анализа. Это позволяет приме нять структуры данных, которые удовлетворяют требованиям их хранения с учетом использования в OLTP-системах и системах анализа. Такое разделе ние позволяет оптимизировать как структуры данных оперативного хранения (оперативные БД, файлы, электронные таблицы и т. п.) для выполнения опе раций ввода, модификации, удаления и поиска, так и структуры данных, ис пользуемые для анализа (для выполнения аналитических запросов). В СППР эти два типа данных называются соответственно оперативными источника ми данных (ОИД) и хранилищем данных.

В своей работе Инмон дал следующее определение ХД: *Хранилище данных — предметно-ориентированный, интегрированный, неизменчи вый, поддерживающий хронологию набор данных, организованный для целей под держки принятия решений.*

Рассмотрим свойства ХД более подробно.

П Предметная ориентация. Это фундаментальное отличие ХД от ОИД. Разные ОИД могут содержать данные, описывающие одну и

ту же пред метную область с разных точек зрения (например, с точки зрения бухгал терского учета, складского учета, планового отдела и т. п.). Решение, при нятое на основе только одной точки зрения, может быть неэффективным или даже неверным. ХД позволяют интегрировать информацию, отра жающую разные точки зрения на одну предметную область.

Предметная ориентация позволяет также хранить в ХД только те данные, которые нужны для их анализа (например, для анализа нет смысла хранить информацию о номерах документов купли-продажи, в то время как их со держимое— количество, цена проданного товара — необходимо). Это существенно сокращает затраты на носители информации и повышает безопасность доступа к данным. ,

П Интеграция. ОИД, как правило, разрабатываются в разное время не сколькими коллективами с собственным инструментарием. Это приводит к тому, что данные, отражающие один и тот же объект реального мира в разных системах, описывают его по-разному. Обязательная интеграция данных в ХД позволяет решить эту проблему, приведя данные к единому формату.

О Поддержка хронологии. Данные в ОИД необходимы для выполнения над ними операций в текущий момент времени. Поэтому они могут не иметь привязки ко времени. Для анализа данных часто бывает важно иметь воз можность отслеживать хронологию изменений показателей предметной области. Поэтому все данные, хранящиеся в ХД, должны соответствовать последовательным интервалам времени.

О Неизменяемость. Требования к ОИД накладывают ограничения на время хранения в них данных. Те данные, которые не нужны для оперативной обработки, как правило, удаляются из ОИД для уменьшения занимаемых ресурсов. Для анализа, наоборот, требуются данные за максимально боль шой период времени. Поэтому, з отличие от ОИД, данные в ХД после за грузки только читаются. Это позволяет существенно повысить скорость доступа к данным, как за счет возможной избыточности хранящейся ин формации, так и за счет исключения операций модификации.

При реализации в СППР концепции ХД данные из разных ОИД копируются в единое хранилище. Собранные данные приводятся к единому формату, со гласовываются и обобщаются. Аналитические запросы адресуются к ХД (рис. 2.1).

Такая модель неизбежно приводит к дублированию информации в ОИД и в ХД. Однако Инмон в своей работе

утверждает, что избыточность данных, хранящихся в СППР, не превышает 1 %! Это можно объяснить следующими причинами.

При загрузке информации из ОИД в ХД данные фильтруются. Многие из них не попадают в ХД, поскольку лишены смысла с точки зрения использования в процедурах анализа.

Информация в ОИД носит, как правило, оперативный характер, и данные, потеряв актуальность, удаляются. В ХД, напротив, хранится историческая информация. С этой точки зрения дублирование содержимого ХД данными ОИД оказывается весьма незначительным. В ХД хранится обобщенная ин формация, которая в ОИД отсутствует.

Во время загрузки в ХД данные очищаются (удаляется ненужная информа ция), и после такой обработки они занимают гораздо меньший объем.

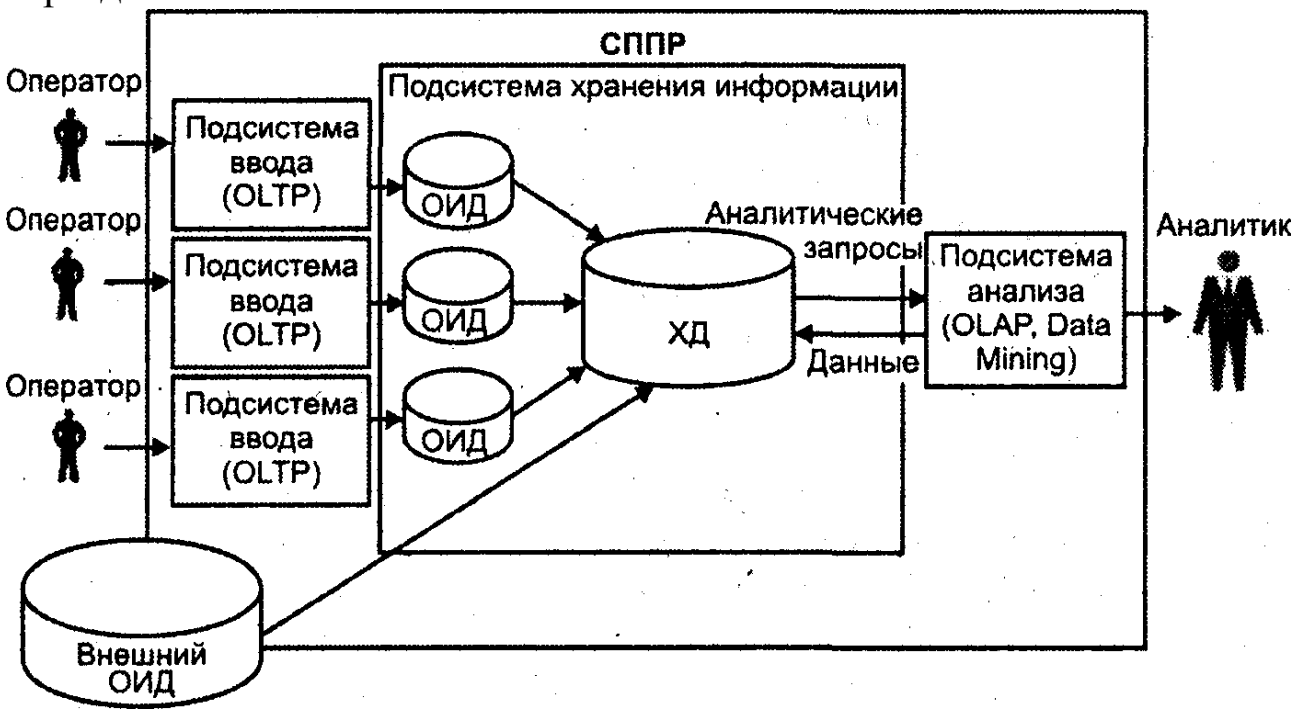

Рис. 2.1. Структура СППР с физическим ХД

Избыточность информации можно свести к нулю, используя виртуальное ХД. В данном случае в отличие от классического (физического) ХД данные из ОИД не копируются в единое хранилище. Они извлекаются, преобразуются и интегрируются непосредственно при выполнении аналитических запросов в оперативной памяти компьютера. Фактически такие запросы напрямую адре суются к ОИД (рис. 2.2). Основными достоинствами виртуального ХД явля ются; ^ '

О минимизация объема памяти, занимаемой на носителе информацией;

О работа с текущими, детализированными данными.

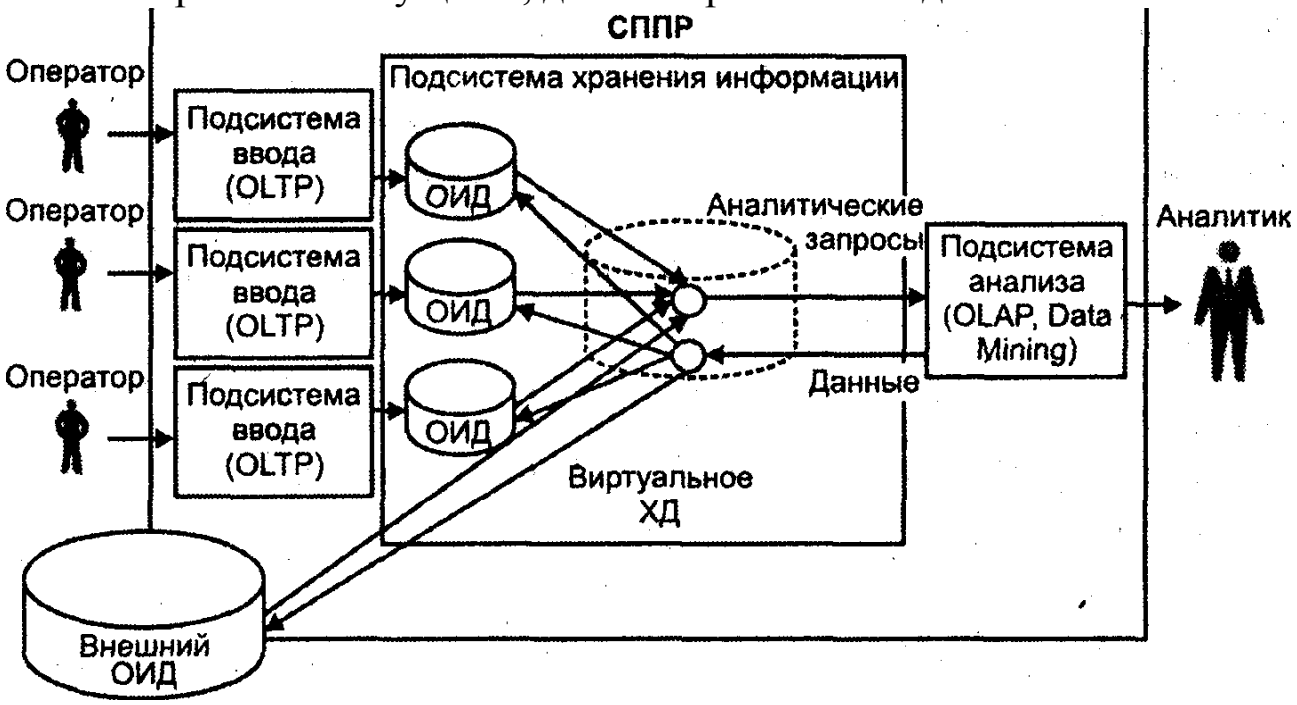

Рис. 2.2. Структура.СППР с виртуальным ХД

Однако такой подход обладает многими недостатками.

Время обработки запросов к виртуальному ХД значительно превышает соот ветствующие показатели для физического хранилища. Кроме того, структуры оперативных БД рассчитанные на интенсивное обновление одиночных запи сей, в высокой степени нормализованы. Для выполнения же аналитического запроса требуется объединение большого числа таблиц, что также приводит к снижению быстродействия.

Интегрированный взгляд на виртуальное хранилище возможен только при выполнении условия постоянной доступности всех ОИД. Таким образом, временная недоступность хотя бы одного из источников может привести ли бо к невыполнению аналитического запроса, либо к неверным результатам.

Выполнение сложных аналитических запросов над ОИД требует значитель ных ресурсов компьютеров. Это приводит к снижению быстродействия ОЬТР-систем, что недопустимо, т. к. время выполнения операций в таких системах часто весьма критично.

Различные ОИД могут поддерживать разные форматы и кодировки данных. Часто на один и тот же вопрос может быть получено несколько вариантов ответа. Это может быть связано с несинхронноетью моментов обновления данных в разных ОИД, отличиями в описании одинаковых объектов и собы тий предметной области, ошибками при вводе, утерей фраАюнтов архивов и т. д. В

таком случае цель — формирование единого непротиворечивого взгляда на объект управления — может быть не достигнута.

Главным же недостатком виртуального хранилища следует признать практи ческую невозможность получения данных за долгий период времени. При отсутствии физического хранилища доступны только те данные, которые на момент запроса есть в ОИД. Основное назначение ОЬТР-систем — оператив ная обработка текущих данных, поэтому они не ориентированы на хранение данных за длительный период времени. По мере устаревания данные выгру жаются в архив и удаляются из оперативной БД.

Несмотря на преимущества физического ХД перед виртуальным, необходимо признать, что его реализация представляет собой достаточно трудоемкий процесс. Остановимся на основных проблемах создания ХД:

О необходимость интеграции данных из неоднородных источников в рас пределенной среде;

О потребность в эффективном хранении и обработке очень больших объемов информации;

О необходимость наличия многоуровневых справочников метаданных;

О повышенные требования к безопасности данных.

Рассмотрим эти проблемы более подробно.

Необходимость интеграции данных из неоднородных источников в рас пределенной среде. ХД создаются для интегрирования данных, которые мо гут поступать из разнородных ОИД, физически размещающихся на разных компьютерах: БД, электронных архивов, публичных и коммерческих элек тронных каталогов, справочников, статистических сборников. При создании ХД приходится решать задачу построения системы, согласованно функцио нирующей с неоднородными программными средствами и решениями. При выборе средств реализации ХД приходится учитывать множество факторов, включающих уровень совместимости различных программных компонентов, легкость их освоения и использования, эффективность функционирования и т. д.

Потребность в эффективном хранении и обработке очень больших объе мов информации. Свойство неизменности ХД предполагает накопление в нем информации за долгий период времени, что должно поддерживаться по стоянным ростом объемов дисковой памяти. Ориентация на выполнение ана литических запросов и связанная с этим денормализация данных приводят к

нелинейному росту объемов памяти, занимаемой ХД при возрастании объема данных. Исследования, проведенные на основе тестового набора TPC-0, по казали, что для баз данных объемом в 100 Гбайт требуется память, в 4,87 раза большая объемом, чем нужно для хранения полезных данных.

Необходимость многоуровневых справочников метаданных. Для систем анализа наличие развитых метаданных (данных о данных) и средств их пре доставления конечным пользователям является одним из основных условий успешной реализации ХД. Метаданные необходимы пользователям СППР для понимания структуры информации, на основании которой принимается решение. Например, прежде чем менеджер корпорации задаст системе свой вопрос, он должен понять, какая информация имеется, насколько она акту альна, можно ли ей доверять, сколько времени может занять формирование ответа и т. д. При создании ХД необходимо решать задачи хранения и удоб ного представления метаданных пользователям,

Повышение требований к безопасности данных. Собранная вместе и со гласованная информация об истории развития корпорации, ее успехах и не удачах, о взаимоотношениях с поставщиками и заказчиками, об истории и состоянии рынка дает возможность анализа прошлой и текущей деятельности корпорации и построения прогнозов для будущего. Очевидно, что подобная информация является конфиденциальной и доступ к ней ограничен в преде лах самой компании, не говоря уже о других компаниях. Для обеспечения безопасности данных приходится решать вопросы аутентификации пользова телей, защиты данных при их перемещении в хранилище данных Из опера тивных баз данных и внешних источников, защиты данных при их передаче по сети и т. п.

Снижения затрат на создание ХД можно добиться, создавая его упрощенный вариант — витрину данных (Data Mart).

Витрина данных (ВД) — это упрощенный вариант ХД, содержащий только тема тически объединенные данные.

ВД максимально приближена к конечному пользователю и содержит данные, тематически ориентированные на него (например, ВД для работников отдела маркетинга может содержать данные, необходимые для маркетингового ана лиза). ВД существенно меньше по объему, чем ХД, и для ее реализации не требуется больших затрат. Они могут быть реализованы как самостоятельно, так и вместе с ХД.

Самостоятельные ВД (рис. 2.3) часто появляются в организации исторически и встречаются в крупных организациях с

большим количеством независимых подразделений, решающих собственные аналитические задачи.

Достоинствами такого подхода являются:

О проектирование ВД для ответов на определенный круг вопросов;

О быстрое внедрение автономных ВД и получение отдачи; ^

П упрощение процедур заполнения ВД и повышение их производительности за счет учета потребностей определенного круга пользователей.

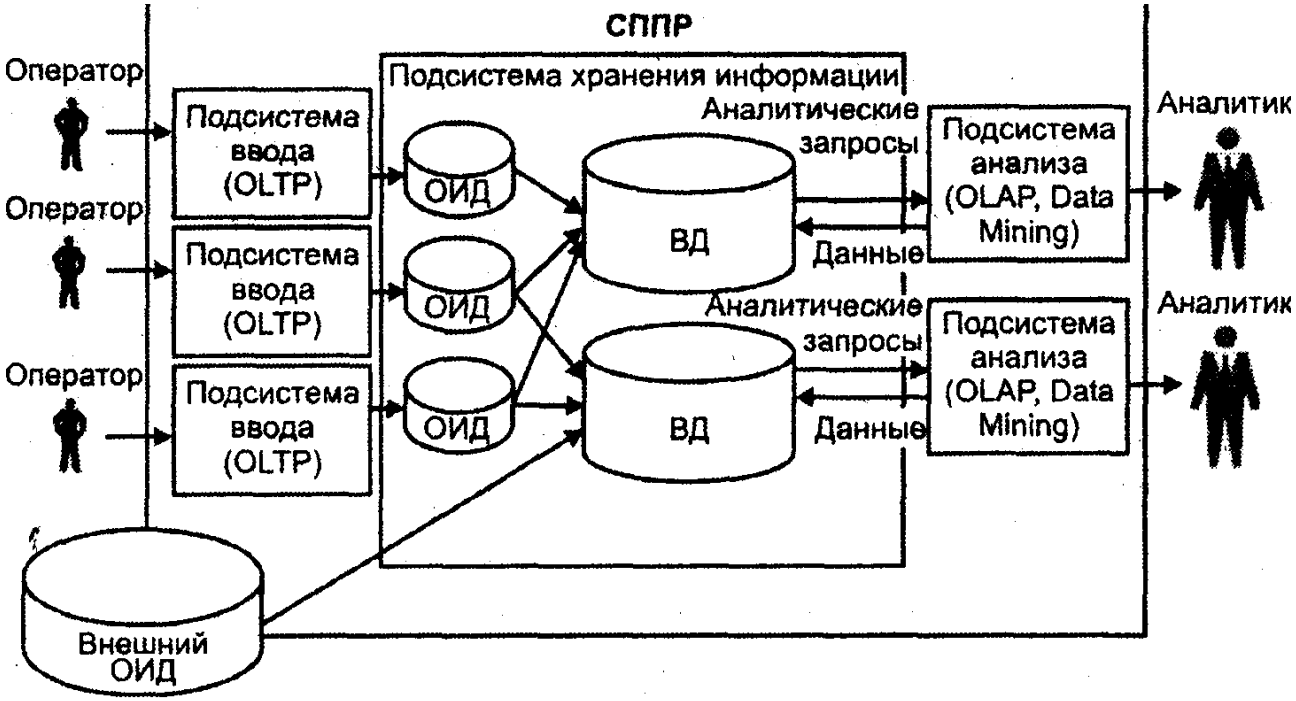

Рис. 2.3. Структура СППР с самостоятельными ВД

Недостатками автономных ВД являются:

С1 многократное хранение данных в разных ВД, что приводит к увеличению расходов на их хранение и потенциальным проблемам, связанным с необ ходимостью поддержания непротиворечивости данных;

О отсутствие консолидированности данных на уровне предметной области, а следовательно — отсутствие единой картины.

В последнее время все более популярной становится идея совместить ХД и ВД в одной системе. В этом случае ХД используется в качестве единственно го источника интегрированных данных для всех ВД (рис. 2.4).

ХД представляет собой единый централизованный источник информации для всей предметной области, а ВД являются подмножествами данных из храни лища, организованными для представления информации по тематическим разделам данной

области. Конечные пользователи имеют возможность дос тупа к детальным данным хранилища, если данных в витрине недостаточно, а также для получения более полной информационной картины.

Достоинствами такого подхода являются:

П простота создания и наполнения ВД, поскольку наполнение происходит из единого стандартизованного надежного источника очищенных данных — из ХД;

О простота расширения СППР за счет добавления новых ВД;

О снижение нагрузки на основное ХД.

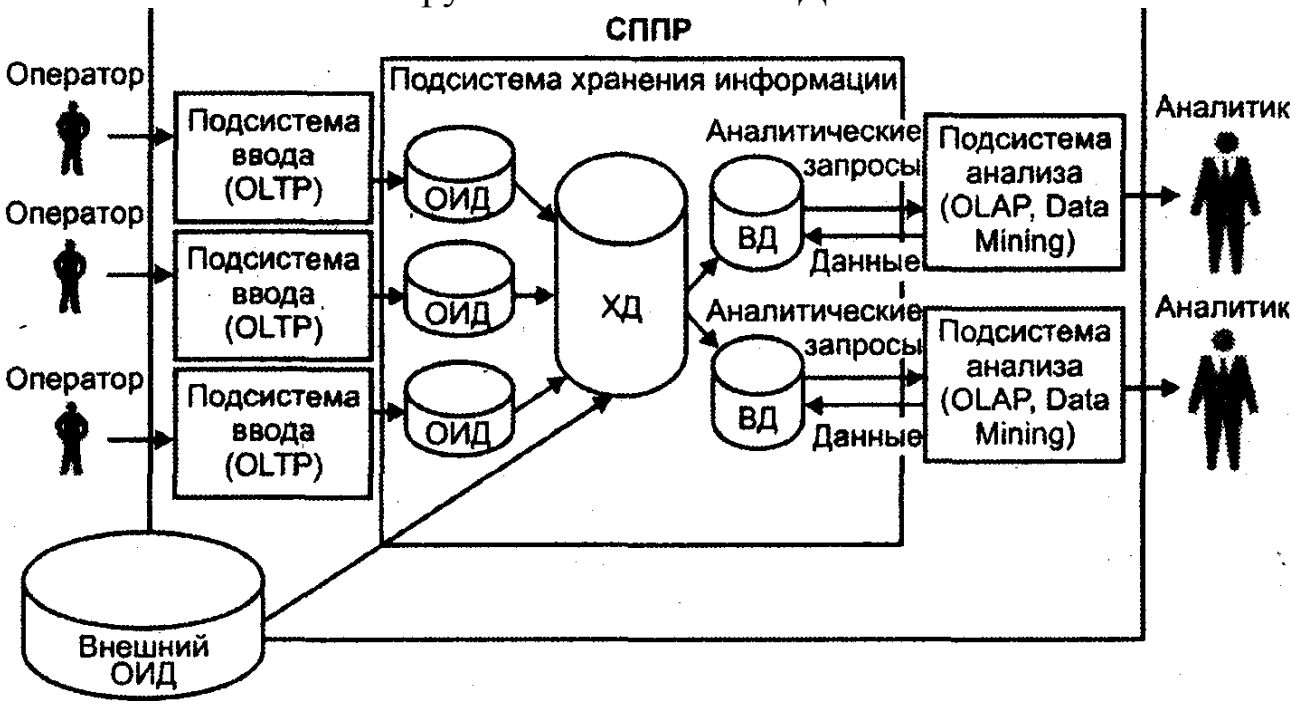

Рис. 2.4. Структура СППР с ХД и ВД

К недостаткам относятся:

О избыточность (данные хранятся как в ХД, так и в ВД);

О дополнительные затраты на разработку СППР с ХД и ВД.

Подводя итог анализу путей реализации СПП^* с использованием концепции ХД, можно выделить следующие архитектуры таких систем:

О СППР с физическим (классическим) ХД (см. рис. 2.1);

П СППР с виртуальным ХД (см. рис. 2.2);

а СППР с ВД (см. рис. 2.3);

П СППР с физическим ХД и с ВД (рис. 2.4).

В случае архитектур с физическим ХД и/или ВД необходимо уделить внима ние вопросам организации (архитектуры) ХД и переносу данных из ОИД вХД.

2.2. Организация ХД

Все данные в ХД делятся на три основные категории (рис. 2.5);

П детальные данные;
_ «
О агрегированные данные;
П метаданные.

Так как метаданные играют важную роль в процессе работы с ХД, то к ним должен быть обеспечен удобный доступ. Для этого они сохраняются в репо-зитории метаданных с удобным для пользователя интерфейсом.

Данные, поступающие из ОИД в ХД, перемещаемые внутри ХД и поступаю щие из ХД к аналитикам, образуют следующие информационные потоки (см. рис. 2.5):

О входной поток (1пйоуу)— образуется данными, копируемыми из ОИД вХД;

О поток обобщения (Приеду) — образуется агрегированием детальных дан ных и их сохранением в ХД;

О архивный поток (Оо\упЛоу/) — образуется перемещением детальных дан ных, количество обращений к которым снизилось;

О поток метаданных (Ме1аР1о^)— образуется переносом информации о данных в репозиторий данных;

СЧ выходной поток (Ои1По\у) — образуется данными, извлекаемыми пользо вателями;

О обратный поток (РееаЬаск Р1о\У) — образуется очищенными данными, за писываемыми обратно в ОИД.

Самый мощный из информационных потоков — входной — связан с перено сом данных из ОИД. Обычно информация не просто копируется в ХД, а под вергается обработке: данные очищаются и обогащаются за счет добавления новых атрибутов. Исходные данные из ОИД объединяются с информацией из внешних источников — текстовых файлов, сообщений электронной почты, электронных таблиц и др. При разработке ХД не менее 60% всех затрат свя зано с переносом данных.

Процесс переноса, включающий в себя этапы извлечения, преобразования и загрузки, называют ЕТЬ-процессом (Е — ехп-аеЦоп, Т — 1:гап5Гогта1юп, Ь — 1оас1т§: извлечение, преобразование и загрузка, соответственно). Программ ные средства, обеспечивающие его выполнение, называются ЕТЬ-системами. Традиционно ЕТЬ-системы использовались для переноса информации

из ус таревших версий информационных систем в новые. В настоящее время ЕТЬ-процесс находит все большее применение для переноса данных из ОИД в ХД иВД.

Рассмотрим более подробно этапы ЕТЬ-процесса (рис. 2.6).

Извлечение данных. Чтобы начать ЕТЬ-процесс, необходимо извлечь дан ные из одного или нескольких источников и подготовить их к этапу преобра зования. Можно выделить два способа извлечения данных:

1. Извлечение данных вспомогательными программными средствами непо средственно из структур хранения информации (файлов, электронных таб-

лиц, БД и т. п. Достоинствами такого способа извлечения данных явля ются: \

• отсутствие необходимости расширять ОЬТР-систему (это особенно важно, если ее структура закрыта);

• данные могут извлекаться с учетом потребностей процесса переноса.

2. Выгрузка данных средствами 01/ГР-систем в промежуточные структуры. Достоинствами такого подхода являются:

• возможность использовать средства ОЫТ-систем, адаптированные к структурам данных;

• средства выгрузки изменяются вместе с изменениями ОЬТР-систем и

оид;

• возможность выполнения первого шага преобразования данных за счет определенного формата промежуточной структуры хранения данных.

Рис. 2.6. ЕТ1--процесс

Преобразование данных. После того как сбор данных завершен, необходимо преобразовать их для размещения на новом месте. На этом этапе выполняют ся следующие процедуры:

О обобщение данных (а§@ге@а1юп) — перед загрузкой данные обобщаются. Процедура обобщения заменяет многочисленные детальные данные отно сительно небольшим числом агрегированных данных. Например, предпо ложим, что данные о продажах за год занимают в нормализованной базе

данных несколько тысяч записей/После обобщения данные преобразуют ся в меньшее число кратких записей, которые будут перенесены в ХД;

О перевод значений (уа1ие 1гап51а1юп) — в ОИД данные часто хранятся в за кодированном виде для того, чтобы сократить избыточность данных и па мять для их хранения. Например, названия товаров, городов, специально стей и т. п. могут храниться в сокращенном виде. Поскольку ХД содержат обобщенную информацию и рассчитаны на простое использование, зако дированные данные обычно заменяют на более понятные описания;

[Т создание полей (Пе1(1 (1ета1юп)— при создании полей для конечных пользователей создается и новая информация. Например, ОИД содержит одно поле для указания количества проданных товаров, а второе — для указания цены одного экземпляра. Для исключения операции вычисления стоимости всех товаров можно создать специальное поле для ее хранения во время преобразования данных;

П очистка данных (с1еагип§) — направлена на выявление и удаление ошибок и несоответствий в данных с целью улучшения их качества. Проблемы с качеством встречаются в отдельных ОИД, например, в файлах и БД могут быть ошибки при вводе, отдельная информация может быть утрачена, мо гут присутствовать "загрязнения" данных и др. Очистка также применяет ся для согласования атрибутов полей таким образом, чтобы они соответст вовали атрибутам базы данных назначения.

Загрузка данных. После того как данные преобразованы для размещения в ХД, осуществляется этап их загрузки. При загрузке выполняется запись пре образованных детальных и агрегированных данных. Кроме того, при записи новых детальных данных часть старых данных может переноситься в архив.

2.3. Очистка данных

Одной из важных задач, решаемых при переносе данных в ХД, является их очистка. С одной стороны, данные загружаются постоянно из различных ис точников, поэтому вероятность попадания "грязных данных" весьма высока. С Другой стороны, ХД используются для принятия решений, и "грязные дан ные" могут стать причиной принятия неверных решений. Таким образом, процедура очистки является обязательной при переносе данных из ОИД в ХД. Ввиду большого спектра возможных несоответствий в

данных их очист ка считается одной из самых крупных проблем в технологии ХД.

2.4. Концепция хранилища данных и анализ

Концепция ХД не является законченным архитектурным решением СППР и тем более не является готовым программным продуктом. Цель концепции ХД — определить требования к данным, помещаемым в ХД, общие принци пы и этапы построения ХД основные источники данных, дать рекомендации по решению потенциальных проблем, возникающих при выгрузке, очистке, согласовании, транспортировке и загрузке данных.

Необходимо понимать, что концепция ХД:

О это не концепция анализа данных, скорее, это концепция подготовки дан ных для анализа;

П не предопределяет архитектуру целевой аналитической системы. Концеп ция ХД указывает на то, какие процессы должны выполняться в системе, но не где конкретно и как они будут выполняться.

Таким образом, концепция ХД определяет лишь самые общие принципы по строения аналитической системы и в первую очередь сконцентрирована на свойствах и требованиях к данным, но не на способах организации и пред ставления данных в целевой БД и режимах их использования. Концепция ХД описывает построение аналитической системы, но не определяет характер ее использования. Она не решает ни одну из следующих проблем:

О выбор наиболее эффективного для анализа способа организации данных;

О организация доступа к данным; ' О использование технологии анализа.

Проблемы использования собранных данных решают подсистемы анализа. Как отмечалось в гл. 1, такие подсистемы используют следующие техноло гии:

О регламентированные запросы;

П оперативный анализ данных;

О интеллектуальный анализ данных.

Если регламентированные запросы успешно применялись еще задолго до по явления концепции ХД, то оперативный и интеллектуальный анализы в по следнее время все больше связывают с ХД.

Выводы

Из материала, изложенного в данной главе, можно сделать следующие вы воды.

О Концепция ХД предполагает разделение структур хранения данных для оперативной обработки и выполнения аналитических запросов. Это позво-

ляет в рамках одной СППР объединить две подсистемы, удовлетворяющие противоречивым требованиям.

О В соответствии с определением Инмона, ХД—это предметно-ориен тированный, интегрированный, неизменчивый, поддерживающий хроно логию набор данных, организованный для целей поддержки принятия ре шений.

О Различают два вида ХД: виртуальное и физическое. В системах, реали зующих концепцию виртуального ХД, аналитические запросы адресуются непосредственно к ОИД, а полученные результаты интегрируются в опе ративной памяти компьютера. В случае физического ХД данные перено сятся из разных ОИД в единое хранилище, к которому адресуются анали тические запросы.

О Облегченным вариантом ХД является ВД, которая содержит только тема тически объединенные данные. ВД существенно меньше по объему, чем ХД, и для ее реализации не требуется больших затрат. ВД может быть реа лизована или самостоятельно, или в комбинации с ХД.

["1 ХД включает в себя: метаданные, детальные, агрегированные и архивные данные. Перемещающиеся в ХД данные образуют информационные пото ки: входной, обобщающий, обратный, выходной и поток метаданных.

1~1 Детальные данные разделяют на два класса: измерения и факты. Измере ниями называются наборы данных, необходимые для описания событий. Фактами называются данные, отражающие сущность события.

О Агрегированные данные получаются из детальных данных путем их сум мирования по измерениям. Для быстрого доступа к наиболее часто запра шиваемым агрегированным данным они должны сохраняться в ХД, а не вычисляться при выполнении запросов.

О Метаданные необходимы для получения пользователем информации о данных, хранящихся в ХД. Согласно принципам Захмана, метаданные должны описывать объекты предметной области, представленные в ХД, пользователей, работающих с

данными, места хранения данных, действия над данными, время обработки данных и причины модификаций данных.

П Наиболее мощным информационным потоком в ХД является входной — поток переноса данных из ОИД в ХД. Процесс переноса, включающий этапы сбора, преобразовайия и загрузки, называют ЕТЬ-процессом.

П Наиболее важной задачей при переносе данных является их очистка. Ос новные проблемы очистки данных можно классифицировать но следую щим уровням: уровень ячейки таблицы, уровень записи, уровень таблицы БД, уровень одиночной БД, уровень множества БД.

О Очистка данных включает следующие этапы: выявление проблем'в дан ных, определение правил очистки, тестирование правил очистки, непо средственная очистка данных. После исправления ошибок отдельных ис точников очищенные данные должны заменить загрязненные данные в ис ходных рид.

П Очищенные данные сохраняются в ХД и могут использоваться для анали за и принятия на их основе решений. За формирование аналитических за просов к данным и представление результатов их выполнения в СППР от вечают подсистемы анализа. От вида анализа также зависит и непосредст венная реализация структур хранения данных в ХД.