

Лабораторная работа №4

Хранилище данных. Детализированные и агрегированные данные.

Цель работы: изучить процесс проектирования хранилища данных, изучить возможность работы с детализированными и агрегированными данными.

Задачи работы:

- изучить методику проектирования семантического слоя хранилища в Deductor Studio;
- изучить приведенный в лабораторной работе пример;
- выполнить контрольное задание.

1. Краткая теория

В хранилищах данных данные хранятся как в детализированном, так и в агрегированном виде.

Детализированные данные

Данные в детализированном виде поступают непосредственно из источников данных и соответствуют элементарным событиям, регистрируемым OLTP-системами. Такими данными могут быть ежедневные продажи, количество произведенных изделий и т.д. Это неделимые значения, попытка дополнительно детализировать которые лишает их логического смысла.

Агрегированные данные

Многие задачи анализа, такие как прогнозирование, требуют использования данных определенной степени обобщения. Например, суммы продаж, взятые по дням, могут дать очень неравномерный ряд данных, что затруднит выявление характерных периодов, закономерностей или тенденций. Однако, если обобщить эти данные в пределах недели или месяца и взять сумму, среднее, максимальное и минимальное значения за соответствующий период, то полученный ряд может оказаться более информативным. Процесс обобщения детализированных данных называется агрегированием, а сами обобщенные данные — агрегированными (иногда — агрегатами). Обычно агрегированию подвергаются числовые данные (факты), они вычисляются и содержатся в хранилище данных вместе с детализированными данными.

Метаданные

Метаданные необходимы для описания значения и свойств информации с целью лучшего ее понимания, использования и управления ею. Буквально, метаданные – это данные о данных.

Метаданные — любая информация, необходимая для анализа, проектирования, построения, внедрения и применения компьютерной информационной системы. Одно из основных назначений метаданных — повышение эффективности поиска. Поисковые запросы, использующие

метаданные, делают возможным выполнение сложных операций по фильтрации и отбору данных.

Метаданные — высокоуровневые средства отражения информационной модели и описания структуры данных, используемой в хранилище данных. Метаданные должны содержать описание структуры данных хранилища и структуры данных импортируемых источников. Метаданные хранятся отдельно от данных в так называемом репозитории метаданных.

Метаданные являются ключевым фактором успеха при разработке и внедрении хранилищ данных. Они содержат всю информацию, необходимую для извлечения, преобразования и загрузки данных из различных источников, а также для последующего использования и интерпретации данных, содержащихся в хранилищах данных.

Можно выделить два уровня метаданных — технический (административный) и бизнес-уровень. Технический уровень содержит метаданные, необходимые для обеспечения функционирования хранилища (статистика загрузки данных и их использования, описание модели данных и т.д.). Бизнес-метаданные обеспечивают пользователю возможность концентрироваться на процессе анализа, а не на технических аспектах работы с хранилищем; они включают бизнес-термины и определения, которыми привык оперировать пользователь.

Фактически бизнес-метаданные представляют собой описание предметной области, для работы в которой создается аналитическая система или хранилище данных. К формированию бизнес-метаданных должны активно привлекаться эксперты и аналитики, которые впоследствии и будут использовать систему для получения аналитических отчетов.

Бизнес-метаданные описывают объекты предметной области, информация о которых содержится в ХД, — атрибуты объектов и их возможные значения, соответствующие поля в таблицах и т.д. Бизнес-метаданные образуют так называемый семантический слой. Пользователь оперирует близкими ему терминами предметной области: товар, клиент, продажи, покупки и т.д., а семантический слой транслирует бизнес-термины в низкоуровневые запросы к данным в хранилище.

Алгоритмы обработки данных

Прежде чем использовать полученные данные для анализа с помощью Microsoft Analyse Manager, следует этап их подготовки и переноса (ETL-процесс). ETL-процесс делится на три этапа (рисунок 1):

Извлечение данных (на этом этапе данные извлекаются из одного или нескольких источников и подготавливаются к преобразованию. Следует отметить, что для корректного представления данных после их загрузки в ХД из источников должны извлекаться не только сами данные, но и информация, описывающая их структуру, из которой будут сформированы метаданные для хранилища);

Преобразование данных (производятся преобразование форматов и кодировки данных, а также их обобщение и очистка);

Загрузка данных — запись преобразованных данных в соответствующую систему хранения.

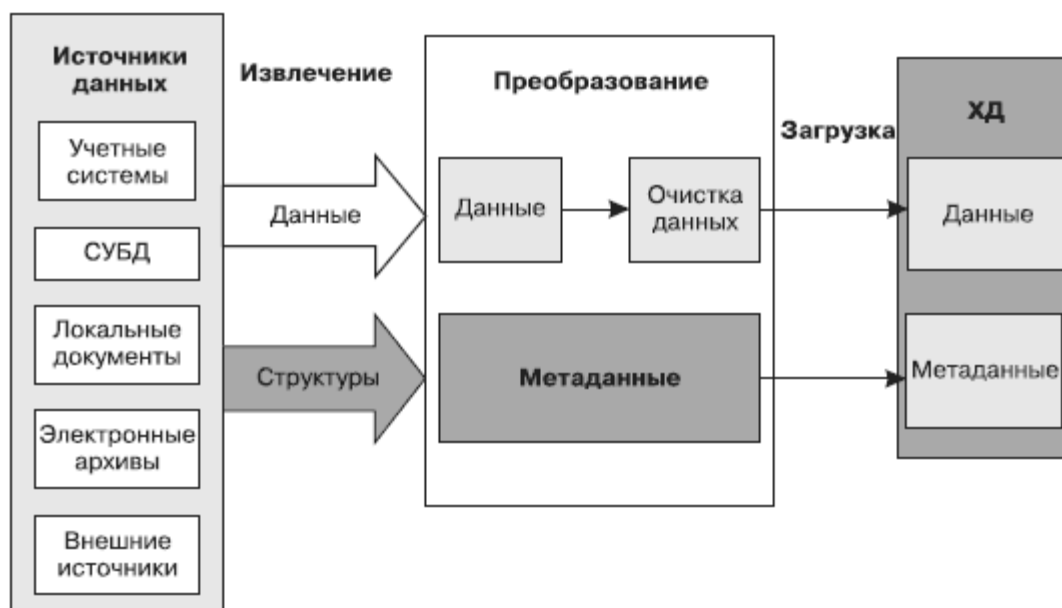


Рисунок 1 – Этапы ETL-процесса

Объекты хранилища данных Deductor Warehouse следующие.

Измерение – это последовательность значений одного из анализируемых параметров. Например, для параметра «время» это последовательность календарных дней, для параметра «регион» - список городов. Каждое значение измерения может быть представлено координатой в многомерном пространстве процесса, например, Товар, Клиент, Дата.

Атрибут – это свойство измерения (т.е. точки в пространстве). Атрибуты как бы скрыты внутри другого измерения и помогает пользователю полнее описать исследуемое измерение. Например, для измерения Товар атрибутами могут выступать Цвет, Вес, Габариты.

Факт – значение, соответствующее измерению. Факты – это данные, отражающие сущность события. Как правило, фактами являются численные значения, например, сумма и количество отгруженного товара, скидка.

Ссылка на измерение – это установленная связь между двумя и более измерениями. Дело в том, что некоторые бизнес-понятия (соответствующие измерениям в хранилище данных) могут образовывать иерархии, например, Товары могут включать Продукты питания и Лекарственные препараты, которые в свою очередь, подразделяются на группы продуктов и лекарств и т.д. В этом случае первое измерение содержит ссылку на второе, второе – на третье и т.д.

Процесс – совокупность измерений, фактов и атрибутов. По сути, процесс и есть «снежинка». Процесс описывает определенное действие,

например, продажи товара, отгрузки, поступления денежных средств и прочее.

Атрибут процесса – свойство процесса. Атрибут процесса в отличие от измерения не определяет координату в многомерном пространстве. Это справочное значение, относящееся к процессу, например, № накладной, Валюта документа и так далее. Значение атрибута процесса в отличие от измерения может быть не всегда определено.

В Deductor Warehouse может одновременно храниться множество процессов, имеющих общие измерения, например, измерение Товар, фигурирующее в процессах Поступления и Отгрузка.

Все загружаемые в хранилище данных данные обязательно должны быть определены как измерение, атрибут либо факт (рисунок 2).



Рисунок 2 – Проектирование структуры хранилища данных

Пример по истории продаж различных товаров (рисунок 3).

В таблице процесса хранится информация о значениях измерений (как правило, это код измерения) и о значениях фактов. В таблице процесса в первой строке содержится информация, что 05.06.2006г. клиент №3 приобрел товар №386 в количестве 100 шт. на сумму 25 000, при этом наценка составила 3 825. Кто такой клиент №3 и что за товар он приобрел, в таблице процесса не указано. Информация с описанием (атрибутами) клиентов и товаров находится в таблицах измерений, которые можно сравнить со словарями, хранящими справочную информацию по измерениям. Столбец Дата является измерением без атрибутов, и поэтому она присутствует только в таблице процесса.

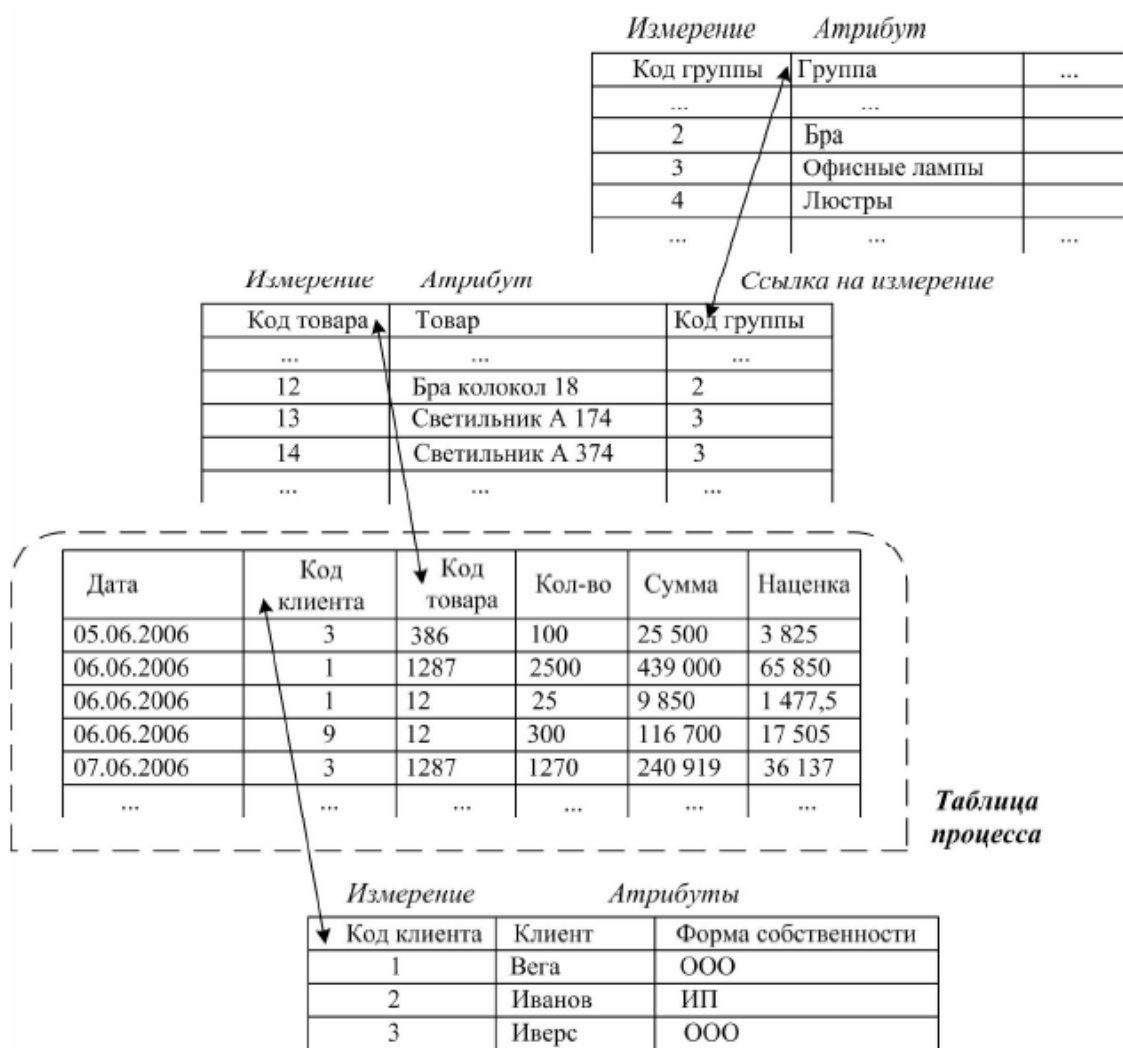


Рисунок 3- Пример схемы «снежинка»

Хранилище данных в Deductor Warehouse соответствует модели ROLAP (схема «снежинка») и может быть развернута на одной из следующих СУБД: Firebird; MS SQL Server; Oracle. С хранилищем на Firebird можно работать локально с использованием специальной библиотеки fbclient.dll.

Выбор той или иной СУБД зависит от многих критериев: стоимость, производительность, сложность администрирования и т.д.

Назначение хранилища данных – своевременно обеспечить аналитика всей информацией, необходимой для проведения анализа, построения моделей и принятия решений. Цель хранилища данных – не анализ данных, а подготовка данных для анализа и их интеграция.

Хранилище данных Deductor Warehouse включает в себя сами исторические данные и специальный семантический слой, содержащий так называемые метаданные (данные о данных). Семантический слой и сами данные физически хранятся в одной базе данных.

Запрос к хранилищу данных осуществляется непосредственно через семантический слой, который через внутреннюю систему команд подбирает

запрашиваемую информацию из многообразия хранимых данных. Работу семантического слоя можно сравнить с работой библиотекаря, который по просьбе читателя достает с разрозненных полок нужные книги, раскрывая их на нужных страницах.

Все данные хранятся в процессах, которые имеют структуру типа «снежинка», где в центре расположены таблицы фактов, а «лучами» являются измерения, причем каждое измерение может ссылаться на другое измерение.



Рисунок 4 - Измерения и факты

Пример 1.

Рассмотрим спроектированное и наполненное данными ХД с продажами аптечной сети (файл `farma.gdb`).

Создадим подключение к хранилищу данных. Для этого на панели Подключения вызовем Мастер подключений.

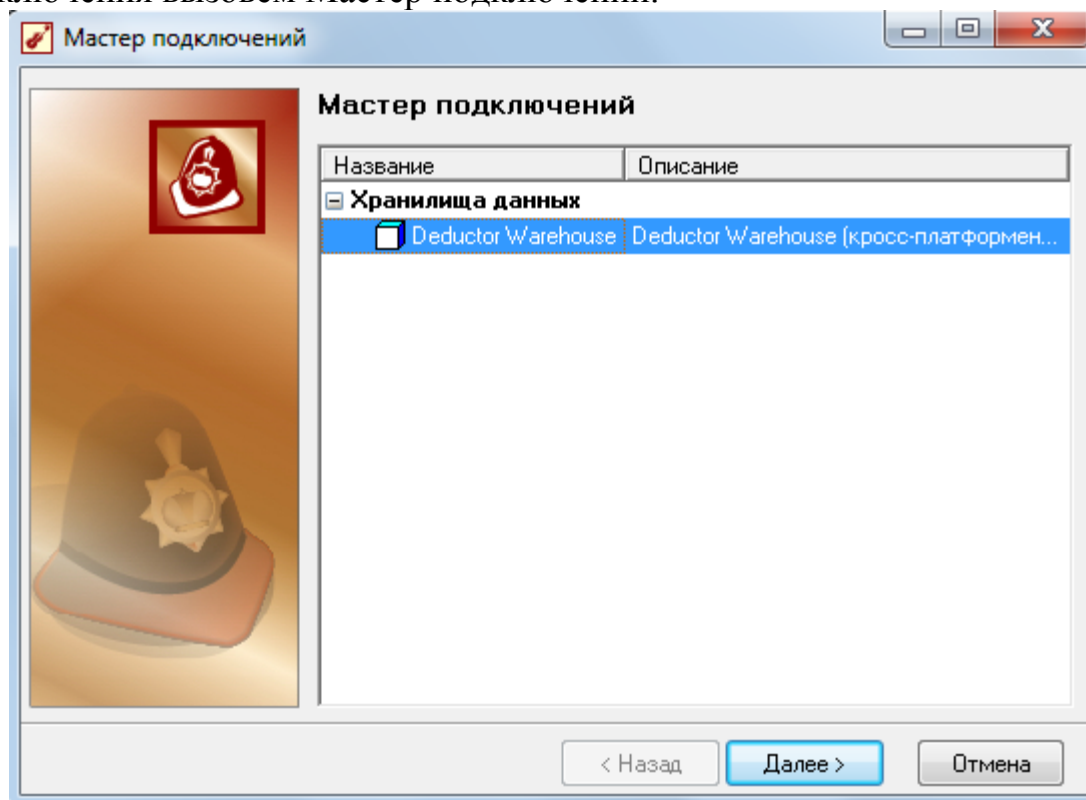


Рисунок 5 – Мастер подключений

Открылось окно выбора возможных источников. Выбираем Deductor Warehouse и переходим к следующему шагу настройки.

Настроим параметры подключения к хранилищу. Зададим строку подключения: она должна содержать полное имя подключаемого хранилища. Путь к нему можно ввести вручную или воспользоваться проводником.

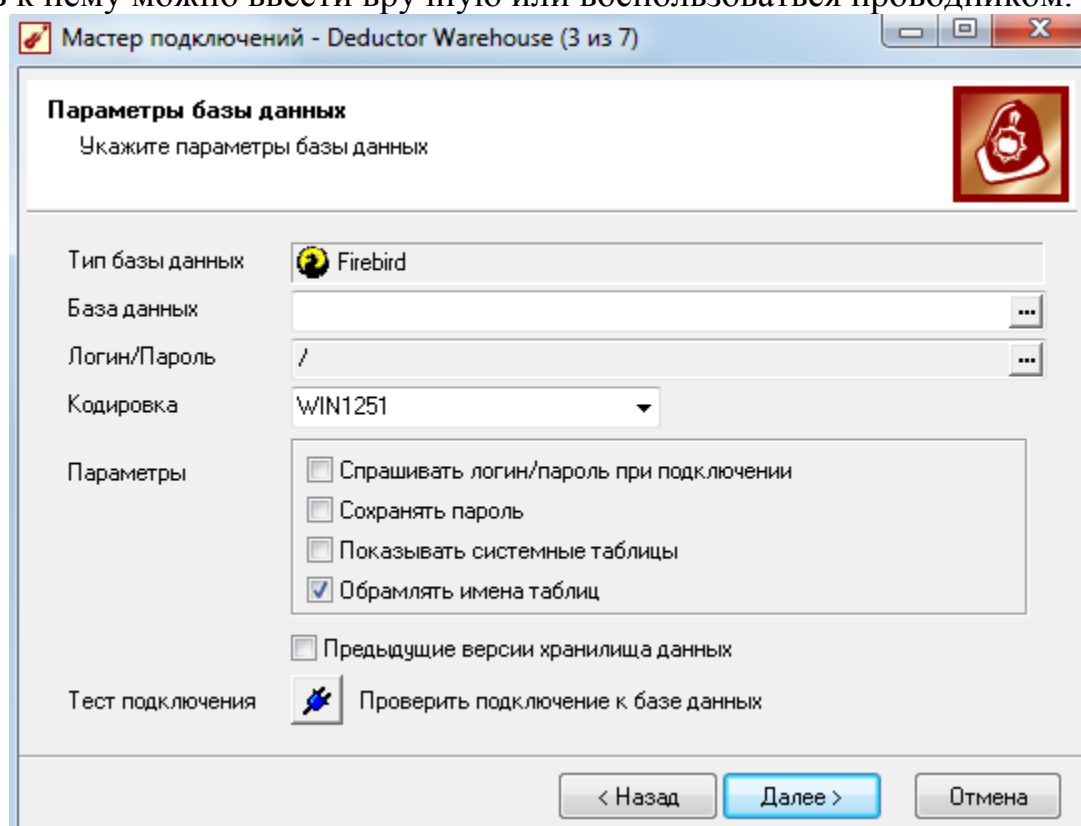


Рисунок 6 – Настройка подключения к хранилищу данных

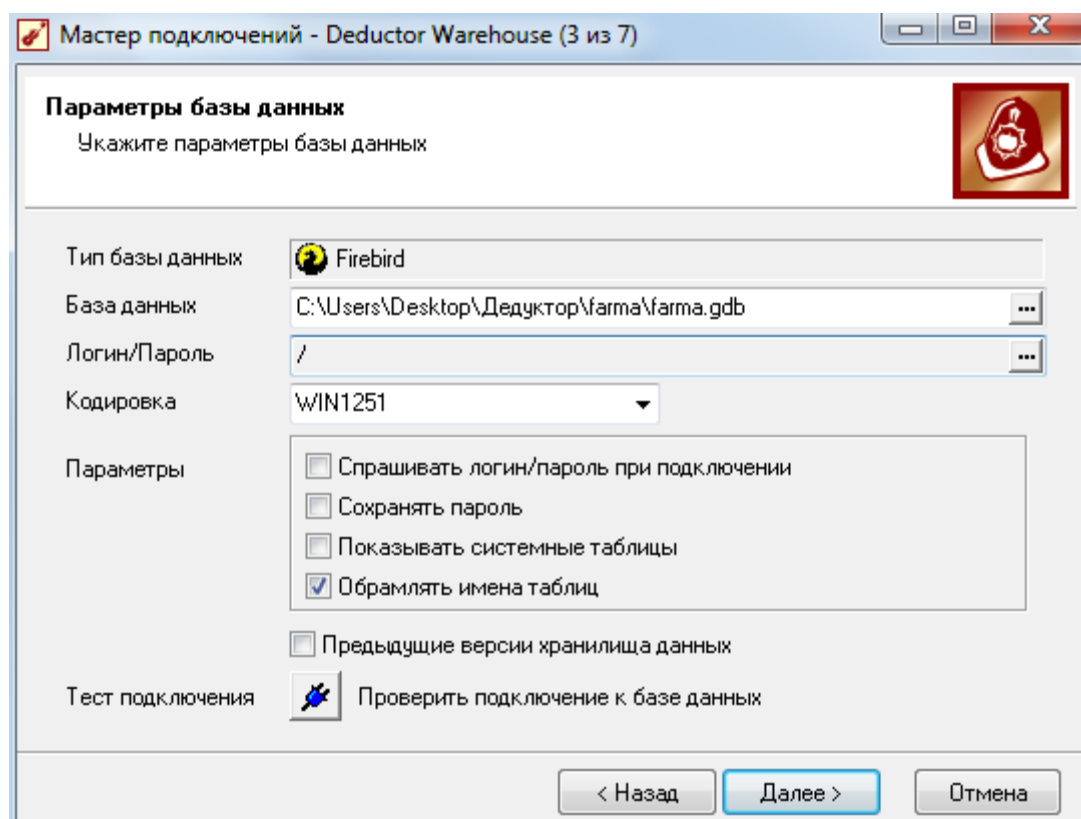


Рисунок 7 - Настройка подключения к хранилищу данных (продолжение)

Зададим имя пользователя и пароль для подключения к нему.

Введем логин sysdba, пароль masterkey. Установим флаг Сохранить пароль.

Настроим способ отображения. Установим все флаги.

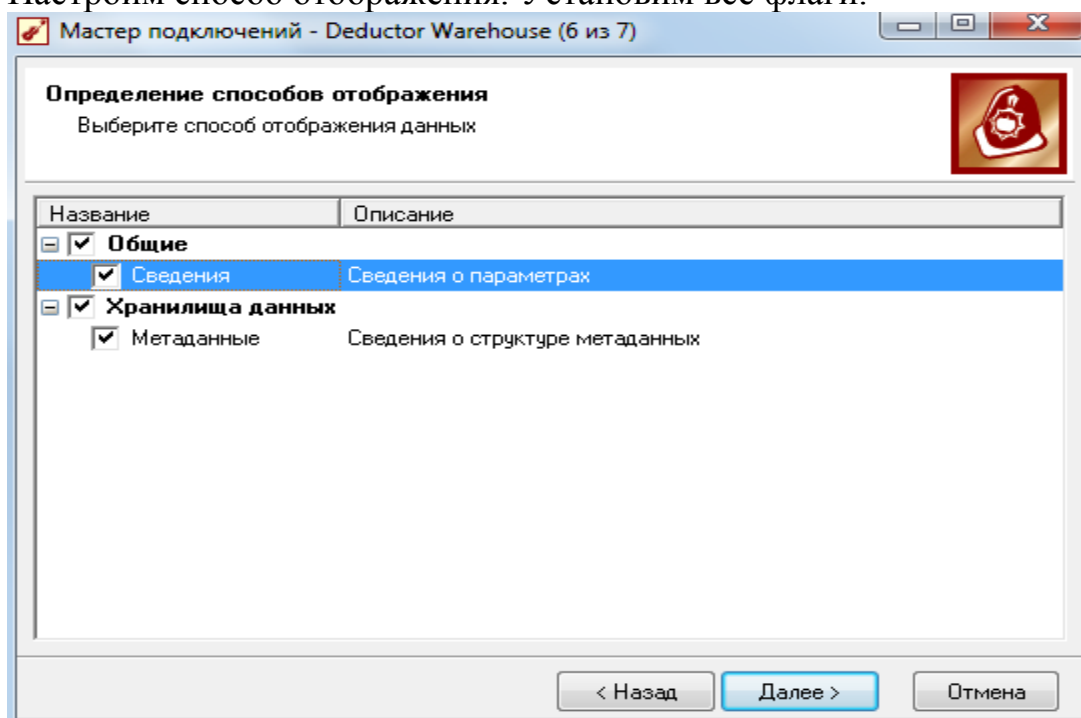


Рисунок 8 –Последний шаг настройки Мастера подключений

При успешном выполнении настроек подключения в списке подключений появилось подключение к хранилищу данных.

Открылись два визуализатора: Сведения и Метаданные.

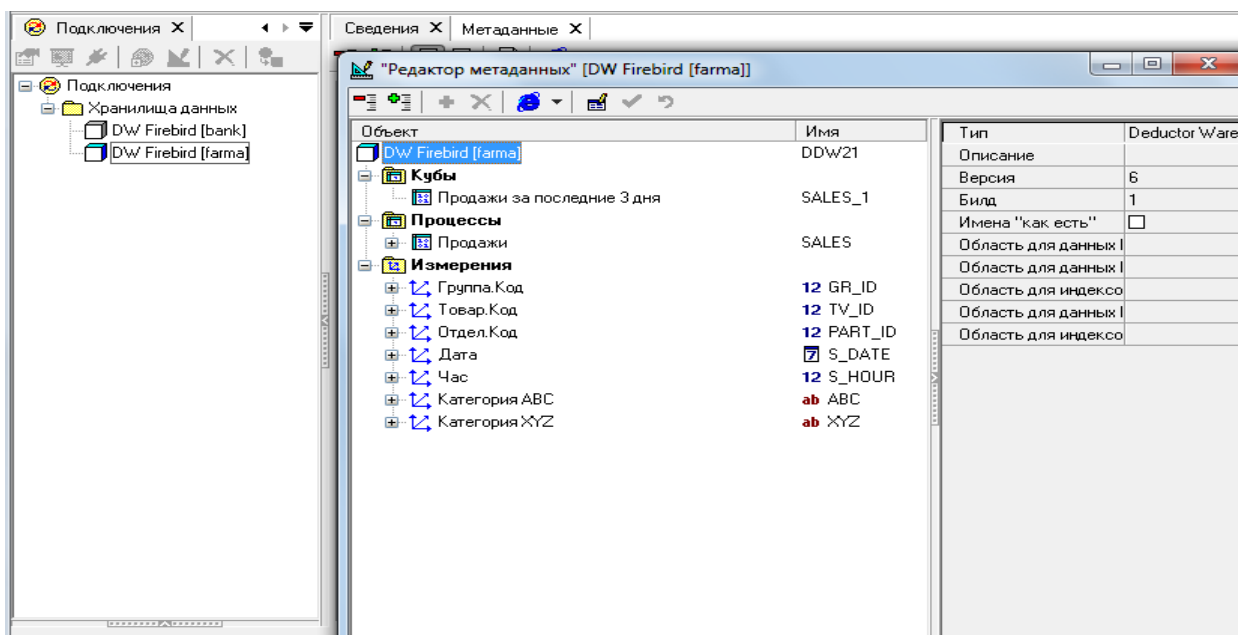


Рисунок 9 – Структура хранилища данных

В первом визуализаторе отображаются параметры хранилища, во втором – его структура (семантический слой).

Основные объекты семантического слоя это: процесс, измерение, факт. Дополнительно могут быть: куб, атрибут процесса, атрибут (измерения).

Процесс – совокупность измерений, фактов и атрибутов. По сути, процесс и есть «снежинка». Процесс описывает определенное действие, например, продажи товара, отгрузки и т.д.

Измерение – это последовательность значений одного из анализируемых параметров. Каждое значение измерения может быть представлено координатой в многомерном пространстве процесса. Измерения содержат только справочную информацию например, Товар, Клиент, Дата.

Атрибут – это свойство измерения (т.е. точки в пространстве). Атрибут как бы скрыт внутри другого измерения и помогает пользователю полнее описать исследуемое измерение. Например, для измерения Товар атрибутами могут выступать Цвет, Вес, Габариты.

Факт – значение, соответствующее измерению. Факты – это данные, отражающие сущность события. Как правило, фактами являются численные значения, например, сумма и количество, скидка.

Атрибут процесса – свойство процесса. Атрибут процесса в отличие от измерения не определяет координату в многомерном пространстве. Это справочное значение, относящееся к процессу, например, № накладной,

Валюта документа и так далее. Значение атрибута процесса в отличие от измерения может быть не всегда определено.

Пример 2.

Рассмотрим для примера наиболее часто встречаемую структуру процесса Продажи.



Рисунок 10 – Процесс Продажи

Измерениями процесса являются: Товар, Клиент, Дата, Филиал. Некоторые измерения входят в иерархию, так Товар в Группа, а Дата в Месяц. Фактами процесса являются: Количество; Сумма.

В Deductor Warehouse одновременно хранится множество процессов. Процессы могут иметь общие измерения, например, измерение Товар, фигурирующее в процессах Поступления и Отгрузка.

Для изучения структуры хранилища перейдем в визуализатор Метаданные. В нем отображается структура хранилища в виде дерева.

В корне дерева находится имя хранилища и его метка. Ниже располагаются три ветви: кубы; процессы; измерения. Все остальные объекты хранилища распределяются по данным сущностям.

Раскройте структуру процесса Продажи.

В основе хранилища данных лежат процессы. Процессов может быть много, их количество зависит от информации, необходимой для решения поставленных бизнес-задач. Данные, находящиеся в каждом процессе, являются основой для решения бизнес-задач, базой для принятия маркетинговых и управленческих решений, или вспомогательными расчетами.

Объекты, входящие в процесс, разделены на 3 группы: атрибуты, измерения, факты.

Откройте список ссылок на измерения процесса. В процессе могут содержаться ссылки только на существующие в хранилище измерения.

Кроме того, любое измерение может входить в один или несколько процессов.

Каждый процесс есть «звезда» или «снежинка». На рисунке отображена структура процесса Продажи. В примере процесс Продажи

содержит историю продаж различных товаров по дням в нескольких торговых отделах (аптеках).

При такой структуре ХД предполагается, что уникальность точки в пространстве определяется совокупностью измерений. В разрезе примера уникальность будет обеспечиваться совокупностью измерений: Дата+Товар+Отдел+Час покупки.



Рисунок 11 - Процесс Продажи

Если в одной и той же аптеке в один и тот же день и час будет совершено несколько покупок например, препарата Анальгин, то в хранилище данных попадает только одна агрегированная по фактам запись, например, общая сумма и количество покупок.

Для более наглядного понимания взаимоотношений измерений, атрибутов и фактов внутри процесса продаж, представим многомерное пространство. Каждое измерение процесса, это ось в многомерном пространстве.

В связи с тем, что визуально можно представить только трехмерное пространство, предположим, что наш процесс состоит всего из трех измерений: Дата, Отдел.Код и Товар.Код.

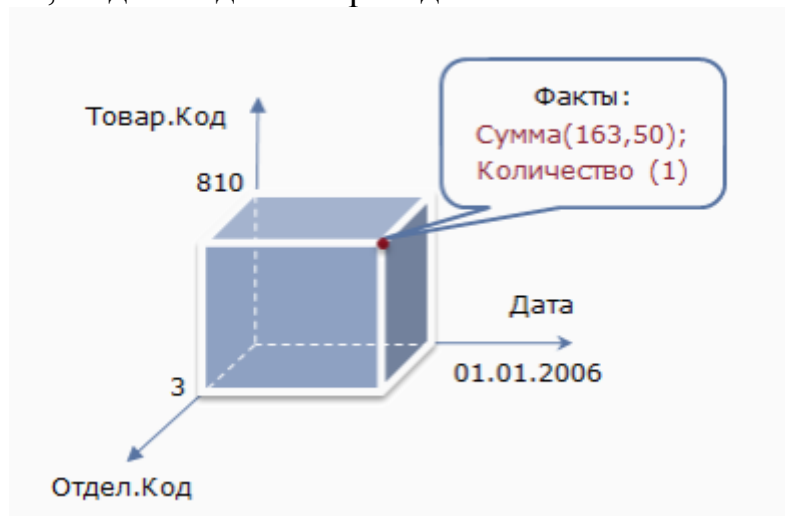


Рисунок 12 – Измерения процесса Продажи

Точка многомерного куба указывает, что 01.01.2006 в Отдел.Код=3 было приобретено товара с кодом 810 на сумму 165,50 у.е. в количестве 1 шт.

Раскройте дерево метаданных хранилища данных.

Рассмотрим список измерений. Измерение Товар.Код имеет ссылку на измерения Группа.Код, это означает, что между этими измерениями существует иерархия. Аналогично Товар.Код имеет ссылки на измерения Категория ABC и Категория XYZ. Зная товар, всегда можно узнать его товарную группу, категорию ABC и XYZ.

Дополним структуру процесса иерархией измерений.



Рисунок 13 – Иерархия измерений процесса Продажи

Структура хранилища данных проектируется на этапе консолидации. Процедуры изменения наполненного хранилища достаточно сложны.

Три измерения из списка имеют по одному атрибуту:

Группа.Код – Группа.Наименование

Товар.Код – Товар.Наименование

Отдел Код – Отдел.Наименование

Измерения могут не иметь ни атрибутов, ни ссылок на другие измерения.

В хранилище имеется еще один объект – куб Продажи за последние 3 дня.

Куб в Deductor Warehouse – это заранее подготовленный срез из ХД. Использование куба оправдано в случае, когда нужно добиться высокой скорости получения ответа на какой-либо сложный запрос из хранилища. Каждый куб, по сути, представляет собой дополнительную таблицу в хранилище данных. Эта таблица формируется в момент загрузки новых данных в ХД, либо может быть создана по команде пользователя. Скорость доступа к информации в кубах максимально быстрая. Каждый раз при пополнении хранилища куб потребуется формировать заново.

Настроим импорт продаж товара из хранилища за последние 3 месяца от имеющихся данных. Процесс получения данных из хранилища осуществляется при помощи Мастера импорта.

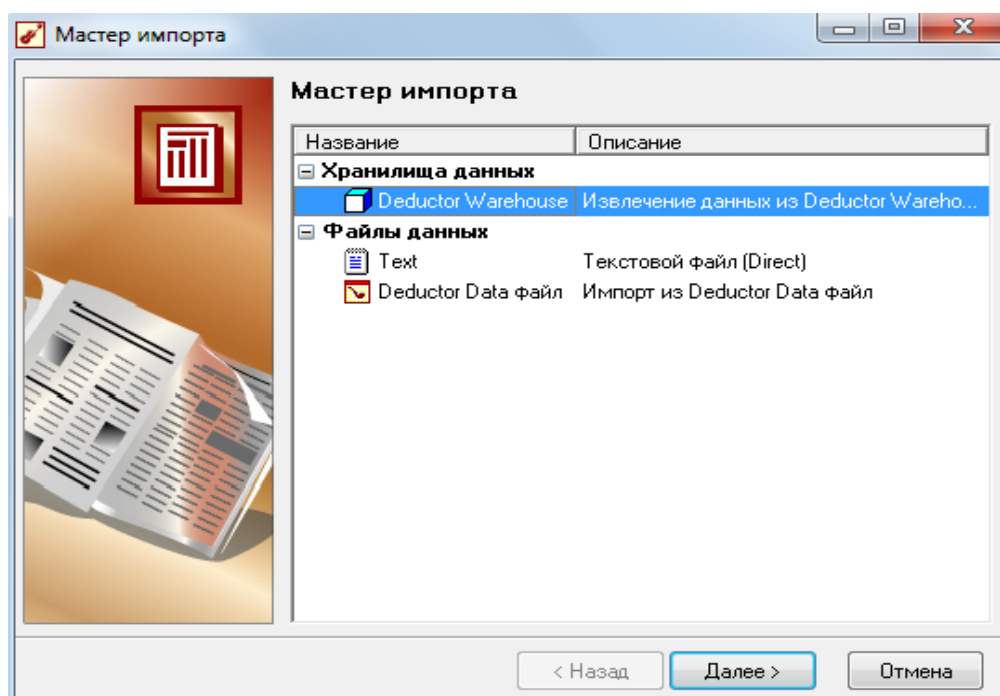


Рисунок 14 - Мастер импорта

Выбираем из списка подключение к Deductor Warehouse. Выбираем из списка хранилище данных, из которого будет осуществляться импорт.

Перед нами окно выбора объекта хранилища данных, из которого будет выполняться импорт. Выбрать можно только один объект.

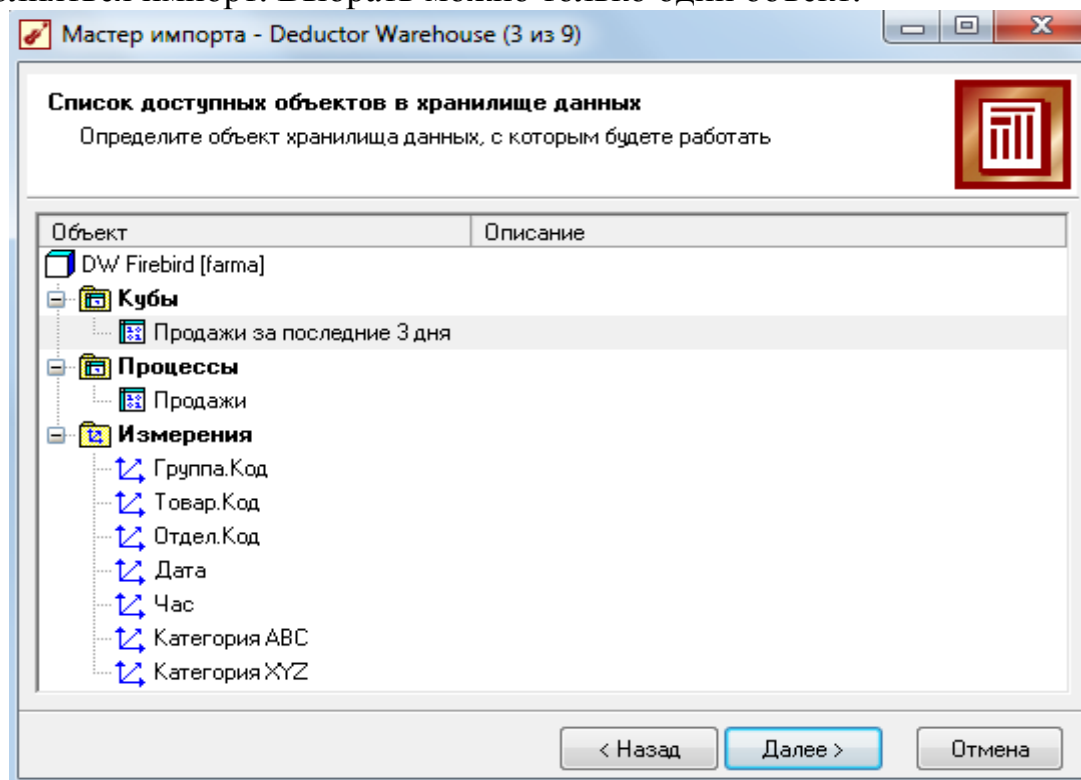


Рисунок 15 - Мастер импорта (продолжение)

Раскроем список всех доступных к импорту измерений процесса Продажи, для этого выберем в списке группу Измерения и нажмем клавишу *. В раскрывшемся дереве можно увидеть, что для импорта доступны сами измерения, их атрибуты и иерархия.

Установите флаги выбора у следующих измерений из списка: Отдел.Наименование; ТоварНаименование; ГруппаНаименование.

При выборе факта дополнительно указывается вид агрегации. Оставьте его по умолчанию.

По умолчанию всегда поставляется вид агрегации – Сумма. Выбираем для факта Сумма вид агрегации – Сумма. Нажимаем кнопку Далее.

В открывшемся окне настройки срезов зададим правила выбора данных. По умолчанию устанавливается фильтр Статистический. Данный тип фильтра обеспечивает импорт данных из хранилища при выполнении узла импорта по заданным правилам без участия пользователя. Для поля Дата зададим правило выбора данных – «3 месяца от имеющихся данных». В списке условий фильтрации выберем условие – последний.

После установки условия стала доступна кнопка выбора значений. Выберем точку отсчета – Имеющиеся данные. При этом нужно иметь в виду, что отбор будет происходить не от последних данных, хранимых в процессе, а от последних данных, хранимых в измерении Дата.

Выполнился импорт продаж за последние 3 месяца от имеющихся данных. Создадим еще один срез, в котором пользователь может задавать параметры импорта процесса продаж. Для этого чтобы не повторять настройки узла, скопируем имеющийся узел импорта и перенастроим его.

На шаге определения срезов выбираем тип фильтра Пользовательский. При запуске узла импорта с активным пользовательским фильтром пользователю каждый раз будет выводиться диалоговое окно, в котором нужно будет установить условия фильтрации. Данная настройка позволяет делать динамические отчеты, в которых пользователь задает конкретные условия фильтрации к заранее определенной информации в момент запроса среза.

Завершаем перенастройку узла импорта и переходим к результатам.

Запустите второй узел на пересчет данных. Открылось окно установки параметров фильтрации, в котором можно изменить имеющиеся условия фильтрации и создать новые. Данное окно будет появляться всегда при запуске этого узла.

Задание для самостоятельной работы:

1. Имеются текстовые структурированные файлы с данными подразделения розничного кредитования банка, описание структуры которых приведено ниже.

Файл «Филиалы.txt»

Код филиала	Название филиала
1	Краснодарское отделение №8619
2	Доп.офис №8619/0127
3	Доп.офис №8619/0129
4	Доп.офис №8619/0131

Файл «Отделения.txt»

Код отделения	Наименование отделения	Код филиала
1	Отделение №1	1
2	Отделение №2	2
3	Отделение №3	3
4	Отделение №4	4

Файл «Тип кредита.txt»

Тип кредита.Код	Тип кредита.Наименование
11	Автокредит
12	Потребительский кредит
13	Ипотечный кредит
14	Кредит на любые цели

Файл «Решение по кредиту.txt»

Решение по заявке.Код	Решение по заявке.Наименование
0	Заявка отклонена
1	Заявка одобрена

Файл «Статус заявки.txt»

Статус заявки.Код	Статус заявки.Наименование
0	Успешное прохождение этапов проверки
1	Отказ по этапу «Андеррайтинг»
...	...

Файл «Заявки.txt»

Заявка	Дата рождения	Пол	Количество детей	Семейное положение	Доходы (мес)	Статус заявки.Код	Тип кредита.Код
15271	04.11.1965	ж	1	Разведен(а)	16 000	0	13
15272	19.05.1984	м	0	Холост/ не замужем	7 500	0	13
...

Файл «Прием заявок.txt»

Заявка	Код отделения	Дата	Час	Заявленная сумма
15271	1	01.07.2014	10	1 500 000
15272	1	01.07.2014	12	1 000 000
...

Файл «Заклученные договора.txt»

Номер договора	Дата	Сумма договора
10987	01.07.2014	1 700 000
10988	01.07.2014	1 200 000
...

Файл «Качество обслуживание долга.txt»

Дата	Номер договора	Суммарные дни просрочки
10.09.2014	10987	0
10.09.2014	10988	1
...

Представленные данные имеют следующие иерархии:

1. Филиал-Отделение.
2. Решение по заявке-Статус заявки-Заявка-Номер договора.
3. Тип кредита-Заявка.
4. Срок кредита-Номер договора.

Последовательность действий:

1. Создайте пустое хранилище данных, назвав его SBER_<Фамилия>.gdb, а метку хранилища – BANK.

2. Спроектируйте структуру хранилища данных в редакторе метаданных. Используйте представленную выше информацию и все имеющиеся в наличии структурированные текстовые файлы (не забудьте про иерархии измерений). В результате должно получиться пустое хранилище данных с готовым семантическим слоем в виде файла SBER.gdb.

3. Сделайте копию пустого файла хранилища, назвав его SBER_1_<Фамилия>.gdb.

4. Напишите сценарий загрузки данных в хранилище SBER. Сохраните сценарий под именем load_<Фамилия>.ded.

5. Создайте три любых среза из хранилища данных, причем один с использованием динамического фильтра, впишите их названия в метки узлов импорта и сохраните все в сценарии Rolap_<Фамилия>.ded.