**Лекция 10:**

**Метаданные в хранилищах данных**

**Цель лекции**

Изучив материал настоящей лекции, вы будете знать:

* что такое *метаданные* в хранилищах данных;
* функции *метаданных* в хранилище данных;
* *элементы метаданных* хранилища данных;
* классификацию *метаданных* хранилища данных;
* что такое модель *метаданных* для хранилища данных;

и научитесь:

* составлять описание *метаданных* фактов, измерений, источников данных;
* проектировать *метаданные* для хранилищ данных.

**Литература**: [[2](https://intuit.ru/studies/courses/599/455/literature#literature.2)], [[3](https://intuit.ru/studies/courses/599/455/literature#literature.3)], [[27](https://intuit.ru/studies/courses/599/455/literature#literature.27)], [[66](https://intuit.ru/studies/courses/599/455/literature#literature.66)], [[68](https://intuit.ru/studies/courses/599/455/literature#literature.68)].

**Метаданные**

К сожалению, часто встречающееся на страницах компьютерной литературы *определение* ***метаданных*** — это "данные о данных" — более вносит путаницу в толкование термина " *метаданных* " (*metadata*), чем поясняет его смысл. Образное *определение* " *Метаданные* — это "тень данных", принадлежащее Б. Инмону и понятное ИТ-специалистам по ХД, также не вносит большой ясности.

Давайте попробуем уточнить смысл термина " *метаданные* ". *метаданные* есть в любой ИС с *БД*, будь то OLTP-система, система складирования данных или корпоративный портал. Чтобы осознать этот факт, нужно вспомнить о том, что любая ИС реализует "вопросно-ответное" *отношение* на конечном алфавите [[62](https://intuit.ru/studies/courses/599/455/literature#literature.62)]. *Метаданные* позволяют пользователям понять, на какие вопросы может отвечать данная ИС.

С технической точки зрения *метаданные* — **это совокупность спецификаций и данных, которая в целом дает ответы на вопросы, какова степень охвата предметной области в ИС, какие данные в ней представлены, какова архитектура системы и т.д**.

В частности, *метаданные* содержат *семантическую интерпретацию* или толкование содержания элементов данных, циркулирующих в ИС. Но это далеко не все. В *метаданные* включается также описание вычислительной среды, предметных областей, информационной безопасности и многое другое, что непосредственно влияет на эффективность использования ИС, в первую *очередь*, конечными пользователями и разработчиками.

Далее **под метаданными будет пониматься совокупность элементов данных и спецификаций, содержащих описание данных ИС и процессов их обработки**.

Неоднозначность толкования термина " *метаданные* " определяется тем обстоятельством, что последние должны удовлетворить технические и семантические потребности всех групп пользователей ИС. Каждая *группа пользователей* имеет свои требования к *метаданным*.

Так, руководство компании (основные пользователи информационных систем руководителей) хочет знать, что оно может получить от системы, как быстро оно получит ответ на интересующий вопрос, и желательно – в терминах, понятных лицам, которые принимают решения. Руководство организации не имеет времени, да и не понимает, зачем ему изучать объемные инструкции по эксплуатации ИС (что совершенно правильно, поскольку руководство организации решает стратегические и тактические задачи организации, а не профессиональные задачи реализации автоматизированных информационных систем).

В то же время специалисты организации, например пользователи бухгалтерских систем, хотят, чтобы такая система "разговаривала" на их профессиональном языке, вплоть до того, что разработчики таких систем (как, например, 1С-бухгалтерия) оснащают свои системы специальным, формальным языком, понятным бухгалтерам. Следует заметить, что бухгалтеры вряд ли будут основательно изучать *SQL*.

Аналитиков компании занимают более сложные вопросы, в частности, о происхождении и достоверности данных. Руководство организации требует от них информации для поддержки принятия обоснованных решений. Цена ошибки может быть велика: значительные убытки или сорванные контракты.

Разработчиков приложений интересует *информация* о модели данных ИС для создания или внедрения дополнительных бизнес-приложений. Они хотят знать, что находится в таблицах *БД* системы и в каком формате.

*Отображение* столь многообразных интересов в *метаданных* порождает большое число элементов, которые составляют *метаданные*. Проектирование и разработка *метаданных* являются одной из самых сложных и трудоемких задач проектирования и разработки ИС.

Разработку *метаданных* можно отнести к ИТ-дисциплине, которую называют "*управление данными*". Решение задач управления данными часто возлагается на *администраторов данных*, которых не следует путать с администраторами *БД* и компьютерных сетей.

Нужно отметить, что некоторые принципы управления данными в ХД и *БД* OLTP-систем имеют существенные отличия. Так, например, характерный для OLTP-систем принцип, состоящий в том, что существует только одно правильно отображающее семантику *определение* данных в системе, не является верным для ХД. Это связано с различными временными горизонтами данных в системах, основанных на ХД.

При проектировании *метаданных* задача проектировщика ХД состоит в:

* идентификации объектов ХД и их атрибутов;
* идентификации источников данных;
* описании семантики данных источников и ХД;
* описании алгоритмов преобразования и агрегации данных;
* описании путей доступа к данным и т.п.

Весь этот комплекс вопросов, которые должен решить проектировщик ХД при проектировании *метаданных*, требует от него достаточно тщательной их проработки еще на начальных стадиях проектирования. Проектирование ХД должно начинаться с проектирования *метаданных* и заканчиваться им же.

Рассмотрим основные функции *метаданных* и их состав, характерный для ХД.

**Функции метаданных в хранилище данных**

Роль *метаданных* для ХД значительно важнее, чем в *системах операционной обработки данных*. Если в *системах операционной обработки данных* *интерфейс* системы настроен на бизнес-процедуры обработки данных конкретными специалистами и понятен им после специального обучения, то *интерфейс* систем складирования данных конструируется таким образом, чтобы помимо всего прочего отвечать на непредопределенные вопросы (*ad hoc*). Как правило, такие вопросы формулируются в терминах *предметной области* и бизнес-процессов, к тому же специалистами, для которых ИТ-технологии не являются основной профессией: аналитиками, менеджерами среднего и высшего уровня.

Таким образом, одним из главных аспектов использования *метаданных* в ХД является их предметная ориентация. Основные вопросы, на которые должны ответить *метаданные*, — это какие данные представлены в системе и как их получить в нужном для анализа данных виде.

Первой основной *функцией метаданных* в ХД является *представление соответствия данных источников и данных ХД*. Как правило, это описание представляет собой фиксацию взаимосвязи атрибутов данных источника и атрибутов данных ХД, правила преобразования первых во вторые, изменение в наименовании данных, в их физических характеристиках и т. д.

Такая *информация* позволяет идентифицировать источники данных для ХД, правильность данных в ХД и их *корректность*.

Вторая основная *функция метаданных* в ХД — *управление данными во времени*. Время жизни данных в ХД, как правило, 5-10 лет, а то и более. А для систем операционной обработки данных время жизни данных — от нескольких дней до нескольких месяцев. Затем данные архивируются в случае необходимости.

Таким образом, временной горизонт данных в ХД гораздо больше, и это обстоятельство изменяет коренным образом некоторые принципы управления данными. Например, в *системах операционной обработки данных* в одно и то же время существует только одно корректное *определение* данных. Для ХД это не так.

*Структура данных* (схема или *модель данных*) в *системах операционной обработки данных* меняется во времени, т.е. данные в таких системах в разное время имеют различные формы представления. Хронология таких изменений должна быть зафиксирована в ХД.

Таким образом, в ХД в одно и то же время может существовать несколько схем данных, отвечающих различным периодам эволюции источников данных. *Запись* о таких структурных изменениях сохраняется в *метаданных* ХД. На основании записей аналитики получают ответы на вопросы, какими данными и за какие периоды они располагают.

Третья, и немаловажная, *функция метаданных* в ХД — это *поддержка версионности*. Эта *функция* тесно связана с управлением данными с большим временным горизонтом. *Метаданные* должны отражать изменения внутренней структуры данных источников и, следовательно, должны сами изменяться, для того чтобы обеспечить непрерывность истории изменения структуры данных ХД.

Таким образом, *поддержка* версионности *метаданных* позволяет в каждый момент времени в прошлом обеспечить правильное описание модели данных, а аналитики получают возможность знать, какие данные, когда и как попали в ХД.

Четвертая основная *функция метаданных* в ХД — это *интерпретация данных в терминах бизнес-пользователей*. *Метаданные* должны поддерживать в запросах понятную для пользователя терминологию, независимо от того, какие правила наименования атрибутов были использованы проектировщиком ХД.

Пятая основная *функция метаданных* — *обеспечение открытости* (доступности другим информационным системам) *системы складирования данных для ее интеграции с другими аналитическими системами организации*. *Опрос* *метаданных* ХД другой системой позволяет последней выяснить структуру данных ХД и поддерживать *обмен данными* между системами.

На [рис. 14.1](https://intuit.ru/studies/courses/599/455/lecture/10173?page=1#image.14.1) просуммированы основные функции *метаданных* для ХД.



**Рис. 14.1.**Основные функции метаданных для хранилищ данных

**Состав метаданных в хранилище данных**

В настоящее время нет строго определенных требований к составу *метаданных* информационных систем, в том числе использующих ХД. О предлагаемых стандартах *метаданных*, и в частности для ХД, мы поговорим несколько позже. Сейчас остановимся на перечне тех *элементов* или компонентов *метаданных*, которые обязательно должны присутствовать в ХД.

Базовые компоненты *метаданных* ХД не сильно отличаются от базовых *компонент* систем операционной обработки данных. Это **описание таблиц, их атрибутов, ключей** и т.д. Существенное отличие для ХД — *поддержка* версионности *метаданных*. Базовые компоненты говорят нам, какие данные сохраняются в ХД.

Следующая, характерная для ХД, *группа* компонентов *метаданных* — **описание преобразований**. Как правило, описание преобразований данных для ХД включает в себя:

* идентификацию полей источников данных;
* соответствие между атрибутами сущностей источников данных и атрибутами объектов ХД;
* преобразования атрибутов;
* физические характеристики преобразований;
* преобразования *таблиц кодировки* и ссылочных таблиц;
* изменения наименований (соответствие имен источников и объектов ХД);
* изменение ключевых атрибутов;
* значение полей по умолчанию;
* логика (алгоритмы) формирования данных ХД из нескольких источников (приоритетность источников);
* алгоритмы трансформации данных и т. д.

Компоненты преобразования говорят нам о том, как данные в ХД были получены.

Немаловажным компонентом *метаданных* ХД является история поступления в него данных. *Компонент* *метаданных* — **история экстрагирования данных** — говорит нам о том, когда данные поступили в ХД, а также позволяет судить о полноте представления данных в ХД. Для проведения анализа данных такая *информация* является очень важной, поскольку на ее основе формируются утверждения пользователей о корректности анализа данных и надежности его результатов.

*Информация* о синонимах, или **терминологические соответствия понятий, — это еще один компонент метаданных ХД**. Он включает в себя *альтернативные* наименования (алиасы) для данных ХД. Такая *информация*, как правило, делает ХД более "дружелюбным" для пользователей.

Следующим важным компонентом *метаданных* является **информация о состояниях и статистике использования данных ХД**. Эта *информация* составляет основу для оптимизации производительности ХД, и к ней относятся данные о числе строк в таблицах, скорости роста таблиц, статистический профиль использования таблиц (среднее и максимальное число запросов на день), *статистика* архивирования и удаления данных, *индексирование таблиц*, частота использования индексов в запросах и т.п.

Еще одним компонентом *метаданных* ХД являются **алгоритмы агрегации и суммирования данных, критерии выборки из источников, правила преобразования данных источников перед загрузкой в ХД, описание взаимосвязей между объектами ХД, их кардинальность** и т.п. Такая *информация* играет важную роль при проведении анализа данных и часто требуется аналитикам для решения вопросов надежности результатов анализа.

*Информация* о том, кто отвечает за **содержание и актуальность различных источников данных**, составляет еще один *компонент* *метаданных*. Эта *информация* важна для группы сопровождения ХД и позволяет организационно решать вопросы качества, точности и надежности данных в ХД.

Часто в *метаданные* включаются компоненты, описывающие шаблоны доступа к данным (когда и как данные мигрировали на другой уровень хранения). Они используются также для оптимизации физического потока данных в ХД и для оптимизации производительности.

Иногда алгоритмы обработки данных в ХД используют информацию об объектах внешних систем — так называемые таблицы расширения (таблицы кодировок и электронных справочников). В этом случае в *метаданных* ХД необходимо фиксировать описание таких таблиц и историю их изменения, поскольку в случае изменения кодов необходимо провести соответствующие изменения в обработке данных ХД, чтобы не потерять исторические связи в данных.

Часто проектировщики ХД включают в состав *метаданных* дополнительную информацию, важную с их точки зрения.

Из сказанного выше ясно, что проектирование *метаданных* ХД является достаточно сложной и креативной задачей для проектировщика ХД, решение которой требует часто литературного мастерства, знания *предметной области* ХД и *много времени*.

На [рис. 14.2](https://intuit.ru/studies/courses/599/455/lecture/10173?page=2#image.14.2) приведены основные *элементы метаданных* для ХД.



**Рис. 14.2.**Основные элементы метаданных для хранилищ данных

**Пример представления метаданных для хранилища данных**

Рассмотрим, как можно описать *метаданные* на примере киоска данных, предназначенного для анализа продаж некоторой гипотетической компании. Предположим, что компания занимается производством и реализацией своей продукции. *Киоск данных* используется аналитиками компании для детального изучения взаимосвязи расходов и доходов компании от реализации продукции и подготовки отчетности о продажах для руководства.

Допустим, что наша гипотетическая компания открыла сеть точек продаж (склады розничной и оптовой торговли) и организовала сеть магазинов, т.е. расширила свою деятельность. Руководство компании хочет оценить эффективность сделанного расширения и иметь более подробную информацию о зависимости между продажами и производством по затратам и доходам.

Далее допустим, что компания выпускает около 200 видов (моделей) некоторой продукции. Каждый продукт имеет базовый набор комплектующих компонентов. Дополнительные комплектующие компоненты используются для создания специфической модели продукта. Рыночная политика компании строится таким образом, что число выпускаемых моделей остается постоянным. Это означает, что количество новых моделей приблизительно равно количеству моделей, снятых с производства.

Для каждой модели каждого продукта в зависимости от спроса применяется гибкая система скидок. Как правило, размер скидки для покупателей больших партий продукции определяет заведующий складом розничной продажи.

Когда принято решение приостановить производство продукции данной модели, информация о ней сохранятся в БД компании в течение 6 месяцев после того, когда вся оставшаяся продукция будет реализована или списана. Данные о продукции удаляются в тот момент, когда удаляются данные о последней модели этой продукции.

Компания поддерживает два способа реализации продукции: через магазин оптовой торговли и через магазин розничной торговли. Магазин оптовой торговли продает товар только оптовым покупателям. Покупатель считается оптовым, если он покупает более 20 партий товара в год. Оптовый покупатель может предоставлять счет либо непосредственно в магазин оптовой торговли, либо по факсу в центральный офис компании. Любой покупатель может покупать в нескольких магазинах компании.

Магазин розничной торговли продает за наличный расчет. Независимо от предоставления скидок, цена товара меняется. Хотя на каждую продажу продукции оформляется счет, компания не ведет учет покупателей в розничной продаже.

*Киоск данных* нашей компании предназначен для решения задач анализа показателей расхода и дохода. Типовые запросы, на которые система должна давать ответы, следующие.

1. Какова величина общих издержек и общей прибыли по каждой модели товара, проданной сегодня, и просуммированной по точкам продажи, типу точки продажи, по региону и по складам оптовой торговли?
2. Какова величина общих издержек и общей прибыли для каждой модели товара, проданной сегодня, и просуммированной по заводам и по регионам?
3. Какой процент моделей получил скидки и какие из моделей были проданы по факту со скидкой (в процентах) в магазинах розничной продажи – для всех продаж на этой неделе? В этом месяце?
4. Для каждой модели товара, проданной в текущем месяце, определить, какой был процент продаж с розничной торговли, с оптовой торговли по безналичному расчету, с оптовой торговли с продавцами.
5. Какие модели и какого типа не продавались в течение последнего месяца? В течение последней недели?
6. Какие пять моделей, проданных за последний месяц, принесли наибольшую прибыль? По продажам за квартал? По всем продажам?

Источником данных для киоска данных является фрагмент БД *системы операционной обработки данных* компании. Одна из возможных структур данных киоска данных, полученная в результате проектирования, приведена на рис. [рис. 14.3](https://intuit.ru/studies/courses/599/455/lecture/10173?page=2#image.14.3).



[увеличить изображение](https://intuit.ru/EDI/20_07_20_2/1595197216-9970/tutorial/632/objects/14/files/09_03.jpg)
**Рис. 14.3.**Многомерная модель киоска данных для анализа продаж компании

Рассмотрим описание *метаданных* для такого киоска данных. Отметим, что приведенное описание является примером одного из возможных подходов, его нельзя считать полным и законченным.

**Логическая структура метаданных хранилища данных**

**Логическая структура метаданных модели**

В этом разделе приводятся логические схемы *метаданных* для ХД для взятого нами примера. Пример не претендует на полноту, но дает ясное представление о подходах к описанию *метаданных*.

Логическая структура модели *метаданных* может быть следующей.

|  |  |
| --- | --- |
| **Имя**: | Продажи (Sales). |
| **Определение**: | Модель *метаданных* содержит метаописание данных о продажах компании для каждого вида продукции, в соответствии с каждым оплаченным счетом, на ежедневной основе. |
| **Назначение**: | Назначением данной модели является предоставление аналитикам и руководству компании возможностей для анализа продаж. |
| **Ответственное лицо за корректность данных**: | Региональный менеджер по продажам. |
| **Измерения**: | Customer (Покупатель), Manufacture (Производитель), Продукт (Product), Продавец (Seller) и Время (Time). |
| **Факты**: | Продажа (Sale). |
| **Метрики**: | Общие издержки (Total cost), Общий доход (Total revenue), Общее количество продаж (Total quantity sold) и Скидка (Discount amount). |

**Логическая структура метаданных фактов**

Логическая структура *метаданных* фактов может быть следующей.

**Имя**: Продажа (Sale).

**Определение**: Этот факт содержит данные о продаже для каждого заказа, который был зафиксирован в оперативной системе обработки заказов для каждого склада розничной и оптовой торговли.

**Альтернативное имя**: Нет.

**Частота загрузки**: Ежедневно.

**Статистика загрузки данных**

* Дата последней загрузки.
* Число загруженных строк.

**Статистика использования данных**

* Среднее число запросов в день.
* Среднее число выбранных записей на запрос.
* Среднее время выполнения запроса.
* Максимальное число запросов в день.
* Максимальное число выбранных записей в запросе.
* Максимальное время выполнения запроса.

**Правила архивирования данных**: Данные будут архивироваться по истечении 36-ти месяцев на ежемесячной основе.

**Статистика архивирования**:

* Последняя дата архивации.

**Правила удаления данных**: Данные будут удаляться по истечении 48-ми месяцев на ежемесячной основе.

**Статистика удаления**:

* Последняя дата удаления.

**Качество данных**: Допускаются ошибки персонала при комплектовании заказов. Однако записи, представленные в БД, являются точными.

**Точность данных**: Метрики этого факта являются на 100% точными, поскольку представляют уже осуществленные продажи.

**Гранулированность измерения "Время"**: Метрики данного факта представляют продажу данного товара по данному заказу.

**Ключевое поле**. Ключом для факта продажи является комбинация ключей измерений: Покупатель (Customer), Производитель (Manufacture), Продукт (Product), Продавец (Seller) и Время (Time).

**Метод генерирования ключа**: Временная часть ключа есть просто дата продажи товара. Ключ товара, ключ производителя, ключ продавца и ключ покупателя выбирается из справочников оперативной БД компании.

**Источники**

* **Наименование**: Таблица заказов (Order Table).
* **Правила преобразования**: Строки из таблицы заказов копируются в таблицу фактов продаж на ежедневной основе.
* **Критерий выборки данных**: Выбираются строки, для которых заказ был завершен на текущую дату.
* **Наименование**: Измерение "Продукт" (Product Dimension).
* **Правила вычисления значения**: Измерение "Продукт" используется для вычисления стоимости модели, проданной в конкретном заказе. Заводская стоимость единицы товара сравнивается с закупочной или отпускной ценой, чтобы определить, была ли дана скидка. Если скидка имела место, то вычисляется ее размер.
* **Критерий выборки**: Перед вставкой строки в таблицу фактов обрабатываются данные о товаре.

**Метрики**: Общая стоимость (Total cost), Общая прибыль (Total revenue), Общее количество продаж (Total quantity sold) и Скидка (Discount amount).

**Измерения**: Customer (Покупатель), Manufacture (Производитель), Продукт (Product) , Продавец (Seller) и Время (Time).

**Сотрудник, ответственный за данные**: Директор завода-производителя.

**Логическая структура метаданных измерений**

Логическую структуру *метаданных* для измерений приведем на примере измерений "Покупатель" и "Время". Она может быть следующей.

*Для измерения "Покупатель"*

**Имя**: Покупатель (Customer).

**Определение**: Покупатель — это любое физическое или юридическое лицо, которое приобретает продукцию компании. Покупатель может приобретать товары в нескольких точках продаж компании.

**Альтернативное имя**: нет.

**Иерархия измерения**: Данные по этому измерению могут суммироваться на двух уровнях. Первый уровень суммирования (нижний) есть адрес *отгрузки товара* покупателю. Данные по каждому адресу юридического лица могут быть позднее просуммированы по каждому покупателю.

**Правила изменения**: Адреса *отгрузки товара* по каждому юридическому лицу вставляются как новые строки в измерение. Изменение существующих адресов покупателей выполняется обновлением непосредственно в таблице измерения.

**Частота загрузки**: Ежедневно.

**Статистика загрузки**

* Последняя дата загрузки.
* Количество загруженных строк.

**Статистика использования**

* Среднее число запросов за день.
* Среднее число выбранных строк на запрос.
* Среднее время выполнения запроса.
* Максимальное число запросов за день.
* Максимальное число выбранных строк на запрос.
* Максимальное время выполнения запроса.

**Правила архивации**: Данные этого измерения не архивируются.

**Статистика архивации**:

* Дата последней архивации.

**Правила удаления**: Покупатели, которые не приобретали продукцию компании в течение последних 5-ти лет, удаляются из таблицы измерения на ежемесячной основе.

**Статистика удаления**:

* Дата последнего обновления.

**Качество данных**: Когда новый покупатель добавляется в измерение, выполняется поиск, чтобы определить, не было ли продаж товара данному покупателю по другому адресу. Независимо от того, были ли такие продажи, покупатель с новым адресом *отгрузки товара* вставляется как новая строка.

**Точность данных**: Допускается пятипроцентная неточность в определении связей между покупателем и его адресами отгрузки.

**Ключ измерения**: Сгенерированное системой число, которое идентифицирует покупателя.

**Метод генерации ключа**: Когда запись о покупателе копируется из подающей системы, выполняется проверка на присутствие покупателя в ХД. Если такого покупателя нет в ХД, новый идентификатор генерируется и запись вставляется в измерение.

**Источники**

* **Имя (Name)**: Таблица "Покупатель" (Customer).
* **Правила преобразования**: Строки из таблицы "Покупатель" подающей системы копируются ежедневно.
* **Критерий выборки**: Выбираются только новые или модифицированные на текущую дату строки.
* **Имя**: Таблица "Адреса покупателей" (Customer Location).
* **Правила преобразования**: Строки из таблицы "Адреса покупателей" копируются ежедневно в таблицу измерения. Для существующих адресов покупателей адрес отгрузки обновляется. Для новых адресов покупателей ключ генерируется и записи вставляются.
* **Критерий выборки**: Выбираются только те записи, которые на текущую дату были обновлены или добавлены.

**Атрибуты**

* **Имя**: Идентификатор покупателя (Customer Key)
* **Определение**: Это есть произвольно выбранное число, гарантирующее уникальность каждого покупателя и его адреса.
* **Правила изменения**: После вставки в измерение значение этого атрибута никогда не изменяется.
* **Тип данных**: Числовой.
* **Домен**: 1 - 999999999
* **Правила вычисления значения**: Сгенерированный системой ключ.
* **Источник**: Генерируется системой.
* **Имя**: Наименование (Name).
* **Определение**: Наименование, под которым покупатель известен компании.
* **Правило изменения**: При изменении наименования покупателя оно обновляется для всего этого измерения.
* **Тип данных**: Символьный.
* **Домен**: Допустимая строка символов.
* **Правила вычисления значения**: Для того чтобы различать покупателей из разных организаций с одинаковым названием, к названию организации будет добавляться число.
* **Источник**: Поле "Наименование" (Name) из таблицы покупателей (Customer) подающей системы.
* **Имя**: Адрес отгрузки (*Ship*-to Address).
* **Определение**: Для юридических лиц — это адрес, по которому отгружается товар. Допускается, что одно юридическое лицо может иметь несколько адресов отгрузки. Для физических лиц и розничных покупателей это поле не поддерживается. Таким образом, для таких покупателей в таблице измерения поддерживается только одна запись.
* **Правила изменения**: При изменении адреса отгрузки выполняется обновление этого значения в измерении.
* **Тип данных**: Символьный.
* **Домен**: Запись адреса в допустимом формате.
* **Правила вычисления значения**: Адрес отгрузки копируется из таблицы источника.
* **Источник**: Поле "Адрес отгрузки" (*Ship*-to Address) из таблицы "Адреса покупателей" (Customer Location) подающей системы.

**Факты**: Продажа (Sale).

**Метрики**: Общая стоимость (Total cost), Общая прибыль (Total revenue), Общее количество продаж (Total quantity sold) и Скидка (Discount amount).

**Ответственный за поставку данных**: Вице-президент по продажам и маркетингу.

*Для измерения "Время"*

**Имя**: Время (Time).

**Определение**: Измерение "Время" содержит моменты времени, когда компания фиксирует данные о продажах.

**Альтернативное имя**: Нет.

**Иерархия измерения**: Наименьший уровень суммирования данных есть день. Данные для этого дня могут быть просуммированы либо за неделю, либо за месяц.

**Правила изменения**: Записи вставляются в измерение один раз за текущий год. Никакие обновления в этом измерении не допускаются.

**Частота загрузки**: По мере необходимости.

**Статистика загрузки**

* Дата последней загрузки.
* Число загруженных строк.

**Статистика использования**

* Среднее число запросов за день.
* Среднее число выбранных строк на запрос.
* Среднее время выполнения запроса.
* Максимальное число запросов за день.
* Максимальное число выбранных строк на запрос.
* Максимальное время выполнения запроса.

**Правила архивации**: Данные этого измерения не архивируются.

**Правила удаления**: По истечении 5-ти лет данные будут удаляться на ежегодной основе.

**Статистика удаления**

* Дата последнего удаления

**Качество данных**: Никаких ошибок в данных этого измерения не предполагается.

**Точность данных**: Данные этого измерения всегда точны.

**Ключ измерения**: Ключ измерения "Время" есть дата в формате ГГГГММДД.

**Метод генерации ключа**: Дата, представленная в строке, используется как значение ключа.

**Источник**

* **Имя**: Календарь, поддерживаемый администратором.
* **Правила преобразования**: Все строки календаря вставляются один раз в год.
* **Критерий выборки**: Все строки выбираются.

**Атрибуты**

* **Имя**: Идентификатор (Time\_ID).
* **Определение**: Это есть дата в формате ГГГГММДД.
* **Альтернативное имя**: нет.
* **Правила изменения**: После вставки значение этого поля никогда не изменяется.
* **Тип данных**: Числовой.
* **Домен**: допустимое знание для даты.
* **Правила вычисления значения**: Дата есть копия значения источника.
* **Источник**: Числовое значение даты из календаря.
* **Имя**: Месяц (Month).
* **Определение**: Номер месяца в году.
* **Альтернативное имя**: нет.
* **Правила изменения**: После вставки значение этого поля никогда не изменяется.
* **Тип данных**: Числовой.
* **Домен**: 1-12.
* **Правила вычисления значения**: Значение копируется из источника.
* **Источник**: Номер месяца в году из календаря.
* **Имя**: Неделя (Week).
* **Определение**: Номер месяца в году.
* **Альтернативное имя**: нет.
* **Правила изменения**: После вставки значение этого поля никогда не изменяется.
* **Тип данных**: Числовой.
* **Домен**: 1-52.
* **Правила вычисления значения**: Значение копируется из источника.
* **Источник**: Номер недели в году из календаря.

**Факты**: Продажа (Sale).

**Метрики**: Общие издержки (Total cost), Общий доход (Total revenue), Общее количество проданного товара (Тotal quantity sold) и Скидки (Discount amount).

**Ответственный сотрудник**: Администратор ХД.

**Логическая структура метаданных для метрик**

Логическую структуру *метаданных* для метрик дадим на примере метрик "Общие издержки", "Общий доход" и "Общее количество продаж". Она может быть следующей.

**Имя**: Общие издержки (Total Cost).

**Определение**: Это есть стоимость всех компонент, используемых для создания данного вида (модели) продукции, которая была продана.

**Альтернативное имя**: нет.

**Тип данных**: Числовой.

**Домен**: 0.01 - 9999999.99

**Правила вычисления значения**: Общие издержки равны произведению стоимости единицы товара (модели) на количество проданных моделей.

**Статистика использования**

* Среднее число запросов в день.
* Максимальное число запросов в день.

**Качество данных**: Эта метрика формируется только исходя из стоимости комплектующих деталей на момент продажи данного вида товара. Никакие другие виды издержек на производство товара не учитываются.

**Точность данных**: Предполагается, что разброс значений в стоимости комплектующих деталей данного вида товара составляет +/- .5%.

**Факты**: Продажа (Sale).

**Измерения**: Покупатель (Customer), Производитель (Manufacture), Продукт (Product), Продавец (Seller) и Время (Time).

**Имя**: Общий доход (Total Revenue).

**Определение**: Общий доход равен произведению проданных единиц товара на отпускную цену этого товара на момент продажи.

**Тип данных**: Числовой.

**Домен**: 0.01 - 999999999.

**Правила вычисления значения**: Общий доход есть произведение отпускной цены модели товара на количество проданных моделей товара.

**Статистика использования**

* Среднее число запросов в день.
* Максимальное число запросов в день.

**Качество данных**: Эта метрика представляет количество проданных моделей товара.

**Точность данных**: С точки зрения построения трендов продаж и шаблонов поведения покупателей высокая точность данных не требуется.

**Факты**: Продажа (Sale).

**Измерения**: Покупатель (Customer), Производитель (Manufacture), Продукт (Product), Продавец (Seller) и Время (Time).

**Имя**: Общее количество продаж (Total Quantity Sold).

**Определение**: Это есть число проданных единиц моделей товара.

**Тип данных**: Числовой.

**Домен**: 1 - 9999999.

**Правила вычисления значения**: Это значение берется непосредственно из графы "количество" для каждой позиции счета.

**Статистика использования**

* Среднее число запросов в день.
* Максимальное число запросов в день.

**Качество данных**: Это поле представляет только количество проданного товара.

**Точность данных**: С точки зрения построения трендов продаж и шаблонов поведения покупателей высокая точность данных не требуется.

**Факты**: Продажа (Sale).

**Измерения**: Покупатель (Customer), Производитель (Manufacture), Продукт (Product), Продавец (Seller) и Время (Time).

**Логическая структура метаданных источников**

Логическая структура *метаданных* источников может быть следующей (на примере описания таблицы "Счет" из подающей системы).

**Имя таблицы**: Счет (Order).

**Метод извлечения данных**: В исходной таблице выбираются записи с законченными на текущую дату операциями для добавления в ХД.

**График извлечения данных**: Ежедневно по завершении рабочего дня.

**Статистика извлечения данных**

* Последняя дата экстрагирования.
* Число строк.

**Стандарты метаданных**

Как правило, любая стандартизация начинается с построения классификации объектов, для которого разрабатывается стандарт. Стандарт в области *метаданных* не является исключением из этого правила. Второй аспект разработки любого промышленного стандарта — это учет предложений ведущих производителей. И третий момент — концепция, которая позволяет связать разрабатываемый стандарт с уже действующими в данной *предметной области* стандартами.

Предметом обсуждения этого раздела будет спецификация "Общая *метамодель* хранилища данных" (Common Warehouse Metamodel, CWM), которая является одним из стандартов [66], использующих *XML*-технологии. Этот стандарт описывает обмен *метаданными* в информационных системах, применяющих технологии ХД, а также в *системах деловой осведомленности* (*Business Intelligence*) и системах управления знаниями (*Knowledge Management*).

**Классификация метаданных**

Как следует из нашего обсуждения *метаданных* в предыдущих разделах, на очень высоком уровне *метаданные* могут быть разделены на две категории: *разделяемые метаданные* (Shared meta data) и *уникальные метаданные* (Unique meta data).

Элементы *разделяемых метаданные* необходимы для точного определения объектов и их семантики для ХД в целом. Так, определение объекта "Покупатель" в нашем примере выше должно быть согласованным и для источников, и для ХД, т.е. иметь одинаковый смысл в рамках информационных систем масштаба организации. *Уникальные метаданные* описывают уникальные объекты системы, например, атрибуты измерений или фактов.

Другой подход к построению классификации *метаданных* — разделение *элементов метаданных* по их функциональному назначению:

* *предметно-ориентированные метаданные* (Business meta data);
* *структурные* *метаданные* (Structural metadata);
* *технические метаданные* (Technical meta data);
* *метаданные процесса обработки данных* (Process meta data).

***Предметно-ориентированные метаданные*** **содержат определения сущностей предметной области в терминах пользователей, логические отображения между данными на различных уровнях их представления в системе, описания словаря ХД, например БД**.

Примером *предметно-ориентированных метаданных* может служить описание какого-либо атрибута сущности предметной области ХД, как то: вес проданного или закупленного товара. Рассмотрим пример на [рис. 14.4](https://intuit.ru/studies/courses/599/455/lecture/10173?page=4#image.14.4).



**Рис. 14.4.**Предметно-ориентированные метаданные: одно имя, одинаковый смысл, различные единицы измерения

В двух киосках данных *системы складирования данных* имеются две таблицы фактов: "Счета покупок" и "Счета продаж". Каждый киоск отвечает за определенное направление деятельности организации. В обеих таблицах имеется атрибут с именем "Вес". Семантический смысл этого атрибута в обоих случаях одинаков: вес товара. Однако в одном *киоске данных* может использоваться единица измерения веса "кг", а в другом - "центнер".

**Cтруктурные метаданные содержат описание структуры различных объектов данных**. Они используются при реализации схем навигации для представления данных пользователям. Например, объект данных "Книга" состоит из элементов "Название", "Автор", "Содержание", "Предисловие", "Глава с тестом". Каждая глава имеет "Заголовок" и, возможно, ряд "Подзаголовков".

***Технические метаданные*** **содержат определения и данные о физических объектах ХД**. Это определения наименований таблиц и их колонок, ограничений, правил физических преобразований данных и т.п. Например, формат и длина поля "Имя покупателя" есть строка переменной длины до 100 символов.

***Метаданные процесса обработки данных*** **содержат информацию, связанную с процессом обработки данных, такую как статистика загрузки, расписание загрузки данных** и т.п.

Исследователями в области проектирования ХД предлагаются и другие подходы к построению классификации, но при этом, как правило, используются названные выше два принципа.

Основные типы *метаданных* приведены на [рис. 14.5](https://intuit.ru/studies/courses/599/455/lecture/10173?page=4#image.14.5).



**Рис. 14.5.**Типы метаданных

**Позиция производителей**

Ведущие производители программного обеспечения в области складирования данных ведут жесткую конкурентную борьбу за лидерство. И выдвижение своих решений в качестве промышленного стандарта для систем этого класса является неотъемлемой составляющей этой борьбы.

В середине 1998 года корпорации IBM, Oracle, Unisys, Hyperion, SAS, Meta Integratiom и ряд других поставщиков программного обеспечения представили в организацию Object Management Group (OMG) спецификацию стандарта "Обмен общими *метаданными* хранилища данных" (Common Warehouse Metadata Interchange, CWMI). Во второй половине 1999 года корпорация Microsoft передала на рассмотрение в консорциум Meta Data Coalition (*MDC*) разработанный ею стандарт "Открытая информационная модель" (Open Information Model, OIM).

В 2000 году произошло слияние *MDC* и OMG, а спустя год, в начале февраля 2001 года, была опубликована первая версия спецификации "Общая *метамодель* хранилища данных" [[67](https://intuit.ru/studies/courses/599/455/literature#literature.67)].

**Концепция стандарта**

CWMI определяет интерфейсы, которые могут быть использованы для обмена *метаданными* между ХД и аналитическими приложениями с помощью инструментальных средств ХД, программно-аппаратных платформ и репозиториев *метаданных* в распределенных гетерогенных вычислительных средах.

CWMI основывается на трех основных стандартах:

* UML — Unified Modeling Language (стандарт OMG для моделирования);
* MOF — Meta Object Facility (стандарт OMG для *метамоделирования* и репозитория *метаданных* );
* *XMI* — XML Metadata Interchange (стандарт OMG обмена *метаданными* ).

Стандарт UML определяет язык *объектно-ориентированного моделирования*, который поддерживает ряд графических нотаций. Стандарт MOF определяет гибкие средства для определения модели *метаданных* и обеспечивает программные средства для хранения и доступа к *метаданным* в репозитории. Стандарт *XMI* определяет спецификации для обмена *метаданными* в формате стандарта XML. Использование этих стандартов не накладывает сильных ограничений при реализации модели *метаданных* в конкретных *системах складирования данных*.

Перечисленные выше стандарты формируют ядро архитектуры репозитория *метаданных* OMG, как показано на [рис. 14.6](https://intuit.ru/studies/courses/599/455/lecture/10173?page=4#image.14.6).



**Рис. 14.6.**Ядро архитектуры репозитория метаданных OMG

**Общая метамодель хранилища данных (обзор)**

Основные элементы общей *метамодели* хранилища данных (CWM) включают в себя:

* четырехуровневую архитектуру *метамоделирования* при работе с *метаданными* в распределенных репозиториях;
* использование нотации UML для представления *метамодели* и моделей данных;
* использование стандартных информационных моделей UML для описания семантики *объектно-ориентированного анализа и проектирования* моделей;
* использование MOF для определения и работы с метамоделями с использованием интерфейсов CORBA;
* использование *XMI* для организации обмена *метаданными*.

Четырехуровневая архитектура *метамоделирования* аналогична общепринятой архитектуре моделирования, как показано в [таблице 14.1](https://intuit.ru/studies/courses/599/455/lecture/10173?page=4#table.14.1).

|  |
| --- |
| Таблица 14.1. Архитектура *метамоделирования* |
| **.Мета уровень** | **Уровень моделирования** |
| M3 | Мета-*метамодель* / мета-мета-метаданные (Meta-Metamodel / Meta-meta-metadata). Например, MOF класс, атрибут, ассоциация |
| M2 | *Метамодель* / мета-метаданные (Metamodel/Meta-metadata). Например, UML класс, атрибут таблицы CWM, колонка |
| M1 | Модель / метаданные (Model/Metadata). Например, Покупатель: Тип\_таблицы\_покупатель: Колонка |
| M0 | Данные / объект (Data/Object). "ООО" ("Флинт") |

Стандарт расширяет базовую *метамодель* метамоделями для реляционных и многомерных данных, для преобразования функций OLAP и ХД, включая процессы и операции. Спецификацию CWM можно рассматривать как язык, предназначенный для определения моделей ХД. Спецификация CWM расширяет язык UML: каждый *метакласс* (*metaclass*) CWM наследуется непосредственно или косвенно из *метаклассов* UML. Так, *метакласс* "Реляционная Таблица" (Relational Table) CWM является непосредственным наследником Класса UML (UML Class), а "Реляционный Столбец" (Relational Column) — прямой потомок Атрибута UML (UML Attribute).

Таким образом, спецификация CWM определяет *метамодель* и для *предметно-ориентированных метаданных*, и для *технических метаданных*. Эта *метамодель* используется для обмена экземплярами *метаданных* между гетерогенным программным обеспечением, поставляемым различными производителями. Системы, поддерживающие *метамодель* CWM, обмениваются данными в форматах, которые согласуются с этой моделью.

Стандарт OMG "Средства метаобъекта" (Meta Object Facility, MOF) определяет общие интерфейсы и семантику для взаимодействующих метамоделей. Являясь подмножеством UML, он представляет собой пример мета-*метамодели*, или модели *метамодели* (подмножество). В сферу действия этого стандарта входит определение языка описания интерфейса (*Interface Definition* Language), который устанавливает правила управления моделями с помощью программных APIs. Все модели CWM выражаются на UML и реализуют семантику MOF.

Стандарт OMG "Обмен *метаданными* XML" (XML Metadata Interchange, *XMI*) устанавливает правила преобразования метамоделей MOF в XML. *XMI* непосредственно задействован в обмене метамоделями. *Метамодели* MOF транслируются в XML DTD, а модели — в XML-документы, которые согласуются со своими DTD.

Таким образом, стандарт CWM состоит из ряда составных метамоделей (суб-метамоделей), которые организованы в виде следующих 4 слоев: базовый слой (Foundation), источники данных (Resources), анализ (Analysis) и управление хранилищем (Management), как показано на [рис. 14.7](https://intuit.ru/studies/courses/599/455/lecture/10173?page=4#image.14.7).



**Рис. 14.7.**Четыре уровня модели CWM

Базовый слой состоит из метамоделей, которые поддерживают моделирование таких различных элементов и сервисов, как типы данных, системное преобразование типов, абстрактные ключи и индексы, выражения, бизнес-информация и включения программного обеспечения, основанного на использовании компонентных объектов.

Слой источников данных предоставляет возможность моделировать существующие и новые источники данных, в том числе реляционные базы данных, ориентированные на запись базы данных (record oriented data-bases), а также XML- и основанные на объектах (*object-based*) источники данных.

Слой анализа предоставляет средства для моделирования сервисов информационного анализа, которые обычно используются в хранилище данных. Он определяет *метамодель* для преобразования данных, OLAP, визуализации информации и исследования данных (data mining).

Слой управления состоит из метамоделей, представляющих стандартные процессы и операции ХД, *журнализации* и планирования работ (например, ежедневной загрузки и выгрузки).

Набор метамоделей CWM является достаточным для моделирования всего ХД.

**Выбор метамодели при проектировании хранилища данных**

Когда требования к *метаданным* собраны и формализованы, можно приступать к разработке *метамодели*. На практике следовать требованиям стандарта часто бывает сложно. Причиной этого является дефицит времени и недопонимание важности проработки *метамодели* руководством компании, особенно когда компания создает ХД силами своего ИТ-подразделения. Руководство по проектированию и разработке *метамодели* CWM насчитывает более 700 страниц [[68](https://intuit.ru/studies/courses/599/455/literature#literature.68)]. Как правило, менеджменту ИТ-подразделения трудно объяснить руководству компании, что для собственной разработки ХД необходимо либо взять новую штатную единицу, либо отправить своего специалиста на обучение.

Заказывает ли компания разработку ХД третьей компании или собирается проводить ее самостоятельно – можно выделить следующие способы создания *метамодели*.

* Построение *метамодели* ХД вручную.
* Построение *метамодели* ХД на основе стандартов.
* Построение как самой *метамодели*, так и репозитория *метамодели* на основе имеющихся инструментальных средств, для интеграции *метаданных* источников.

Чтобы построить *метамодель* ХД вручную, необходимо собрать правильные определения сущностей, их атрибутов и взаимосвязей между сущностями. Для разработки такой *метамодели* может быть применено либо *объектно-ориентированное моделирование*, либо ER-*моделирование*.

Если для построения *метамодели* ХД проектировщик ориентируется на использование стандарта, то у него есть возможность задействовать либо спецификацию "Открытая информационная модель" (*Open* *Information* *Model* OIM), либо спецификацию "Общая *метамодель* хранилища данных" (Common Warehouse Meta-*Model*, CWM). CWM описывает обмен *метаданными* в *системах складирования данных*, управления знаниями и деловой осведомленности. OIM является набором спецификаций *метаданных* для использования в разработке приложений ХД. Обе спецификации основываются на промышленных стандартах, таких как *UML*, *XML* и *SQL*.

Выбор подхода к проектированию *метаданных* во многом определяется набором инструментальных средств проектировщика ХД и выбором несущей *СУБД*. Ясно, что разработанная вручную *метамодель* имеет важное преимущество: она, как правило, наиболее полно отражает *представление* *метаданных* компании. Но у такой модели есть большой недостаток — ее нужно сопровождать и поддерживать постоянно в актуальном состоянии, как правило, вручную. Модели, разработанные с учетом стандартов, учитывают большинство требований по представлению *метаданных* компании в ХД. Кроме того, они расширяемы и поддерживаются ведущими производителями средств разработки (*Oracle*, *IBM*, Microsoft).

*Репозиторий* *метаданных* ХД следует поддерживать при использовании любого метода проектирования *метаданных*. При этом важно выбрать для него архитектуру (централизованный он будет или распределенный) и способы поддержки его в актуальном состоянии (поскольку *метаданные* связывают между собой семантику всех *компонент* *системы складирования данных*).

Программные компоненты *системы складирования данными* через *репозиторий* обмениваются *метаданными* в процессе своей работы. Для организации обмена *метаданными* стандарт CWM позволяет детализировать архитектуру репозитория. При этом формат обмена *метаданными* есть *XML*-документ.

Пример использования AllFusion ERwin *Data Modeler* для создания и поддержки *метамодели* ХД подробно рассмотрен в [[37](https://intuit.ru/studies/courses/599/455/literature#literature.37)]. Далее мы изучим вопрос, как можно построить модель *метаданных* вручную.

**Проектирование логической модели метаданных хранилища данных**

Обращаясь к изучению вопроса логического проектирования модели *метаданных*, мы преследуем цель разобраться в сути процесса представления *метаданных* в ХД и получить навыки построения модели *метаданных*, которые можно в дальнейшем применить для управления *метаданными* через репозитории *метаданных*, поставляемых производителями программного обеспечения для создания ХД.

В разделе "*Логическая структура* *метаданных* хранилища данных" мы рассмотрели примеры описания *метаданных* для различных объектов ХД. Мы будем использовать эти примеры при создании нашей модели *метаданных*.

Создадим сначала логическую *модель данных* для *метаданных* таблиц фактов. Она может быть такой, как на [рис. 14.8](https://intuit.ru/studies/courses/599/455/lecture/10173?page=5#image.14.8).



[увеличить изображение](https://intuit.ru/EDI/20_07_20_2/1595197216-9970/tutorial/632/objects/14/files/09_08.jpg)
**Рис. 14.8.**Модель метаданных для таблицы фактов

*Метаданные* о таблице фактов целесообразно разместить в двух сущностях, одна из которых — "Таблицы фактов" — содержит практически не меняющуюся информацию о таблице фактов, а другая — "История загрузки" — содержит данные, которые меняются согласно параметру "Частота загрузки".

В сущность "Таблицы фактов" включена *информация*:

* о наименовании таблицы фактов;
* о бизнес-определении таблицы фактов;
* об альтернативном имени;
* об ответственном сотруднике;
* о качестве данных;
* о точности данных.

В сущность "История загрузки" включена *информация*:

* о статистике загрузки данных;
* о статистике использования данных;
* о правилах архивирования данных;
* о статистике архивирования;
* о правилах удаления данных;
* о статистике удаления.

Оставшиеся элементы метаописания таблицы фактов находятся с сущностью "Таблицы фактов" в отношении наследования. Для представления этих элементов метаописания на рисунке введены дополнительные сущности, которые мы будем моделировать далее.

*Таблица* фактов многомерной модели содержит данные о фактах и метриках. Построим соответствующий фрагмент модели *метаданных* ([рис. 14.9](https://intuit.ru/studies/courses/599/455/lecture/10173?page=5#image.14.9)).



[увеличить изображение](https://intuit.ru/EDI/20_07_20_2/1595197216-9970/tutorial/632/objects/14/files/09_09.jpg)
**Рис. 14.9.**Модель метаданных для метрик таблицы фактов

Сущности "Факты", "Метрики фактов", "Метрики" и "Поля схемы "*звезда*" представляют описание характеристик фактов. Сущности "Атрибуты" и "Домены атрибутов" представляют физические определения метрик фактов в ХД.

Заметим, что в сущности "Атрибуты" присутствует *атрибут* "*Ключ* измерения", поскольку эта сущность также должна описывать атрибуты измерений.

Рассмотрим *моделирование* *метаданных* измерений ([рис. 14.10](https://intuit.ru/studies/courses/599/455/lecture/10173?page=5#image.14.10)).



[увеличить изображение](https://intuit.ru/EDI/20_07_20_2/1595197216-9970/tutorial/632/objects/14/files/09_10.jpg)
**Рис. 14.10.**Модель метаданных для таблиц измерений

Сущности "Измерения" и "Измерения и метрики" описывают измерения и их связи с метриками таблицы фактов. Сущность "Атрибуты" описывает атрибуты измерений.

Дополним разработанную модель *метаданных* для нашего примера информацией об источниках данных, как показано на [рис. 14.11](https://intuit.ru/studies/courses/599/455/lecture/10173?page=5#image.14.11).



[увеличить изображение](https://intuit.ru/EDI/20_07_20_2/1595197216-9970/tutorial/632/objects/14/files/09_11.jpg)
**Рис. 14.11.**Модель метаданных для хранилища данных

Источники данных для ХД описаны в сущностях "Источники данных", "Таблицы", "Колонки", "*Загрузка* данных" и "Правила преобразования".

Таким образом, мы построили *логическую модель метаданных* ХД для нашего примера из раздела "*Логическая структура* *метаданных* хранилища данных" настоящей лекции.

Заметим, что не все элементы описания *метаданных* были использованы при конструировании модели *метаданных* ХД. Это право проектировщика ХД, основанное на изучении требований к системе складирования данных.

Также обратим внимание на то, что была создана частная модель, которая не учитывает ряд требований, предъявляемых к модели *метаданных*. Как правило, при построении модели *метаданных* ХД должен быть учтен ряд обязательных элементов представления *метаданных* в модели, а именно:

* предметная направленность бизнес-окружения ХД. Этот фрагмент модели *метаданных* должен описывать направления административно-хозяйственной деятельности организации, т.е. описание бизнес-модели деятельности организации;
* описание качества данных. Этот фрагмент модели *метаданных* должен описывать требования к качеству данных, загружаемых в ХД;
* описание программно-аппаратной среды ХД. Этот фрагмент модели *метаданных* должен описывать таблицы, колонки, ключи и процессы обработки данных в ХД;
* описание информационной безопасности. Этот фрагмент модели *метаданных* описывает процедуры разграничения доступа и аутентификации пользователей в системе, которая использует доступ к ХД.

Обратим внимание на то, что вопросам представления информации об информационной безопасности в этой лекции не было уделено никакого внимания. Как правило, программно-аппаратные решения в области обеспечения информационной безопасности носят конфиденциальный характер, и давать какие-либо общие рекомендации по их описанию в модели *метаданных* нецелесообразно. Это будет определяться руководителем ИТ-проекта создания ХД.

**Резюме**

В настоящей лекции мы рассмотрели понятие *метаданных* как совокупности спецификаций и элементов данных, содержащих описание данных ИС и процессов их обработки. Были определены основные функции и дана классификация *метаданных* в ХД. Был дан краткий обзор спецификации "Общая *метамодель* хранилища данных".

На примере конкретного киоска данных было подробно показано, как формировать *метаданные* для модели, таблиц фактов, фактов, таблиц измерений и источников данных. Приведенное описание *метаданных* послужило основой для логического проектирования модели *метаданных* ХД.

*Метаданные* — это *информация* о данных, которая требуется для управления ХД, а управление *метаданными* — существенный *компонент* архитектуры хранения. К *техническим метаданным* относится вся *информация*, которая требуется для настройки и использования ХД. *Предметно-ориентированных метаданных* включают в себя бизнес-термины и определения данных ХД. **Структурные метаданные** – это описание объектов ХД и их характеристик. *Метаданные процесса обработки данных* — это *информация*, собранная во время работы ХД, такая как происхождение перенесенных и преобразованных данных; статус использования данных (активные, архивированные или удаленные); данные мониторинга, такие как *статистика* использования, сообщения об ошибках и результаты аудита.

*Метаданные* часто размещаются в репозитории, который позволяет совместное использование *метаданных* различными инструментами и процессами при проектировании, установке, применении, эксплуатации и администрировании ХД.