DWH – Task 6

Content

[1. The initial scheme 2](#_Toc498024704)

[2. 4 steps Dimensional Process 2](#_Toc498024705)

[3. Star Schema 3](#_Toc498024706)

[4. Snowflake Schema 5](#_Toc498024707)

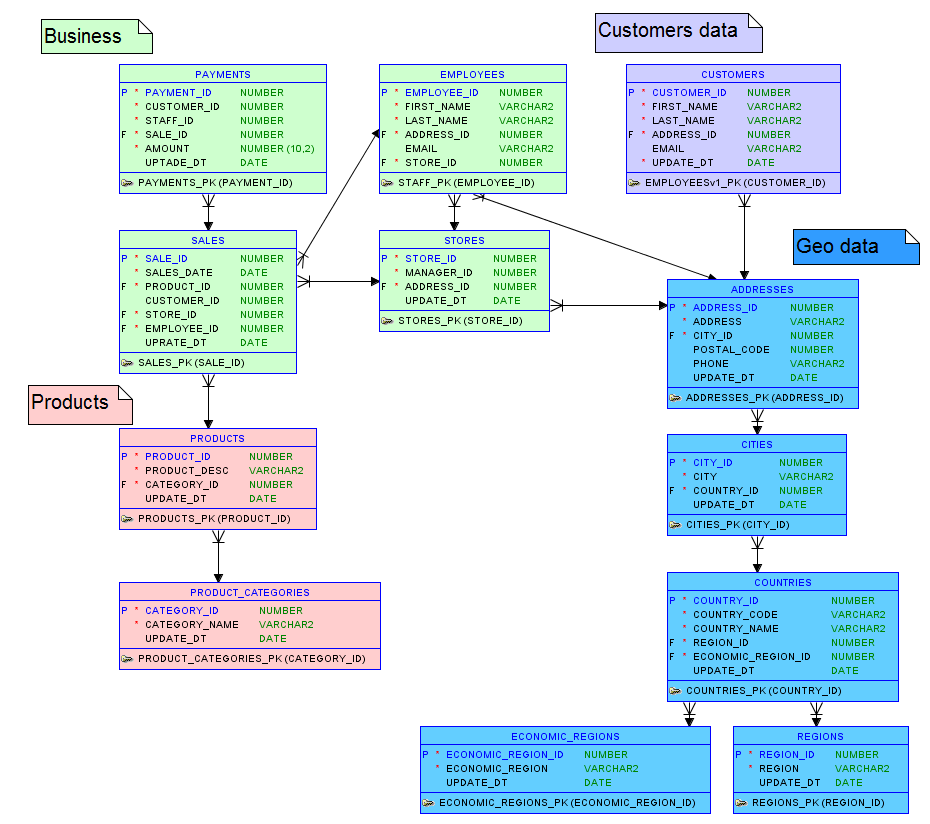
[5. Analytical task 10](#_Toc498024708)

[5.1. Business background 10](#_Toc498024709)

[5.2. Problems because of poor data management. 10](#_Toc498024710)

[5.3. Benefits from implementing a Data Warehouse 10](#_Toc498024711)

# The initial scheme



# 4 steps Dimensional Process

*Шаг 1. Выберите бизнес-процесс.*

Исходная схема отображает учет продаж магазинов. Sales – это центральная таблица предложенной схемы, а единственное числовое значение, на котором можно строить анализ – это Amount – сумма платежа из таблицы Payments. Одна продажа Sales состоит из нескольких платежных документов Payments. Все остальные таблицы идут как справочники, раскрывающие сущность продажи. В итоге получается нужный мне бизнес-процесс – это ежедневные суммы продажи товаров.

*Шаг 2: Объявите зерно Grain.*

Повторюсь, одна продажа Sales состоит из нескольких платежных документов Payments. Одна продажа Sales относится к одному товару Products и может быть осуществлена одним сотрудником Employees в одном магазине Stores для одного клиента Customers. Везде имеет место отношение 1 ко многим. Для дальнейшей работы я это и выберу за зерно. Т.е. одна строка будет отражать одну продажу Sum(Amount) одного продукта осуществленную одним сотрудником в конкретном магазине одному конкретному заказчику в один определенный день.

*Шаг 3: Определите Dimensions.*

Dimension Dates – содержит необходимые данные для дальнейшего анализа в разрезе времени – дата, месяц, квартал, полугодие, год.

Dimension Products – содержит необходимую информацию о продукте.

Dimension Stores – описывает магазины и информацию о них.

Dimension Customers – описание заказчиков и информации о них.

Dimension Employees – описание сотрудников и информации о них.

Вся информация содержащаяся в таблицах product\_categories, addresses, cities, countries, regions, economic\_regions не характеризует напрямую факт продаж, но тем не менее является дополнительной информацией, по которой можно проводить группировку и фильтрацию данных. В зависимости от типа схемы данные из этих таблиц будут либо включены в основные dimensions, либо выделены в отдельные. Так, например, dimension Products в схеме звезда будет содержать информацию о категории товара, а в схеме снежинка только id на категорию, а информация по категории будет выделена в отдельный dimension.

*Шаг 4: Определите факты Facts.*

Таблица фактов будет содержать поле Sum(Amount) – названо так для наглядности. Ключи для ссылки на dimensions - Date\_Id, Products\_Id, Stores\_Id, Customers\_Id и Employees\_Id. А также поле Update\_dt, которое будет в дальнейшем использоваться для инкрементальной загрузки.

# Star Schema

При разработке схемы Звезда я убрала таблицу Payments, перенеся в таблицу фактов значением Amount. При этом в Sales будет записываться именно Sum(Amount), т.к. несколько платежек могут формировать одну продажу. Остальная информация из этой таблицы не нужна для моей задачи.

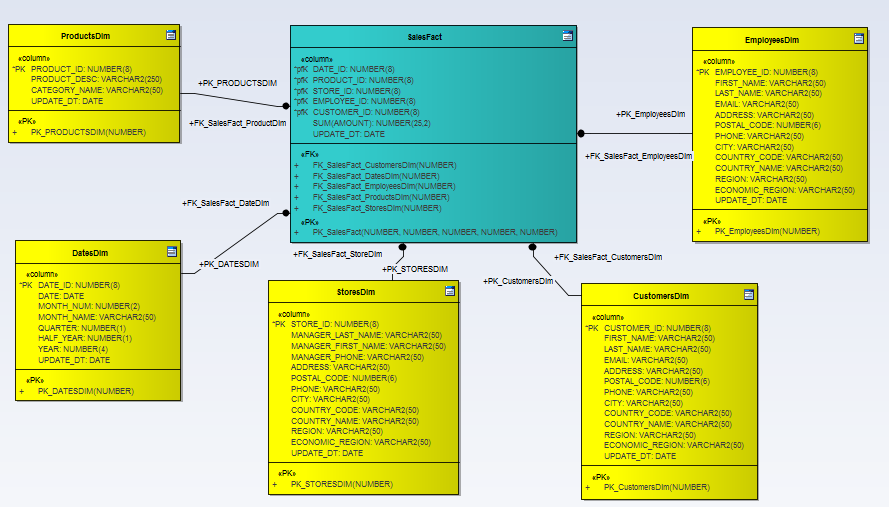
Все геоданные распределены в dimensions Stores, Customers и Employees, исходя из базовой схемы.

В dimension Products включены данные о категории товара.

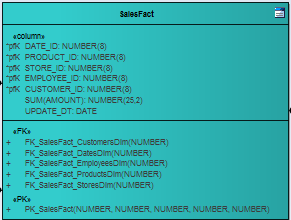
В dimension Stores я не включала поле manager\_id, т.к. для моей задачи этот столбец не несет никакой информативности.

Dimension Dates я создала с нуля, т.к. это основной dimension, который только может быть в DWH. Я включила поля необходимые для поставленной цели – дата, номер месяца, название месяца, квартал, полугодие, год.

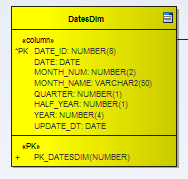
Также, каждая таблица содержит поле Update\_dt, которое, как говорилось выше, будет в дальнейшем использоваться для инкрементальной загрузки.



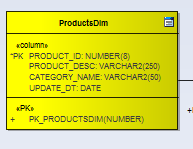
Facts:



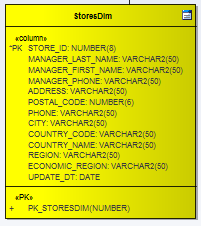
Dimension Dates:



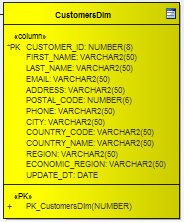
Dimension Products :



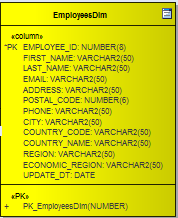
Dimension Store:



Dimension Customers:



Dimension Employees:



# Snowflake Schema

В схеме Снежинка в сравнении со схемой Звезда некоторые dimensions делятся на несколько таблиц, чтобы быть ближе к третьей нормальной форме NF.

Направление Dates я разделила на два dimensions. Я отдельно выделила информацию по месяцу MonthsDim и остальные данные DatesDim, включающий дату, ид месяца, квартал, полугодие и год.

Ветка Products состоит непосредственно из Products и ProductsCategories.

Направление Stores состоит из dimension Stores и dimensions описывающих его геоданные, - StoresAddresses, Cities, Countries, Regions и Economic Regions. Аналогичная ситуация с веткой Customers.

Направление Employees я разделила на два dimensions. Я отдельно выделила информацию фамилию, имя и email и отдельно данные по локации сотрудника – адрес, почтовый индекс, город, страна, регион. Первая часть – это dimention EmployeesDim, вторая часть - это dimention EmployeeslocationDim. Информация из первого будет часто использоваться в шапке отчета по сотруднику, информация из второго – крайне редко для специфических задач маркетинга.

Также как и в схеме Star, каждая таблица содержит поле Update\_dt, которое будет в дальнейшем использоваться для инкрементальной загрузки.

Snowflake схема:

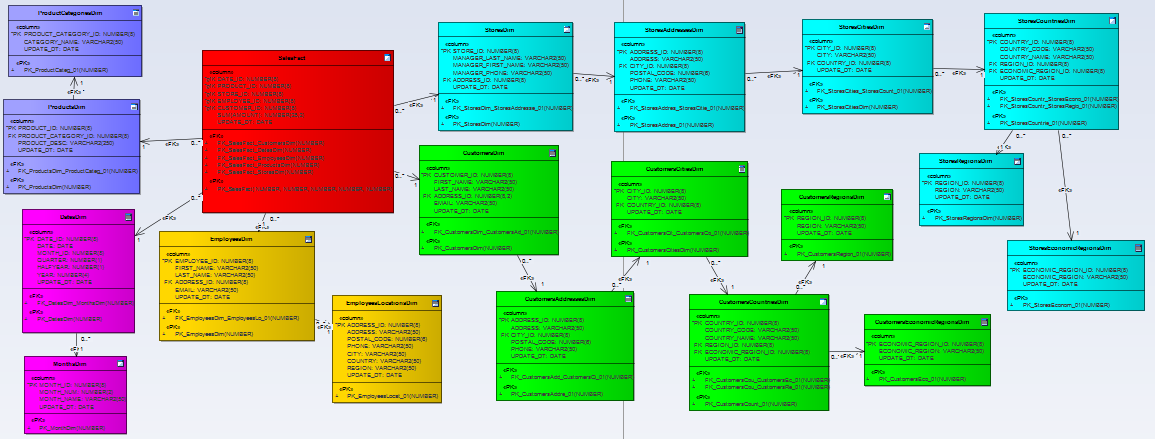
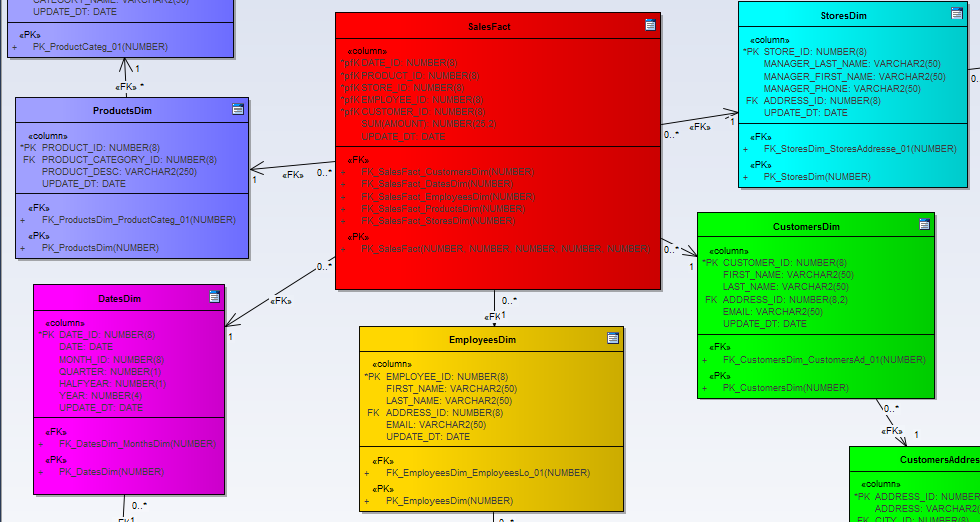
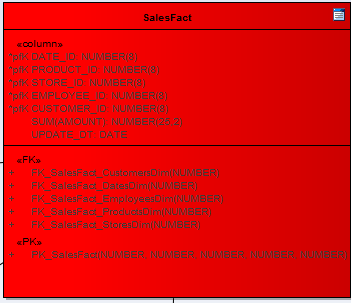


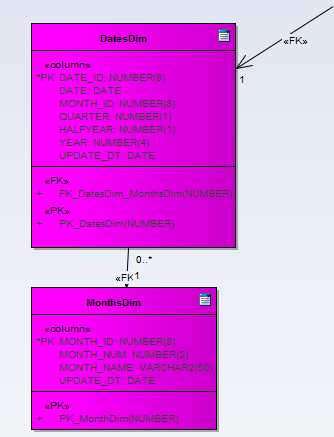
Таблица Facts и “близлежащие” dimensions:



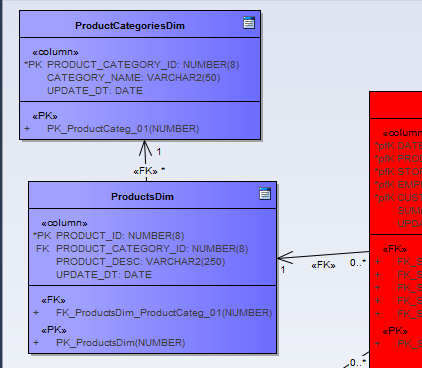
Facts:



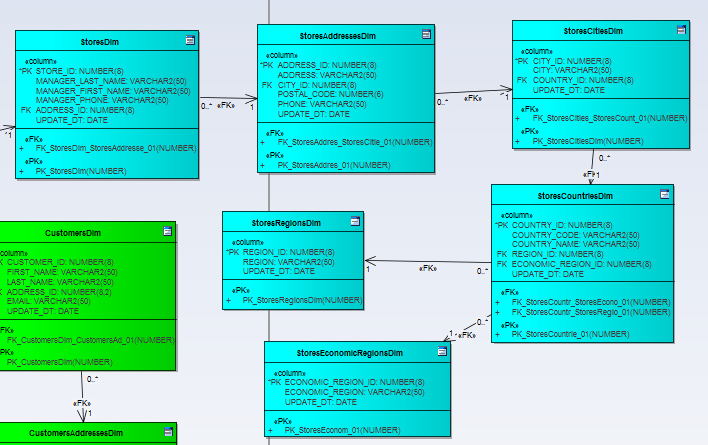
Dimension Dates и Dimension Month:



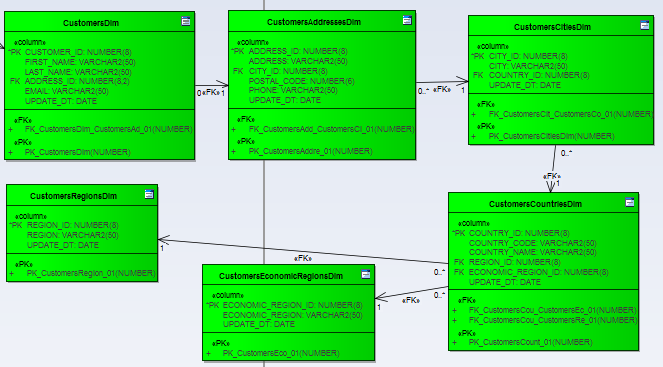
Dimensions Products и ProductsCategories:



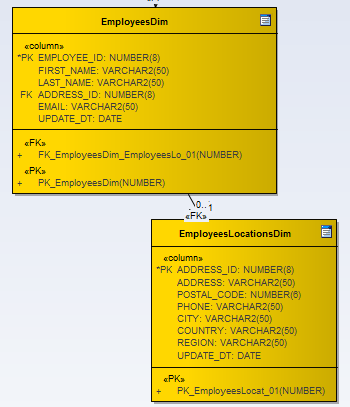
Dimension Stores и dimensions, описывающие его геоданные:



Dimension Customers и dimensions, описывающие его геоданные:



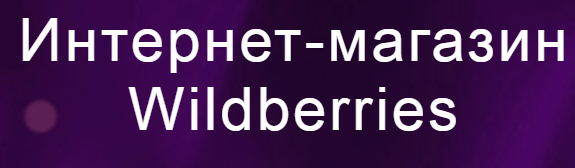
Dimension Employees и dimension EmployeesLocations, описывающий месторасположение и контактные данные сотрудника:



# Analytical task

## Business background

Хранилище данных будет реализовано для нужд белорусского online-магазина Wildberries. Wildberries – это один из крупнейших интернет-магазинов модной одежды, обуви, аксессуаров, косметических средств, существующий уже более 12 лет. Магазин насчитывает более 7 тысяч брендов на сайте и более 100 тысяч заказов в день. Из года в год он развивается, расширяет географию и улучшает качество обслуживания!



## Problems because of poor data management.

С каждым днем количество товаров, отделений магазина и клиентов растет в геометрической прогрессии. Это требует пересмотра обработки данных. На данный момент база приложения недостаточно производительна, чтобы отрабатывать сложные запросы, поэтому на получение одного простейшего отчета, с которым хорошо построенное DWH справляется за считанные секунды, уходят часы. В такой ситуации многие запросы управляющего сектора не находят отклика, нет возможности быстрого реагирования на изменение рынка, реактивного изменения стратегий продаж, открытия новых. В результате теряется скорость принятия решений, теряются деньги, растет недоверие к ИТ решениям.

## Benefits from implementing a Data Warehouse

Внедрение DWH поможет навести порядок и упростит использование данных для анализа. Сложные отчеты будут быстро формироваться, а, следовательно, не будут нагружать приложения, будут чаще использоваться и потому помогут в развитии бизнеса. Увеличится скорость реагирования на изменения рынка, что приведет к принятию стратегически важных решений, улучшится качество обслуживания, увеличится количество клиентов и прибыли соответственно.

Расширение бизнеса приведет к увеличению количества рабочих мест, как в самом онлайн магазине, так и у поставщиков и производителей приобретаемой продукции.