



Тема:

Повышение эффективности работы баз данных. Обработка транзакций OLTP-OLAP системы мониторы транзакций.

КАБУТОВ ХОДЖИМУРОД

МАХМАДИЕВИЧ

ФТЭС 3-КУРС 2-ГРУППА

Индексы в стандарте языка

Индекс – это набор ссылок, упорядоченных по определенному столбцу таблицы, который в данном случае будет называться индексированным столбцом. Хотя индекс и связан с конкретным столбцом (или столбцами) таблицы, все же он является самостоятельным объектом базы данных. Индексы обычно создаются с целью удовлетворения определенных критериев поиска после того, как таблица уже находилась некоторое время в работе и увеличилась в размерах. Создание индексов не предусмотрено стандартом SQL, однако большинство диалектов поддерживают как минимум следующий оператор:

```
CREATE [ UNIQUE ] INDEX имя_индекса  
ON имя_таблицы(имя_столбца[ASC|DESC][, ...n])
```

ИНДЕКСЫ

В СРЕДЕ SQL SERVER РЕАЛИЗОВАНО
НЕСКОЛЬКО ТИПОВ ИНДЕКСОВ:

КЛАСТЕРНЫЕ ИНДЕКСЫ ;

НЕКЛАСТЕРНЫЕ ИНДЕКСЫ ;

УНИКАЛЬНЫЕ ИНДЕКСЫ.

Некластерный индекс

Некластерные индексы – наиболее типичные представители семейства индексов. В отличие от кластерных, они не перестраивают физическую структуру таблицы, а лишь организуют ссылки на соответствующие строки .

Для идентификации нужной строки в таблице некластерный индекс организует специальные указатели, включающие в себя:

информацию об идентификационном файле, в котором хранится строка ;

идентификационный номер страницы соответствующих данных;

номер искомой строки на соответствующей странице;

содержимое столбца.

```
CREATE UNIQUE NONCLUSTERED INDEX index_klient2
ON Клиент (Фамилия DESC,Имя DESC)
WITH FILLFACTOR=30,
STATISTICS_NORECOMPUTE
ON PRIMARY
```

Создание уникального некластерного индекса.

Кластерный индекс

Кластерный индекс может включать несколько столбцов. Однако количество таких столбцов рекомендуется по возможности свести к минимуму.

Необходимо избегать создания кластерного индекса для часто изменяемых столбцов, поскольку сервер должен будет выполнять физическое перемещение всех данных в таблице, чтобы они находились в упорядоченном состоянии, как того требует кластерный индекс. Для интенсивно изменяемых столбцов лучше подходит некластерный индекс.

При создании в таблице первичного ключа (PRIMARY KEY) сервер автоматически создаст для него кластерный индекс, если его не существовало ранее или если при определении ключа не был явно указан другой тип индекса.

```
CREATE UNIQUE CLUSTERED INDEX index_klient1
ON Клиент (Фамилия)
WITH DROP_EXISTING
ON PRIMARY
```

Создание уникального кластерного индекса.

Уникальный индекс

Уникальный индекс является своеобразной надстройкой и может быть реализован как для кластерного, так и для некластерного индекса . В одной таблице может существовать один уникальный кластерный и множество уникальных некластерных индексов.

Уникальные индексы следует определять только тогда, когда это действительно необходимо. Для обеспечения целостности данных в столбце можно определить ограничение целостности UNIQUE или PRIMARY KEY, а не прибегать к уникальным индексам. Их использование только для обеспечения целостности данных является неоправданной тратой пространства в базе данных. Кроме того, на их поддержание тратится и процессорное время.

OLAP и OLTP системы

OLTP –

оперативная транзакционная обработка данных

OLAP –

оперативная аналитическая обработка данных

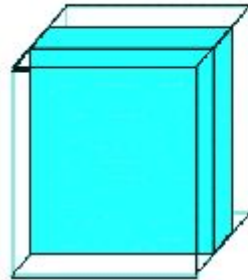
Характеристики OLTP системы

- Большой объем информации
- Часто различные БД для разных подразделений
- Нормализованная схема, отсутствие дублирования информации
- Интенсивное изменение данных
- Транзакционный режим работы
- Транзакции затрагивают небольшой объем данных
- Обработка текущих данных – мгновенный снимок
- Много клиентов
- Малое время отклика – несколько секунд

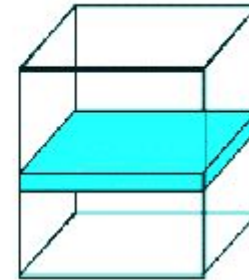
Характеристики OLAP системы

- Большой объем информации
- Синхронизированная информация из различных БД с использованием общих классификаторов
- Ненормализованная схема БД с дубликатами
- Данные меняются редко, Изменение происходит через пакетную загрузку
- Выполняются сложные нерегламентированные запросы над большим объемом данных с широким применением группировок и агрегатных функций.
- Анализ временных зависимостей
- Небольшое количество работающих пользователей – аналитики и менеджеры
- Большее время отклика (но все равно приемлемое) – несколько минут

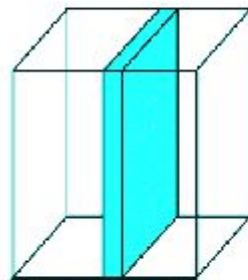
Основной способ логического представления данных – МНОГОМЕРНЫЕ КУБЫ (OLAP – кубы)



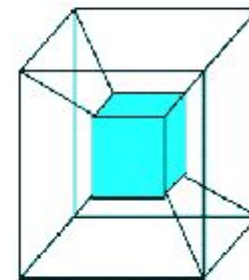
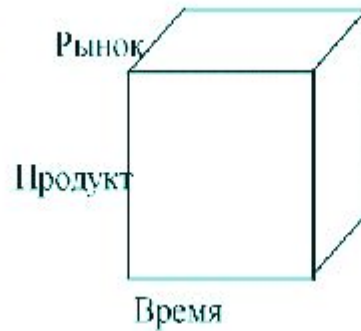
*Вид для менеджера
по рынкам*



*Вид для менеджера
по продуктам*



*Вид для финансового
менеджера*



Выборочный вид

OLAP – куб и срезы данных

	Март		
	Февраль		
	Январь		
	США	Канада	Мексика
Напитки	10 000	2000	1 000
Продукты питания	5000	500	250
Прочие товары	5000	500	250

	США	Канада	Мексика
Январь	20 000	4000	2000
Февраль	30 000	6000	3000
Март	50 000	10 000	5000

	США	Канада	Мексика
Unit Sales	2000	400	200
Store Sales	30 000	6000	3000
Store Cost	10 000	2000	1000

	Январь			Февраль		
	США	Канада	Мексика	США	Канада	Мексика
Unit Sales	500	100	50	500	100	50
Store Sales	7500	1500	750	7500	1500	750
Store Cost	2500	500	250	2500	500	250

Правила Кода для реляционных БД

1. Правило информации.
2. Правило гарантированного доступа.
3. Правило поддержки недействительных значений.
4. Правило динамического каталога, основанного на реляционной модели.
5. Правило исчерпывающего подязыка данных.
6. Правило обновления представлений.
7. Правило добавления, обновления и удаления.
8. Правило независимости физических данных.
9. Правило независимости логических данных.
10. Правило независимости условий целостности.
11. Правило независимости распространения.
12. Правило единственности.

Правила Кодда для OLAP

1. Концептуальное многомерное представление.
2. Прозрачность.
3. Доступность.
4. Постоянная производительность при разработке отчетов.
5. Клиент-серверная архитектура.
6. Общая многомерность.
7. Динамическое управление разреженными матрицами.
8. Многопользовательская поддержка.
9. Неограниченные перекрестные операции.
10. Интуитивная манипуляция данными.
11. Гибкие возможности получения отчетов.
12. Неограниченная размерность и число уровней агрегации.

Реализация OLAP

Типы OLAP - серверов

MOLAP (Multidimensional OLAP) - и детальные данные, и агрегаты хранятся в многомерной БД.

ROLAP (Relational OLAP) - детальные данные хранятся в реляционной БД; агрегаты хранятся в той же БД в специально созданных служебных таблицах.

HOLAP (Hybrid OLAP) - детальные данные хранятся в реляционной БД, а агрегаты хранятся в многомерной БД.

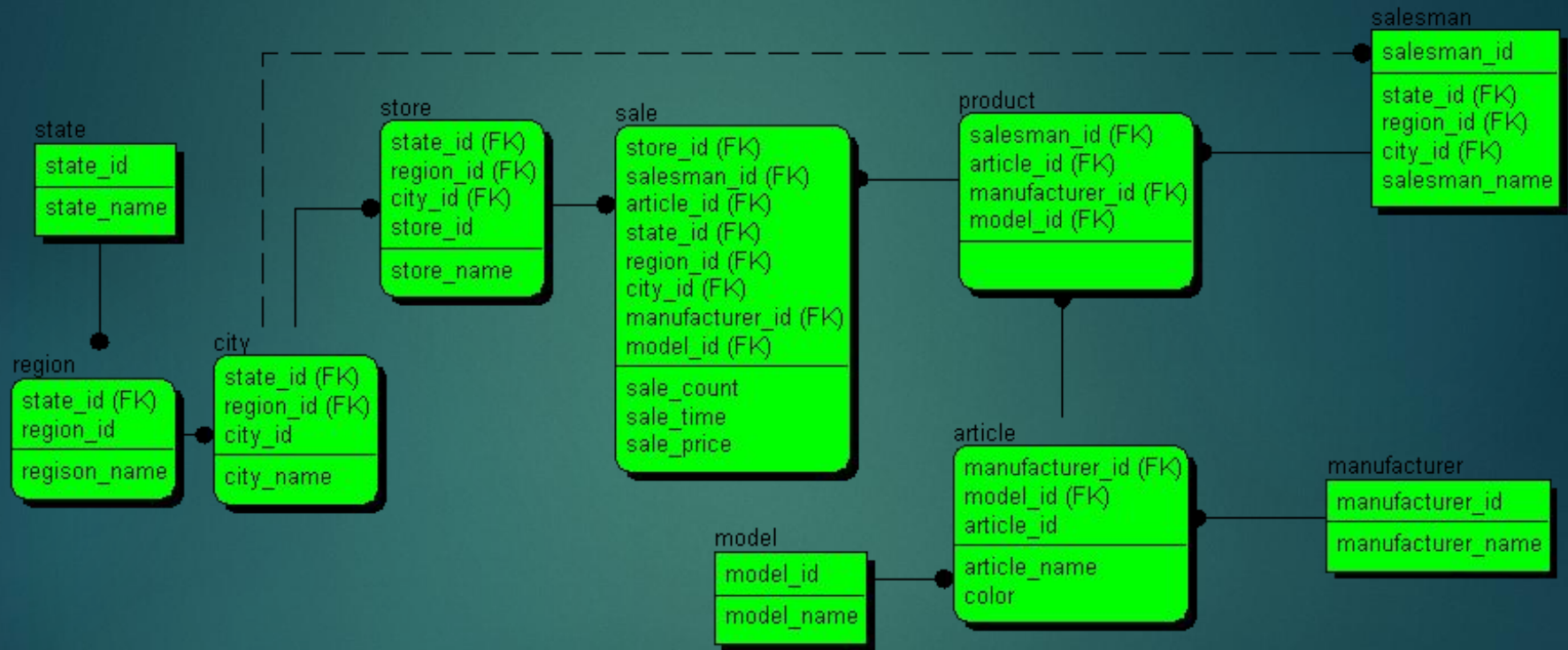
OLTP схема базы данных

Моделируются оптовые продажи на склад

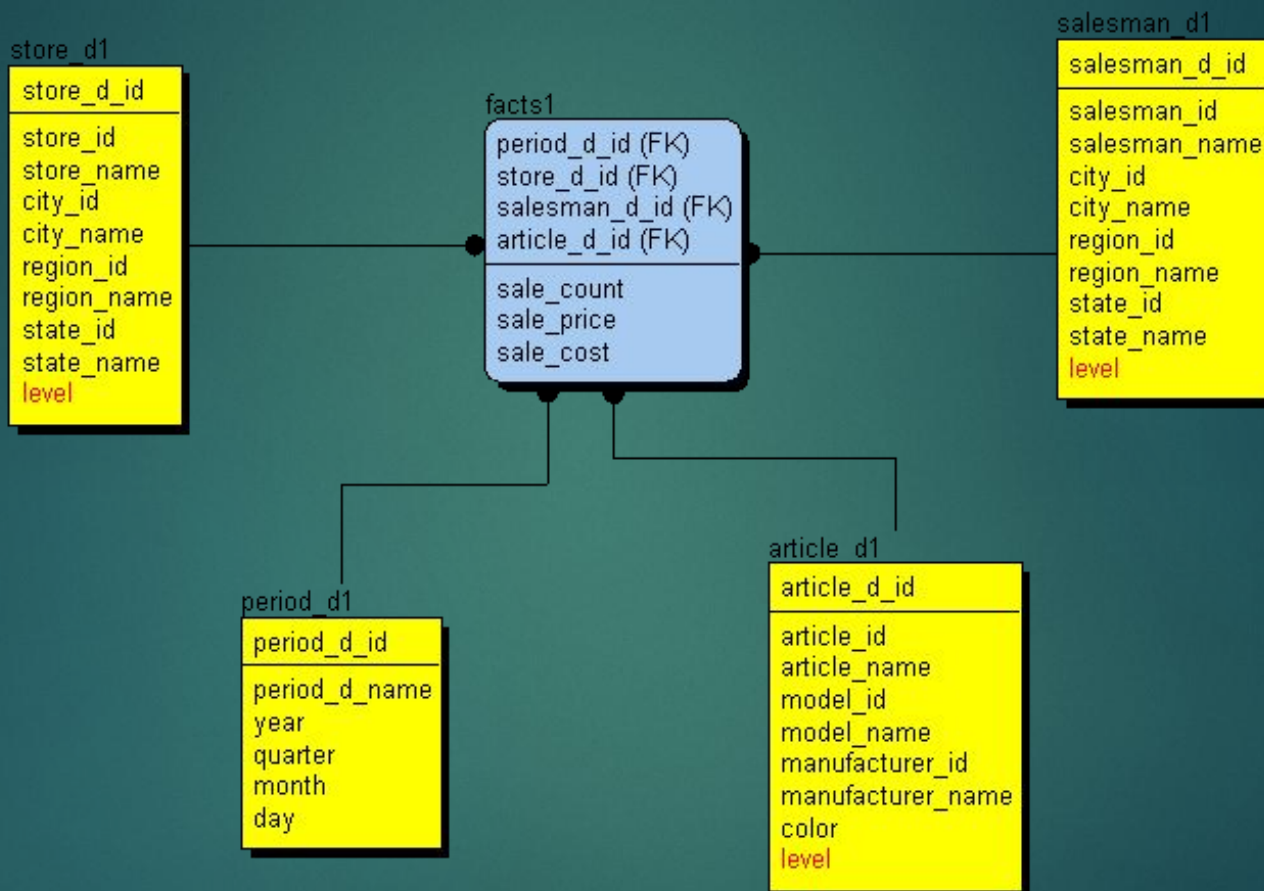
Объекты

1. Склады
2. Категории товаров (модель)
3. Производители
4. Товары
5. Продавцы
6. Оптовые продажи на склад

Оперативная схема БД оптовых продаж на склады



ROLAP – схема типа звезда



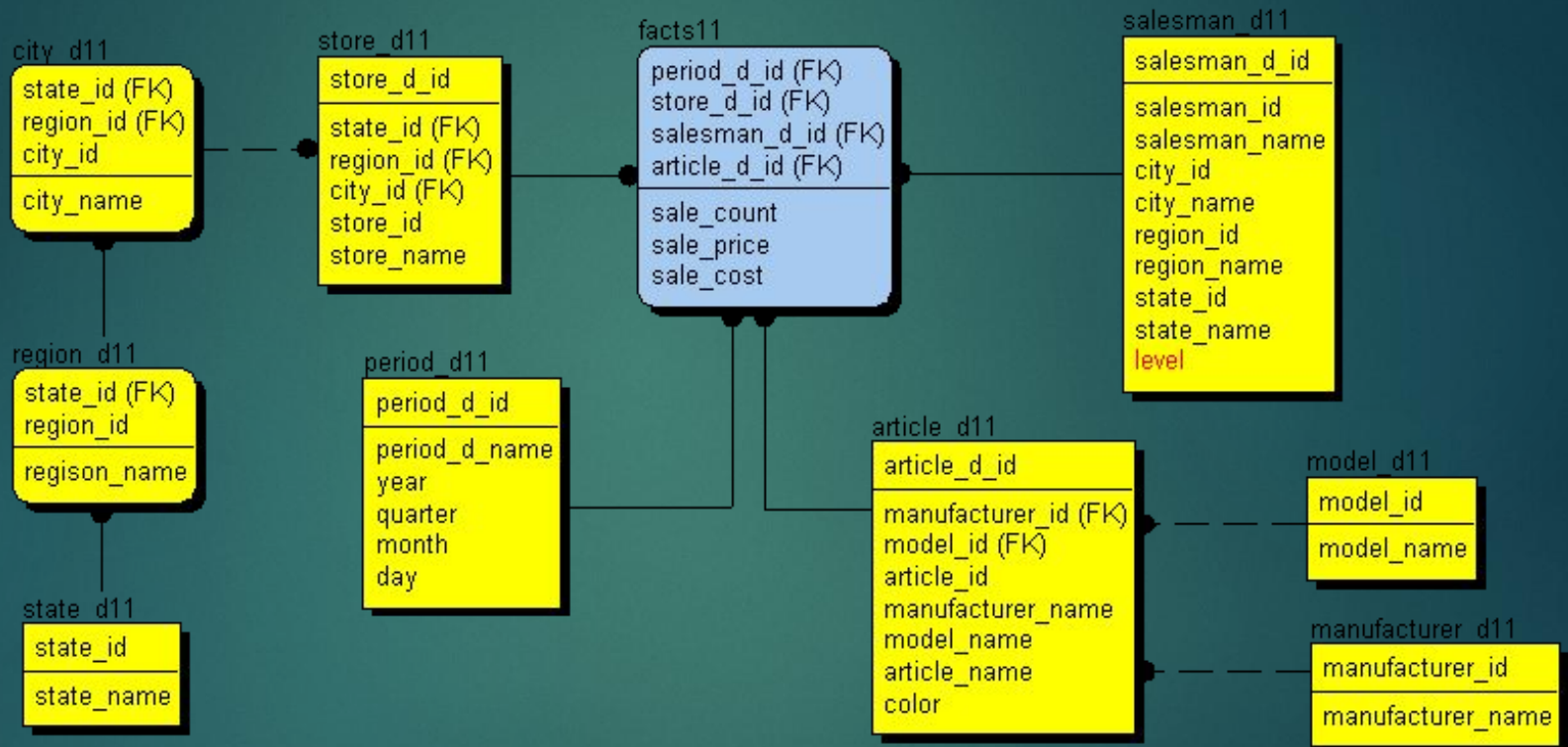
Особенности ROLAP – схемы типа звезда

1. Одна таблица фактов, которая сильно денормализована
2. Несколько таблиц измерений, которые также денормализованы
3. Первичный ключ таблицы фактов является составным и имеет по одному столбцу на каждое измерение
4. Агрегированные данные хранятся совместно с исходными

Недостатки

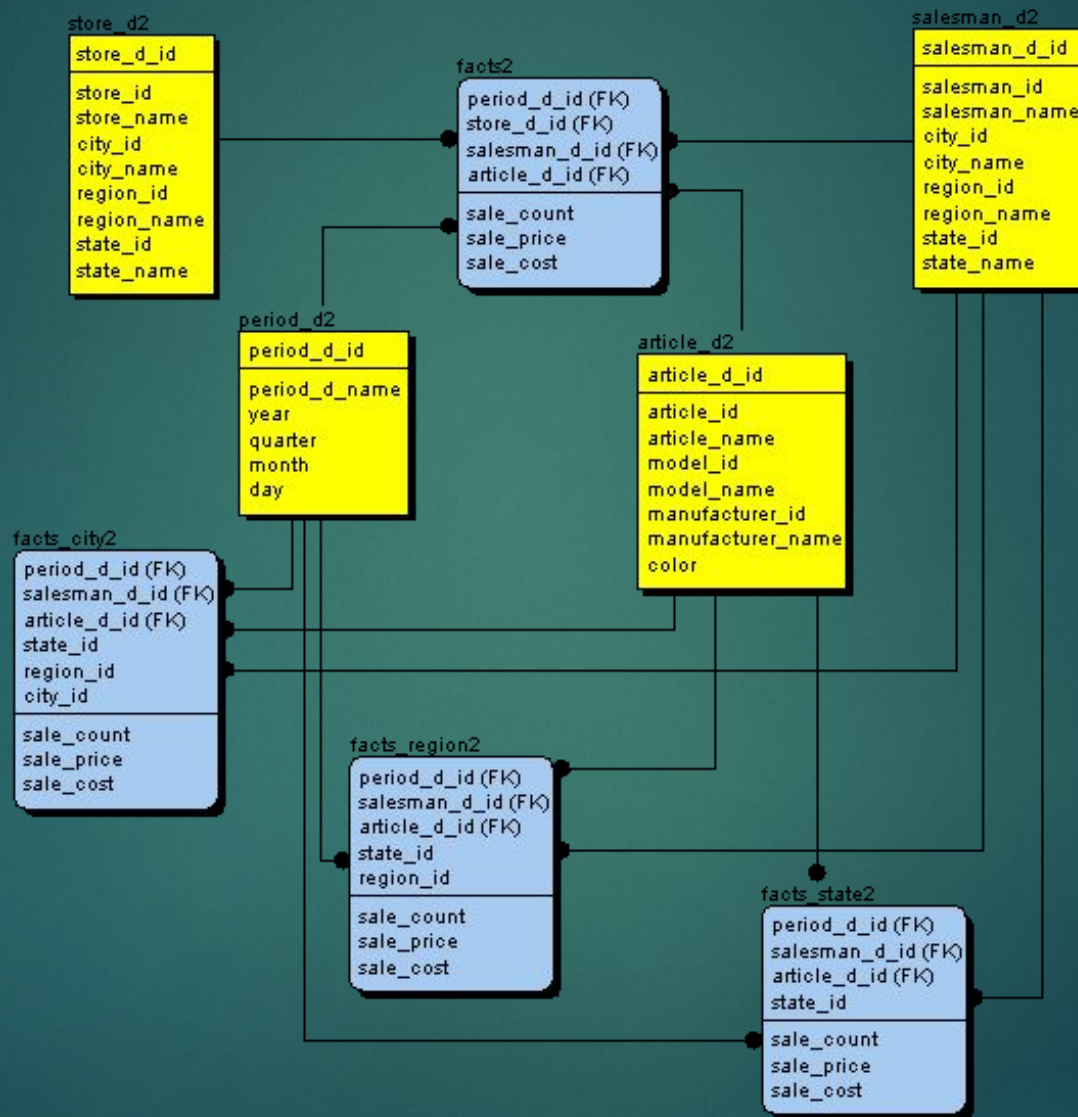
Если агрегаты хранятся совместно с исходными данными, то в измерениях необходимо использовать дополнительный параметр – уровень иерархии

ROLAP – схема типа снежинка с нормализованными измерениями



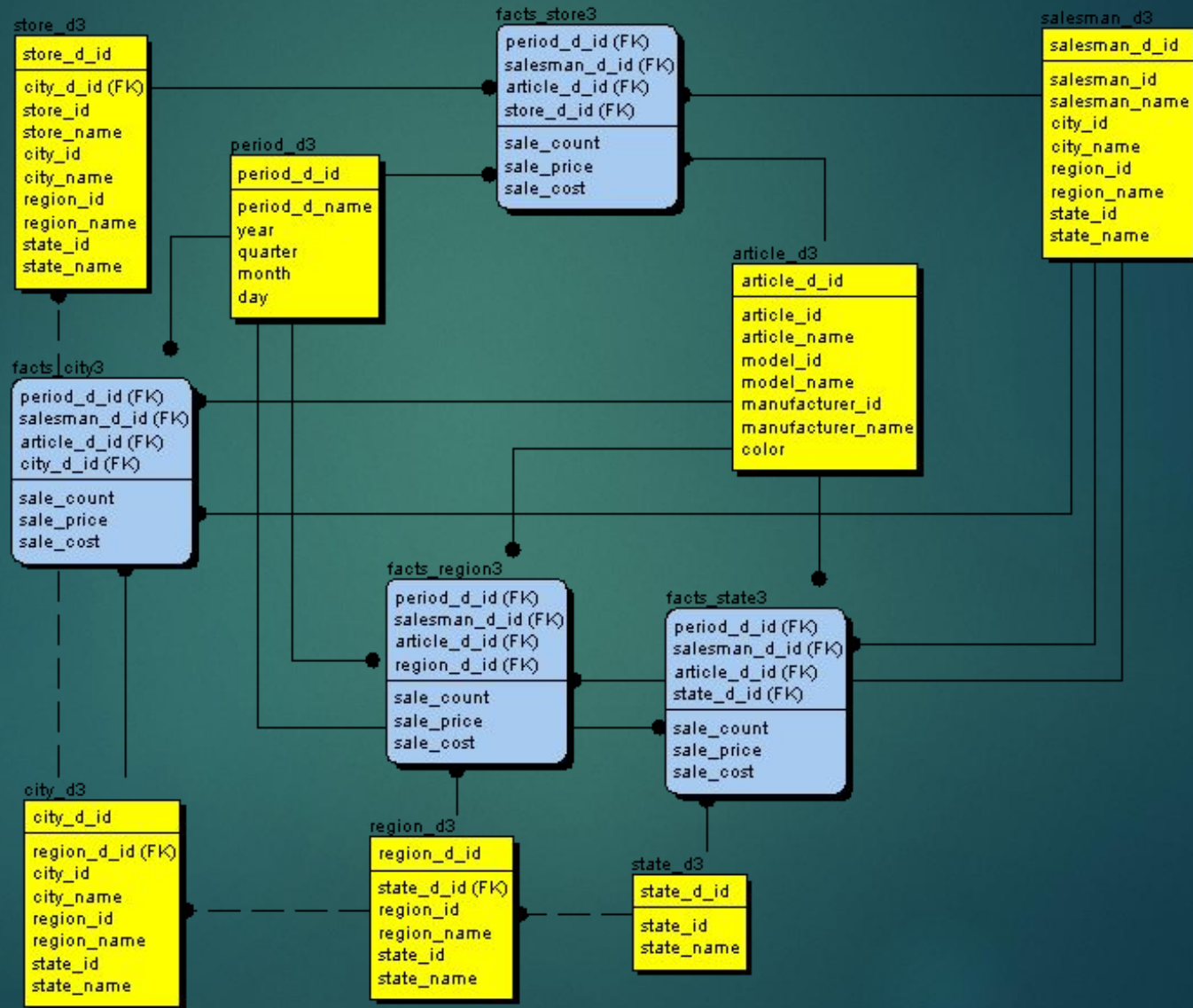
ROLAP – схема типа снежинка

с выделением агрегированных таблиц

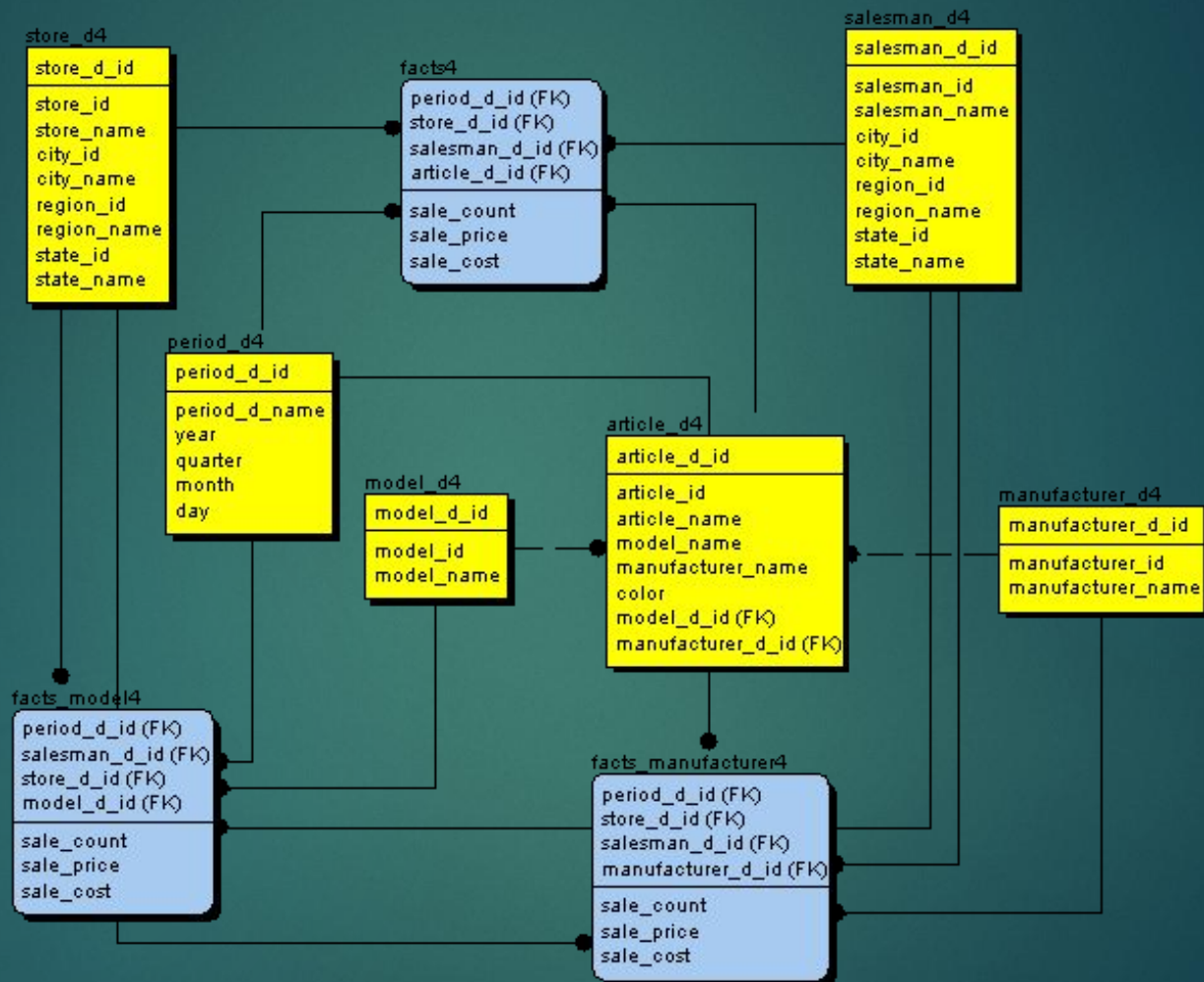


ROLAP – схема типа снежинка

с выделением агрегированных таблиц и нормализованными измерениями

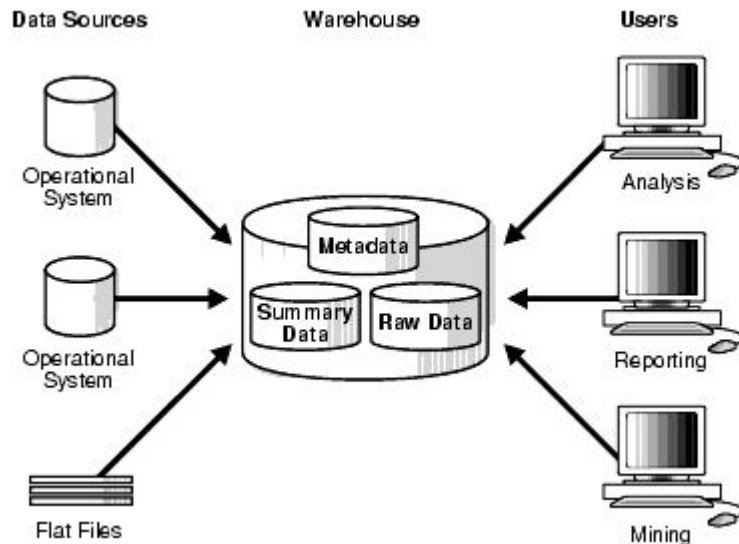


Агрегирование по производителю и модели товара



Состав хранилищ данных

1. Метаданные
2. Исходные данные
3. Предварительно просуммированные данные



Основные метаданные OLAP

Куб

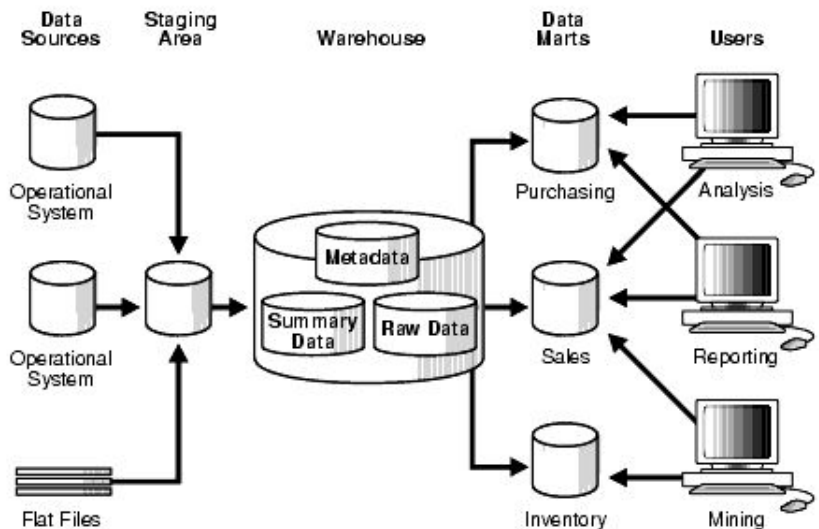
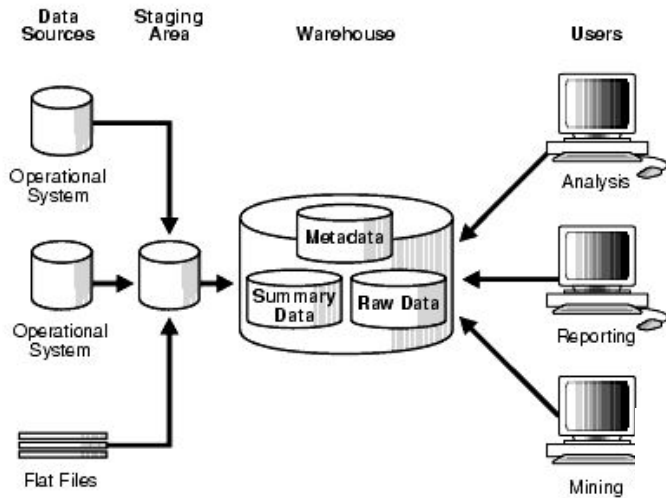
Факты

Измерения

1. Уровни
2. Иерархии
3. Атрибуты

Общая структура хранилища данных

1. Источники данных
2. Процедуры выгрузки, преобразования и загрузки данных
3. Хранилище данных
4. Витрины данных
5. Аналитические приложения



Структура хранилища в ORACLE

