МОДЕЛИРАНЕ И АНАЛИЗ НА СОФТУЕР

Павел Кюркчиев
Ас. към ПУ „Паисий Хилендарски“
https://github.com/pkyurkchiev
@pkyurkchiev

ЛОГИЧЕСКОТО МОДЕЛИРАНЕ НА ДАННИ

Въпроси, на които ще получим отговори

- Какво е модела на данни?
- Защо модела на данни е толкова важен?
- Какво прави модела на данни добър(ефективен)?
- Коя точно част от софтуерната разработка обхваща моделирането на данни?
- Кой са основните стратегии за създаване на модел на данни?

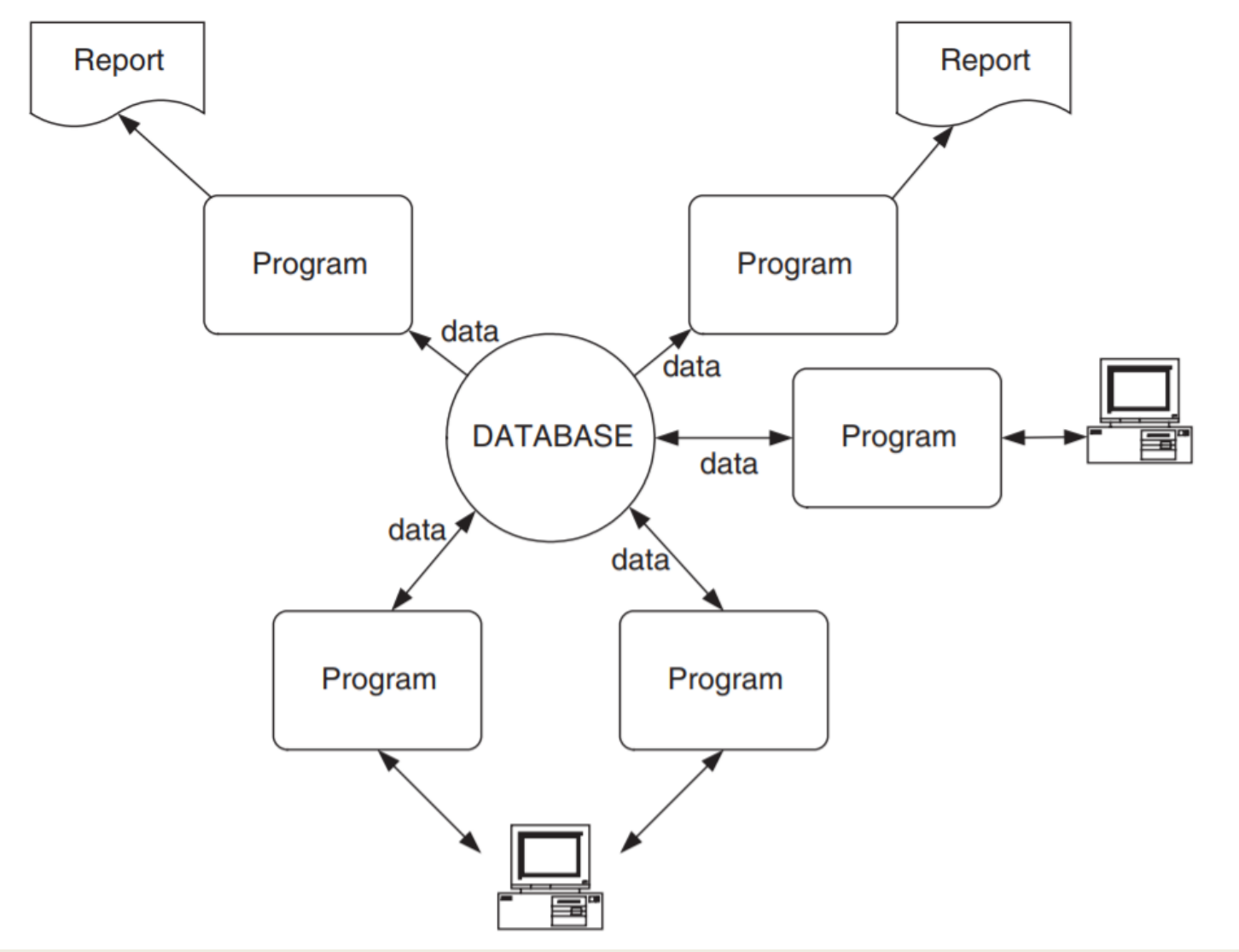
- Как се моделират данните, свързани с дизайна на базата данни?
- Кой участва в моделирането на данни?
- Какво е въздействието на новите технологии и техники върху моделирането на данни?

Какво е модела на данни?

Модел на данни

- Модела на данни е представянето на данни, необходими за поддържане на определен процес или набор от процеси. Някои учени използват "модел на данните", за да опишат конкретен начин за представяне на данни: например в таблици (Relational Model), йерархично (Object-Role Model) или като мрежа (Network Model).
- Моделирането на данни е подзадача от създаването на база от данни.

A Data-Centered Perspective



Пример 1.0

POLICY TABLE

Policy Number	Date Issued	Policy Type	Customer Number	Commission Rate	Maturity Date
V213748	02/29/1989	E20	HAYES01	12%	02/29/2009
N065987	04/04/1984	E20	WALSH01	12%	04/04/2004
W345798	12/18/1987	WOL	ODEAJ13	8%	06/12/2047
W678649	09/12/1967	WOL	RICHB76	8%	09/12/2006
V986377	11/07/1977	SUI	RICHB76	14%	09/12/2006

CUSTOMER TABLE

Customer Number	Name	Address	Postal Code	Gender	Age	Birth Date
HAYES01	S Hayes	3/1 Collins St	3000	F	25	06/23/1975
WALSH01	H Walsh	2 Allen Road	3065	M	53	04/16/1947
ODEAJ13	J O'Dea	69 Black Street	3145	M	33	06/12/1967
RICHB76	B Rich	181 Kemp Rd	3507	M	59	09/12/1941

Първи преглед на дата модела

- Информацията е разделена в две таблици Policy и Customer.
- На пръв погледни никакви технически специфики по дата модела не се забелязват

Подробен преглед на дата модела

- Какво представлява Customer? Може би е човек, който би се възползвал от застрахователната полица или е човека, който ги изплаща. А може и да става дума за семейство. В тези случай как колоните Age, Gender и Birth Date биха се интерпретирали?
- Наистина ли е необходимо да пази Age? Нямали да е по - лесно да го изчисляваме от колоната Birth Date?

- Commission Rate винаги ли е една и съща за определен Policy Type? Например E20 е 12% за всеки ред. Ако е това е вярно, записването на тази информация многократно е безсмислено. И как бихме могли да запишем нов Commission Rate за нов Policy Type, ако все още не сме продали такъв?
- Customer Number е изграден от името на клиент + двуцифрен номер, за различаване на клиентите с еднакви имена. Добър избор ли е?
- Нямали да е по – добре да се държат отделно инициалите от фамилията? Колона Name.
- Форматът на попълнената информация в колона Address не е консистентен. „Road“ и „Street“ са написани по няколко различни начина.

Защо дата модела е важен?

Leverage

- Малка промяна в модела може да окаже сериозно въздействие върху системата като цяло. Система работи със съхранената информация. Лошо съхранена информация може да окаже сериозно забавяне в системата, заради добавянето на допълнителни обработки.

Постоянство (Conciseness)

- Моделът за данни е много мощен инструмент за изразяване на изискванията и възможностите на информационните системи. Силата му се дължи отчасти на неговата кратка прецизност. Разбирането на модел за данни изисква много по – малко време от анализ на цяла система.

Качество на данните (Data Quality)

- Данните, съхранявани в база данни, обикновено представляват ценен бизнес актив, изграден в продължение на дълъг период от време. Неточните данни (лошото качество на данните) намаляват стойността на актива и могат да бъдат скъпи или дори невъзможни за коригиране.

Какво прави модела на данни добър(ефективен)?

Пълнота (Completeness)

- Модела трябва да подържа цялата информация необходима за разработката на софтуер.

Ако вземем за Пример 1.0 можем да кажем, че липсва част от информацията и модела не е пълен. Например колона за заетост на клиента или колона за записване на премии. Ако такава информация е необходима на приложението. Друг пример е невъзможността за запис на комисионна без да е била продадена застрахователна полица с нея.

Несъответствие (Non Redundancy)

- Информацията трябва да бъде съхранявана само на едно място и да се избягват повторенията.

Пример за това са колоните Age и Birth Date, които носят една и съща информация. Както и запазването на Commission Rate на повече от един ред. Всичко това изисква специална поддръжка на информацията при нужда от промяна. Това от своя страна увеличава и нуждата от по – голяма пространство и повече изчислителни ресурси.

Прилагане на бизнес правилата (Enforcement of Business Rules)

- В основаната си част модела на данни налага определени бизнес правила върху системата. Добре наложените правила правят системата ефективна.

Като пример можем да погледнем таблицата Policy. Която определя, че всяка застрахователна полица може да бъде притежава от точно един клиент на системата. На всеки ред в таблицата Policy е записан точно един Customer Number.

Преизползваемост на данните(Data Reusability)

- Модела на данни трябва да предоставя възможност за преизползване на информацията.

Първоначално застрахователната компания може да записва данни за застрахователните полици в подкрепа на функцията за таксуване. Но е възможно отделът за продажби да желае да използва данните при изчисляване на комисионните; както и маркетинговия отдел би желал да следи демографските показатели и т.н.

Стабилност и гъвкавост(Stability and Flexibility)

- Модела на данни трябва да има възможност за надграждане при промяна на бизнес изискванията.

Пример 1.0 е малко вероятно да се справи с разширение на асортимента на застраховките или някаква промяна. При промяна такава промяна на изискванията за предпочитане е изнасянето на Policy Type в отделна таблица.

Елегантност (Elegance)

- Ако таблицата ни за клиенти включваше само осигурени лица, а не бенефициенти, може да имаме нужда от отделна таблица на бенефициента. За да избегнем записването на факти за един и същ човек в двете таблици, ще трябва да изключим бенефициентите, които вече са били записани като клиенти. А таблица Beneficiary тогава ще съдържа "бенефициенти, които не са клиенти", това много вероятно ще доведе до тромава система. Елегантното решение би било създаване на таблица Person.

Общуване (Communication)

- Модела трябва да бъде максимално опростен за да предоставя лесно и разбираемо бизнес логиката на разработчиците на системите. Качеството на окончателния модел ще зависи в голяма степен от информирана обратна връзка от бизнес хората и разработчиците.

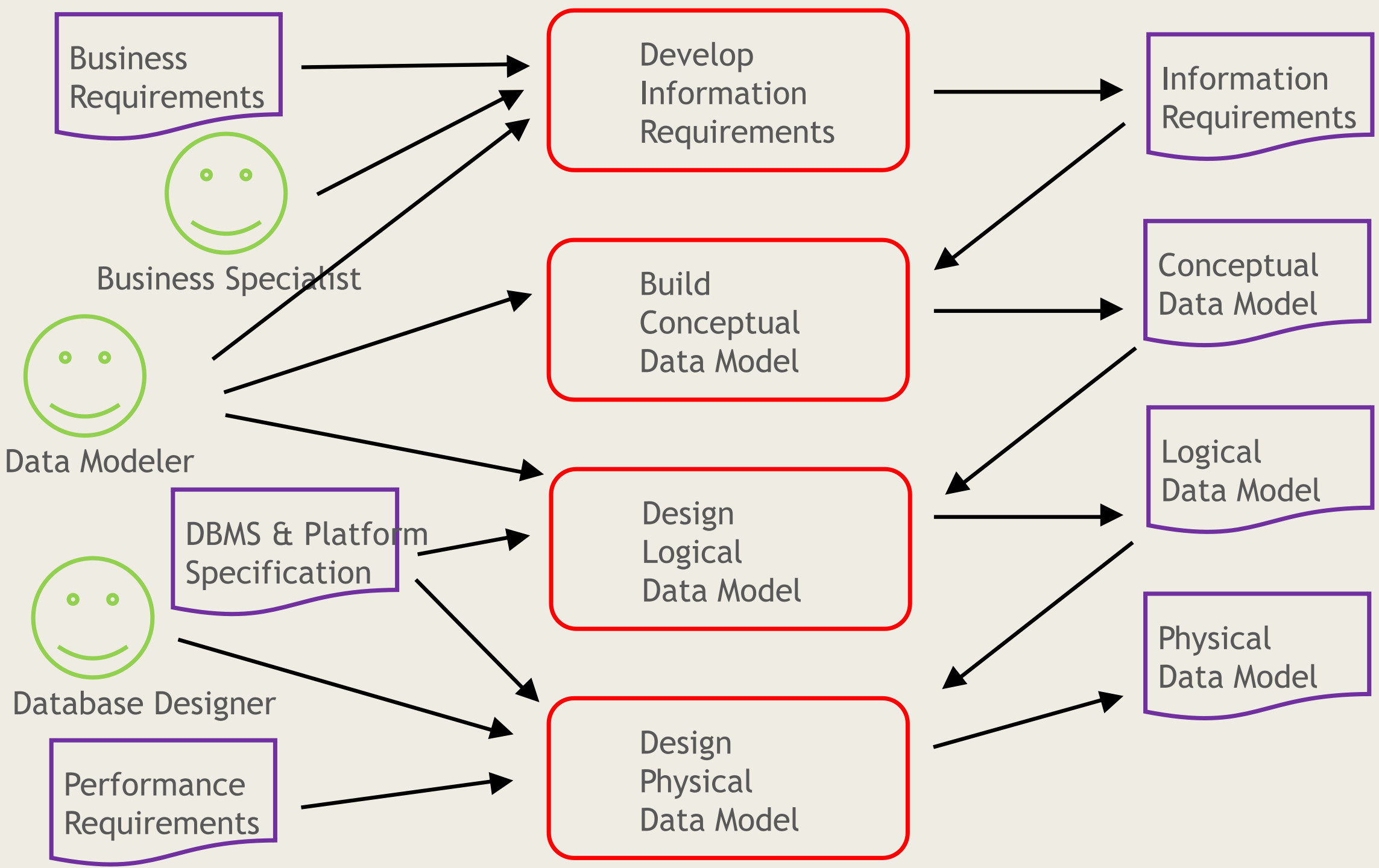
Интеграция (Integration)

- Модела на данни трябва да предоставя възможност за интеграция и разширение с други вече съществуващи модели на данни в системата.

Конфликтни цели (Conflicting Objectives)

- Описаните по горе 8 цели в голямата си част могат да бъдат в конфликт една с друга. Нашата обща цел е да разработим модел, който осигурява най-добрия баланс сред тези евентуално противоречиви цели.

Коя точно част от софтуерната разработка обхваща моделирането на данни?
Кой участва в моделирането на данни?


Business Requirements
Business Specialist
Data Modeler
DBMS & Platform Specification
Database Designer
Performance Requirements
Develop Information Requirements
Build Conceptual Data Model
Design Logical Data Model
Design Physical Data Model
Information Requirements
Conceptual Data Model
Logical Data Model
Physical Data Model

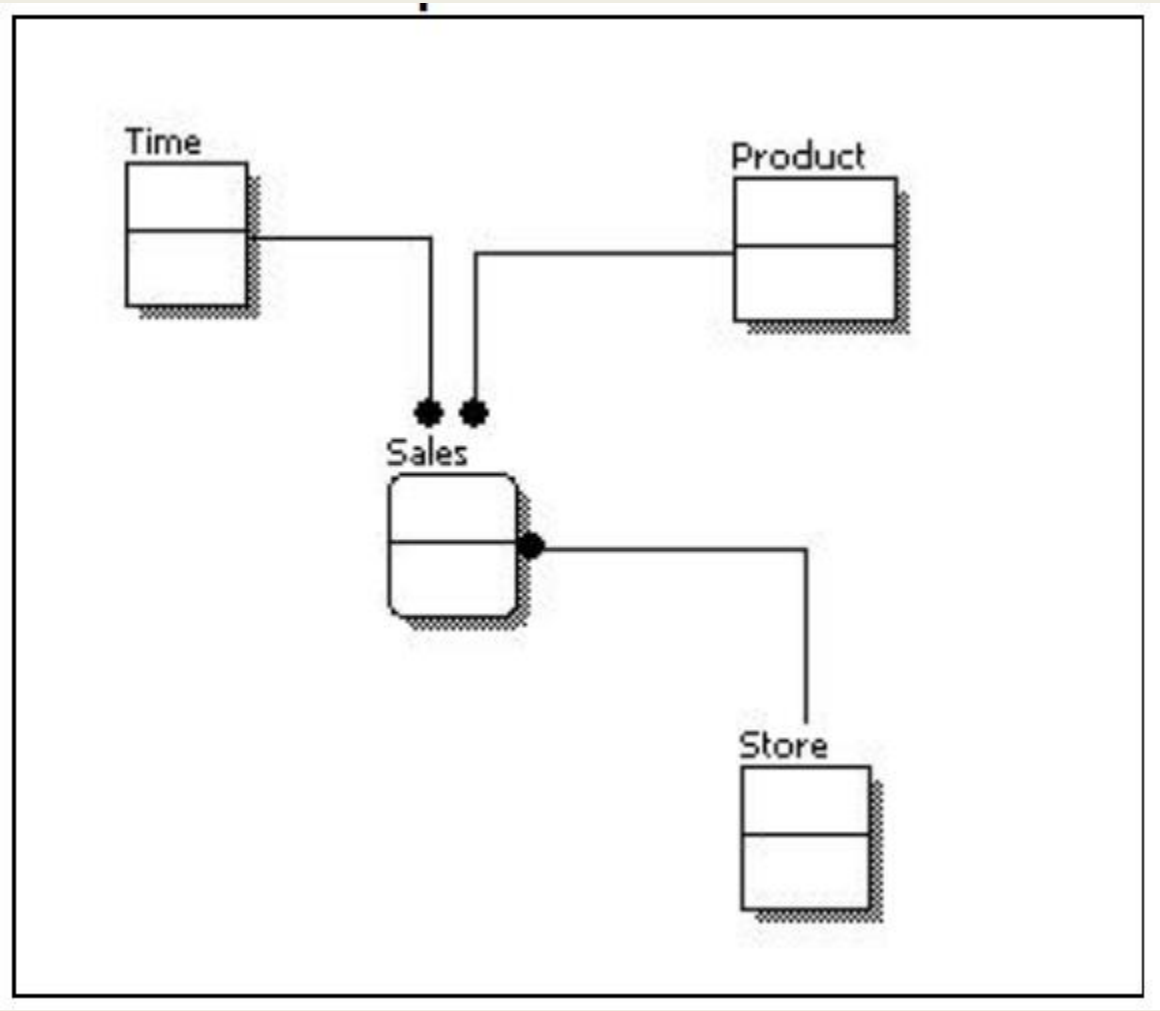
Концептуален модел на данните (Conceptual Data Model)

- Концептуалния модел на данни е (относително) независима от технологията спецификация на данните, които се съхраняват в базата данни. Фокусира се върху комуникацията между модератора на данни и заинтересованите страни от бизнеса и обикновено се представя като диаграма с подкрепяща документация.

Особености

- Включва важните обекти и взаимоотношенията между тях.
- Не се посочват всички атрибути. Само определящите, ако е необходимо.
- Не се задават първични ключове.

Концептуален модел на данните



Логически модел на данните (Logical Data Model)

- Логическият модел за данни описва данните с възможно най-големи подробности, без значение от това как те ще бъдат физически приложени в базата данни.

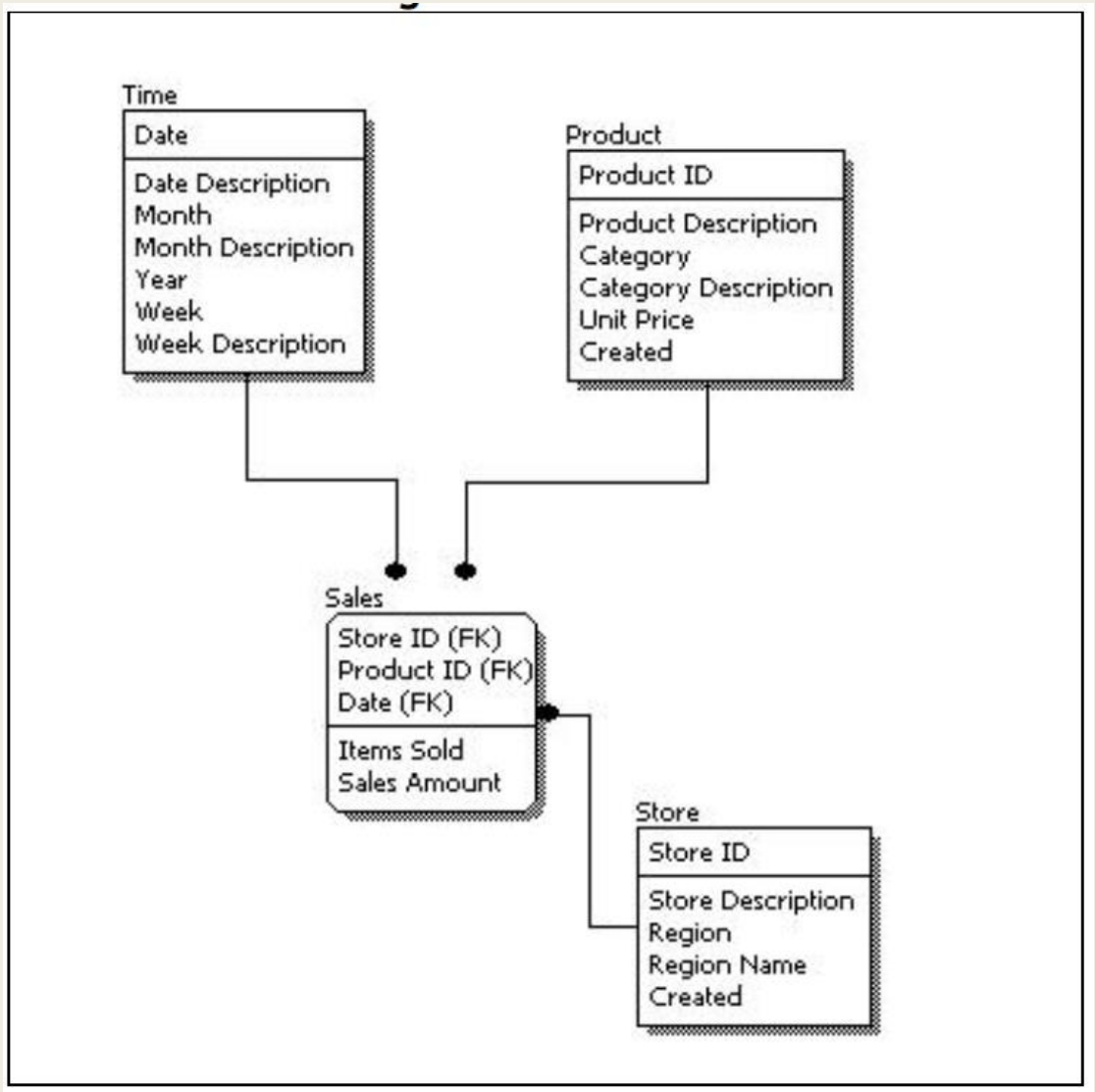
Особености

- Включва всички обекти и отношения между тях.
- Всички атрибути за всеки обект са посочени.
- Определен е първичният ключ за всеки обект.
- Изброяват се чужди ключове.
- Нормализирането се извършва на това ниво.

Стъпки по стъпка

- Посочете първичните ключове за всички обекти.
- Намерете връзките между различните обекти.
- Намерете всички атрибути за всеки обект.
- Решаване на много-много отношения.
- Нормализиране.

Логически модел на данните



Физически модел на данните (Physical Data Model)

- Физическият модел представя как моделът ще бъде изграден в базата данни. Физическият модел на база данни показва всички таблични структури, включително име на колона, тип на колона, ограничения на колоните, първичен ключ, чужд ключ и отношения между таблици.

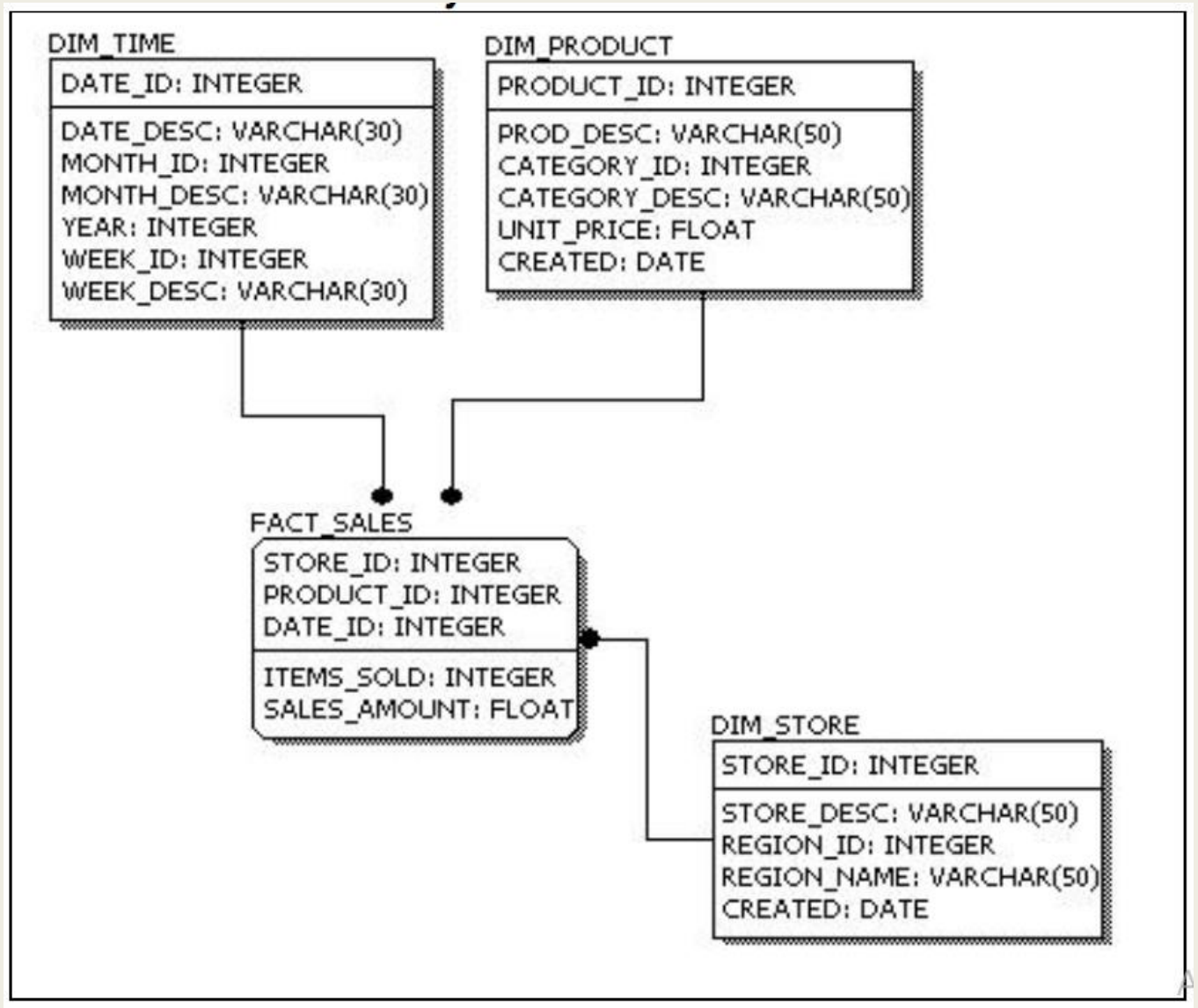
Особености

- Спецификация на всички таблици и колони.
- Чуждите ключове се използват за идентифициране на взаимоотношенията между таблиците.
- Денормализация може да възникне въз основа на изискванията на потребителя.
- Физическите съображения могат да накарат физическия модел на данни да бъде съвсем различен от този на логическия модел.
- Физическият модел ще бъде различен за различните RDBMS. Например: MySQL и SQL Server.

Стъпки по стъпка

- Преобразувайте обектите в таблици.
- Преобразувайте връзките в чужди ключове.
- Конвертирайте атрибутите в колони.
- Промяна на физическия модел на данни въз основа на физически ограничения / изисквания.

Физически модел на данните



Особености на моделирането

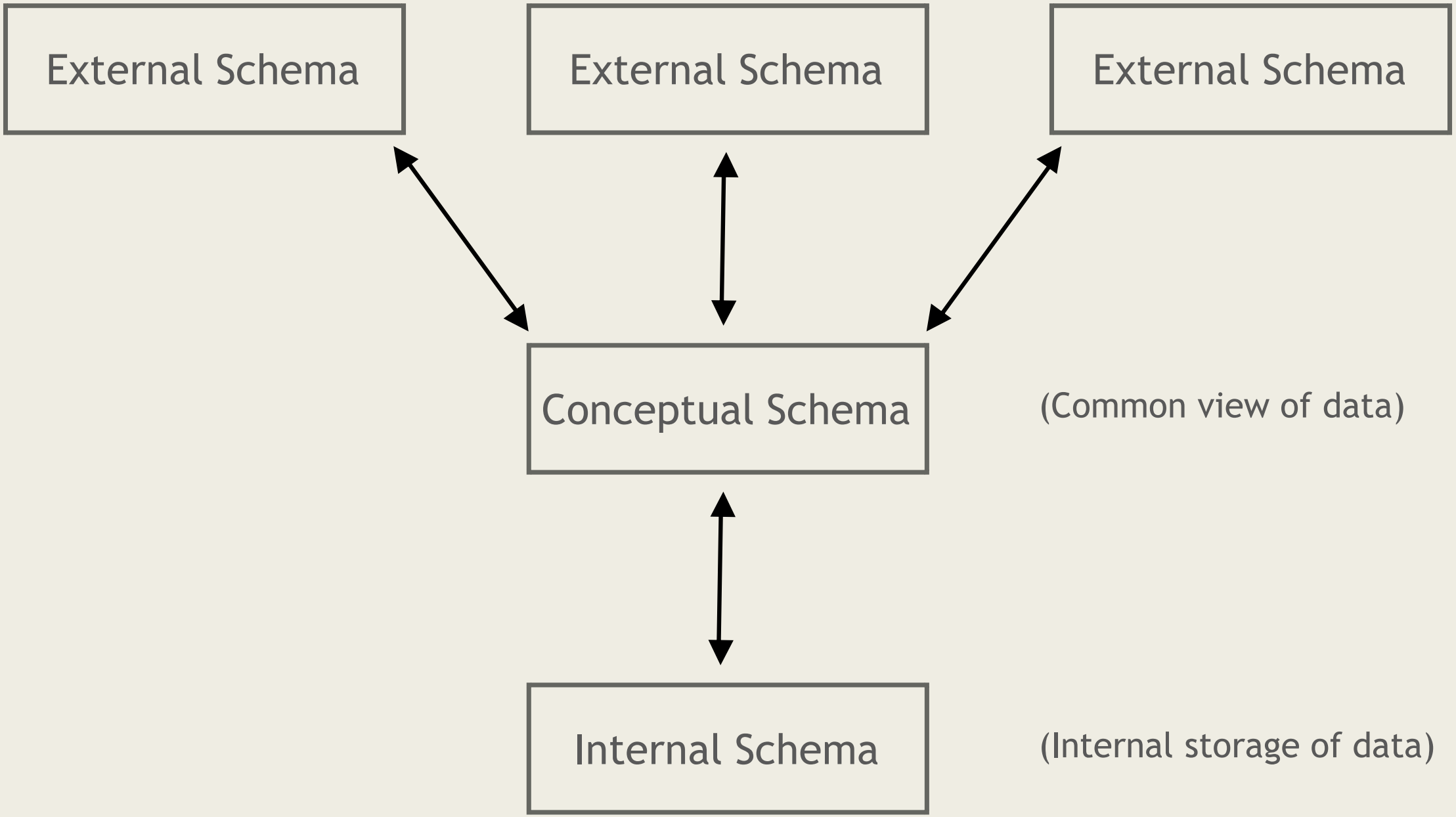
Особености	Концептуален	Логически	Физически
Entity Names	✓	✓	
Entity Relationships	✓	✓	
Attributes		✓	
Primary Keys		✓	✓
Foreign Keys		✓	✓
Table Names			✓
Column Names			✓
Column Data Types			✓

The Three-Schema Architecture and Terminology

Трислойната архитектура е поддържана от всички популярните DBMS,а това предоставя две предимства:

- Той изолира програмистите и крайните потребители на базата данни от начина, по който данните се съхраняват физически в компютъра.
- Позволява на различни потребители на данните да виждат само подгрупата от данни, които са от значение за тях, организирани така, че да отговарят на техните конкретни нужди.

(User views of data)



(Common view of data)

(Internal storage of data)

Кой са основните стратегии за създаване на модел на данни?

Моделирането на данни взема участие във всяка една методология за работа. Разликата е дали е в началото, края или паралелно с процес модел.

- Process-Driven Approaches
- Data-Driven Approaches
- Parallel (Blended) Approaches
- Object-Oriented Approaches
- Prototyping Approaches
- Agile Methods

ВЪПРОСИ ?