

Компютърни технологии за управление на информация



● Технология

практика по приложение на теоретични методи и технически средства за решаване на проблеми на хората

● Компютърна технология

практика по приложение на **математически** методи и **компютърни** средства за решаване на **проблеми на хората**

● Приложение

■ конкретна реализация, решение

● Бизнес-приложение

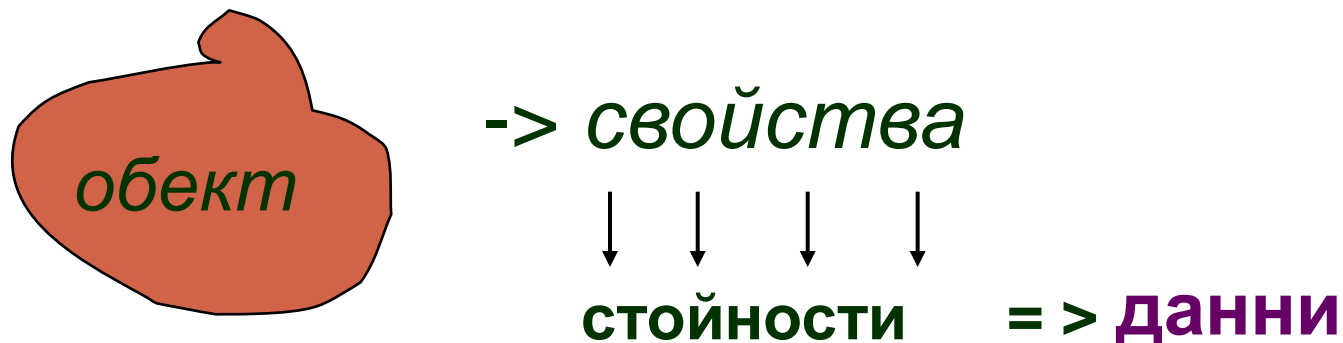
■ приложение за решаване на проблеми в бизнеса



Основни зависимости

- Компютърът е **средство** на човека, инструмент, машина
 - **идеята**, инициативата, програмата за прилагане идва от човека, а не от инструмента
- Компютърът е машина за
 - обработка на **данни и информация**
- Компютърът се състои от
 - устройства и **програми** (hardware + software)
- **Информация** е резултатът от обработка на **данни** посредством **програми**





данни: *регистрирани факти - стойности на свойства на обекти (числа, думи, код, ...)*

предназначение: *за обработка от компютър*

пример:

обект	прозорец	
свойства	<i>ширина</i>	<i>височина</i>
данни	1.60	1.20 (числа реални)



Информация

- **тълкуване, интерпретация, придаване на значение на данните**
- **значението зависи от ситуацията на използване**
- **предназначение: за разбиране и използване от хора, в съответствие с техни цели**
 - за вземане на решения
 - за предприемане на действия
- **качества:**
 - адекватност
 - пълнота
 - точност
 - актуалност



- **Данни – регистрирани стойности, обработвани от компютър**
- **Информация – интерпретация на данните, използвана от хора**
- **Информацията е резултат от операции на компютъра с данните**



Обработка на данни

- Събиране
- Регистриране
- Съхраняване
- Оперирание
- Представяне
- Разпространяване



Технологии

● Йероглифи, стенописи

...

● Книги, тетрадки, тефтери

...

● Сметала, калкулатори

...

● Компютри



Представяне като масив от записи

Type

```
row = record
```

```
    Key : integer;
```

```
        lme : string[40];
```

```
        Pol : boolean;
```

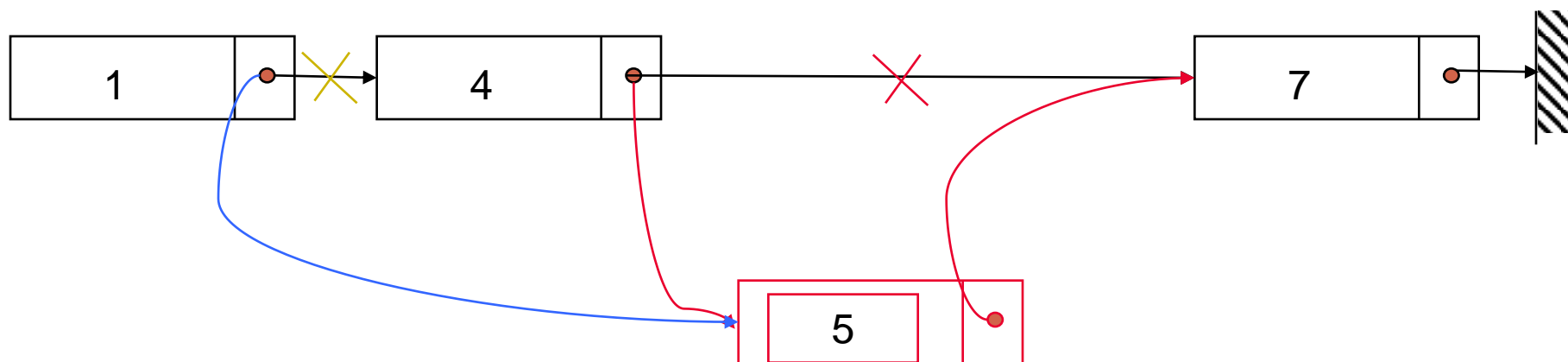
```
    ...
```

```
end;
```

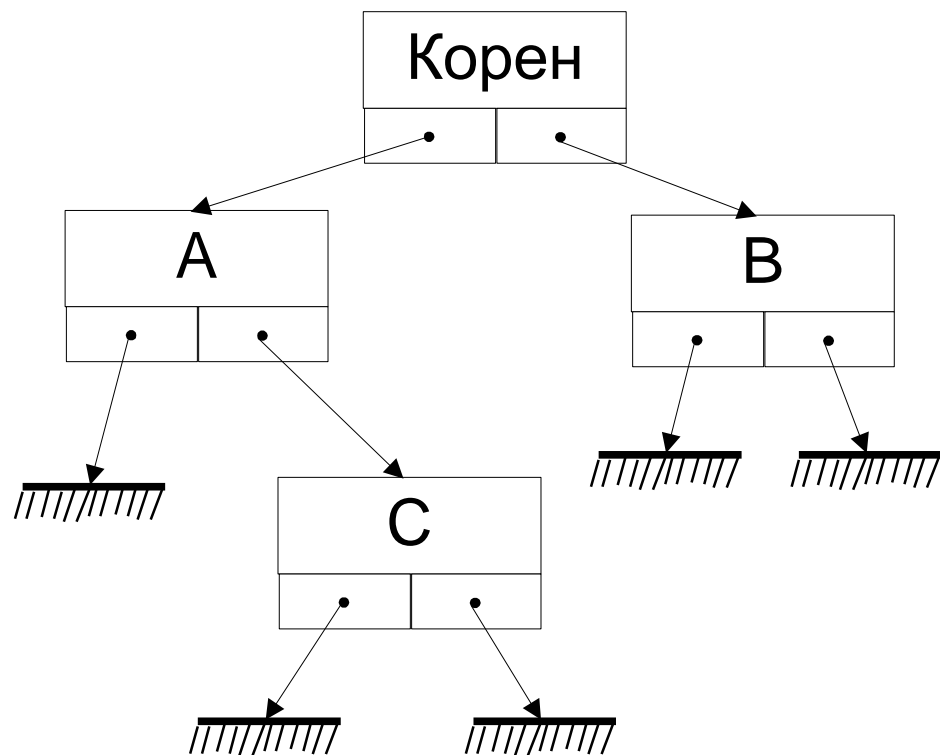
```
Table = array [1..Maxrow] of row;
```



Представяне като линеен списък



Представяне като двоично дърво



B и B+ дървета

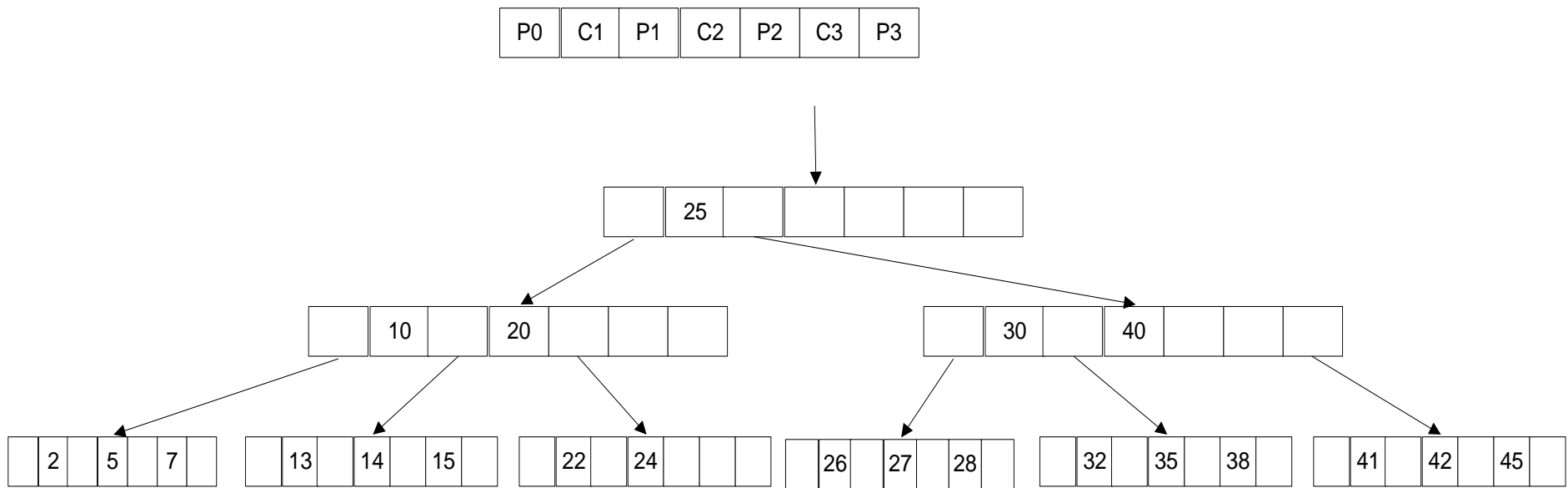
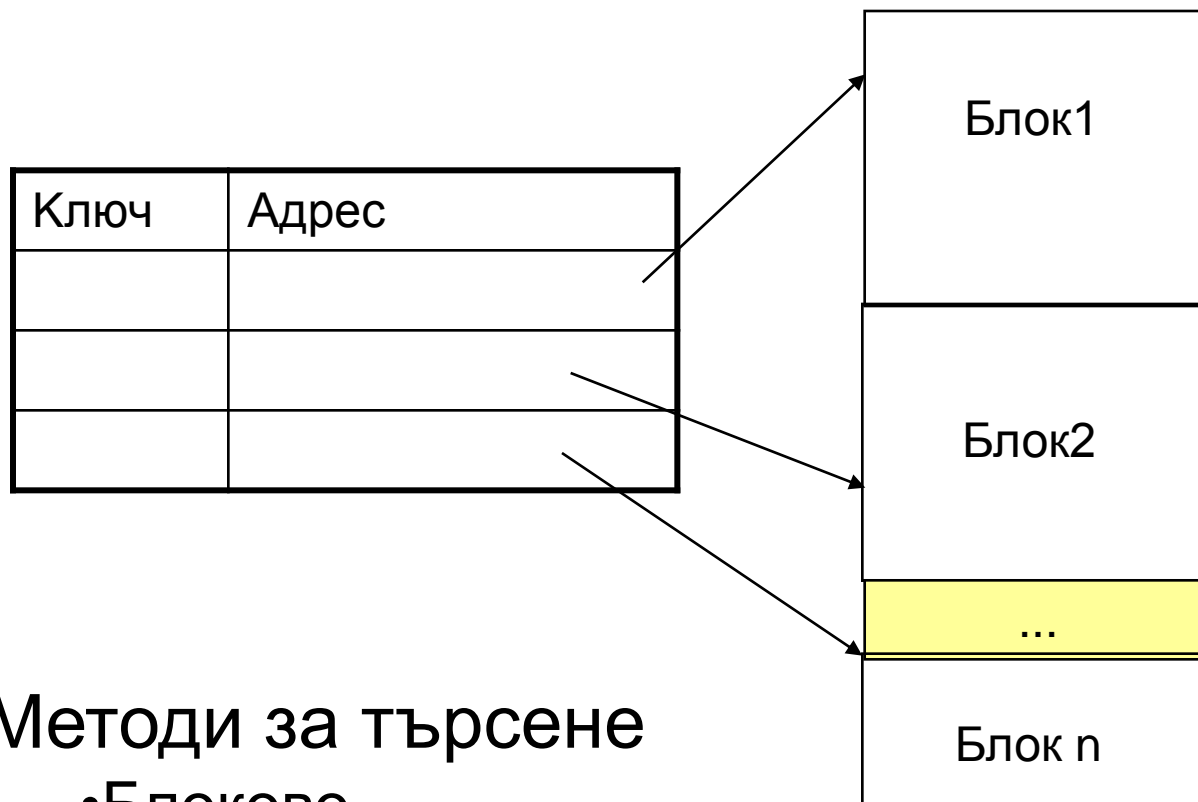


Таблица във външна памет



Методи за търсене

- Блоково
- Индексно



Файлова система

● Файлова система

програма+ файлове с данни => файл с резултати

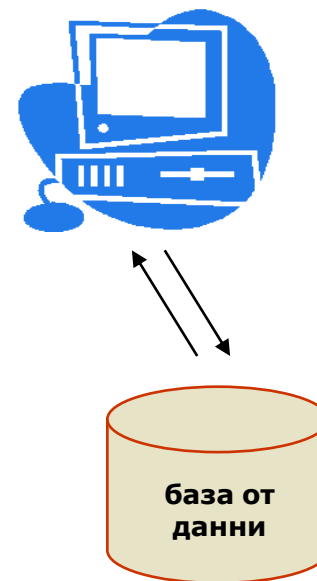
дублиране на данни

несъвместимост

трудно поддържане

● Системи с база от данни

файлове с данни + СУБД + програма => резултат



Определения

- Съвкупност от логически свързани данни, структурирани за ефективна обработка
- БД е структурирано множество от данни, записано на носител, който е достъпен за четене и запис чрез компютър с цел да обслужи едновременно много потребители поотделно и за разумно време.
- Множество от логически свързани и съхранявани заедно данни без излишък, които могат да бъдат използвани от различни приложения; начинът на съхранение е независим от приложенията и се използват общи процедури за промяна, добавяне и за търсене.
- Множество от данни структурирани според една схема от данни.
- Данни, които директно и едновременно са достъпни от различни, взаимно конкуриращи се потребители



данните са организирани в обособени части -
фамилия, име, адрес, телефон
частите са подредени в определен ред -
азбучно
в колони

т.е. данните имат **структура и контекст**

фамилия	име	адрес	телефон
Крайников	Йосиф А.	Х. Димитър 66	454529
Крайниковски	Асен И.	Ивац Войвода 20	449949
Крайнин	Христо К.	Червена роза 31	668819
Крайничанец	Драган Д.	В. Левски 52	539282



Информационни системи



Организация на хора и компютърни технологии за управление на информация

Компоненти:

- **база от данни**
- **управляващи програми**
- **потребителски интерфейс**
- **комуникационна среда**



Според областите на приложение
административно обслужване
стопанско управление
финанси и счетоводство
търговия и услуги
научни изследвания
и др.



Според източника на данни

- за обработка на данни, произхождащи от хора и организации
 - данните се въвеждат се ръчно в компютъра
- за обработка на данни, генерирани от компютър
 - програмни процедури и CASE софтуер



Според предназначението

- за обработка на данни
 - за рутинна обработка на голямо количество данни, напр. плащания
 - активират се периодично и автоматично
- за управление **MIS**
 - за операционно управление
 - за тактическо управление
 - за стратегическо управление
- за подпомагане на вземането на решения **DSS**
 - насочени към конкретна група специалисти и задачи
 - решенията се вземат от хората
- експертни системи **KBS**
 - системи, базирани на знания
 - препоръчват на хората готови решения
 - използват предварително въведени експертни знания и език за извличане
- други системи с изкуствен интелект **AI**
 - системи за откриване на знания **Data Mining**
 - системи, основани на софтуерни агенти



Участници

- Потребители **end users**
- Специалисти
 - системни анализатори
 - проектанти
 - програмисти
 - специалисти по тестване
 - системни администратори



Различни роли

Разнородни умения

Разнородни дейности

управленски

технически

аналитични

оперативни

междучелностни

социални

и др.



Системни анализатори

Role	Responsibilities
Business analyst	Analyzing the key business aspects of the system Identifying how the system will provide business value Designing the new business processes and policies
Systems analyst	Identifying how technology can improve business processes Designing the new business processes Designing the information system Ensuring that the system conforms to information systems standards
Infrastructure analyst	Ensuring the system conforms to infrastructure standards Identifying infrastructure changes needed to support the system
Change management analyst	Developing and executing a change management plan Developing and executing a user training plan
Project manager	Managing the team of analysts, programmers, technical writers, and other specialists Developing and monitoring the project plan Assigning resources Serving as the primary point of contact for the project



- Програмни средства и системи
 - за подпомагане на създаването на програмни средства и системи
- Интегрират дейностите по създаване
- Предимства от използването им
 - увеличават продуктивността на специалистите
 - подобряват комуникациите между потребителите и специалистите
- Езици за комуникация
 - графични средства – графични примитиви
 - видове диаграми



- Структура и поведение на системата и нейните компоненти

- Разглеждат се два вида архитектури

 - архитектура на системата

 - архитектура на данните

- Архитектурата на данните
влияе върху / зависи от
архитектурата на системата



● Архитектура на системата

- базата от данни + цялата програмна и комуникационна среда на използването и
- програми за търсене и визуализиране на данните, организация на комуникациите между клиенти и сървъри, организация на сървърите и т.н.

● Архитектура на данните

- модели за структуриране и управление на данните
- съдържание: структура + език за описание на поведението и правилата



Системна архитектура

● Абстрактна архитектура на обектите в системата и отношенията между тях

● Видове архитектури

- клиент-сървър
- трислойна
- разпределена
- обектно-разпределена
- и др.



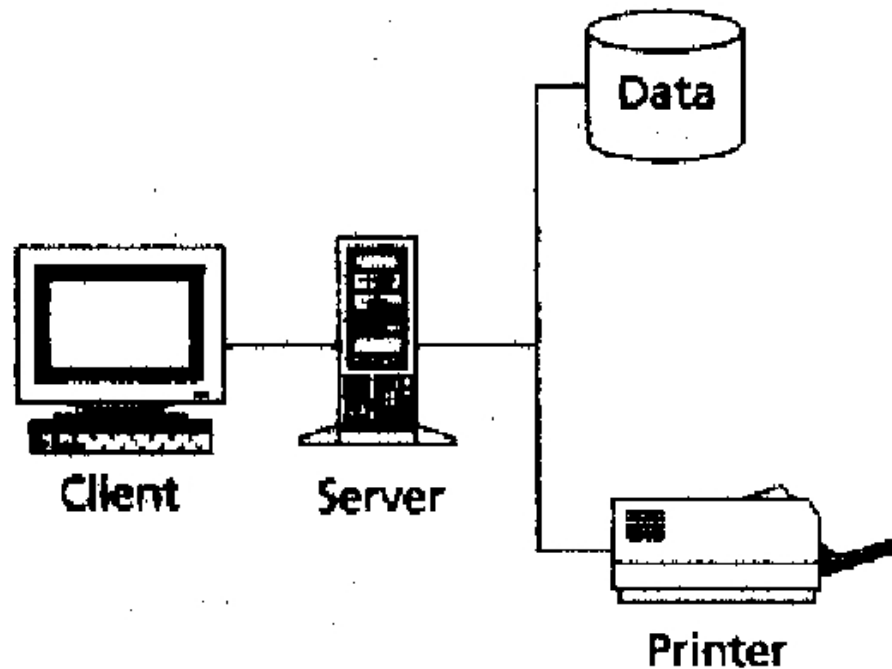
Клиент-Сървер

Възниква във връзка с развитието на компютърните комуникации, 1992

Системата се дели на две части:

клиенти, които ползват данните

сървър, който обслужва заявки на клиентите



Клиент-Сървер

- Цел: намаляване на трафика от данни по мрежата
 - вместо цял файл, по мрежата се предават само необходимите на клиента данни
- Решение: разделяне на задачите на три:
 - визуализация;
 - обработка;
 - управление на данни.
- Влияе върху логическото и физическото ниво на проектиране на БД



- Приложната програма на сървъра започва първа и чака контакт от клиента
- Клиентът се свързва със сървъра по местоположението му и инициира комуникация
- Клиентът и сървърът си обменят съобщения (данни)
- Клиентът и сървърът си изпращат eof за прекъсване на комуникацията



● На приложно ниво: две приложни програми (приложения)

■ На клиента

активно инициира комуникацията

■ На сървъра

пасивно чака КОНТАКТ



Разпределение:

визуализация на данни -

web browser

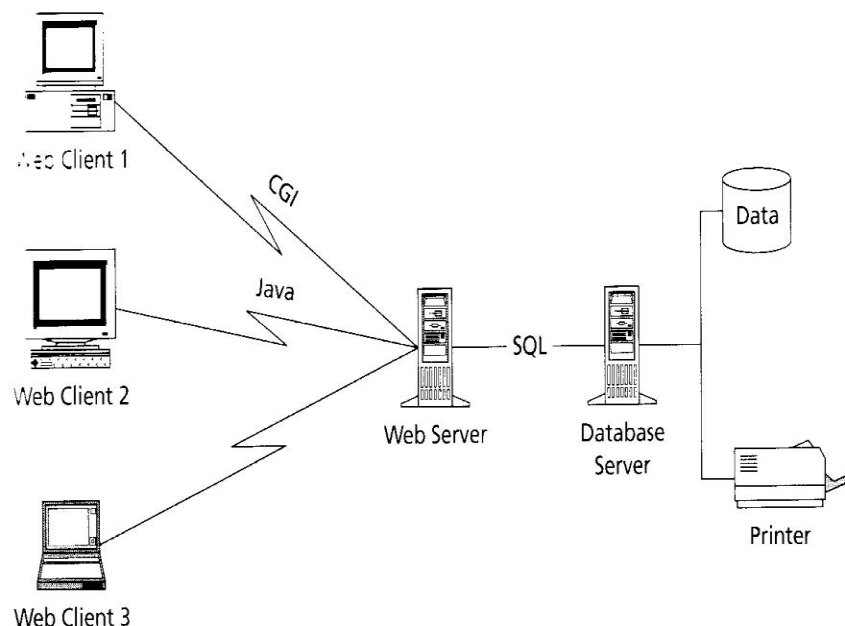
управление на данни - DB-

сървър

обработка на данни: web-

сървър + DB-сървър

Web-сървър: клиент на
DB-сървър

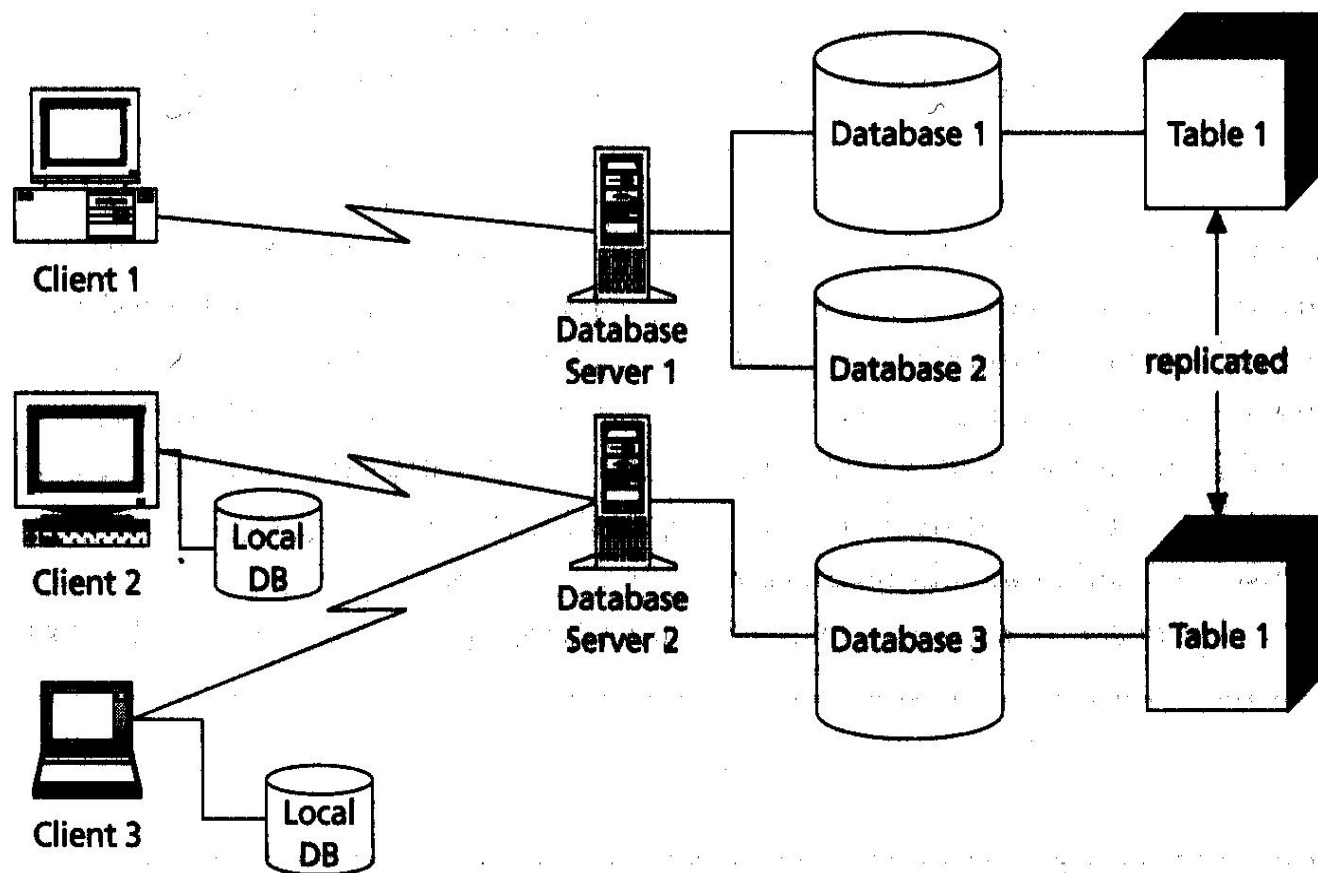


Трислойна архитектура клиент-сървър

● Програмни средства:

- потребител / web browser: **GUI**
- web browser / web-сървър: **script**
- Web-сървър / DB-сървър: **SQL**

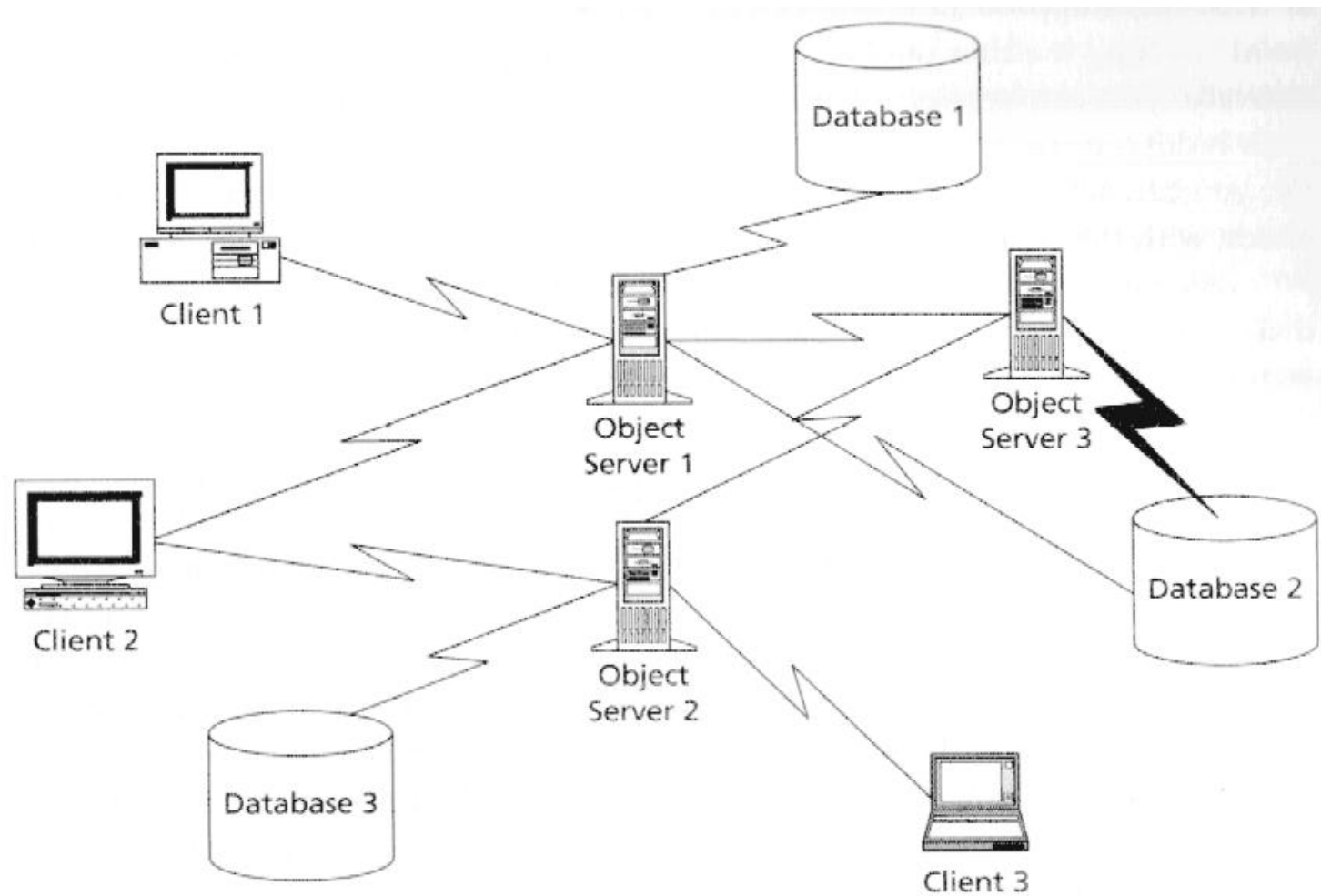




Разпределена архитектура

- Данните са разпространени в мрежа
- Свързват се посредством комуникационни връзки
- Поддържа се синхронизация *replica*
- Прилагат се в складове за данни *Data Warehouse*
- Три особености на разпределена база от данни:
 - прозрачност (transperancy) на операциите
 - управление на транзакциите
 - оптимизация на заявките, изпълнявани от различни сървъри
- Влияе върху физическото ниво на проектиране на БД чрез изискванията за фрагментация на данните и конкуренция на заявките





● Common Object Request Broker Architecture CORBA, 1992

● Софтуерна технология за разделяне на приложенията на части, изпълнявани от различни сървъри

● Влияе върху всички нива на проектиране на БД



● Съдържание

- структура на данните
- описание на поведението им
- правила за отношенията между тях

● Влияе върху:

- логическата структура на данните
- физическата организация на базата от данни
- езика за достъп до данните
- структурата на приложенията



Три-схемна архитектура

- ANSI / SPARC, 1975
- Системата се представя чрез **3 модела**:
 - външен
 - концептуален
 - вътрешен
- Схема: описание на модел (метамодел)
- Всяка схема = структура + отношения
- Схемите са независими помежду си
- Резултат: програмна и апаратна независимост на данните



ВЪНШНИ - визуализация на данните за потребителите

● концептуален - логическа организация на данните

● вътрешен - физическа организация на запомнящата

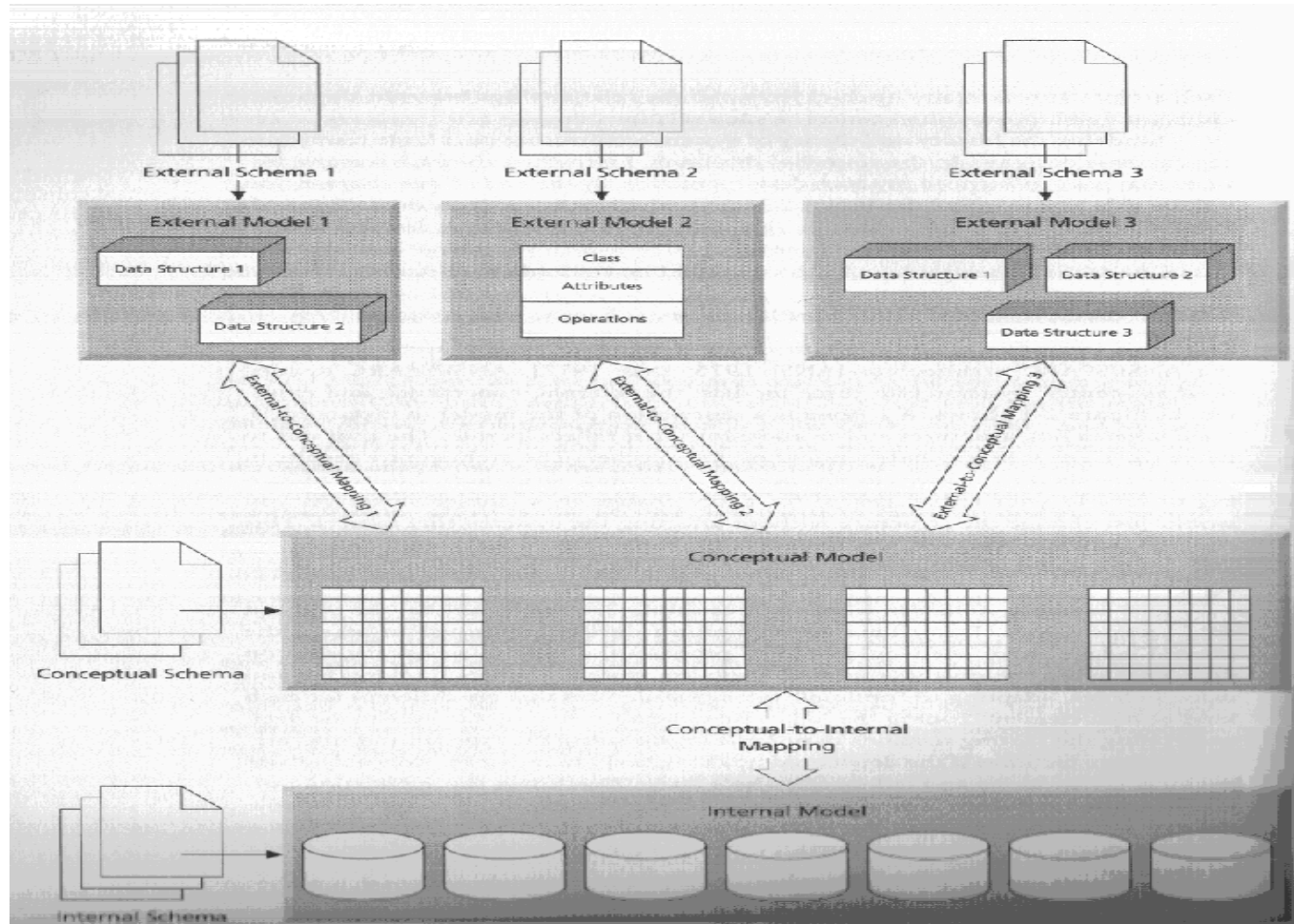
среда



ВЪНШНИ

КОНЦЕПТУАЛЕН

ВЪТРЕШЕН



● Според технологията на реализация

■ релационни - **RDBMS**

таблици, правила + SQL

■ обектно-ориентирани – **OODBMS**

класове и методи на класовете

■ обектно-релационни - **ORDBMS**

релационни структури + допълнителни типове обекти и методи,
описващи тяхното поведение



● Определение

- Формален процес на създаване на информационна система, следващ определени правила, процедури и стандарти

● Съдържание

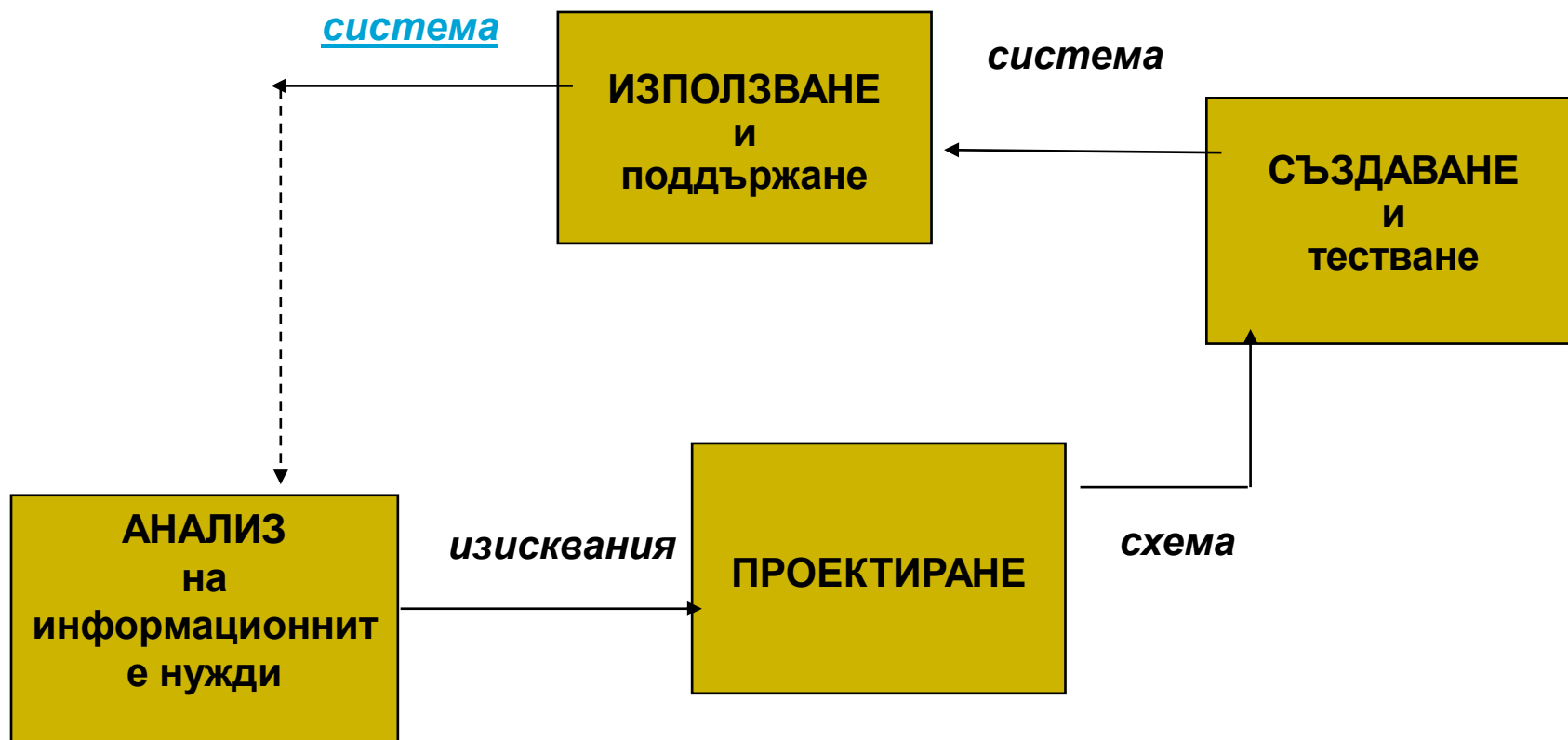
- състои се от етапи (фази)
- всеки етап има
 - определена цел
 - средства и
 - изпълнители



- всеки етап се състои от стъпки, които
 - водят до постепенно уточняване
 - завършват с конкретен резултат
 - прилагат определени технологии
 - се документират



Жизнен цикъл



- Цел: дефиниране на изисквания
- Обект: потребителите и техните информационни нужди
 - “крайни” потребители: ползватели
 - “системни” потребители: програмисти, администратори
- Информационни нужди -> изисквания към БД
 - ползвателите задават **съдържанието** на БД
 - професионалистите определят **структурата** на БД



Въпроси

- защо е необходима информационна система, каква ще бъде нейната стойност за бизнеса?
- кой и как ще я използва?
- къде и кога ще бъде използвана?
- кой и как ще я създаде – план, персонал, техника, управление



Основни стъпки

- стратегия за анализ
- събиране на данни
- дефиниране на изисквания
- инициране на проект



Вътрешен модел

- Единствен физически модел
- Осигурява трансформирането на структурите на концептуалния модел в запомнящи структури
 - файлове, записи, полета, променливи
- Осигурява методи за търсене и достъп до данните
 - индексирание, сортиране
- Осигурява физическа независимост на данните
 - промяна на външен модел не изисква промяна на физическия модел



Структури памет

Методи за достъп

Оптимизация и настройка

Управление на риска



Основни дейности

- Събиране на данни – събиране на документи, проучване, наблюдение, интервю
- Анализ на фактите – статистика, приоритети, сложност, трудоемкост, риск
- Установяване на недостатъци на съществуващата система
- Изследване за осъществимост
- Инициране на проект



● Резултати

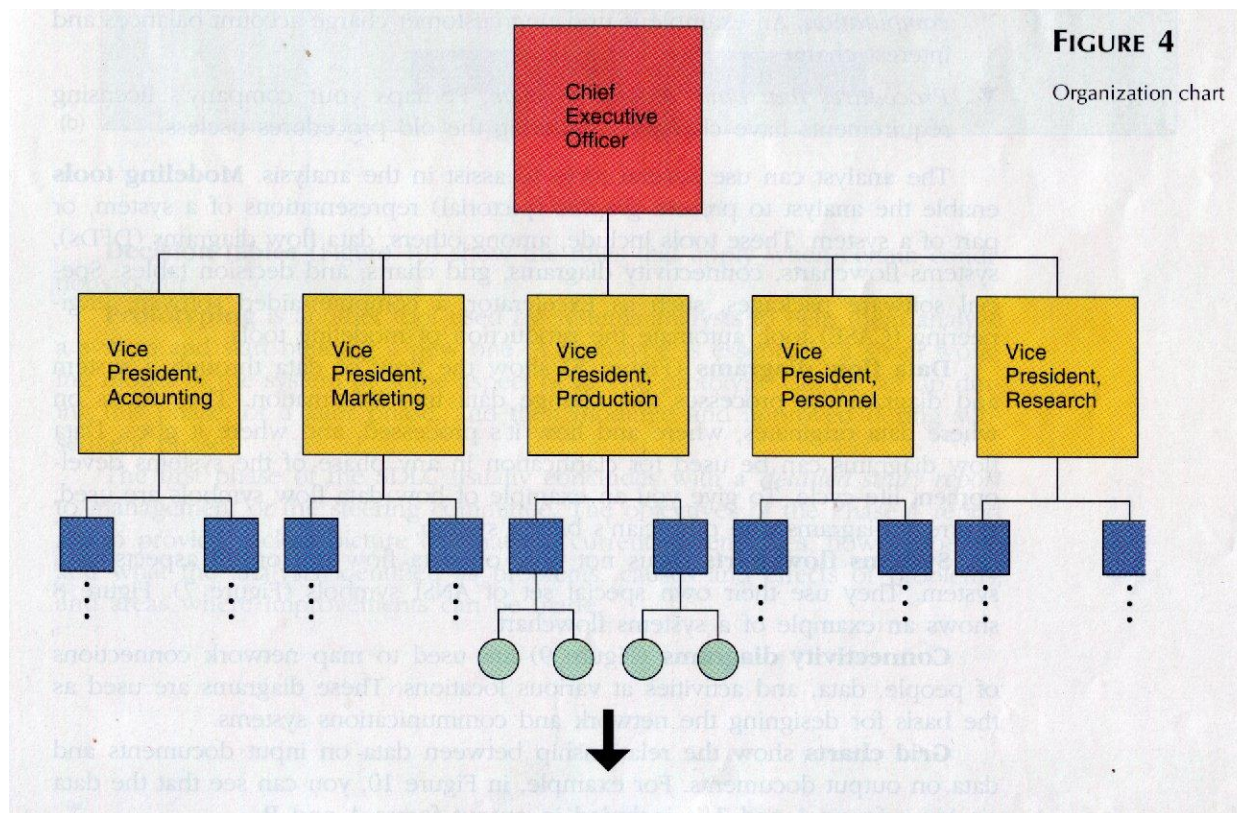
- Дефиниране на изисквания – БД, софтуер, хардуер
- Избор на алтернативен модел
- Изготвяне на прототип, който се оценява от потребителите

● Средства

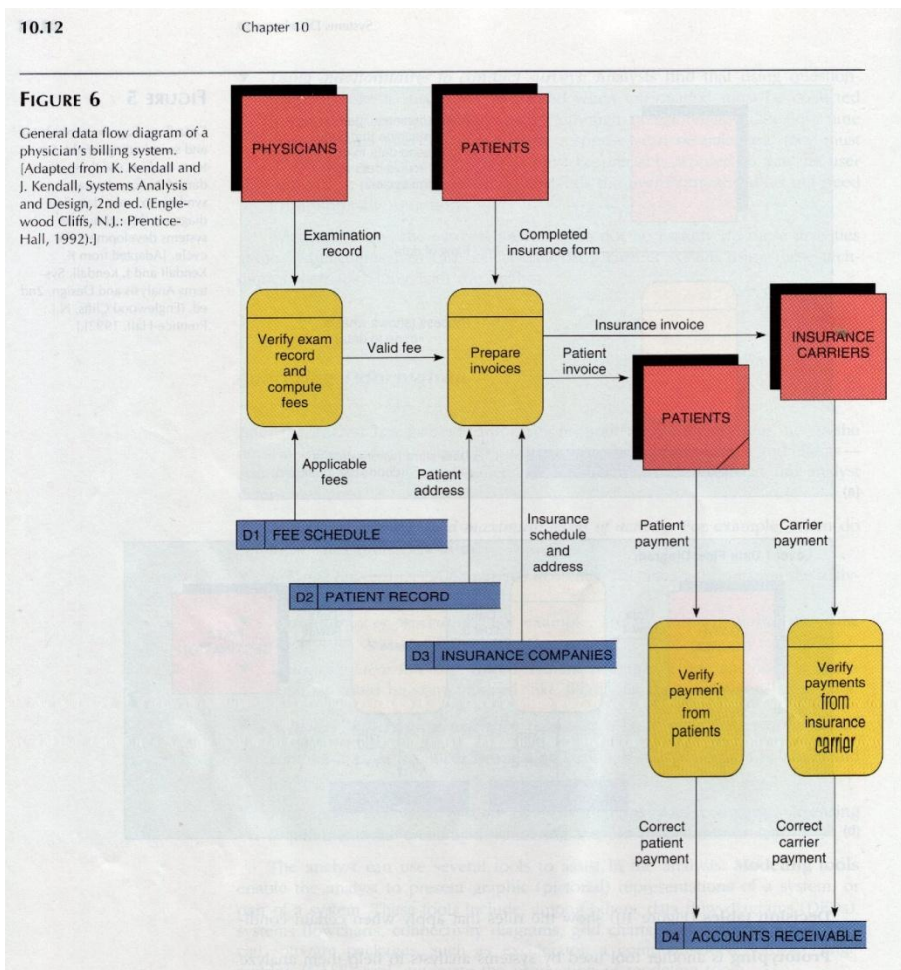
- DFD, flowcharts, connectivity, grid, decision table, prototypes



• Організаційна діаграма



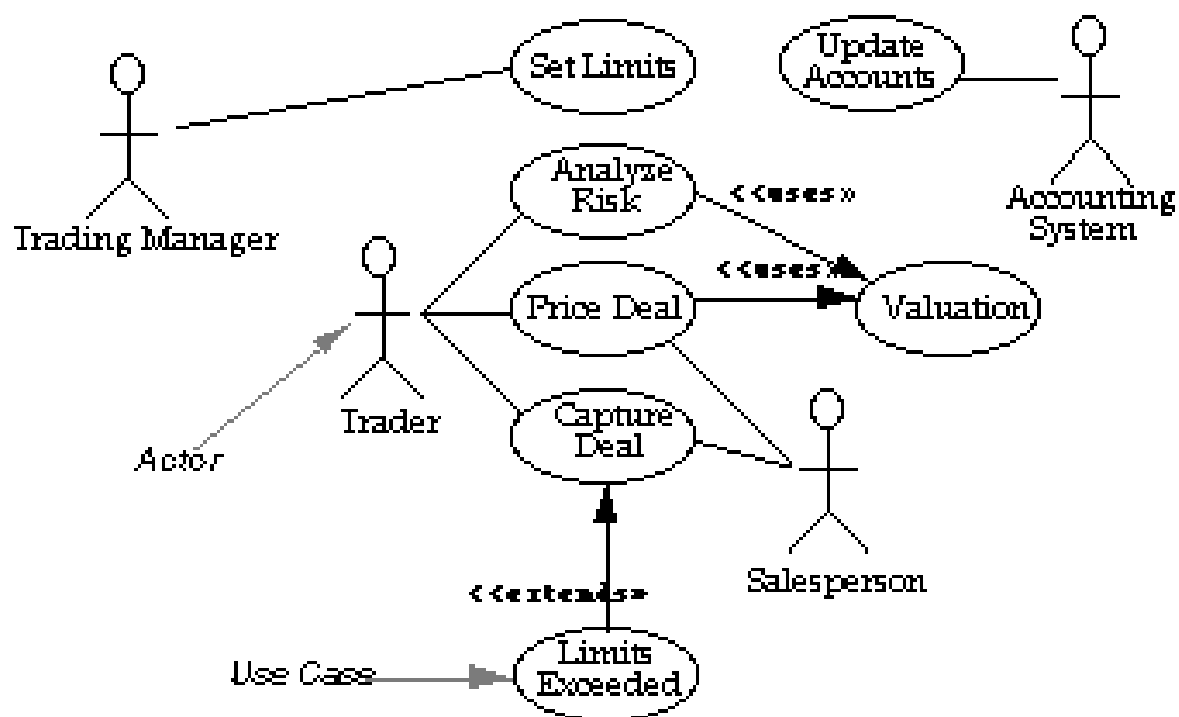
Data Flow Diagram DFD



- **Interface** – външен потребител на системата, който може да дава данни или да получава информация (напр. Лекар, Пациент, Здравна каса)
- **Process** – дейност, при която се генерира, използва, трансформира или изтрива информация (напр. Преглежда, Плаща, Пише)
- **Data store** – контейнер за информация. Има връзка с процеси – процесът ползва информация или я променя (Ценоразпис, Картон, Запис в ЗК).
- **Data flow** – обмен на информация между процеси, дискретни пакети от информация, които се предават от един процес на друг (напр. запис на данни от прегледа, търсене на цена, попълване на формуляр)



- UML Uniform Modeling Language



● Елементи

■ Actor

- Роля на външен за системата обект, който взаимодейства с нея
- Един физически обект може да играе няколко роли – всяка роля се моделира отделно

■ Use Case

- Транзакция – логическа единица работа на актьор
- За всеки актьор се определя кои транзакции инициира

● Подобно на сценарий, описват какво трябва да се прави

● Показват

- транзакциите в системата
- използване на данните, прилагане на ограниченията
- ролята и значението на БД в цялата система
- средство за контрол и оценка на софтуер



● Таблица на решенията (Decision Table)

Conditions

IF ... and IF ... (Y Y N Y N)

Decision Rules (Then Do ...)

● Таблица на операциите (Grid Chart)

■ вход / изход

● Прототипи на интерфейсите и задачите

■ документи

■ екрани



● Цел

- Изготвяне на проект
- Установяване на среда за функциониране и контрол

● Дейности

- Стратегия на проектирането
- Избор на системна архитектура – хардуер, софтуер, мрежи
- Проектиране на базите от данни
- Проектиране на приложните програми

● Резултат

- схема на БД
- алгоритми

● Средства

- CASE (Computer Aided Software Engineering)



Въпроси

- как ще оперира системата?
- в каква среда ще оперира
 - хардуер
 - софтуер
 - мрежи
- какъв потребителски интерфейс е необходим
 - екранни форми
 - печатни документи
- какви специфични програми и данни са необходими?



- Различни модели за различни потребители

Nancy Davolio

Company Info | Personal Info

Employee ID: 1

First Name: Nancy


Last Name: Davolio

Title: Sales Representative

Reports To: Fuller, Andrew

Hire Date: 01-May-1992

Extension: 5467



Add/Change

Re **Bill To:** Alfreds Futterkiste

Ship To: Alfreds Futterkiste

Obere Str. 57

Berlin 12209

Germany

Salesperson: Peacock, Margaret

Ship Via: Speedy United Federal

Order ID: 10692 **Order Date:** 03-Oct-1997 **Required Date:** 31-Oct-1997 **Shipped Date:** 13-Oct-1997

Product	Unit Price	Quantity	Discount	Extended Price
▶ Vegie-spread	\$43.90	20	0%	\$878.00
*			0%	

Print Invoice

Subtotal:	\$878.00
Freight:	\$61.02
Total:	\$939.02

Products

[Preview product list](#)

Output product list as HTML

Product ID: 1

Product Name: Cha

Supplier: G'day, Mate

Category: drinks

Quantity Per Unit: 10 boxes x 20 bags

Unit Price: \$18.00

Units In Stock: 39

Units On Order: 0

Reorder Level: 10

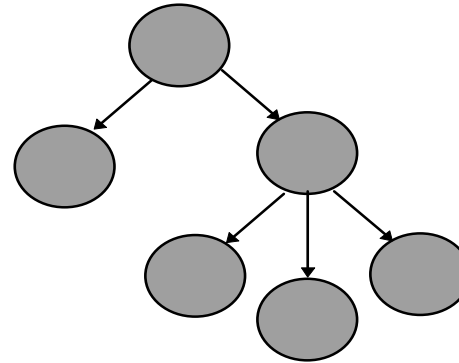
Discontinued:



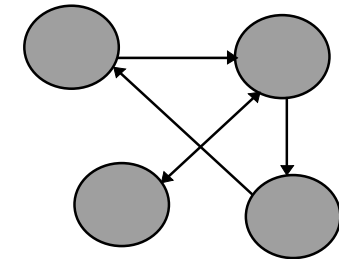
- Единствен логически модел
- Отразява структурата на данните
- Видове
 - йерархичен
 - мрежов
 - релационен
- Осигурява логическа независимост на данните -
промяна на външен модел не изисква промяна на
логическия модел



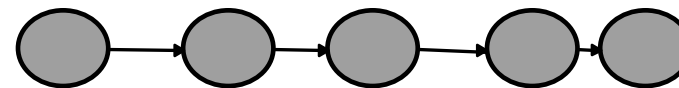
- йерархични
структура “дърво”



- мрежови
структура “граф”

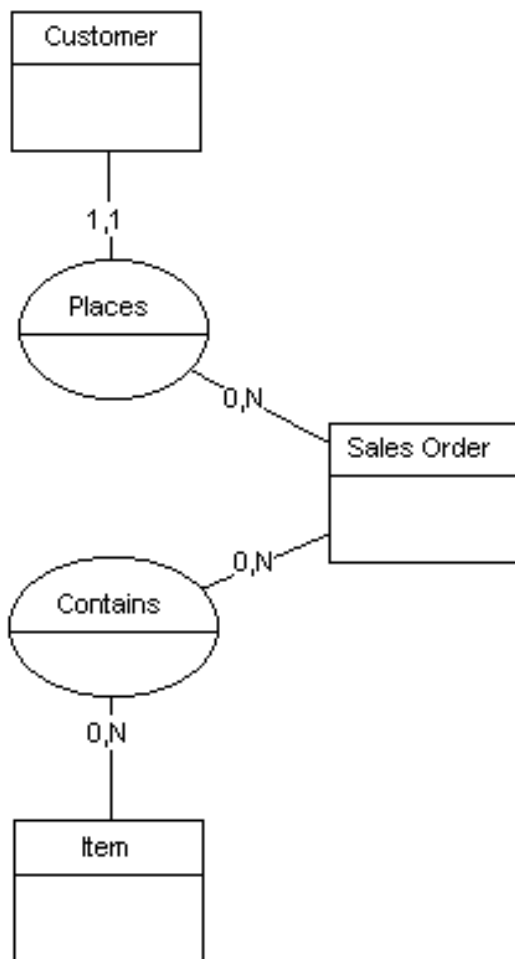


- релационни
структура “таблица”



- Преобразуване на моделите от един вид в друг

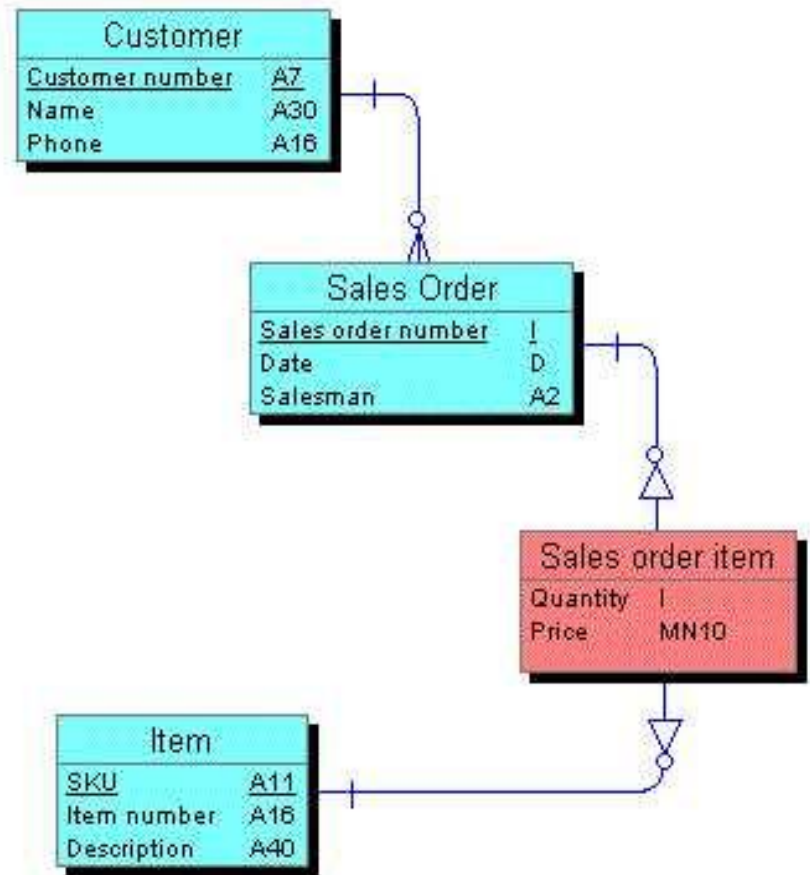


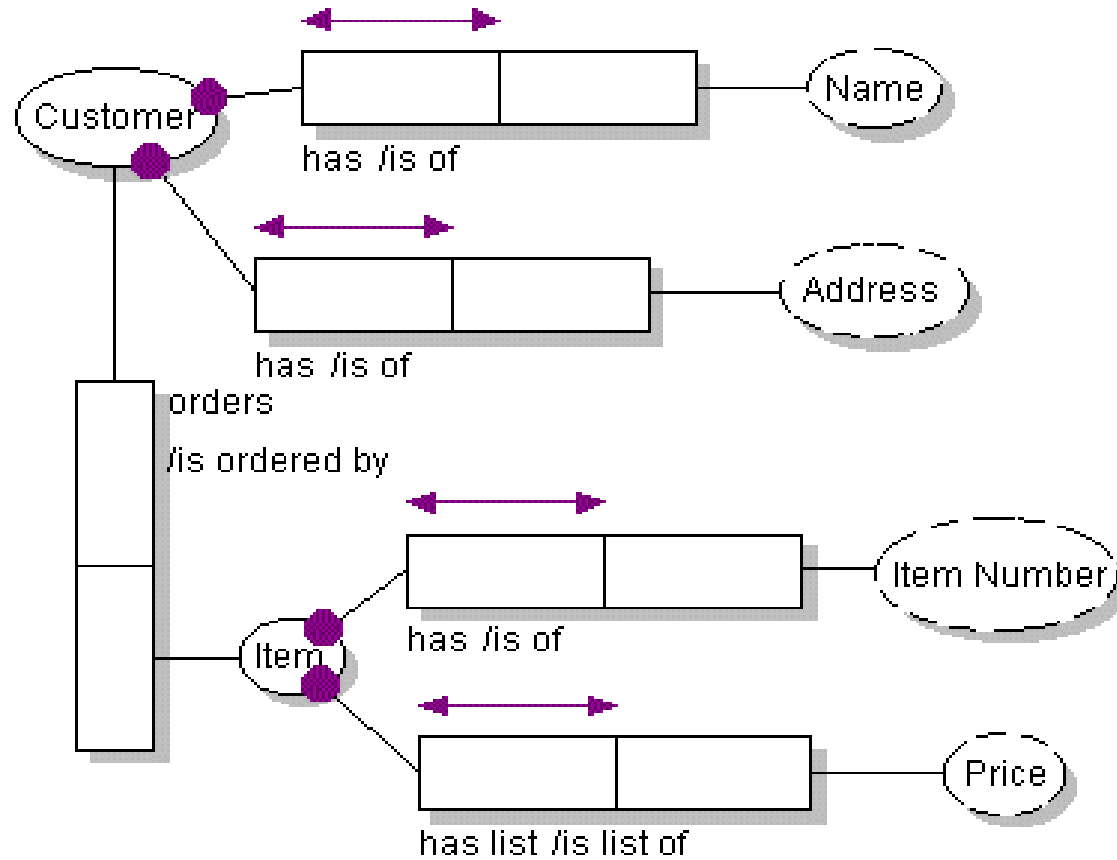


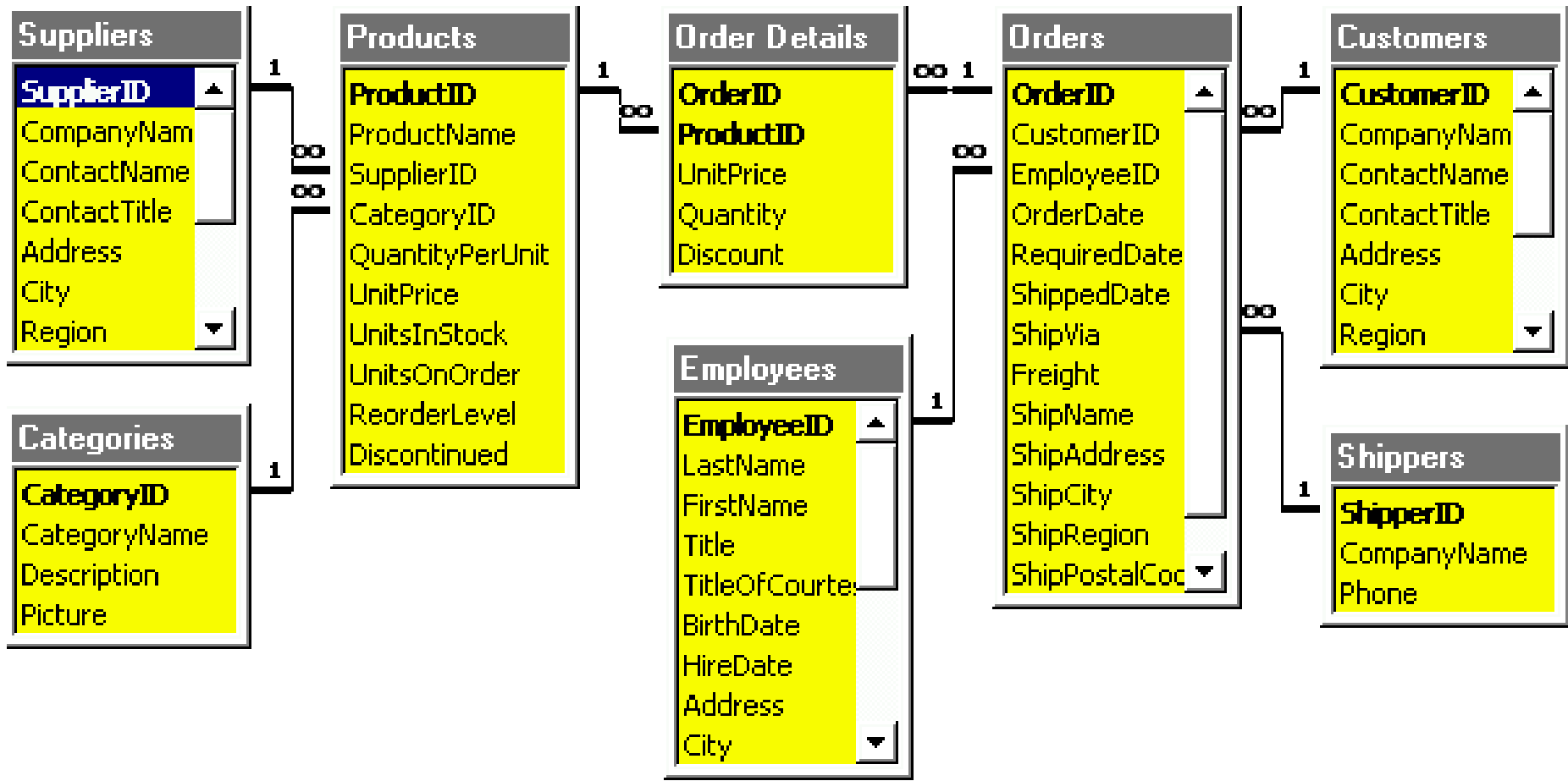
- Entity Relationship Model ER
 - класове обекти
 - отношения между тях



- Information Engineering IE
 - класове обекти
 - отношения между тях







- Единствен физически модел
- Осигурява трансформирането на структурите на концептуалния модел в запомнящи структури
 - файлове, записи, полета, променливи
- Осигурява методи за търсене и достъп до данните
 - индексирание, сортиране
- Осигурява физическа независимост на данните
 - промяна на външен модел не изисква промяна на физическия модел



- Структури памет
- Методи за достъп
- Оптимизация и настройка
- Управление на риска



Създаване на БД

● Цел

- създаване на система

● Дейности

- Програмиране
- Тестване
- Сертифициране
 - + Инсталиране на хардуер
- Обучение на потребителите



Тестване на БД

● План на тестване

● Обекти на тестване

■ съдържанието

- как се въвеждат данните, правила и ограничения

■ структурата

- съдържа ли БД структурата от модела на данни
- съдържа ли физическата структура (индекси, таблици на символите, правила за достъп и др.)

■ поведението

- тестване на процедури и програмни модули

● Тестови данни

- специално подбрани данни
- реални данни



● Същност

- определяне на използваемостта в различни приложения (*reuse*)

● Съдържание

- оценка на степента на риска при различни приложения -> данни от тестовете
- дефиниране на функциите на БД -> документация на моделите и схемите
- определяне на отговорностите по поддържане -> деклариране на собствеността и правата за ползване



Изготвяне на документация

Процедури за въвеждане в експлоатация

директно
паралелно
постепенно
пилотно



● План за поддръжане

- Интерактивен и итеративен процес

- видове поддръжка

 - при повикване

 - периодично

 - on-line

● План за обучение на потребителите



- Цел: установяване на съответствие с изискванията
- Задачи:
 - дефиниране на система от критерии за оценка
 - установяване на недостатъците
 - анализ на ефективността на поддържане
- Резултат: води до първия етап от жизнения цикъл



Методология

- Методология – формален подход за реализиране на жизнения цикъл
- Конкретната реализация зависи от приоритета на
 - бизнес-процесите
 - данните



Видове методологии

● Основани на процесите

- Основни обекти за процесите в бизнеса
- Системата се представя като множество от процеси, към и от които тече информация

● Основани на данните

- Основни обекти са данните, тяхната организация и съхранение
- Системата се представя като структура, чието ядро са модели на данните

● Обектно-ориентирани методологии

- Балансиране на фокуса между процесите и данните
- Системата се представя колекция от обекти, които отразяват данните и процесите едновременно (UML)

