

# Databázové systémy

---

1. úvod do DBS, základné pojmy,  
údajový model, E-R model



**KKUI**  
Katedra kybernetiky  
a umelej inteligencie

# Odporúčana literatúra:

- E. Ocelíková a kol. : Databázove systémy a jazyk SQL
- Database SQL Language Reference - [oracle.com \(link\)](#)
- Database SQL Language Quick Reference - [oracle.com \(link\)](#)
- SQL Developer User's Guide - [oracle.com \(link\)](#)
- Database Sample Schemas - [oracle.com \(link\)](#)

# Vyučujúci

- Prednášky:
  - Ing. Erik Kajáti, PhD.
- Cvičenia:
  - Ing. Nikola Hrabovská
  - Ing. Lenka Kališková
  - Ing. Erik Kajáti, PhD.

*(uprednostnite komunikáciu cez MS Teams – chat, príspevky do fóra)*

# Bodovanie predmetu

- 40b zápočet
  - 25b zadanie
  - 7b písomka(y)
  - 5b bonus (na cvičeniach)
  - 3b dochádzka na cvičeniach
- 60b skúška

# Úvod do databázových systémov

- DBS sú súčasťou každého informačného systému aj každodenného života t.j. smartfóny, e-shopy, organizácie, ...
- Ciele tohto predmetu:
  - priblížiť základné vlastnosti DBS
  - naučiť sa komunikovať s DBS pomocou jazyka SQL

# Databáza

- organizovaný súbor údajov, ktoré spolu logicky súvisia
- zjednodušeným odrazom skutočností z reálneho sveta (spoločnosť, zamestnanec, účet, príjem)
- určená pre špecifický účel, ktorý závisí od charakteru ukladaných údajov (účtovníctvo, plánovanie ciest)

# Vývoj DBS

- 60-te roky 20. storočia prvé DBS

v oblasti leteckých rezervačných systémov, bankových systémov, evidencie údajov vo veľkých spoločnostiach – klasické spracovanie súborov t.j. údaje sú uložené v lokálnych súboroch s vlastnou štruktúrou

- nevýhody klasického spracovania dát

- Problém redundancie a nekonzistencie dát
- Problém obtiažného prístupu k požadovaným dátam
- Problém izolovanosti dát
- Problém prístupu viacerých používateľov k dátam
- Problém ochrany dát
- Problém integrity dát

# Nevýhody klasického spracovania dát

- Riešenie: oddelenie súborov s údajmi od programov



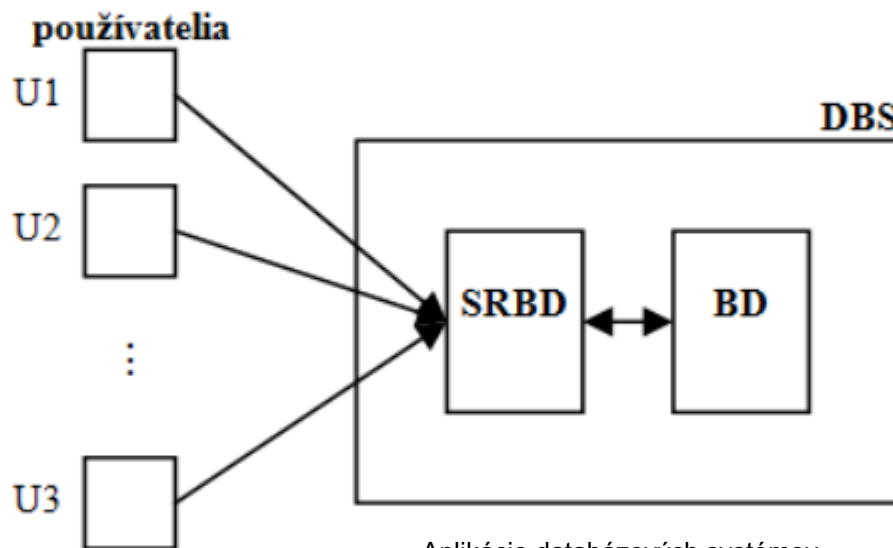
# Vývoj DBS

- 1970: Edgar Frank Codd: *A Relational Model of Data from Large Shared Data Banks*
- 1977: Lawrence J. Ellison, Robert N. Miner a Edward Oates spoločnosť Software Development Laboratory; projekt ORACLE pre CIA
- 1979: 1.komerčne dostupnú relačnú DB - Oracle 2
- 1980: SDL -> Oracle Corporation

# DBS: základné pojmy

Databázový systém pozostáva z:

- *báza dát (BD) resp. databáza* – organizovaná množina údajov
- *systém riadenia bázy dát (SRBD)* – systém programov pre prístup a prácu s BD



# Báza dát (Database)

- Komplex (základňa) dát medzi ktorými existujú vzťahy
- dáta sú prístupne výlučne pomocou SRBD
- okrem dát samotných obsahuje aj metadata, teda údaje o týchto dátach (zoznam tabuliek, zoznam stĺpcov, miesto uloženia,..)
- navyše zahŕňa relačné vzťahy medzi jednotlivými prvkami DB, obmedzenia integrity dát a schému dát

# Báza dát (Database)

Kritéria bázy dát:

- **neredudantnosť dát**
  - dáta sa zbytočne neopakujú
  - výnimka hodnoty kľúčov potrebné pre prepojenie dát, niektoré štatistické údaje
- **zdieľanie dát**
  - každý údaj je prístupný pre viacerých užívateľov opakovane alebo súčasne v tom istom čase, zabezpečené SRBD
- **konzistentnosť dát**
  - určitý údaj má rovnakú hodnotu pre každý výskyt svojho uloženia, zabezpečené neredudantnosťou dát
- **integrita bázy dát**
  - dáta spĺňajú podmienky integritného obmedzenia
- **nezávislosť bázy dát**
  - Programy na prístup nezávisia na mieste a spôsobe uloženia dát
- **ochrana dát**
  - Pridelovanie práv užívateľom na rôznych úrovniach

# System riadenia bázy dát (Database Managment System(DBMS))

- systém umožňujúci používateľovi pracovať s dátami
- užívateľ nemusí poznať fyzickú štruktúru uložených dát
- komunikácia sa realizuje pomocou jazyka SQL (deklaratívny jazyk)

# **Systém riadenia bázy dát (Database Management System(DBMS))**

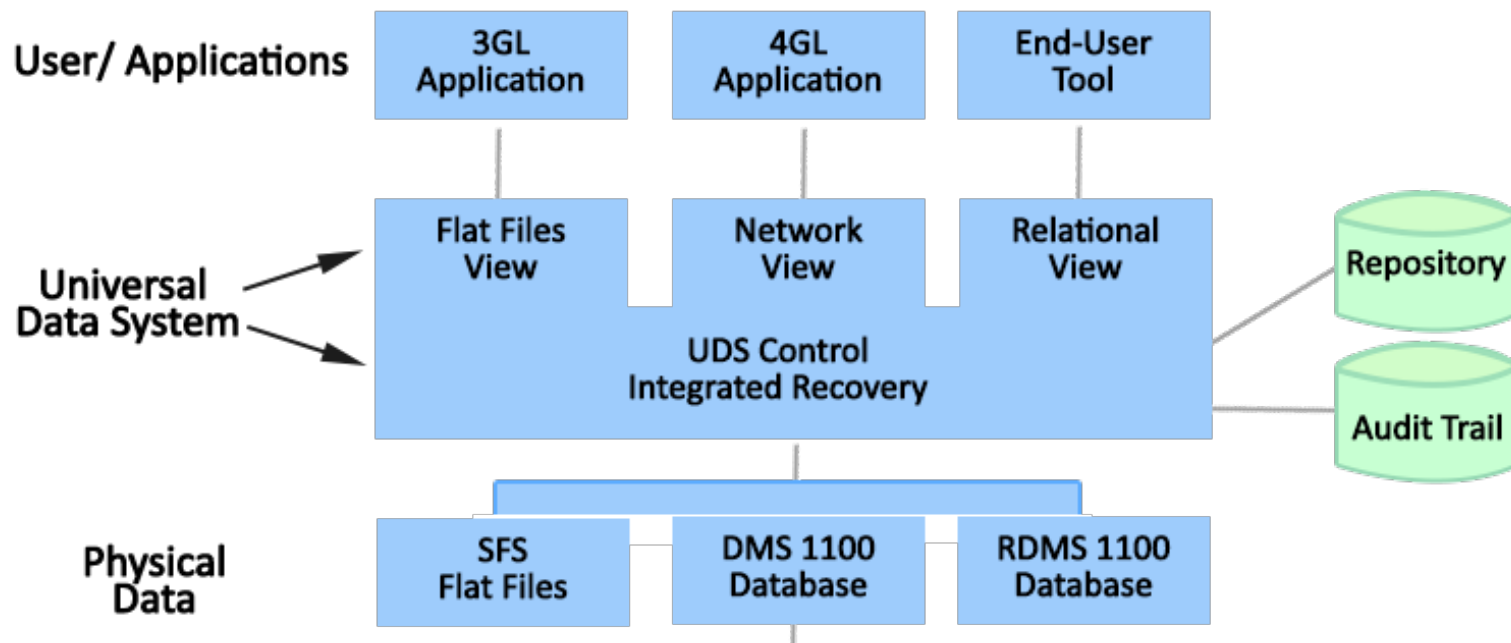
Základné funkcie sa delia na:

- riadiace funkcie pre kontrolu a riadenie systému:
  - dekodujú a kontrolujú príkazy systému
  - poskytujú informácie o chybách pri formulácii príkazov
  - realizujú prevzatie riadenia po ukončení príkazu
- výkonové funkcie umožňujú užívateľovi:
  - definovať štruktúru dát,
  - ukladať, vyberať, aktualizovať dáta
  - chrániť ich proti zneužitiu alebo poškodeniu

# Trojúrovňová architektúra DBS

1. Fyzická úroveň
  - najnižšia úroveň, detailne popisuje uloženie dát na disku
  - za inštaláciu a konfiguráciu DB programu a súborov dát zodpovedá DBA
2. Konceptuálna úroveň
  - popisuje aké dáta sú v DB uložené a ako navzájom súvisia
  - popis DB na konceptuálnej úrovni sa nazýva schéma; obsahuje popis všetkých tabuľkových údajov, typov a vzťahov, ktoré v DB sú
  - na úrovni schémy DB pracuje s pojmami:
    - typ objektu: učiteľ, predmet, miestnosť ....
    - vlastnosti typu objektu: r.č., meno, priezvisko ....
    - vzťahy medzi typmi objektov: učiteľ - predmet, zákazník - konto, ...
3. Používateľská úroveň
  - predstavuje najvyššiu úroveň abstrakcie dát
  - zložená z užívateľských pohľadov = subschéma
  - zobrazuje dáta podľa požiadaviek užívateľa

# Database Architecture





# Jazyk pre popis a manipuláciu s dátami

- **SQL(Structured Query Language)** – základ pre súčasné relačné databázy
  - **jazyk pre popis dát – DDL (Data Definition Language)**
    - modifikácia používateľských dát
    - popisuje typy dát, napr. ide o reťazec max. dĺžky 10
    - umožňuje definovať konzistenčné obmedzenia, ktoré dáta musia spĺňať napr. hodnota konta nesmie klesnúť pod určitú hranicu
    - sú uložené v slovníku dát (Data Dictionary)
  - **jazyk pre manipuláciu s dátami - DML (Data Manipulation Language)**
    - výber údajov z DB
    - vkladanie nových údajov
    - vymazanie niektorých údajov
    - aktualizácia údajov

# Modelovanie dát

**Model dát** je súhrn prostriedkov pre popis dát na logickej úrovni a to pre

- samotný popis dát
- popis väzieb medzi dátami
- popis obmedzení pre jednotlivé dáta

Pre vytvorenie modelu je potrebné určiť:

- typy entít

- triedy objektov rovnakého typu napr. pracovník auto, kniha atď.

*Entita je objekt reálneho sveta, ktorý môže existovať sám o sebe a dokážeme ho odlíšiť od iného objektu toho istého typu. Entitou je napr. Janko Mrkvička, r.č. 125625/1234*

# Modelovanie dát

- typy vzťahov, ktoré môžu medzi typmi entít existovať
  - napr. určitý pracovník(entita) vlastní(vzťah) určité auto(entita)
- atribúty, ktoré môžeme priradiť jednotlivým typom entít a vzťahov
  - popisujú vlastnosti entít a vzťahov napr. PRACOVNÍK (entita) : meno, priezvisko; VLASTNÍŤ(vzťah): dátum kúpy auta
  - množina všetkých rôznych hodnôt určitého atribútu sa nazýva doména. Rôzne atribúty môžu mať rovnakú doménu.
- integritné obmedzenia
  - typ atribútu a jeho doména napr. MESIAC v roku celé číslo z intervalu  $<1, 12>$
  - či je atribút kľúčový t.j. či je súčasťou identifikačného kľúča napr. CISLO\_OP
  - či môže mať daný atribút prázdnu hodnotu t.j. nemusí byť zadany napr. RODNE\_CISLO musí byť
  - či musí mať atribút unikátnu hodnotu t.j. danú hodnotu môže mať len jeden objekt napr. RODNE\_CISLO

# Modely dát

- **objektovo orientované** (E-R model)
- **záznamovo orientované** (relačný model)
- **fyzické** (modely implementované na najnižšej úrovni)

*Vytvorenie modelu dát na konceptuálnej úrovni*  
spočíva v tom, že z reálneho sveta si vyberiem tie typy entít, ich vzťahy a atribúty, ktoré nás zaujímajú a sú pre aplikáciu dôležité.

# Identifikačný kľúč

- jednoznačne odlišujú objekty a vzťahy medzi nimi.
- identifikačný kľúč môže tvoriť jeden alebo viacero atribútov, ktoré jednoznačne odlíšia objekty toho istého typu.
- podľa počtu atribútov rozlišujeme:
  - **jednoduchý kľúč** - jeden unikátny atribút  
(pracovník – rodné číslo)
  - **zložený kľúč** - viacero atribútov, ich počet by mal byť minimálny  
(pracovník – priezvisko, meno, dátum narodenia)
- všetky možnosti vytvorenia kľúča sa nazývajú **množina kľúčov**. Jej prvok nazývame **kandidát kľúča**. Vybraný kľúč nazývame **primárny**, ostatné nazývame **sekundárne**.

# Slabé a silné entity




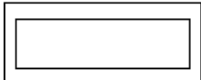

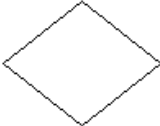
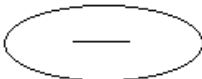
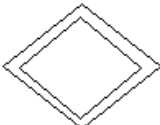
- Silné entity
    - objekty, ktoré kľúč majú, zodpovedajú im dominantné objekty
  - Slabé entity
    - objekty, ktoré nemajú kľúč, zodpovedajú im podriadené objekty
- V prípade slabých entít sa primárny kľúč vytvorí z primárneho kľúča jeho dominantného objektu (**identifikačný vlastník**) a vhodne zvoleného atribútu slabej entity (**diskriminátor**)
- Napr. meno mesta/obce môžeme identifikovať pomocou kódu regiónu, v ktorom sa nachádza (identifikačný vlastník) a mena danej obce (diskriminátor)

# E-R model

- Entity-Relationship model
- je vhodný pre návrh štruktúry DB
- graficky zobrazuje množiny entít a vzťahy medzi nimi
- jeho realizáciou vznikne schéma
- podobný význam ako vývojový diagram pri programovaní

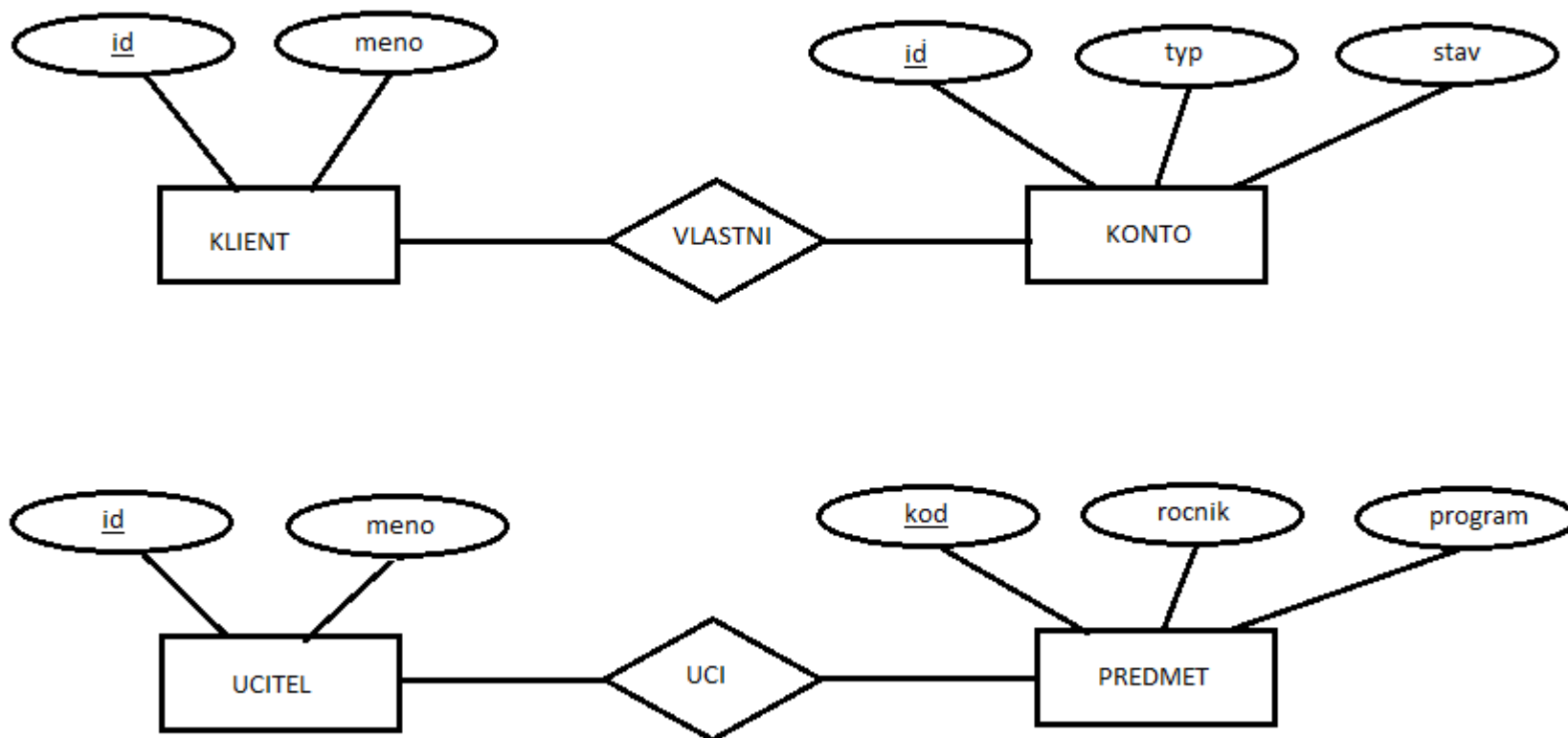
# E-R model

- Na zobrazenie objektov a vzťahov medzi nimi slúžia nasledujúce značky:

	doména hodnôt atribútu		entita
	atribút		slabá entita
	atribút s viacerými hodnotami		vzťah
	kľúčový atribút		vzťah so slabou entitou



# E-R model



# Vzt'ahy

- matematicky popísané ako relácie; sú to n-tice objektov, ktoré daný vzťah tvoria
- zobrazenie objektov (vzorov) do obrazov
- počet obrazov, do ktorých sa zobrazí jediný vzor sa nazýva **kardinalita**

# Typy vzťahov

- ***vzt'ah 1:1***
  - jeden vzor ma jeden obraz
  - napr. každý občan má jedno rodné číslo
- ***vzt'ah 1:N***
  - jeden vzor má viacero obrazov
  - napr. jeden klient má viacero účtov
- ***vzt'ah M:N***
  - viac vzorov sa zobrazí do viacerých obrazov
  - napr. viacero učiteľov môže vyučovať viacero predmetov

# Vzt'ahy

Grafické znázornenie vzťahov:

	1 : 1		M : 1
	1 : N		M : N

Príklad vzťahu M:N

