

Razvoj podatkovnih baz

0. obdobje pred pojavitvijo podatkovnih baz – čas datotek

- Do 60. let prejšnjega stoletja
- Datoteke: omogočajo direktni dostop do podatkov in trajno hranjenje razmeroma velikih količin podatkov
- Aplikacijski sistemi so uporabljali ločene datoteke – vsaka datoteka je bila nek ločen 'podatkovni otok'
- Edini pogled na podatke = fizični pogled
- Vse operacije nad podatki so prilagojene fizični predstavitvi podatkov
- Rast števila aplikacij → narašča število deljenih datotek, različnih formatov zapisa
- Problematična izmenjava podatkov med aplikacijami
- Večkratne kopije istih podatkov (redundanca) – brez centralnega nadzora!
- Podatki so postajali neskladni (nekonsistentni)
- Pomanjkljiva skrb za varnost podatkov, težavno sledenje dogodkom
- Nefleksibilni sistemi

1. obdobje PB – hierarhični in mrežni podatkovni model

- Konec 60. let - začetek komercialne uporabe PB
- Pojavljajo se in uvajajo **mehanizmi za opis podatkov – podatkovni modeli**:
 - Hierarhični podatkovni model
 - Mrežni podatkovni model
 - V 70. letih sta predstavljala standard za uporabo PB
- Delitev pogledov na podatke:
 - Fizični pogled
 - Logični pogled
- Lažje posodabljanje sistemov
- Avtomatiziran nadzor nad sočasnim dostopom do podatkov
- Zagotovljena možnost obnavljanja PB
- Avtomatizirano beleženje dogodkov v sistemu

2. obdobje PB – relacijski podatkovni model

- 1970. Edgar F. 'Ted' Codd predstavi koncept relacijskega podatkovnega modela
- Bistvo modela
 - PB se uporabnikom predstavi kot množica podatkov, ki so organizirani v tabelah
 - uporabniki relacijskega sistema se ne obremenjujejo s strukturo shranjevanja
 - striktno ločevanje med logičnim in fizičnim pogledom na podatke
- Za tabele Codd vpelje naziv **relacija**
- Poizvedbe se podajo v visoko-nivojskih jezikih → večja učinkovitost programerjev
- Posledica: razvili so zmogljiv deklarativni jezik, namenjen delu s podatkovno bazo imenovan **Structured Query Language**

2. obdobje PB – relacijski podatkovni model (nad.)

- Razvijalci sistemov za upravljanje z relacijsko PB: IBM + univerze
- Komercialne uspešnice:
 - INGRES, ki je bil razvit na University of California, Berkeley in
 - System R (ki je pozneje prerasel v DB2), razvit v IBM San Jose laboratorijih.
- 80. leta: relacijski model postane osrednji in najbolj razširjen model podatkov
- Še danes (v poslovnem okolju) prevladujejo podatkovne baze, ki temeljijo na relacijskem modelu
- 80. leta – razvoj porazdeljenih PB

3. obdobje – objektno relacijski in objektni podatkovni model

- začetek 90. let
- dodana je podpora objektom - pogled podatkov v PB hranimo tudi metode (postopke) za delo s podatki
- uporaba: specializirana področja, ki zahtevajo procesiranje zelo kompleksnih podatkov
- aplikacije za delo s PB so se razširile tudi na področje svetovnega spleta in e-poslovanja
- uvajajo se novi standardi za beleženje podatkov, kot je to XML (eXtended Markup Language)
 - XML = le za nov način beleženja (zapisovanja) podatkov // to NI podatkovni model!

4. obdobje – post-relacijski podatkovni sistemi

- konec 90. let
- post-relacijski podatkovni sistemi
- veliko novih funkcionalnosti:
 - podatkovno skladiščenje
 - OLAP (on-line analytical processing)
 - podatkovno rudarjenje
- podpirajo kompleksnejše podatkovne tipe (temporal, prostorske podatke)
- pojavljajo se t.i. aktivne podatkovne baze, ...

Prednosti uporabe PB:

- Omogoča nadzorovano redundanco podatkov, manj naporno vzdrževanje podatkov.
- Omogoča skupno rabo podatkov (načeloma) neomejenemu številu uporabnikov.
- Preprečuje nepooblaščen dostop do podatkov
- V objektnih bazah omogoča hranjenje tudi metod za delo s podatki (objekti)
- Zagotavlja tako strukturo shranjevanja, ki omogoča učinkovito izvajanje poizvedb.
- Zagotavlja storitve arhiviranja in restavriranja podatkovnega sistema in podatkov.
- Omogoča večkratne uporabniške vmesnike (pogleda) za dostop do podatkov.
- Vzdržuje kompleksne sisteme povezav med podatki.
- Vsiljuje izvajanje integritetnih omejitev nad podatki.

Prednosti uporabe PB: (nad.)

- Podpira in spodbuja uveljavljanje standardov v organizaciji. // Standardi se lahko nanašajo na imena podatkov, formate prikaza podatkov, strukture poročil, opise podatkov in tudi na poslovna pravila organizacije.
- Skrajša čas razvoja in vzdrževanja aplikacij (posebej pri dodajanju novih modulov/funkcij programa).
- Zagotavlja fleksibilnost pri spreminjanju strukture podatkov.
- Zagotavlja dostop do 'svežih' trenutnih podatkov.
- Lahko pomaga tudi pri iskanju in odpravljanju podvojenih virov v organizaciji → posredno tudi zmanjšanju stroškov organizacije.

Slabosti uporabe PB

- Cena SW (SUPB) (ki ne sodijo med prosto dostopno programsko opremo, je njihova cena). // pri prosto dostopnih SUPB-jih natančno preberi licenčen pogoje
- Odvisnost od proizvajalca/dobavitelja SUPB-ja.
- Zahteva čas za učenje in privajanje na sistem.
- Šibke točke SUPB-ja (lahko) vplivajo na šibkost PB (optimizacija kode, varnost)
- Računalniški sistem, na katerem bo nameščen SUPB mora biti zmogljivejši //praviloma večprocesorski sistem, ki ima veliko RAM-a in zelo hiter disk