

II. Softwarové inženýrství

Update: 5. května 2018

Obsah

1	Modelování databázových systémů, konceptuální modelování, datová analýza, funkční analýza; nástroje a modely.	2
2	Relační datový model, SQL; funkční závislosti, dekompozice a normální formy.	3
3	Transakce, zotavení, log, ACID, operace COMMIT a ROLLBACK; problémy souběhu, řízení souběhu: zamykání, úroveň izolace v SQL.	4
4	Procedurální rozšíření SQL, PL/SQL, T-SQL, trigger, funkce, procedury, kurzory, hromadné operace.	5
5	Základní fyzická implementace databázových systémů: tabulky a indexy; plán vykonávání dotazů.	6
6	Objektově-relační datový model a XML datový model: principy, dotazovací jazyky.	7
7	Datová vrstva informačního systému; existující API, rámce a implementace, bezpečnost; objektově-relační mapování.	8
8	Distribuované SŘBD, fragmentace a replikace.	9

1 Modelování databázových systémů, konceptuální modelování, datová analýza, funkční analýza; nástroje a modely.

Konceptuální model

Konceptuální model je jednoduchý popis entit a jejich vzájemných vztahů používaný v databázích. Jedná se o jakýsi prvotní jednoduchý návrh námi vytvářené databáze. Je kladen důraz na zobrazení všech entit a jejich vztahů. Nezávislý na SŘBD.

Základní pojmy

- **Entita** - objekt reálného světa.
- **Atribut** - vlastnost entity (možné hodnoty jsou označeny jako doména atributu)
- **Entitní typ** - množina entit se stejnými atributy.
- **Vztah** - vztah mezi **2 entitními** typy.
- **Popis** entitních typů, jejich atributů a vztahů mezi nimi v rámci daného zadání.
- **E-R diagram** - grafické znázornění konceptuálního modelu, neexistuje standard (používá se např. crow foot notace).
- **Kardinalita vztahu** - Dělení vztahů podle počtu entit vstupujících do vztahu:
 - 1:1
 - 1:N
 - M:N

Funkční analýza

- Cílem je popsat vytvářený systém jako „černou skříňku“, definovat její **vnější chování** a strukturalizovat **okolí systému**, které se systémem komunikuje. **Popsat všechny funkce, které se budou s daty provádět.**
- **Vnitřní:** rozpracované algoritmy pro jednotlivé akce.
- **Vnější:** náhled na strukturu a hierarchii funkcí.
- Úkolem je získat od zadavatele **seznam všech funkcí**, které bude od IS požadovat.
- Sestaví se tabulka se sloupci událost a reakce, do ní se formou krátkých textů zapisují všechny možné vnější události, podněty působící na systém a jim odpovídající reakce systému.

Otázky na požadavky

- **PROČ** nový systém.
- **ČEMU** má sloužit.

- **KDO** s ním pracuje - běžně, příležitostně, pravidelně zřídka.
- **VSTUPY** – objekty, atributy
- **VÝSTUPY** – výstupní sestavy, požadované informace
- **FUNKCE** – jaké výpočty, odvozování, výběry, třídění, ...
- **Vazby na OKOLÍ systému** – odkud data a kam.

Nefunkční požadavky

- Požadavky na **výsledný program**
- **Vnější požadavky**: ostatní nefunkční implementační požadavky, použití **standardů**, **cenová** omezení, **časové** požadavky.

Diagram datových toků (DFD)

- Několik úrovní podrobnosti.
- Definuje **hranici systému**.
- Definuje **všechny akce**, které mezi systémem a jeho okolím probíhají.

Minispecifikace = algoritmy elementárních funkcí

- Pro každou nerozložitelnou funkci z DFD **existuje minispecifikace**.
- Popisuje postup, jak jsou vstupní **data transformována na výstupní**.
- Popisuje, co **funkce znamená** věcně, ne, jak se to spočítá.
- Používá se **přírozený jazyk** s omezeným množstvím jasně definovaných pojmů.
- Musí být **srozumitelná** analytikovi, uživateli i programátorovi.

Datová analýza

- Zabývá strukturou obsahové části systému (strukturou databaze).
- Datový model, obsahuje již samotné atributy - **Závislý na SŘBD**.
- **Popis struktury dat IS**, jejich **vazby**, uchovávané vlastnosti a možné hodnoty těchto vlastností.
- **Využívá se**: lineární zápis entit a atributů, ER diagram, Konceptuální schéma, datový slovník (tabulky atributů).

Problémy agendového zpracování

- **Redundance**: některé informace ve více souborech opakují, jsou redundantní. Redundance je zdrojem mnoha dalších problémů.

- **Konzistence:** vzájemná **shoda údajů**. Postupem času - vlivem nedostatečné kontroly v programech se stejné hodnoty na různých místech v datových souborech, začnou rozcházet.
- **Integrita:** data aktuální, odrážejí skutečnost z reálného světa. Problémem tedy je zabezpečit, aby chybou či nedůsledností uživatele nebyla porušena integrita a konzistence dat.
- **Izolovanost dat:** data roztroušena v různých souborech, soubory mohou být různě organizovány, data různě formátována. To komplikuje tvorbu nových aplikačních programů a možnost realizovat vazby mezi datovými strukturami.
- **Současný přístup více uživatelů:** větší systémy vyžadují současný přístup k datům více uživatelů. Pak je nutné, aby programy vzájemně spolupracovaly, jejich činnosti byly koordinovány.
- **Ochrana proti zneužití:** při zpracování důvěrných či tajných dat není přípustné, aby měl kdokoli přístup ke všem informacím. Při klasickém zpracování však musí mít programátor aplikačních programů k dispozici tolik podrobností, že to ochranu dat prakticky znemožňuje.

2 Relační datový model, SQL; funkční závislosti, dekompozice a normální formy.

- 3 Transakce, zotavení, log, ACID, operace COMMIT a ROLLBACK; problémy souběhu, řízení souběhu: zamykání, úroveň izolace v SQL.

- 4 Procedurální rozšíření SQL, PL/SQL, T-SQL, trigger, funkce, procedury, kurzory, hromadné operace.

- 5 Základní fyzická implementace databázových systémů: tabulky a indexy; plán vykonávání dotazů.

6 Objektově-relační datový model a XML datový model: principy, dotazovací jazyky.

- 7 Datová vrstva informačního systému; existující API, rámce a implementace, bezpečnost; objektově-relační mapování.

8 Distribuované SŘBD, fragmentace a replikace.