

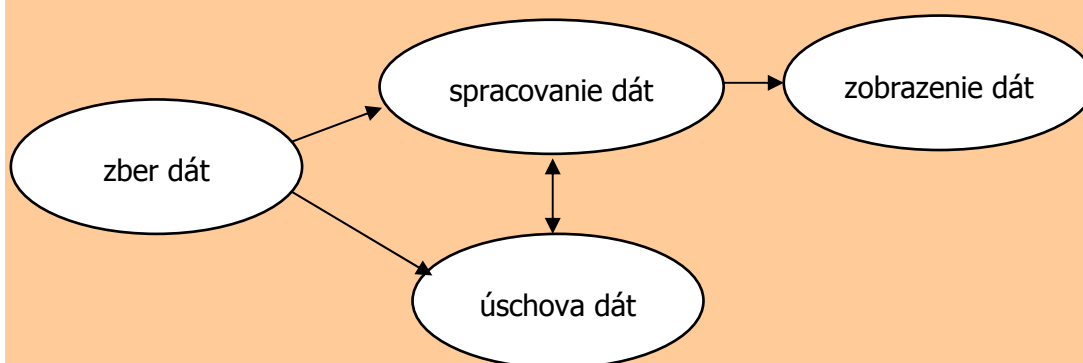
Informačné systémy

© HALÁSZ, Jozef

Zdroj: RUSKO, Miroslav - HALÁSZ, Jozef, 2011: *Environmentálne orientované informačné systémy*. - Žilina: Strix, Edícia EV-64, Prvé vydanie, ISBN 978-80-89281-76-3, 220 s.

Základné pojmy a definície

Informačný systém je funkčný celok zabezpečujúci cieľavedomé a systematické zhromažďovanie, spracúvanie, uchovávanie a sprístupňovanie informácií. Každý informačný systém zahŕňa informačnú základňu, technické a programové prostriedky, technológie a pracovníkov ¹.



Obr. 1 Hlavné činnosti zabezpečované v IS

Podľa zamerania sa delia informačné systémy na:

- Transakčné systémy, ktorých cieľom je zautomatizovať spracovanie bežných úloh, ktoré zahŕňajú vedenie účtovníctva, spracovanie miezd a plátov, rôzne evidenčné systémy, systémy skladového hospodárenia a pod. Niekedy sem zahrňujeme aj systémy pre riadenie technologických procesov v reálnom čase. Typickým atribútom je skutočnosť, že veľká časť celkového spracovania v týchto systémoch je vykonávaná už pri vkladaní dát, resp. tesne po ňom. Táto kategória systémov je v súčasnosti najviac rozvinutá a rozšírená a to hlavne vďaka ich jasnej špecifikácii a evidenčnému charakteru. V prípade, že transakčný systém je modulárnym riešením pokrývajúcim väčšiu časť oblastí podnikových aktivít, hlavne výrobu, predaj, financie a personalistiku, nazývame ho „podnikový informačný systém“ (**ERP** – Enterprise Resource Planning).
- Systémy pre riadenie (**MIS** - Management Information Systems) ich hlavnou úlohou je poskytovať riadiacim pracovníkom rôzne prehľady a agregované zostavy podľa rôznych časových, priestorových a iných hľadísk, hlavne za účelom dodania informácií riadiacemu pracovníkovi a tým mu uľahčiť proces kontroly podriadených oddelení. Vstupnými informáciami sú dáta z transakčného systému.

¹ Zákon č. 256/1992 Zb. o ochrane dát v informačných systémoch

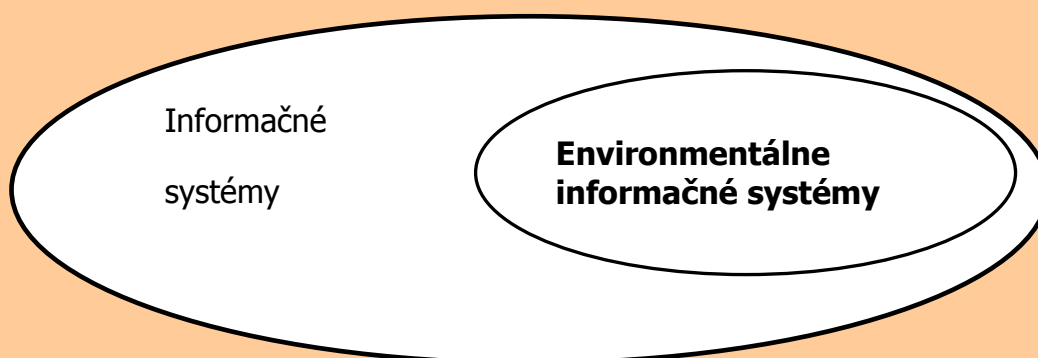
- Systémy pre podporu rozhodovania (**DSS** - Decision-Support Systems), ktoré sú nadstavbou MIS. Určené sú hlavne pre rôznorodé analýzy a ich využitie v procese rozhodovania riadiacich pracovníkov. Zastrešujú všetky predošlé. Dá sa povedať, že na rozdiel od predchádzajúcich systémov sú určené hlavne pre získanie podkladov pre strategické rozhodovanie vo finančnej a personálnej oblasti.

Environmentálne informačné systémy môžeme považovať za špecializované informačné systémy, zamerané najmä na údaje o životnom prostredí. Životné prostredie je tá časť sveta, s ktorou je človek vo vzájomnom pôsobení (interakcii), ktorú používa, ovplyvňuje a ktorej sa prispôsobuje. Zjednodušene je možné povedať, že životné prostredie je prostredím, v ktorom existuje a vyvíja sa život, pričom sa často zdôrazňuje život ľudskej spoločnosti. Ide teda o priestor na povrchu Zeme, kde žijú organizmy od najjednoduchších foriem až po človeka.

Pre oblasť životného prostredia je typická rozsiahla škála najrôznejších tematických informácií, ktoré majú významné priestorové vzťahy. Priestorovo lokalizované informácie je nutné použiť pri rozhodovaní o znížení zdravotných rizík, obmedzení znečisťovania krajiny, zásahoch do prírody a krajiny, využití prírodných zdrojov.

Environmentálne informačné systémy (EIS) spravujú údaje o pôde, vode, ovzduší, rastlinách a živočíšnych druhoch v prírode. Zber a spracovanie týchto dát je základným komponentom každej účinnej environmentálnej stratégie. Väčšinu týchto dát potrebujú riadiaci pracovníci pre rozhodovanie, spravidla v predspracovanej (kondenzovanej, agregovanej) podobe. EIS umožňujú integrovať informácie získané z digitalizovaných máp, leteckých a družicových snímok a špecializovaných databáz do jediného funkčného celku. Umožňujú pracovať s priestorovými modelmi terénu, geologických a hydrologických telies, koncentrácie škodlivín v ovzduší, vo vode, v pôde a pod.

V užšie vymedzenej oblasti EIS (podľa vyššie uvedenej definície) patrí vedúca pozícia nemeckým odborníkom. V Nemecku existuje od roku 1984 pracovná skupina pre environmentálnu informatiku, ako súčasť nemeckej počítačovej spoločnosti. Každoročne sa konajú konferencie, workshopy a iné aktivity. Bázu tvorí zhruba 150 výskumných pracovníkov. Vzhľadom na odlišný prístup k problematike, v USA pracuje v oblasti EIS podstatne menej špecialistov. Tu sa vytvárajú riešiteľské skupiny na konkrétne projekty.



Obr. 2 Vzťah EIS a IS

Úzke spojenie alebo skoro stotožnenie EIS a GIS charakterizujú aj knižné publikácie, hlavne amerického pôvodu. Problematika GIS má pomerne bohatú odbornú

literatúru, ale oblasť EIS v našom ponímaní pokrýva len niekoľko publikácií, z ktorých treba vyzdvihnúť základné dielo prof. O. Günthera ².

V SR bola schválená koncepcia rozvoja informačného systému štátnej správy a harmonogram konkrétnych krokov pre realizáciu úloh. Rezort Ministerstva životného prostredia SR spracoval koncepciu rezortnej časti štátneho informačného systému, ktorej cieľom je objasniť štruktúru informačného priestoru rezortu, definovať základné subsystemy, ich obsahy a vzájomné väzby a určiť postup realizácie tak, aby pri autonómnom rozvoji jednotlivých subsystemov nedochádzalo k údajovej alebo technologickej nekompatibilite a aby boli informácie o životnom prostredí prístupné všetkým, ktorí ich potrebujú. Koncepcia ďalej stanovuje spôsob realizácie navrhnutých zámerov a metódy riadenia ďalšieho rozvoja informatizácie rezortu.

Príprava návrhu IS

Informačné systémy spracúvajú dáta, aby mohli poskytovať užitočné informácie pre všetkých používateľov systému. Túto základnú požiadavku môžu dobre plniť len správne navrhnuté informačné systémy. Požiadavka navrhnuť informačný systém orientovaný na potreby používateľa si vyžaduje najprv analyzovať systém z hľadiska celkovej štruktúry a vzťahov medzi informáciami a potom podrobnejšie špecifikovať štruktúru dát a potrebné prostriedky (programy).

Tvorba informačného systému je vlastne komplexom mnohých rozhodovacích procesov. Takýto pohľad na tvorbu informačného systému zdôrazňuje uskutočňovanie voľby medzi viacerými alternatívami riešenia, a to na všetkých úrovniach a vo všetkých fázach tvorby informačného systému ³. Každá voľba by mala byť zdôvodnená a zdokumentovaná. Rozhodnutia o vytvorení a cieľoch informačného systému sú zásadné, najmä strategické rozhodnutia na úrovni vedenia organizácie. Do tejto kategórie môžeme zaradiť:

- tvorbu dlhodobých plánov rozvoja informačného systému v organizácii,
- rozhodnutia o celkovej architektúre informačného systému organizácie,
- rozhodnutia o štandardoch a zásadách platných pre tvorbu a činnosť informačného systému v organizácii,
- voľbu priorít pri realizácii dlhodobých plánov.

Rozhodnutia o procese tvorby informačného systému sa týkajú spôsobu uskutočnenia aplikačného projektu, predovšetkým voľby metodických postupov, techník a prostriedkov. Patrí sem aj výber riešiteľského kolektívu, organizovanie spolupráce s používateľmi, spôsob riadenia projektu a pod.

Rozhodnutia v rámci procesu tvorby informačného systému, ktoré sa týkajú už vlastného uskutočnenia jednotlivých fáz tvorby informačného systému, zahŕňujú konkrétne kroky podrobnejšej identifikácie cieľov a funkcií informačného systému, jeho dekompozície, vypracovania logického a fyzického návrhu, implementácie a overenia.

Jednotlivé kategórie rozhodovacích procesov sú úzko previazané.

Požiadavky na informačný systém sa dajú získať:

- dopytovaním sa budúcich používateľov systému,
- odvodením z existujúceho informačného systému,
- syntézou charakteristík používateľského (objektového) systému,
- na základe experimentovania s vyvíjaným systémom.

² GÜNTHER, O., 1998: *Environmental Information Systems*. - Berlin : Springer Verlag

³ RABENSEIFER, A., 1993: *Moderné navrhovanie informačných systémov*. - Bratislava : Veda, s. 67

Na základe zistených požiadaviek vytvárame abstraktné modely informačného systému, ktoré postupne pretvárame na konkrétne štruktúry a postupy.

Medzi prvé monografie, ktoré vyšli v slovenčine z oblasti navrhovania IS patrí kniha Björna Langeforsa ⁴. Po desaťročiach, ktoré uplynuli od jej napísania sú ešte stále aktuálne postrehy autora o základných problémoch pri navrhovaní IS. Uvedieme preto aspoň niektoré z nich:

- Ľudia majú tendenciu zľahčovať význam alebo existenciu vecí, ktoré nie sú schopní vidieť alebo vnímať (ignorovať môžeme hlavne to, čo nevidíme alebo nevieme odvodiť).
- S každým detailom systému a systémovej práce na jeho opise sú spojené určité náklady.
- Neprehľadný systém je systém, ktorého počet častí a ich vzájomných väzieb je taký veľký, že celú štruktúru tohto systému nemožno sledovať naraz.

Z uvedeného vyplýva:

- Pri neprehľadných systémoch je tendencia podhodnocovať ich zložitosť, t.j. počet ich častí a relácií.
- Pri neprehľadných systémoch sa podhodnocujú náklady na ich činnosť, ako aj náklady na systémovú prácu.
- Manuálna práca sa občas podhodnocuje. (Práca človeka sa podobá zložitému systému, rozmanitosť činností je veľká).
- Jedinou možnosťou navrhnuť systém tak, aby mal špecifikované vlastnosti, je kombinovať systém z množiny podsystémov, ktoré majú potrebné vlastnosti a sú prepojené tak, že výsledný systém má špecifikované vlastnosti.

Spoločné zistenie požiadaviek na informačný systém je stále jedným z kľúčových predpokladov úspechu. Existujú tri hlavné príčiny ťažkostí pri získaní korektného a úplného súboru požiadaviek:

- obmedzenia (ohraničenia) človeka ako spracovateľa informácií a riešiteľa problémov,
- rozmanitosť a zložitosť informačných požiadaviek,
- zložitosť interakcie medzi užívateľmi a analytikmi pri definovaní požiadaviek.

Z uvedeného vyplýva, že neexistuje univerzálny spôsob zisťovania požiadaviek, ktorý by sa mohol aplikovať pre všetky informačné systémy. Konkrétne aplikácie vyžadujú rôzne metodiky podmienené špecifickými podmienkami. Metodika pre stanovenie informačných požiadaviek by mala pomôcť analytikovi vymedziť a štruktúrovať problémový priestor a poskytnúť istotu, že požiadavky sú úplné a korektné.

Analýza a syntéza informačného systému

Spracovanie dát zahŕňa v sebe dve podstatné zložky: dáta, ktoré sú predmetom spracovania a procesy, ktoré transformujú vstupujúce dáta na výsledné informácie. Vznikli preto aj rôzne prístupy k analýze a navrhovaniu informačného systému: procesne orientovaný prístup, dátovo orientovaný prístup a objektový prístup ⁵.

⁴ LANGEFORS, B., 1981: *Teoretická analýza informačných systémov*. - Bratislava : Alfa

⁵ MOOS, P., 1993: *Informační technologie*. - Praha : ČVUT

Procesne (funkčne) orientovaný prístup vychádza z výsledných cieľov navrhovaného systému a základom jeho filozofie je účelovosť informačného systému. Chápe IS ako integrovanú súčasť prostredia, ktoré má podporovať, alebo inak povedané, ako súčasť riadiaceho mechanizmu organizácie. Informácie sa tu chápu, ako dáta, ktoré boli vybrané, spracované a zoskupené tak, aby slúžili špecifickému účelu. Informácie teda plnia určité funkcie.

Riešiteľ informačného systému si kladie predovšetkým otázku, aký problém má riešiť, čo má informačný systém robiť, aká bude jeho funkcia. Ak ide o zložitejší systém, pôjde o komplex funkcií. Riešenie si preto vyžiada postupnú dekompozíciu problému na čiastkové funkcie, činnosti a procesy, označovanú ako funkčná dekompozícia. Konečný vyprojektovaný systém predstavuje potom množinu procedúr, ktoré operujú s dátami.

Z celkovej filozofie takéhoto prístupu vyplýva, že dekompozícia sa deje postupne prístupom zhora nadol (top-down).

Najvýznamnejším predstaviteľom procesne orientovaného prístupu, resp. funkčnej dekompozície sú metódy **štruktúrovanej analýzy a štruktúrovaného navrhovania**, ktoré predstavujú prevažujúcu orientáciu metodík IS. Štruktúrovaná analýza a navrhovanie priniesli niektoré špecifické prostriedky a techniky pre analýzu a dokumentáciu systému. Sú to najmä diagramy tokov dát, diagramy prechodu stavov, slovníky dát a štruktúrovaný jazyk.

Všeobecne oceňovanou prednosťou procesnej orientácie je explicitné riešenie zložitosti problému rozkladom na menšie časti, pri rešpektovaní zásad minimalizácie počtu vzájomných väzieb a maximalizácia súdržnosti jednotlivých častí. Jej technická realizácia prostredníctvom štruktúrovaného postupu súčasne zavádza disciplínu do postupu analýzy i návrhu systému.

Nedostatkom procesne orientovaného prístupu je previazanosť navrhnutého systému s konkrétnymi funkciami, čo znižuje prispôbitelnosť systému vývojovým zmenám a v tejto súvislosti sťažuje aj znovu použiteľnosť jednotlivých častí systému. Tento postup nekladie dostatočný dôraz na štrukturalizáciu a riadenie dát.

Dátovo orientovaný prístup chápe IS ako obraz, mapu, model reálneho objektového systému. Riešenie informačného systému organizácie predstavuje potom modelovanie, štruktúrovanie dát používaných v organizácii. Jeho vedúcou ideou je flexibilita systému umožňujúca mnohostranné využitie dát.

Podľa filozofie dátovo orientovaného prístupu sú dáta základnými stavebnými blokmi systému. Vychádza sa z predpokladu, že ak vieme identifikovať a klasifikovať dátové prvky (entity a atribúty), ktoré existujú v určitej situácii, potom sme identifikovali pravú povahu systému. Dáta sú statické, resp. málo podliehajú zmenám. Analýza dát nie je orientovaná na riešenie vlastného problému. Usiluje sa definovať základ systému v termínoch dát a ich relácií a na tento účel používa logický model dát organizácie. Dátovo orientovaný prístup sa vyvíjal najmä na základe výskumu dátových modelov a v súčasnosti využíva predovšetkým poznatky entitno-relačného modelu, techniku zoskupovania entít a model života entity. Vo väčšine štruktúrovaných metodík sa stretávame s kombináciou oboch prístupov.

Objektový prístup možno v určitom zmysle pokladať za ďalší vývojový stupeň dátového prístupu. Pri objektovom prístupe je informačný systém považovaný za množinu vzájomne komunikujúcich objektov. Objekty obsahujú údaje o sebe, o svojom stave a procedúry, ktoré sú určené na spracovanie údajov objektu. Stav objektu je vyjadrený jeho atribútmi a procedúry sú spojené so službami poskytovanými iným objektom pri prevádzke informačného systému. Takýmto

spôsobom je dosahovaná nezávislosť každého objektu systému, čo umožňuje vykonávať zmeny vybraného objektu, bez nutnosti meniť ostatné objekty. Objekty môžu byť súčasťou hierarchicky štruktúrovanej množiny objektov. Pritom každý objekt prijíma (dedí) atribúty a procedúry nadradeného objektu, t.j. svojej triedy.

Komunikácia medzi objektmi sa uskutočňuje ako volanie podprogramov alebo funkcií, ktoré sú súčasťami iných objektov. Výsledky sa vracajú volajúcemu objektu v podobe množiny parametrov.

Štruktúrovaný návrh IS

Základným princípom štruktúrovaného návrhu je hierarchická dekompozícia systému na menšie celky (moduly, subsystemy, procedúry). Je to vlastne realizácia zásady projektovania zhora a výstavby zdola (top-down analysis and bottom-up design), čo znamená postupný rozklad systému na menšie a menšie celky tak, aby pritom bola zachovaná celistvosť a kompaktnosť systému, ale vlastná výstavba, t.j. realizácia systému sa robí postupne zdola, implementáciou jeho jednotlivých častí. Tento postup zaručuje celkový integrovaný pohľad na všetky entity, poskytuje základňu pre štrukturalizáciu dátového modelu do ľahšie realizovateľných databáz a v priebehu výstavby umožňuje ľahkú integráciu čiastkových databáz do uceleného informačného systému. Významným princípom štruktúrovaného prístupu je princíp abstrakcie a princíp jednoduchosti. Štruktúrované techniky pre návrh systému obsahujú najmä tieto nástroje a postupy:

- Funkčný model systému vytvorený pomocou diagramov dátových tokov **DFD** (Data Flow Diagram) a podrobným slovným popisom funkcií.
- Dátový model systému vytvorený pomocou entitno-relačných diagramov **ERD** (Entity-Relationship Diagram).
- Model prechodu stavov vytvorený pomocou diagramu stavov a prechodov **STD** (State Transition Diagram).
- História života entity (Entity Life History - **ELH**) znázorňuje životný cyklus entít. Sleduje, ktoré udalosti ovplyvňujú činnosť systému, na ktoré entity pôsobia a ako menia ich stav.

Pri tvorbe základného modelu IS sa zaoberáme premietnutím požiadaviek používateľa do činností systému, bez toho aby sme sa príliš venovali problémom implementácie IS. Základný model tvorí model prostredia a model správania sa systému.

Model prostredia definuje rozhrania systému voči prostrediu. Model prostredia tvorí: stručná definícia účelu systému, diagram kontextu a zoznam udalostí. Navrhovaný informačný systém sa chápe ako účelový systém, ktorý reaguje na udalosti a stimuly prostredia. Kritickým aspektom modelu prostredia je identifikácia udalosti v prostredí, na ktoré musí systém reagovať. Hlavným prostriedkom znázornenia modelu prostredia systému je diagram kontextu, vyjadrený symbolmi diagramu toku dát. Sú na ňom uvedené väzby navrhovaného systému na všetky externé entity.

Model správania sa systému tvoria diagramy tokov dát, entitno-relačné diagramy, diagramy prechodu stavov, zápisy v slovníku dát a špecifikácie procesov.

Poradie činností pri tvorbe modelu správania sa systému je nasledovný:

- vytvorí sa proces pre každú udalosť na zozname udalostí, vypracovanom v rámci modelu prostredia;
- proces sa pomenuje opísaním reakcie systému na príslušnú udalosť;
- určia sa vstupy a výstupy tak, aby bola možná požadovaná reakcia procesu. Vytypujú sa potrebné súbory dát pre komunikáciu medzi procesmi;

- výsledný (hrubý) diagram toku dát sa porovná s diagramom kontextu a so zoznamom udalostí, aby sa skontrolovala úplnosť a konzistencia.

Paralelne a nezávisle možno vypracovať predbežnú verziu entitno-relačného diagramu (**ERD**) a diagramu dátových tokov (**DFD**) tak, aby súborom na DFD zodpovedali objekty (entity) na ERD a naopak. Podobne položky na zozname udalostí sa môžu stať entitami pre ERD. Tento predbežný model systému nie je určený pre používateľa. Je to rámec, na základe ktorého možno vypracovať konečnú verziu systému.

Poradie činností pri tvorbe upresneného (konečného) modelu správania sa systému:

- Hierarchické rozčlenenie diagramov tokov dát až po úroveň primitívnych procesov.
- Skompletizovanie slovníka dát. Postupne sa kontroluje úplnosť a konzistentnosť zápisov.
- Vypracovanie špecifikácií procesov. Špecifikácia procesov opisuje, čo sa odohráva v každom procese na najnižšom stupni DFD (primitívny proces). Na opis procesov je k dispozícii viacero techník: štruktúrovaný jazyk, rozhodovacie tabuľky, postupové diagramy a pod.
- Dopracovanie dátového modelu.
- Dokončenie diagramu prechodu stavov.

Vypracovanie implementačného modelu zahrňuje nasledujúce činnosti:

- Určenie hraníc automatizácie - čo automatizovať, čo robiť ručne. Rozhodujúce je stanovisko užívateľa. To je miesto, kde treba zvážiť vplyv ľudského faktora, resp. pôsobenie automatizovaného systému na človeka. Treba upraviť DFD a ERD a vypracovať špecifikácie neautomatizovaných procesov.
- Určenie rozhrania človek-stroj:
 - a) voľba vstupných a výstupných zariadení,
 - b) formát všetkých vstupov z externých entít do systému,
 - c) formát všetkých výstupov zo systému pre externé entity,
 - d) poradie a časovanie vstupov a výstupov, ak systém pracuje v reálnom čase.
- Identifikácia dodatočných ručných podporných aktivít.
- Špecifikácia operačných obmedzení:
 - e) určenie objemov dát,
 - f) stanovenie časov odpovedi na rôzne vstupy,
 - g) zohľadnenie obmedzení vyplývajúcich z uplatňovania záujmov používateľa,
 - h) zohľadnenie obmedzení prostredia, ako sú teplota, vlhkosť, prašnosť a pod.
 - i) požiadavky na bezpečnosť a spoľahlivosť.
- Návrh technického riešenia:
 - j) Ako bude model systému rozdelený na jednotlivé technické a programové procesory, riešenie komunikácie, priradenie procesov a dátových súborov uvedených na DFD. Do úvahy sa pritom berú náklady, účinnosť, bezpečnosť, spoľahlivosť a operačné obmedzenia.
 - k) Pre každý procesor treba priradiť procesy a dátové súbory k individuálnym úlohám (task, partition, job step). Vyrieši sa komunikácia medzi úlohami.
 - l) Pre každú individuálnu úlohu treba vypracovať štruktúrny diagram (structure chart), ktorý ukáže hierarchickú organizáciu modulov v úlohe.
- Vypracuje sa návrh programových modulov. Pritom sa uplatnia zásady minimalizácie počtu väzieb medzi modulmi (coupling) a maximalizácia vnútornej

súdržnosti modulov (cohesion). Ďalej nasleduje programovanie, testovanie, konverzie súborov, inštalácia systému a zaškolenie pracovníkov.

Metodológie návrhu IS

Dnes už tradičnou metodológiou návrhu informačného systému je *Yourdonova štruktúrovaná metodológia YSM* (Yourdon Structured Method). Podstatou YSM je modelovanie, vytváranie obrazu reality špecifickými prostriedkami⁶. Cieľom je zachytiť všetky objektívne javy a skutočnosti tak, aby sme získali esenciu popisovanej reality. Tzv. esenciálny model obsahuje všetko potrebné pre implementáciu do konkrétneho programového a technického prostredia.

Yourdonova štruktúrovaná metodológia zdôrazňuje tri činnosti:

- modelovanie,
- iteráciu,
- dekompozíciu.

Yourdonova metodológia vytvára rôzne modely (dátový model, funkčný model, model riadenia systému a model štruktúry programového systému), z ktorých každý znázorňuje iný uhol pohľadu na systém. Prvé tri sú výsledkom systémovej analýzy, posledný je výsledkom konštrukcie systému.

Dátový model systému je vyjadrením statického pohľadu na realitu. Opisuje objekty, ktoré nás zaujímajú a vzájomné vzťahy medzi nimi. Najpoužívanejší je entitno-relačný model. Pri tvorbe modelu reality je nutné:

- určiť relevantné entity,
- opísať entity pomocou významných atribútov,
- vytvoriť užitočné triedy entít,
- definovať vzťahy medzi entitami a triedami entít.

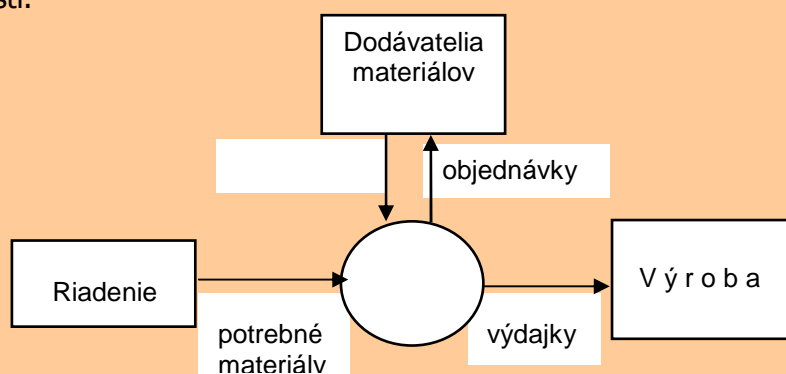
Báza dát má obsahovať také údaje o riadenom objekte a jeho okolí, aby v nej boli zachytené všetky podstatné skutočnosti potrebné pre používateľov. Dáta pritom musia odrážať tieto skutočnosti verne a v časovom súlade s vývojom reality.

Funkčný model systému predstavuje dynamický pohľad na systém. Vyjadruje činnosť systému tak, že si všíma miest, kde je potrebné transformovať dáta do inej formy a označuje ich ako funkcie. Všíma si, ktoré dáta sú pre danú funkciu vstupné a ktoré výstupné. Pre funkčné modelovanie sa v YSM používajú diagramy funkčnej štruktúry, diagramy dátových tokov **DFD** (Data Flow Diagram) a podrobné slovné opisy funkcií. Pri tvorbe diagramu funkčnej štruktúry položíme na jednu hierarchickú úroveň funkcie, ktoré pracujú s objektmi zhruba rovnakej zložitosti. Uvažujeme o postupnosti vykonávania funkcií (dynamické hľadisko) a v tomto poradí ich ukladáme do FSD (Function Structure Diagram). Názvy funkcií tvoríme ako holé vety, kde podmet reprezentuje objekt spracovania a prísudok charakterizuje činnosť.

Model riadenia systému sa skladá z dvoch častí. Prvú tvorí diagram stavov a prechodov **STD** (State Transition Diagram). Je to opis všetkých stavov systému spolu s opisom podmienok prechodu medzi jednotlivými stavmi. Každý prechod systému z jedného stavu do druhého je uskutočnený akciou, ktorá sa začne až po splnení daných podmienok, alebo po zaregistrovaní daných udalostí. Rozdiel medzi

⁶ MOLNÁR, Z., 1992: *Moderní metody řízení informačních systémů*. - Praha : Grada

podmienkou a udalosťou je vlastne len vo formulácii. Podmienka je statické vyjadrenie udalosti.



Obr. 3 Príklad konceptuálneho diagramu dátových tokov

Diagram prechodov a stavov musí byť doplnený o riadiace toky v DFD (druhá časť modelu riadenia).

Riadiace toky a riadiace funkcie umožňujú priamo v DFD zapísať, ako sa k sebe majú chovať výkonné funkcie, ktorá z nich a za akých podmienok má byť aktívna.

Model štruktúry programového systému (Structure Chart) je už nástrojom časti YSM nazývanej System Design – návrh systému. Je to súhrn programových modulov a väzieb medzi nimi. Diagram má hierarchickú štruktúru, v ktorej sú povolené spoje aj medzi rôznymi úrovňami a vetvami.

Fázy vývoja systému podľa Yourdona:

[A] Prieskum uskutočniteľnosti (predbežná analýza), má dať odpoveď na otázku, či nový systém môže vyhovieť všetkým požiadavkám. Delí sa na nasledujúce etapy:

- Predbežný prieskum, počas ktorého sa získavajú základné údaje o skúmanom objekte a upresňujú sa ciele projektu.
- Analýza vonkajšieho prostredia, definovanie a skúmanie väzieb objektu na okolie, dekompozícia na podsystémy.
- Analýza existujúceho informačného systému, v rámci ktorej sa môžu navrhnúť racionalizačné zmeny organizačnej štruktúry.
- Vypracovanie plánu ďalšieho postupu (časový harmonogram projektu, odhad ekonomickej efektívnosti).

[B] Špecifikácia systému (analýza systému), ktorá pozostáva z nasledujúcich 5 etáp:

- 1) Vypracovanie modelu prostredia založeného na relevantných údajoch, tzv. esenciálneho modelu:
 - a) definovanie hraníc medzi objektom a okolím,
 - b) definovanie cieľov systému,
 - c) vypracovanie tzv. kontextového diagramu,
 - d) zostavenie zoznamu činností,
 - e) opis vonkajších informačných tokov a dátových pamätí,
 - f) zostavenie ERD,
 - g) zostavenie hrubého dátového modelu.
- 2) Vypracovanie modelu chovania, ktorý obsahuje:
 - a) opis chovania systému a podsystémov pod vplyvom vonkajších udalostí,
 - b) zostavenie diagramu dátových tokov,

- c) zostavenie diagramu prechodových stavov systému,
 - d) opis dát v Slovníku dát,
 - e) kvantifikácia systému (objem a intenzita dátových tokov).
- 3) Vypracovanie hrubého návrhu testovania systému.
- 4) Ekonomické zhodnotenie návrhu.
- 5) Zostavenie modelu systému, ktorý je v podstate koncepciou navrhovaného informačného systému a skladá sa z:
- a) modelu funkcií (hierarchické diagramy funkcií),
 - b) modelu dát (ERD a DFD),
 - c) modelu riadenia (STD).

[C] Návrh systému.

- Modulárna konštrukcia (návrh optimálnej modulovej štruktúry systému).
- Návrh vstupov a výstupov (formuláre a zostavy).
- Návrh plánu testovania pre moduly a celok informačného systému.

[D] Implementácia (implementačné modelovanie).

Špecifikácia programov na základe opisu modulov a funkcií na najnižšej hierarchickej úrovni:

- programovanie,
- príprava testov,
- testovanie programov a ich doladenie,
- interpretačné testy, testovanie činnosti informačného systému, ako celku.

Metodológia SSADM

SSADM (Structured System Analysis and Design Methodology) bola vytvorená v roku 1981 firmou Learmonth Burchett Management Systems (LBMS), výrobcom úspešných CASE produktov.

SSADM pokrýva len proces analýzy a návrhu informačného systému. Člení proces analýzy a navrhovania informačného systému na 6 etáp, ktoré sa ďalej členia na kroky a úlohy. Prvé tri etapy sú zamerané na analýzu, ďalšie na návrh systému.

SSADM kladie veľký dôraz na overovanie výsledkov každej etapy, aby sa chyby a nedostatky nepreniesli do výsledného riešenia⁷.

Metodológia SSADM obsahuje tri základné zložky, ktoré do určitej miery kompenzujú nedostatky vyplývajúce z toho, že pokrýva iba fázu analýzy a návrhu:

- Slovník produktov pre riadenie projektu, pre technické produkty a pre zaistenie kvality. Slovník obsahuje zoznam všetkých dokumentov projektu, ich opis a vzájomné väzby.
- Štruktúrovaný model, udávajúci postupnosť krokov pri analýze a návrhu.
- Techniky používané pri návrhu.

Etapy navrhovania systému v SSADM

V metodológii SSADM sú fázy analýzy a návrhu systému rozdelené do 6 etáp, ktoré sa skladajú z konkrétnych krokov tvorených jednotlivými úlohami.

⁷ RABENSEIFER, A., 1993: *Moderné navrhovanie informačných systémov*. - Bratislava : Veda

Fáza analýzy pozostáva z etáp:

- Analýza existujúceho systému a aktuálnych problémov.
- Stanovenie požiadaviek.
- Voľba variantu technického riešenia.

Fáza návrhu obsahuje etapy:

- Návrh usporiadania dát.
- Návrh postupov.
- Návrh fyzického riešenia.

Špecifikácia jednotlivých etáp:

- 1. Analýza existujúceho systému a aktuálnych problémov.*
 - 1.1. Začiatok analýzy.*
 - 1.2. Funkčná analýza (analýza existujúceho systému).*
 - 1.3. Analýza štruktúry dát systému.*
 - 1.4. Vypracovanie zoznamu problémov (požiadaviek).*
 - 1.5. Revízia a schválenie výsledkov 1. etapy.*
- 2. Stanovenie požiadaviek.*
 - 2.1. Logická definícia systému.*
 - 2.2. Konsolidácia požiadaviek používateľa.*
 - 2.3. Výber riešenia.*
 - 2.4. Podrobnejšie definovanie zvoleného variantu.*
 - 2.5. Vytvorenie požadovanej štruktúry dát systému.*
 - 2.6. Stanovenie poradia zmien dát. Podstatou je vytvorenie histórie života entít (entity life histories ELH).*
 - 2.7. Špecifikácia a prototypy dialógov.*
 - 2.8. Revízia špecifikácií požadovaného systému.*
- 3. Voľba variantu technického riešenia.*
 - 3.1. Vytvorenie technických variantov.*
 - 3.2. Kompletizácia a revízia systémových špecifikácií.*
 - 3.3. Stanovenie výkonnostných cieľov.*
- 4. Návrh usporiadania dát.*
 - 4.1. Relačná analýza dát.*
 - 4.2. Vytvorenie Composite Logical Data Design.*
- 5. Návrh procesov a postupov.*
 - 5.1. Revízia ELH.*
 - 5.2. definovanie logických procesov.*
- 6. Návrh fyzického riešenia.*

Objektovo orientovaný prístup k tvorbe IS

Pri objektovo orientovanom prístupe (**OOP**) sa chápe IS ako súhrn objektov, ktoré sa nachádzajú v tzv. problémovom priestore. Analýza a návrh systému na vysokej úrovni sa uskutočňujú nielen v pojmoch týchto objektov, ale aj v pojmoch služieb, ktoré si tieto objekty navzájom poskytujú. Objekty spolupracujú medzi sebou prostredníctvom správ, ktoré postupujú informácie, vyzývajú objekty implementovať určitú procedúru a pod.

Jedným z cieľov OOP je vytváranie všeobecných tried objektov, ktoré možno uložiť v knižnici programov pre opakované použitie. Súčasné uplatnenie prístupov zdola nahor a zhora nadol môže viesť k vytvoreniu veľmi robustných systémov.

Základnými zložkami objektovo orientovanej metodiky sú: objekt, trieda a dedenie, ktoré sa objavujú vo všetkých fázach projektovania systému. Polymorfizmus a dynamické viazanie sa používajú vo fáze podrobného návrhu a vo fáze implementácie.

Objekty sú základnými entitami objektovo orientovaného systému. Objekt poskytuje služby klientovi. Klientom môže byť program alebo užívateľ.

Objektovo orientovaná analýza

Skúsenosti z navrhovania IS ukázali, že objektovo orientovaný návrh a programová implementácia sa nedajú efektívne uskutočniť na základe predchádzajúcej štruktúrovanej analýzy. Štruktúrovaná analýza sťažuje identifikáciu tried, nevyužíva možnosti dedenia a znemožňuje uchovať charakteristiky zapamätania a správania sa. Riešením tohto problému je vývoj odpovedajúcich metód objektovo orientovanej analýzy.

Po ukončení objektovo orientovanej analýzy IS, by mali byť k dispozícii potrebné znalosti o objektoch a správaní sa systému, o vzťahoch medzi objektmi a o dynamike interakcií medzi nimi. Tieto údaje musia byť špecifikované použitím jasnej a dobre definovanej terminológie, akceptovanej v danej problémovej oblasti.

V súčasnosti sa používajú rôzne metódy objektovo orientovanej analýzy, ktoré vykazujú rozdiely v jednotlivých etapách analýzy. Niektoré prístupy navrhujú v prvom kroku identifikovať objekty. Objektmi sú predmety reálneho sveta alebo myšlienkové konštrukcie pomenované v špecifikácii požiadaviek na systém podstatnými menami. Logické vlastnosti objektov možno odvodiť z prídavných mien a ich správanie zo sloviess. Voči tomuto prístupu sa namieta, že pre veľké systémy obvykle neexistujú úplné a formálne korektné špecifikácie požiadaviek⁸.

Iným prístupom je tzv. analýza správania sa objektov (objekt behaviour analysis), kde sa zdôrazňuje nevyhnutnosť poznania, čo sa v systéme deje, ako sa správa v rôznych situáciách. V priebehu analýzy sa priradí správanie jednotlivým častiam systému a zisťuje sa, kto správanie iniciuje a kto sa na ňom podieľa. Iniciátori a účastníci, ktorí hrajú v systéme dôležitú úlohu, sa definujú ako objekty a prisúdi sa im zodpovednosť za príslušné úlohy.

Uvedené prístupy ovplyvňujú poradie činností pri analýze.

Metodika vypracovaná Rumbaughom a kol., ktorá zodpovedá prvému z uvedených prístupov, uvádza toto poradie krokov:

1. Modelovanie objektov:

- identifikovanie objektov a tried,
- príprava slovníka dát,
- identifikovanie asociácií (vrátane agregácií) medzi objektmi,
- identifikovanie atribútov objektov a väzieb,
- organizovanie a zjednodušovanie tried objektov použitím princípu dedenia (inheritance),
- overenie, že existuje prístupová cesta pre pravdepodobné dopyty,
- iterovanie a zjemnenie modelu,
- zoskupenie tried do modulov.

2. Dynamické modelovanie:

- príprava scenárov poradia typických interakcií,
- identifikovanie udalostí medzi objektmi,
- príprava sledovania udalostí pre každý scenár,

⁸ RABENSEIFER, A., 1993: *Moderné navrhovanie informačných systémov*. - Bratislava : Veda

- vypracovanie stavových diagramov,
- porovnanie udalosti medzi objektmi, aby sa overila konzistencia.

3. Funkčné modelovanie:

- identifikovanie hodnôt vstupov a výstupov,
- vytvorenie diagramov tokov dát pre znázornenie funkčných závislostí,
- opísanie funkcií,
- identifikovanie obmedzení,
- špecifikovanie optimalizačných kritérií.

Je vidieť, že viaceré metodiky objektovo orientovanej analýzy si vypomáhajú prostriedkami štruktúrovanej analýzy: modelovanie objektov sa dopĺňa modelovaním funkcií s využitím diagramov tokov dát. Podobne v niektorých metodikách sa používajú diagramy prechodu stavov alebo entitno-relačné diagramy.

Analýzy správania sa objektov vyžaduje takýto postup analýzy:

1. Vytvorenie kontextu pre analýzu (predetapa):

- určenie účelu systému (business goals) a čiastkových cieľov,
- identifikácia primeraných zdrojov pre analýzu,
- identifikácia hlavných oblastí činnosti,
- vypracovanie predbežného plánu analýzy.

2. Porozumenie problému:

- vypracovanie scenára činnosti systému,
- podrobnejšie opísanie operácií systému ako množiny dobre definovaných služieb, ktoré si vzájomné poskytujú entity systému (iniciátori a účastníci služieb),
- vytvorenie slovníkov,
- určenie atribútov iniciátorov a účastníkov služieb.

3. Definovanie objektov:

- určenie objektov podľa stanovených kritérií,
- opísanie objektov a ich správania, vrátane prevzatých (dedených) atribútov a činností.

4. Klasifikácia objektov a identifikácia vzťahov:

- opísanie vzťahov,
- organizácia objektov do hierarchii.

5. Modelovanie dynamiky systému:

- vytvorenie zoznamu (slovníka) definícií stavov pre každý objekt,
- vypracovanie životného cyklu každého objektu a identifikovanie udalostí, ktoré ho ovplyvňujú,
- určenie poradia operácií v systéme.

Objektovo orientovaný návrh

Za jednu z významných predností objektovo orientovaného prístupu sa považuje plynulý prechod medzi fázou analýzy a návrhu, takže je niekedy ťažké viesť deliacu čiaru medzi oboma fázami. Z metodického hľadiska môžeme fázu objektovo orientovaného návrhu rozdeliť do troch krokov:

Podrobnejší opis internej štruktúry objektov postupom zdola nahor. Objekty sa vytvárajú z knižníc primitívnych objektov. Súčasne môže začať implementácia tried objektov na najnižšej úrovni.

Zavedenie hierarchických vzťahov dedenia. Vzhľadom na to, že v podrobnejšom návrhu systému, ku ktorému sa dospelo v predchádzajúcich krokoch, vznikli ďalšie

objekty, je potrebné realizovať prehodnotenie celej množiny tried. Prostredníctvom iteratívnej analýzy sa zistí, či nie je účelné zaviesť nové nadtriedy alebo podtriedy. Pritom sa uplatňujú princípy agregácie a generalizácie. Vyhotovia sa diagramy dedenia (inheritance diagrams).

Agregácia a/alebo generalizácia tried. Agregácia a/alebo generalizácia uskutočnená v predchádzajúcom kroku môže vyžadovať iteratívny návrat do predchádzajúcich krokov, prípadne až do opisu systému vytvoreného v etape analýzy. Nakoniec sa vytvorené komponenty systému dopracujú tak, aby boli dostatočne všeobecné a robustné a uložia sa do knižnice komponentov.

Nástroje objektovo orientovaného návrhu

Objektovo orientovaný prístup oveľa viac závisí od použitia podporných nástrojov a vývojového prostredia ako procesne orientovaný prístup. Uvedieme prehľad týchto nástrojov členený na rôzne kategórie:

- Pojmové nástroje
 - generalizácia/špecializácia, uplatňuje ich mechanizmus dedenia,
 - komponenty, jedna trieda sa môže použiť ako časť reprezentácie inej triedy,
 - kontrakty, špecifikujú del'bu zodpovednosti medzi dvoma triedami, ktoré spolupracujú,
 - zoskupenia/podsystemy/framework, techniky na vytváranie množín tried, ktoré majú koncepčné väzby.
- Prostriedky prístupu k definíciám, implementácii a štruktúre tried. Znovu použiteľnosť komponentov je silnou stránkou objektovo orientovaného prístupu. Táto znovu použiteľnosť je podmienená nástrojmi, ktoré sú prispôsobiteľné rôznym spôsobom použitia tried.
- Nástroje počítačového navrhovania s objektovo orientovaným prístupom.
- Nástroje na implementáciu, programovacie jazyky napr. Smalltalk, C++.

Perspektívne podporné nástroje by mali umožniť integrovanie návrhu na vysokej i nízkej úrovni. Na vysokej úrovni by mali podporovať rozoznávanie a zaznamenávanie vzťahov medzi entitami v problémovom priestore. Tieto vzťahy reprezentujú interakcie medzi komplexnými entitami a v dôsledku toho majú nekonečnú varietu. Výsledkom je zložitá štruktúra závislostí. Nástroj by mal poskytnúť takú reprezentáciu tejto štruktúry, ktorá môže byť základom pre aktívnu manipuláciu. Samozrejmosťou by malo byť využívanie bázy znovu použiteľných programových komponentov. Predovšetkým by ho mali užívateľ a vybaviť prostriedkom navigácie, ktorý bude vedieť sledovať všetky väzby vytvorené medzi triedami. Takýto nástroj musí byť veľmi všeobecný, pretože vzťahy, ktoré sa vyskytujú v objektovo orientovanom návrhu, sa pohybujú od aplikačne orientovaných vzťahov po štandardné vzťahy, ako je dedenie.

Počítačom podporovaná tvorba IS (CASE)

Pod často používanou skratkou **CASE** (Computer Aided System/Software Engineering) sa uvádzajú produkty pre počítačom podporovaný vývoj softvérových a informačných systémov. Všeobecná definícia považuje CASE za súbor nástrojov a metód podporujúcich inžiniersky prístup k vývoju programových produktov na všetkých stupňoch procesu. Pod inžinierskym prístupom sa rozumie dobre definovaná, koordinovaná a opakovateľná činnosť so všeobecne prijatými prostriedkami znázorňovania (zobrazovania), pravidlami navrhovania a štandardmi kvality.

V ďalšom sa budeme venovať len použitiu CASE pre vývoj informačného systému. Cieľom CASE je maximálne odbremeniť tvorcov informačných systémov od únavnej mechanickej práce a poskytnúť im voľnosť pre tvorivú činnosť. Zároveň CASE je technológia náročná na dodržiavanie pravidiel a postupov daných použitou metodológiou, na disciplínu pri projektovaní. CASE sa dá veľmi dobre použiť pre analýzu a návrh systému, v ktorom sa vyskytujú aj nepočítačové spôsoby výmeny informácií. Pomocou CASE môžeme napr. navrhovať organizačné štruktúry veľkých podnikov alebo inštitúcií⁹.

CASE umožňuje:

- kontrolu väzieb medzi jednotlivými časťami projektu, čím sa zvýši kvalita projektovaného systému a uľahčí jeho údržbu a ďalší rozvoj po implementácii,
- skvalitnenie a uľahčenie tvorby a udržiavania dokumentácie,
- vylepšenie komunikácie medzi používateľom a riešiteľom systému,
- zefektívnenie a skrátenie procesu projektovania automatizáciou zdĺhavých a pracovne náročných rutinných činností,
- zlepšenie koordinácie projekčných prác, najmä pri veľkých projektoch.

CASE ako technológia automatizovanej tvorby IS predstavuje súbor programov, ktorý je obvykle zviazaný s určitou metodikou analýzy a projektovania IS. Podporuje všetky, príp. iba niektoré časti tzv. životného cyklu informačného systému, počnúc od plánovania, cez systémovú analýzu, vypracovanie projektu, jeho implementáciu až po jeho údržbu.

Unifikačné trendy v navrhovaní IS

Európsky integračný proces sa prejavil aj na úrovni metodiky navrhovania IS. Projekt ***Euromethod*** má za cieľ vytvorenie metametodiky, zastrešujúcej a zosúladijúcej používané metodiky tvorby IS. V Európe sa dnes používa najmenej dvadsať medzinárodne známych a komerčne dodávaných metodík. Vznikla tým dosť neprehľadná situácia, ktorá sťažuje výber vhodnej metodiky na riešenie konkrétnej úlohy. Porovnávanie a vyhodnocovanie projektov IS pri konkurzoch je veľmi ťažké. Euromethod má byť nadstavbou umožňujúcou porovnať, prípadne spoločne využívať vybraný okruh metodík.

Cieľom projektu Euromethod je riešiť uvedené problémy tromi spôsobmi:

- Odstránením sémantických prekážok medzi tvorcami IS, vytvorením slovníka referenčných výrazov.
- Zlepšením vzťahu medzi dodávateľmi a odberateľmi metodík a prostriedkov tvorby IS vypracovaním smerníc pre analýzu ponúk.
- Vytvorením štruktúrovaného modelu tvorby informačného systému, ktorý poskytne spoločnú charakteristiku jednotlivých činností a výsledkov tohoto procesu.

Práce na projekte Euromethod započali v roku 1989. Po úvodných štúdiách uskutočniteľnosti bola v roku 1992 vypracovaná tzv. počiatočná verzia projektu Euromethod.

⁹ RUSSEV, S.- ADAMEC, M.- BRDJAR, J., 1993: *Softvérové inžinierstvo a systémy CASE..* - Bratislava : Ekonomická Univerzita Bratislava