

Kapitola 1

Vysvětli pojmy: data, informace, znalost. Jaký je rozdíl mezi pojmy data a informace?

- **DATA** jsou atributy (vlastnosti) objektů vyjádřené pomocí znaků nebo signálů. Data nemusí obsahovat informaci. Vyjádřením dat pomocí znaků nebo signálů a uspořádáním podle syntaxe (skladby) vznikají zprávy.
- **ZPRÁVY** jsou DATA vyznačující se syntaxí (skladbou) a sémantikou (smyslem obsahu zprávy). Pokud zpráva nepřináší příjemci nic nového, pak neobsahuje informaci. V opačném případě je zpráva nositelem informace.
- **INFORMACE** je ZPRÁVA, která snižuje entropii (míru neurčitosti) systému, tzn. že doplňuje poznatky o jistých skutečnostech. Informace je nehmotná. Lze ji členit z hlediska času (statické, plánované, prognostické, operativní), působení na systém (vnitřní, vnější, aktivní, pasivní), opakovatelnosti, uchování, významu (nezbytné, důležité, zajímavé).
- **ZNALOSTI** jsou INFORMACE pamatované v podobě soudů. Znalosti jsou tvořeny praktickými poznatky a teorií a mají vliv na rozhodování. Mohou být převzaty od člověka, skup. lidí, případně z jiných zdrojů.

Jak lze definovat systém?

Systém je účelová množina prvků a vazeb mezi nimi, které společně určují jeho vlastnosti. Prvek je dále nedělitelná část celku.

Na jednom objektu (např. podniku) lze definovat mnoho systémů (informační, organizační, výrobní, vzdělávací...)

Co je to IS a jeho význam pro podnik

Informační systém je systém metod, lidí a prostředků zabezpečujících informační procesy.

Základním úkolem informačních systémů je získat informace, zpracovat informace a v potřebný čas, požadovaném rozsahu a formě, poskytovat na rozhodovací místa.

Informační systém musí poskytovat potřebné informace všem úrovním řízení, tj. *vrcholovému, taktickému* i *operativnímu* řízení.

Význam: IS jsou jedním z nejvýznamnějších faktorů ovlivňujících ekonomiku vyspělých zemí, protože informacemi jsou podmíněny znalosti. Bez IS a IT je dnes rychlá inovace výrobků a služeb nepředstavitelná. Výpočetní technika je dnes využívána ve fázi výzkumu, vývoje, technické a technologické přípravě výroby, řízení výroby až po odbyt a dodání výrobku zákazníkovi.

- IS umožňuje managementu podniku rychle se rozhodovat a reagovat na změněné podmínky na trhu.
- Slouží k jednotné a přesné informovanosti všech pracovníků podniku
- Pomáhá při řešení obtížných rozhodovacích problémů
- uchovává získané informace pro zabezpečení konzistentního chování podniku při zvýšené migraci pracovníků (tj. aby s odchodem pracovníka "neodešly" i potřebné informace);
- zvyšuje konkurenceschopnost využitím nových kvalitních služeb informačního systému

Trendy IS při řízení podniku

1. Zplošťování organizačních struktur

- jistý odklon od hierarchických struktur

- zvýšení pružnosti podniku a zvýšení iniciativy nižších článků organizační struktury

a) **Reengineering** = tvorba relativně nezávislých jednotek zaměřených na určitý hlavní druh činnosti, tzn. efektivní komunikace, zpracování a přenos dat.

b) **Dočasné pracovní týmy** = pružná reakce.

c) **Virtuální pracovní týmy** = bez kanceláře, Internet

(IS umožňuje zplošťování organizačních struktur podniku.)

2. Vznik útvarů informatiky. Význam těchto útvarů se zvyšuje a nastává posun k vrcholovému řízení. Zřizují se funkce ředitele informatiky, který je přímo podřízen generálnímu řediteli podniku.

3. Vznik nových funkcí - správce počítačové sítě, správce databáze, informační manažeři, auditoři informačních systémů apod.

4. Minimalizace vlastního vývoje, tj. přechod na zajištění vývoje dodavatelským způsobem, a to buď nákupem typového ASW nebo projektováním specifického IS.

5. Zvyšování nároků na spolehlivost software a ochranu dat - organizační opatření podniku:

- archivace dat a tvorba bezpečnostních kopií na přenositelných médiích,
- dodržování zákonných ustanovení ve vztahu ke zpracování dat,
- systém školení a doškolování pracovníků podniku,
- systém konzultačních služeb při provozu IS,
- provozní řád včetně nastavení změn přístupových práv k službám IS atd.

Trendy v oblasti základního a aplikačního SW

▪ ZSW

Vývoj ZSW je veden směrem ke *standardizaci funkcí a uživatelského rozhraní operačních systémů*.

- a. *Využití grafického rozhraní a sílicí podpora, zvukového a video rozhraní*, výrazně snižují požadavky na odborné znalosti uživatele, i když na druhé straně zvyšují nároky na výkon počítače.
- b. *Rozvoj komunikačního software a s ním spojených služeb*. Např. elektronická pošta, přenos souborů z přístupných adresářů (FTP), služby pro prohlížení hypertextových dokumentů, vyhledávací služby v celosvětové síti INTERNET.
- c. *V databázových systémech* je zřejmým trendem přesun od hierarchických a síťových systémů k relačním:
 - správa malých databází (dbase, FoxPro, Acces)
 - správa rozsáhlých, často distribuovaných databází (Oracle)
 - podpora různých typů dat (strukturovaná data, grafy, obrázky, texty, audio, video)
 - zpracování dat spravovaných jiným systémem řízení báze dat atd.

▪ ASW (typový SW)

- a. Tvorba *technologicky orientovaného* aplikačního software (ASW). Např. OIS (Office Information Systems - informační systém kanceláře) podporuje standardní administrativní činnost. Tvorba textových dokumentů (WORD, TEX), elektronická pošta, tvorba tabulek a grafů, tvorba obrázků (COREL), schémat a prezentací, práce s malou databází (Access), plánovací kalendář, CAD systémy a softwarové systémy pro projektování IS (CASE), atd.
- b. Významnou vlastností technologicky orientovaných programových produktů je jejich vzájemná provázanost. Do tvořeného dokumentu lze převzít informace z databáze, doplnit grafy a výsledek zaslat elektronickou poštou adresátovi.
- c. *Zvyšování role typového aplikačního software* vyvinutého specializovanou firmou. Architektura typového aplikačního software bývá obvykle modulová, kde každý modul postihuje nějakou vymezenou část podnikových procesů. Např. konstrukční příprava výroby (kusovníky), technologická příprava výroby (postupy), operativní plánování, skladové hospodářství (resp. zásoby), doprava, prodej apod.
- d. Nasazování typových řešení probíhá obvykle ve spolupráci s dodavatelem, který provede nastavení modulů, vyškolí pracovníky a případně pomůže převést data z původních podnikových aplikací.
- e. *Zvyšování vnitřní integrace* modulů typového aplikačního programového vybavení například modulu účtnictví na příjem a výdej materiálu a fakturaci při odbytu výrobků, při plánování zakázky do výroby se mohou automaticky kontrolovat stavy potřebných materiálových položek na skladě a vystavovat objednávky na chybějící materiál apod. Integrace neprobíhá pouze na úrovni programů. Integrovaný IS vyžaduje i *integrovanou datovou základnu* podniku, jinak řečeno jednotnou databázi.
- f. Rostoucí podpora elektronické výměny dat (EDI - Electronic Data Interchange) mezi podnikovým IS a okolím (dodavatelé, odběratelé, banky apod.).

Jak lze využít IS pro zvýšení kvality výrobků nebo služeb?

Automatizací a optimalizací jednotlivých částí (příjem zakázky, kalkulace nákladů, optimalizace dopravy snížením spotřeby zpracování faktur, e-banka, snížení zásob, zvýšení kvality vyhodnocením zmetků, zvýšení prodeje reklamou na webu).

Vysvětlit konkrétní příklady snížení nákladů podniku v důsledku využití kvalitního IS.

- a) efektivní komunikace mezi středisky i navenek a zpracování a přenos dat,
- b) méně zaměstnanců (automatizace činností),
- c) menší náklady na provoz (práce doma).

Jak si lze vysvětlit přínos (pro uživatele VT) standardizace hardware.

Standardizace HW umožňuje provozovat stejné aplikace na počítačích různé velikosti. S výměnou HW a SW lze zachovat uživatelské rozhraní bez větších finančních nároků na změnu aplikace.

V čem vidíte přínos využití grafického uživatelského rozhraní?

Grafické rozhraní snižuje požadavky na odborné znalosti uživatele. Jedná se tak o jednoduché a nejširší zpřístupnění výpočetní techniky.

Proč jedním z trendů z ekonomického hlediska je podpora strategického řízení?

Kvůli jeho důležitosti, obtížnosti a stále ještě nedostatečné SW podpoře. Přínos je ve zrychlení cyklu výzkum-vývoj-výroba-prodej. Zvýšení konkurenceschopnosti. Kde a jak ušetřit čas.

Kapitola 2

Co je to systémová integrace? Jaké byly důvody jejího vzniku?

Systémová integrace je proces tvorby integrovaného informačního systému, tj. proces cílevědomého propojení všech komponent HW, SW, podnikových procesů a prostředí do fungujícího systému. To vše v souladu s informační strategií podniku. Je to proces, který nikdy nekončí a je nedílnou součástí konkurenčního boje. Systémová integrace je relativně mladou disciplínou softwarového inženýrství. Vznikla jako reakce na aktuální problémy praxe.

Hlavní příčiny vzniku:

- **rostoucí význam informací** (vnitřních i z okolí) pro prosperitu podniku a s tím související masové nasazování informačních technologií,
- **rychlý vývoj informačních technologií** (HW i SW),
- **vysoké nároky na kvalifikaci** a praktické zkušenosti pracovníků vyvíjejících a provozujících IS a informační technologie.

Jaké jsou základní požadavky na IS, aby se projevil jako strategický faktor?

- **Musí podporovat strategické cíle podniku**
- **Musí fungovat jako otevřený systém** pochopitelný jak řešitelům, tak uživatelům.
- Vedením podniku **musí být chápán jako jedno z hlavních jmění podniku.**

Jak si představujete HW integraci?

Je to integrace různých komponent počítače a přídavných zařízení, sítí atd. pro korektní funkčnost IS (ovladače k HW a aby IS fungoval na daném HW). Spočívá také ve školení uživatelů na daný HW. Důležitou roli hraje standardizace.

Charakterizujte integraci s okolím podniku:

Je to optimální přizpůsobení podniku měnícímu se okolí a zpracování informací vstupujících z okolí do IS podniku. Je to výměna požadovaných informací s významnými obchodními partnery (zákazníci, dodavatelé), státními orgány, bankami a prezentace/získávání informací pomocí Internetu.

Jak si představujete datovou integraci?

Jako vytvoření jednotné datové základny, kde data jsou vedena pouze jednou a pro potřeby všech středisek podniku. Nemělo by docházet k duplicitám.

Co představuje technologická integrace?

Je to integrace různorodých komponent IS (HW i SW). Rysem je distribuované zpracování dat (realizace distribuované datové základny v rámci počítačové sítě), která může být homogenní i heterogenní, LAN, MAN i WAN.

Co si představujete pod pojmem vnitropodniková integrace?

Úkoly v rámci integrace vnitropodnikových procesů spočívají hlavně v jejich zefektivnění (zkrácení doby procesu, minimalizace zdrojů a zkvalitnění procesů). Tento postup se nazývá Business Process Reengineering. Jeho součástí je také integrace podnikových procesů s funkcemi IS prostřednictvím subjektů, které jsou v interakci s IS prostřednictvím jeho rozhraní.

Uveďte reprezentativní postupy vývoje a realizace IS

1. Vlastní vývoj ASW, nákup ostatních komponent, integrace vlastními silami.

Klady: IS je šitý na míru podniku, jeho funkce odpovídají požadavkům podnikových procesů, IS lze realizovat postupně dle potřeb a požadavků podniku, detailní znalost IS pracovníky podniku a tudíž i snadné provádění změn, konkurence nezná slabé a silné stránky podniku,

Nedostatky: vysoké náklady, dlouhá doba řešení, obvykle nižší kvalita ASW (v důsledku nižší kvalifikace domácích řešitelů), nízká parametrizace (řešitelé jednoduše plní požadavky, které nemusí být dostatečně zobecněny) a tím možné prodražování budoucí údržby při fluktuaci řešitelů jsou značná rizika nekonzistence dílčích modulů systému

2. Vývoj ASW externí firmou, nákup ostatních komponent, integrace vlastními silami.

Klady: IS navržený „na míru“, jeho funkce odpovídají potřebám podnikových procesů, postupný nárůst je možný dle potřeby, konkurence nezná silné a slabé stránky IS, optimální využití znalostí interních i externích specialistů.

Nedostatky: vysoké náklady (obvykle vyšší než v 1. variantě), obvykle dlouhá doba řešení (ale kratší než ve variantě 1.), rizika úniku informací mimo podnik.

3. Nákup všech modulů včetně typového ASW od vhodných výrobců, integrace vlastními silami.

Klady: rychlá realizace, nižší náklady než v předchozích variantách, možná volba osvědčených řešení pro každou část IS, typové ASW je parametrické - snadnější implementace budoucích změn.

Nedostatky: procesy podniku se musí přizpůsobit typovému ASW, při nákupu od různých dodavatelů ASW je obtížnější integrace, mohou být obtíže s vazbami mezi aplikacemi a tím i relativně nízká stabilita IS, realizace netypových požadavků podniku může být obtížná.

4. Nákup celého IS od generálního dodavatele, včetně systémové integrace.

Klady: nejrychlejší realizace, nízké náklady, profesionální řešení každého modulu, lze vybrat osvědčená řešení, ASW obsahuje nejlepší zkušenosti širokého okruhu uživatelů, ASW je parametrické, což usnadňuje realizaci nových požadavků pouhým nastavením parametrů (pokud s takovým požadavkem ASW počítá), inovace IS garantována dodavatelem, rozložení rizik mezi podnik a dodavatele.

Nedostatky: nutné přizpůsobení procesů podniku typovému ASW, velká závislost na generálním dodavateli (schopnostech, serióznosti, stabilitě), rizika úniku informací.

5. Tvorba IS, integrace i provoz IS (části nebo celého) externím dodavatelem.

Klady: stejné jako ve var. 4. a navíc snížení nároků na provozní personál – více se může věnovat hlavním činnostem, provozuje-li externí provozovatel více IS, mohou být náklady podniku na provoz nižší než ve var. 4., možnost využívání nejprogressivnějších informačních technologií, protože systémový integrátor může rychleji obměňovat svůj technologický park.

Nedostatky: stejné jako var. 4. a navíc další růst závislosti na systémovém integrátorovi další zvýšení rizika úniku informací

Závěr: vlastní vývoj unikátního ASW je nejméně efektivní.

Co je to outsourcing, jeho výhody a nevýhody.

Outsourcing znamená využití externích služeb. To umožňuje podniku soustředit se na svou hlavní činnost a činnosti vedlejší, nebo některé z nich, zabezpečovat pomocí externího dodavatele. Opačný stav, kdy podnik obhospodařuje své zdroje interně, je pak nazýván *insourcingem*. *Outsourcing jako proces* se realizuje na základě pečlivě propracovaného projektu, který může podnik vypracovat samostatně nebo ve spolupráci s odbornou firmou.

Výhody:

1. **Možnost více se soustředit na hlavní činnost podniku.**
2. **Využití nejmodernějších komponent IS/IT.** Partnerství s poskytovatelem služeb na světové úrovni může přinést následující výhody:
 - přístup k novým technologiím, nástrojům a technikám,
 - eliminaci nákladů spojených se sledováním světového trendu v IT a s náklady na školení nových pracovníků,
 - umožnit lepší pracovní příležitost pracovníkům, kteří přešli k poskytovateli služeb IS/IT,
 - využití kvalitnějších metodiky a zkušeností pracovníků získaných od ostatních klientů poskytovatele outsourcingu.
3. **Efektivnější provádění reengineeringu.** (Při reengineeringu se podnik snaží zbavit podpůrných procesů, tj. procesů, které nejsou součástí hlavních činností podniku)
4. **Sdílení rizik.** Zadavatel výrazně redukuje rizika vyplývající z rychlého vývoje informačních technologií i vlastního provozu IS. Vzniká ovšem riziko vyplývající ze závislosti na poskytovateli, jehož výše je závislá na serióznosti a stabilitě poskytovatele služeb. Tyto faktory zřejmě nejvíce ovlivňují využití outsourcingu v ČR.
5. **Uvolnění zdrojů.** Trvalým úkolem managementu je co nejeftivnější využití zdrojů. To znamená, že je žádoucí přesouvat zdroje z okrajových činností do oblastí hlavní činnosti.
6. **Nedostupnost zdrojů.** V případě IS/IT jsou tímto deficitním zdrojem obvykle lidé, tj. specialisté na vývoj a provoz informačního systému.

Nevýhody:

- závislost na poskytovateli outsourcingu,
- možná nízká operativnost,
- nutnost řízení vztahu "podnik – poskytovatel",
- rizika zadavatele (snížení úrovně služby, zvýšení ceny služeb, možnost krachu poskytovatele),
- nekontrolovatelnost informačních toků mimo podnik.

Definuj Business Intelligence:

Poprvé byl termín „BI“ použit Gartnerem a dále pak popularizován Howardem Dresnerem jako „*proces zkoumání doménově strukturovaných informací za účelem zjištění trendů nebo jiných společenských informací pro tvorbu závěrů*“. V současné době lze nalézt celou řadu definicí, jako například:

1. BI – cesta k informacím a znalostem.
2. BI – cesta jejíž cílem je zlepšení rozhodování využitím široké škály informačních systémů, aplikací a technologií sběru, uchování, analýzy a zpřístupnění dat.
3. BI pokrývá širokou kategorii aplikací a technologií sběru, uchování, analýzy a poskytování přístupu k datům, které by umožnily uživatelům zlepšit podnikatelské rozhodování.
4. BI je proces profesionálního sběru, zpracování a šíření informací pro podporu rozhodování vedení podniků.
5. BI je proces zaměřený na zvýšení konkurenceschopnosti podniků inteligentním využitím přístupných dat v rozhodovacím procesu.

Vypište alespoň 5 příčin neúspěchů IS:

- Chybně postavená globální podniková strategie. Je-li globální podniková strategie chybně postavena, nelze ani od IS očekávat přínosy.
- Nerespektování vlastnických a organizačních změn v podniku
- Podcenění významu IS pro zajištění konkurenceschopnosti podniku.
- Malá angažovanost vrcholového vedení při inovaci IS.

- Řízení IS delegováno na příliš nízkou úroveň podnikové hierarchie.
- Řízení IS je odděleno od řízení organizačních záležitostí.
- Povrchní specifikace požadavků na IS.
- Budování IS bez jednotné koncepce.
- Nejednotná datová základna IS.
- Nedůsledné řízení projektů.
- Nedodržení termínu realizace a finanční náročnosti IS.

Vysvětlete nejdůležitější kritéria volby SW (ASW).

- **funkčnost** – tj. plnění požadovaných funkcí,
- **provazovatelnost** (v různém prostředí, např. aplikace pro prostředí Windows XP nejsou provazovatelné v prostředí OS UNIX),
- **spolehlivost** (ošetření chyb uživatele, žádné nedefinovatelné zastavení),
- **ochrana dat**
- **otevřenost a modularita** (pro perspektivní rozšíření funkcí),
- **náročnost na obsluhu** (příjemné uživatelské prostředí),
- **kvalita servisních služeb**
- **přijatelná doba řešení úloh**
- dobrá cena.

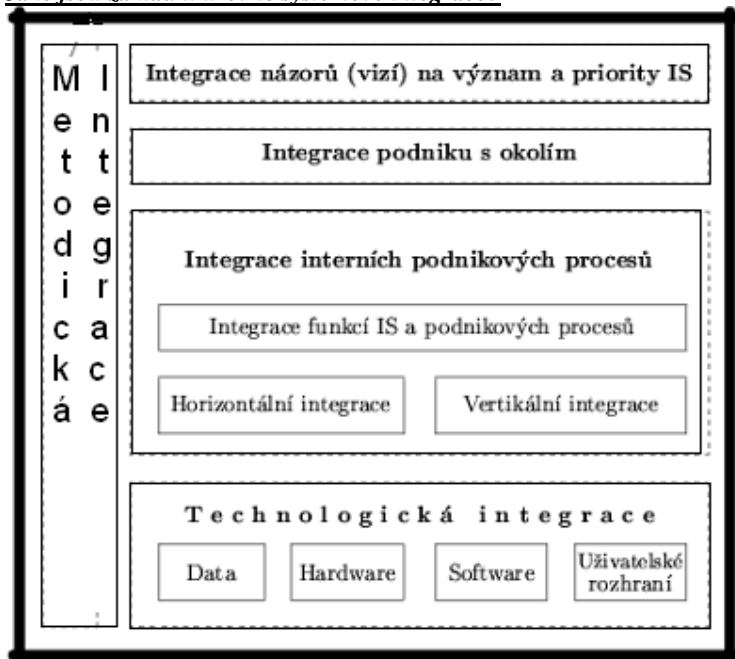
Jaké jsou základní principy systémové integrace?

1. **KOMPLEXNOST** – IS musí být řešen jako *komplexní systém*, který podporuje všechny významné vnitropodnikové procesy a je složený z řady různých komponent: počítačů a přídatných zařízení, sítí, základního SW, aplikačního SW, interních datových a externích datových zdrojů.
2. **STANDARDIZACE** – IS by měl být realizován na bázi mezinárodních standardů garantujících podniku jistou nezávislost na výrobcích techniky a software.
3. **JEDNOTNÁ KONCEPCE** – tj. IS je vyvíjen na základě jednotné koncepce (resp. Informační strategie) a metodiky.
4. **DISCIPLINOVANOST** – IS musí být nejen tvořen podle určitých pravidel, ale i provozován podle jednotných pravidel dodržovaných všemi uživateli systému. Tato pravidla jsou obvykle formulována v provozním řádu.

Proč je důležité akceptování mezinárodních standardů při tvorbě IS?

Garantuje podniku jistou nezávislost na výrobcích techniky a software, není dobrá přílišná tvořivost vývojářů.

Jaké jsou základní úrovně systémové integrace?



Co je ASP, výhody a nevýhody.

ASP (Application Service Providing) znamená pronájem přístupu k běžným komerčním aplikacím. Nabízí zákazníkům přístup do nového aplikačního prostředí bez nutnosti nákupu SW licencí, serverů a najímání vyškolených specialistů.

Výhody - nabízí odstranění nákladného procesu na zdokonalování a pořizování nového vybavení, vysokých výdajů na získání a udržení kvalifikace interních specialistů

Nevýhody - neexistuje žádná vazba mezi tvůrcem ASW a uživatelem, a tedy uživatel nemůže očekávat změnu funkčnosti aplikace podle jeho potřeb, vazba mezi provozovatelem aplikace a tvůrcem je uživatelem neovlivnitelná, atd. Je tedy otázkou, zda riziko pronájmu ASW pro podporu rozhodujících činností firmy je adekvátní přínosům.

Kdo je systémový integrátor?

Může to být osoba, skupina odborníků, nebo firma. Systémový integrátor zajišťuje komplexní realizaci systémové integrace. Většinou se jedná o externí firmu.

Uveďte příklad špatné podpory globální strategie informační strategií.

K neúspěchu tvorby IS dojde, je-li podniková strategie postavena správně a nesprávně je odvozena informační strategie, tj. informační strategie nepodporuje globální strategii podniku.

Příklady:

- 1) neustálé snižování výrobních nákladů,
- 2) budování dobrého jména firmy, výrobku nebo služby,
- 3) stálá orientace na jeden neměnný předmět činnosti,
- 4) orientace na velké trhy,
- 5) restrukturalizace založená na odprodeji částí podniku.

Kapitola 3

K čemu slouží globální architektura IS?

Je to základní schéma vyjadřující hrubou podobu budoucího IS. Má svoji *grafickou i verbální podobu*. Základem jsou stavební bloky reprezentující nějakou aplikaci a jí odpovídající funkci, datovou základnu, aplikační software, základní software a hardware.

Pro každý blok je charakteristická jeho:

- věcná orientace (např. účetnictví, sklady, TPV, prodej, personalistika atd.),
- vztah k úrovni řízení (operativní, taktické, strategické),
- způsob pořízení (typový ASW, vlastní vývoj nebo na zakázku),

- parametry softwarových produktů na trhu a podmínky jejich provozu.

Vymenujte a stručně charakterizujte dílčí architektury IS (strukturovaného přístupu k tvorbě SW).

- funkční- Představuje hierarchii funkcí jednotlivých bloků globální architektury. Je to verbální popis činností, které musí daná aplikace pokrýt. Neuvádí se zde ale jakým způsobem.
- procesní – zobrazuje se ve tvaru DFD, což je grafický model IS, který obsahuje následující základní prvky: procesy, datové toky, zásobníky dat, terminátory.
- datová – představuje návrh datové základny IS, nejčastěji je zobrazován jako relační model databáze.
- technologická - rozhoduje o technologickém řešení aplikace. Propojuje softwarovou, hardwarovou a datovou architekturu. Definuje způsob zpracování jednotlivých aplikací (*dávkové, interaktivní, řízené událostmi, v reálném čase*).
- softwarová – určuje, z jakých SW komponent bude IS postaven
- hardwarová - určuje typy, počty a vzájemné vazby HW komponent (serverů, osobních počítačů, tiskáren) a dalších přídatných zařízení

Procesní architektura, tvar zobrazení, zobrazení, význam, základní prvky.

- Zobrazuje se ve tvaru DFD (Gane & Sarsonova notace), což je grafický model IS, nebo jako esenciální diagram,

- je to jistý zjednodušený pohled na IS

- DFD obsahuje následující základní prvky:

- procesy – reprezentují místa transformace dat
- datové toky – reprezentují vazby mezi prvky systému
- zásobníky dat – neboli úložiště dat – slouží k uchování potřebných dat, označují se DMT
- terminátory (resp. externí entity) – jsou to prvky, které zobrazují prostředí okolo IS

esenciální model - Jako výchozí diagram prvků DFD se obvykle tvoří tzv. **esenciální diagram**, ve kterém je analyzovaná/navrhovaná část IS zobrazena jako jediný proces s vazbou na externí entity. *Tedy uvedeným diagramem se zobrazuje a definuje pouze okolí analyzované/navrhované části IS a vazby (datové toky) na okolí.* Tento globální proces se obvykle vůbec nepopisuje nebo se popisuje velmi stručně verbálně, protože definovat jeho algoritmus by mohlo být příliš složité.

Čím je dán konec dekompozice procesu v procesní architektuře?

Pokud lze jednoduše popsat algoritmus transformace vstupních datových toků na výstupní.

Jakým způsobem se definují procesy DFD?

Jejich popis lze provádět všemi vhodnými způsoby popisu algoritmů. Nejčastěji je prováděn prostým verbálním popisem.

Co je datový slovník?

Je to vyjádření datových struktur. Obsahuje **položky** (identifikátor, strukturu a komentář) a datové **elementy** (identifikátor, délka údaje, popis a typ). U rozsáhlého systému strukturuje datový slovník do několika částí (datové toky, zásobníky, datové prvky, číselníky).

Jakým způsobem lze popsat obsah datových toků a zásobníků dat?

= logickou strukturou. Pro def. log. struktury využíváme Backus-Naurovu syntax

K čemu slouží datová architektura a jak se zobrazuje?

- představuje návrh datové základny IS

- zobrazuje se jako DFD nebo jako ERD - podle složitosti se tvoří buď relační (pouze normalizovaná soustava logických struktur) nebo entitně-relační model (obsahuje grafickou část se zobrazením entit a vazeb mezi nimi). Tyto modely v případě relační struktury databáze, jinak nutno zvolit jiný model. **Prvky ERD:** 1) entity abstrakce reprezentující množiny objektů se stejnou logickou strukturou (stejnými atributy). Logický obsah entity se vyjadřuje názvem (podst.jm). 2) vazby = souvislost mezi objekty jedné (*unární*), dvou (*binární* vazba) nebo několika entit (*n-ární*). Logický smysl vazby se uvádí její názvem (sloveso).

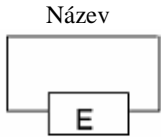
Jaké jsou základní vazby ERM (entitě relační model) + konkrétní příklady

1) unární = vazba sama na sebe, zvláštní případ bin. vazby,

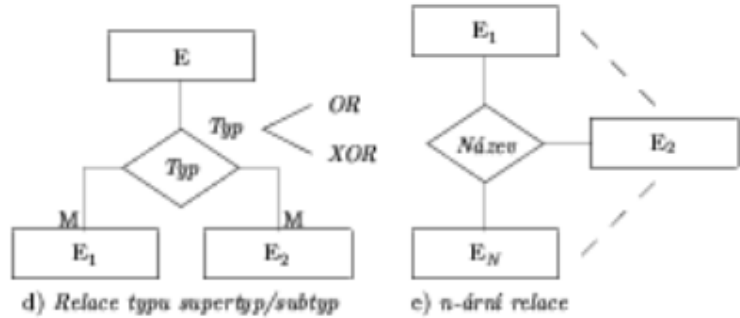
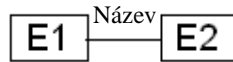
- 2) binární =vazba mezi dvěma entitami,
 3) n-ární =mezi několika entitami,
 4) supertyp/subtyp =spec. typ n-ární vazby. Reprezentuje větvení záznamů, které obyčejnou binární vazbou nelze znázornit.

Kardinalita (mohutnost) vazby, stanovuje na každé straně vazby (u každé entity). Vyjadřuje se přirozeným číslem a může být 0, 1, 5, 0...1, 1...3, 0...M, 1...M, atd.

a) Unární



b) Binární



Co obsahuje HW architektura?

Konfigurace HW je závislá na SW (standardním i aplikačním), který na nich bude provozován. Minimální i doporučená konfigurace je stanovena výrobcem.

Definitivní konfigurace =splněny doporučené optimální požadavky všech aplikací + rezerva pro další rozvoj IS, ne minimální. Definice kapacity operační paměti (velikost datových souborů + záložní soubory), diskové paměti a výkonu procesoru.

Konfigurace počítačové sítě =částečně z požadavků aplikací, zda bude síť homogenní nebo heterogenní, na jejím rozsahu, topologii, metodách přístupu, uložení dat, poskytovaných službách atd.

Co je obsahem technologické architektury?

Propojuje SW, HW a datovou architekturu. Definiuje způsob zpracování jednotlivých aplikací (dávkové, interaktivní, řízené událostmi, v reálném čase), který určuje jakými podněty jsou jednotlivé funkce spouštěny, jaká je doba odezvy, jaký vztah má zpracování funkcí k procesům reálného světa (zda aplikace probíhá ve stejném čase jako podnikové procesy).

Jaké jsou základní způsoby zpracování dat?

- 1. Dávkové zpracování** =požadavky i data se definují před spuštěním aplikace (v def. souboru). Čas spuštění aplikace není vázán na podnikové procesy a není rozhodující délka zpracování (odezvy) aplikace.
- 2. Interaktivní zpracování** =uživatel v přímém kontaktu s počítačem, požadavky zpracovávají okamžitě. Při sociálně ekonomické aplikaci (vystavení faktury, objednávky, počítačové hry, psaní dopisů, CAD, CAM, CASE).
- 3. Aplikace řízené událostmi** jsou spouštěny událostmi, které reálně nastávají. Jsou datové (zásoba materiálu klesla pod normu =objednávka), časové (v 17h uzavřít pokladnu a odeslat peníze na účet) a mimořádné (přerušení dodávky el. proudu =záložní zdroj).
- 4. Centralizované zpracování dat** =využívá pro uložení dat i programů jeden hlavní počítač (server). Pracovní stanice jednoduché, levné řešení.
- 5. Decentralizované zpracování dat** =na samostatných počítačích (není komunikace). Přenos dat na strojově čitelném médiu (disketa) nebo na papíře. Není operativní.
- 6. Distribuované zpracování** =několik vzájemně propojených počítačů do sítě (klient-server nebo peer-to-peer). Aplikace rozděleny podle místa vzniku dat a odpovědností za jejich aktualizaci a zpracování. Efektivní využití veškeré kapacity HW.
- 7. Kooperativní zpracování** =vyšší forma distribuovaného zpracování. Zpracování dotazu na spoustě počítačů a uživatel neví, které to jsou. Využívá síť.

Uveďte příklady použití dávkového a kooperativního zpracování dat?

Dávkové zpracování - při dávkovém zpracování se požadavky i data definují před spuštěním aplikace. Používá se u těch aplikací, které nevyžadují přímý kontakt s uživatelem a mají dlouhou odezvu (např. výpočet mezd).

Kooperativní - např. když se na zpracování dotazu pro vyhledání jisté informace podílí celá řada počítačů a uživatel ani neví, které to jsou. Nejrozvinutější forma je outsourcing.

Co je to primární klíč a způsoby jeho stanovení

Primární klíč – může to být tzv. přirozený klíč – rodné číslo osoby, osobní číslo zam-ce, DIČ firmy, SPZ vozidla, kód výrobku atd.

Stanovení primárního klíče

Tvorbu vložených klíčů je potřebné provést jako prioritní operaci.

Za primární klíč lze automaticky považovat datový prvek struktury. Nelze-li pro tvorbu primárního klíče entity využít ani prvků vložených klíčů, je nutno zvážit některou z následujících možností:

- rozšířit logickou strukturu entity o takový reálný prvek (skupinu), který bude plnit funkci primárního klíče. Nelze-li jinak, často se volí datum nebo čas vytvoření záznamu, případně pořadové číslo záznamu,
- zvolit pseudoprimární klíč. Tj. zvolit jeden nebo několik prvků, které sice v obecném případě nelze za primární klíč považovat, ale v daném případě jsou vyhovující,
- ponechat entitu bez primárního klíče. Tato varianta se může vyskytovat v případech, že její primární klíč nebude součástí jiné logické struktury.

Kapitola 4

Co si představujete pod organizační přípravou návrhu nového IS?

- stanovení základních principů tvorby a údržby integrovaného IS,
- funkce informačního manažera,
- vytvoření útvaru informatiky (pokud je potřebný),
- ustavení pracovního týmu pro formulaci informační strategie,
- výběr externích spolupracovníků a
- výběr systémového integrátora.

Cílem tohoto organizačního aktu je delegování pravomocí a odpovědnosti za stav a rozvoj IS na jednu osobu, která je schopna zabezpečit veškeré předpoklady (organizační, finanční i personální) pro rozvoj IS v souladu s globální strategií organizace. Nedílnou součástí pravomocí by mělo být právo disponovat s finančními prostředky určenými pro údržbu a rozvoj IS.

Jaké jsou povinnosti informačního manažera?

- odpovědnost za přípravu, realizaci informační strategie,
- sledování stavu IS, účinnosti informační strategie a navrhování inovací IS,
- odpovědnost za tvorbu vnitropodnikových norem, které souvisí s provozem IS,
- odpovědnost za realizaci zákonných ustanovení a předpisů týkajících se IS,
- zabezpečování zdrojů financování inovace a provozu IS,
- zabezpečení odborné přípravy všech uživatelů IS,
- spolupráce se systémovým integrátorem.

Co je strategické řízení a co je jeho úkolem?

Strategické řízení IS je kontinuální proces, který musí budovat a neustále udržovat integritu IS v úrovni vizí o IS ve vrcholovém managementu, okolí podniku, interních podnikových procesů, technologické (datová, hardwarová, softwarová, uživatelského rozhraní) a metodické.

Cílem strategického řízení je dát informačnímu systému a informačním technologiím podnikatelskou hodnotu, která se projeví v přidané hodnotě zboží a služeb. Tato podnikatelská hodnota IS musí být nejdříve formulována vizí, která dá smysl a cíl všem dalším aktivitám.

Úkolem strategického řízení IS je:

- formulovat vizí, cíle a podnikatelské hodnoty budoucího IS,
- určit cestu realizace cílového stavu,
- řídit přechod od současného do cílového stavu tak, aby byla neustále zachována integrita IS na všech dříve zmíněných úrovních.

Co obsahuje informační strategie, jaké jsou principy její tvorby?

Je to dílčí strategie navazující na globální strategii (další dílčí strategie: finanční, personální, marketingová, výrobní). Je to dlouhodobá orientace podniku v oblasti informačních zdrojů, služeb a technologií

Cílem je optimální podpora cílů podniku a podnikových procesů pomocí IT.

Obsah:

- a) popis a hodnocení současného stavu IS,
- b) definice cílového stavu IS,
- c) návrh možných cest transformace současného stavu do stavu cílového.

Principy tvorby:

- Řešení informační strategie nesmí trvat dlouho (ani u velkých podniků by nemělo převýšit cca 3 měsíce)
- Informační strategii tvoří tým složený z vrcholových pracovníků podniku, informatiků podniku a z externích konzultantů. Vedoucím týmu je ředitel informatiky, nebo není-li v podniku tato funkce vytvořena, pak pověřený odborný ředitel.
- Informační strategie pokrývá celý podnik, všechny jeho lokality, pobočky a předměty činnosti. Jinak by informační strategie nemohla zajišťovat integritu celého IS podniku.
- Informační strategie se zpracovává na období dvou až tří let.
- Informační strategie se mění tehdy, změnila-li se globální podniková strategie nebo se objevil nový trend v informačních technologiích, který by podnik mohl použít k získání konkurenční výhody, apod.

Popište základní fáze životního cyklu IS.**Kaskádový model ŽCIS**

U kaskádového modelu každá hlavní fáze cyklu ústí do následující nižší fáze projektování:

1) Studijní fáze. V návaznosti na informační strategii organizace se provádí přesnější specifikace problémů organizace a hledá se nejlepší řešení. Pokud informační strategie neexistuje, základem studijní fáze je provedení rámcové analýzy stavu informačního systému, definování základních problémů ve vazbě na strategii podniku, resp. hlavní podnikatelské cíle. Hlavním výstupem této fáze je **zpráva** (Úvodní studie, resp. Realizační studie, angl. Feasibility study), ve které jsou definovány základní problémy a způsob jejich řešení. Podle vypracované studie musí být management (resp. informační manager) podniku schopen rozhodnout o dalším postupu projektu. Po schválení studie vstupuje projektování do další fáze životního cyklu.

2) Analýza. V této fázi musí již analytik naprosto přesně analyzovat požadavky a řešit problémy související s jejich implementací do stávajícího systému. Tento návrh není ještě závislý na implementovaném prostředí.

je potřebné definovat:

- okolí procesu – tj. odkud získáme potřebné vstupy (jiné procesy, externí entity, úložiště dat) a kam budou směřovány výstupy procesu a s jakým obsahem.
- vazby tohoto procesu (vstupy, výstupy a jejich strukturu), tj. co potřebujeme pro realizaci požadavku a co bude jeho výsledkem,
- samotný proces – algoritmus transformace vstupů na výstupy,
- Jedná-li se o změnu funkce již existujícího procesu, pak je nutno dle potřeby modifikovat vstup/výstupy, externí entity, úložiště dat a algoritmus.

3) Návrh (nového systému). Činnosti prováděné v této fázi jsou potřebné pro získání informací nezbytných k vlastní realizaci softwarových komponent a jsou zaměřeny na:

- tvorbu hardwarové, softwarové a technologické architektury,
- pořízení a osvojení hardware a software (základního i typového),
- návrh (úprava) datové architektury, datových souborů, vstupních formulářů, tiskových sestav, obrazovek,
- příprava testovacích úloh,
- návrh organizačních směrnic (včetně změn organizační struktury pokud jsou žádoucí).

Hlavním výstupem je podrobná specifikace programů nového informačního systému, hardwarová a technologická architektura.

4) Realizace. Hlavní náplní této fáze projektování je vlastní tvorba a testování programových modulů navrhovaného informačního, realizace a ověření hardwarové architektury systému a příprava instalace nového systému.

Další aktivity jsou zaměřeny na:

- dokumentování programů,
- tvorbu provozní dokumentace,
- školení pracovníků uživatele,
- příprava instalačního plánu.

5) Instalace. Jedná se o poslední fázi před rutinním využíváním nového systému. Nainstaluje se nový hardware, software, provede se konverze nezbytných datových souborů ze starého systému (pokud existují) a zpracování zkušebních úloh. Hlavním výstupem této fáze je pouze protokol o instalaci a průběhu testování systému. Tím je zahájen zkušební provoz nového systému.

6) Vyhodnocení (zkušebního provozu) představuje závěrečnou fázi ŽCP. Doba trvání zkušebního provozu je závislá na náročnosti projektu. Hlavním výstupem této fáze je protokol o případných závadách systému a způsobu jejich odstranění.

7) Provoz. Tato fáze reprezentuje vlastní využívání nového informačního systému. Zde končí životní cyklus projektování.

Charakterizujte spirálový životní cyklus.

Spirálový životní cyklus je tvorba IS po částech.

Základem všech přírůstků IS je strategická analýza podniku a na ni navazující studie proveditelnosti. Nelze postihnout složitější model IS jako celek.

1.Návrh prototypu =zahrnuje etapu analýzy a návrhu včetně modelů. Důležitá součást je odsouhlasení návrhu uživatelem.

2.Realizace prototypu jak bude prezentována uživateli. Pokud nevyhovuje, upraví se návrh a následně realizace. Provádí se dokud není odsouhlasen uživatelem.

3.Implementace =integrace částí IS do komplexu IS, vyškolení obsluhy a vlastní provoz. Pokud se ve zkušebním provozu vyskytne závada, opakují se dle potřeby předchozí fáze.

Specifikujte náklady na tvorbu IS.

Jednorázové náklady. Vyskytují se pouze jednou v průběhu implementace informačního systému a obsahují:

- *Náklady na vývoj informačního systému* (u podniků, které provádí vývoj samostatně) – pořízení specifického HW a SW potřebného pouze pro etapu vývoje, výdaje za konzultace, oponentní posudky, mzdové náklady spojené se zaměstnáním externích pracovníků, dočasné vybavení pracovišť, cestovné, školení a kurzy analytiků a programátorů, atd.
- *Náklady na investice* – hardware, software, výstavba nových případně adaptace starších budov, vybavení kanceláří, realizace počítačové sítě, atd.
- *Náklady na vlastní implementaci informačního systému* (náběh jeho rutinního využívání) –konverze dat (prvotní záznam nebo z médií starého informačního systému na média a do podoby nového informačního systému), školení pracovníků podniku, konzultace expertů a souběžné zpracování informací podle starého a nového informačního systému (kde je potřebné).

Roční provozní náklady informačního systému:

- *Stálé* – mzdové náklady stálých zaměstnanců, energie, údržba hardware a software na základě smlouvy s příslušnou firmou, platby za licence, leasing (hardware, dopravní prostředky), pronájem prostor a úklid atd.
- *Variabilní* – penále, slevy, doprava, kurzy, semináře, cestovné, mzdové náklady externích zaměstnanců, spotřeba provozního materiálu, nákup specifického software, atd.

Nejvyšší položku nákladů tvoří obvykle mzdové náklady profesionálních pracovníků z oboru výpočetní techniky, kteří pracují na projektu, a investiční náklady na pořízení technického vybavení.

V čem spočívá analýza efektivnosti IS a k čemu se využívá?

Analýza efektivnosti zdůvodňuje rozhodnutí, proč přikročit k implementaci IS. Prostřednictvím analýzy efektivnosti (přínosů) se zkoumá, jak implementace IS ovlivní důchod (příjmy) podniku a využívá se ke stanovení doby návratnosti vynaložených prostředků na realizaci IS.

Přínosy se dělí na:

- a) nekvantifikovatelné – nedají se vyčíslit – přesnost a operativnost zpracovávaných informací, lepší řízení, lepší přístup k informacím, kvalitnější evidence, zlepšení jména firmy, pracovního prostředí
- b) kvantifikovatelné – odprodej projektu IS nebo jeho dílčí části, zvýšení výroby, odbytu, snížení nákladů

Uveďte příklady kvantifikovatelných a nekvantifikovatelných přínosů dílčích subsystémů IS (např.

ÚČETNICTVÍ, SKLADY, OPERATIVNÍ PLÁNOVÁNÍ, MZDY, HIM, NIM, atd.).

1.Nekvantifikovatelné =vlastnosti nebo charakteristiky systému, je obtížné je vyčíslit.

Př: přesnost a operativnost zpracovávaných informací se projeví v lepším řízení a v jeho důsledku snížením výrobních nákladů, penalizace a pokut, atd.

- a) kvalitnější evidence, přesnější a operativnější zpracování informací,
- b) zlepšení jména firmy,

- c) zlepšení pracovního prostředí,
- d) vyšší sortiment a kvalita informací pro zákazníky

2. Kvantifikovatelné

- 1) Jednorázové = odprodej projektu IS nebo jeho dílčích částí,
- 2) Pravidelné roční
 - a) *zvýšení výroby* = zlepšení plánování, zkvalitnění výrobní dokumentace a konstrukce výrobků, optimalizace výrobních postupů atd.,
 - b) *zvýšení odbytu* = zrychlení vyřizování objednávek,
 - c) *snížení nákladů* = úspora pracovníků využitím moderních IT, snížení nákladů na skladování a dopravu, úspory surovin a materiálu, atd.

Vysvětlete postup výpočtu doby návratnosti investic do IS.

Kvantifikovatelné přínosy se odrážejí v zisku firmy: $Zisk = (0,2Max + 0,5Pru + 0,3Min) \cdot Z$, kde: Max = optimistický odhad ročního nárůstu objednávek, Prum = průměrný roční nárůst objednávek, Min = pesimistický odhad ročního nárůstu objednávek, Z = průměrný zisk z jedné realizované objednávky. Firmu zajímá za jakou dobu se firmě vrátí investované finanční prostředky? Zjednodušeně (bez ohledu na inflaci): $N_J + T_N \cdot N_R - (P_J + T_N \cdot P_R) = 0$, kde N_J = jednorázové náklady na realizaci informačního systému, N_R = pravidelné roční náklady na provoz informačního systému, P_J = jednorázový přínos informačního systému, P_R = pravidelný roční přínos informačního systému. Z toho pak vypočítáme **dobu návratnosti vložených investic**: $T_N = (N_J - P_J) / (P_R - N_R)$.

Kapitola 5

Co jsou CASE nástroje?

Nástroje pro podporu tvorby IS

Zkratka CASE je akronym odvozený ze slov

- C – Computer
- A – Aided, Assisted, Automated
- S – Software, Systems
- E – Engineering

Která kombinace se zvolí záleží na tom co se autorům příslušného nástroje více hodí nebo líbí. Jednoduše řečeno CASE je nástroj, který slouží k vývoji, modernizaci a údržbě software.

Jaké výhody nabízí využití CASE nástrojů při tvorbě SW?

- eliminaci neproduktivního času projektantů (jednoduché provádění oprav, průběžná kontrola navrhovaných komponent, automatická archivace dat a tvorba dokumentace),
- zvýšení kvality software (projektant se může více soustředit na tvůrčí práci),
- urychlení procesu vývoje (zvýšení produktivity),
- jednodušší provádění změn (v důsledku automatizace uchování a editace všech dat o příslušném projektu),
- využití vhodných strukturovaně nebo objektově orientovaných technik,
- práci v příjemném interaktivní prostředí.

Z uvedeného vyplývá, že využívání CASE může mít značný dopad na pracnost tvorby software v jednotlivých fázích životního cyklu. Značně se ovlivní především systémová analýza a syntéza, ale i vlastní realizace software.

Co poskytují ICASE?

Nástroje CASE se dělí podle toho, které fáze životního cyklu podporují. Vývoj CASE nástrojů jde ovšem směrem k integrovaným CASE, tedy I-CASE, které podporují ve větší, či menší míře všechny fáze životního cyklu projektu.

Co lze považovat za nejjednodušší CASE?

Lower CASE – pro podporu fyzické, tj. programové realizace systému. Někdy se jim říká CAP (Computer Aided Programming)

Proč se v poslední době zdůrazňuje prototypování a co se od něj očekává?

Tvorba prototypu software se začala prosazovat od počátku osmdesátých let, a to současně s názorem, že strukturální metody neumožňují vhodně zobrazovat požadavky kladené na software. Jako řešení daného stavu se prosazuje objektivě orientované projektování a programování a prototypování.

Prototypování je v jiných oborech velmi rozšířené. Např. při vývoji automobilů, letadel, obráběcích strojů, elektronických zařízení (jmenovitě počítačů) atd. Prototyp se podrobí přísným zkouškám. Získané poznatky se pak promítnou do konstrukce a technologie pro sériovou výrobu.

Výroba a zkoušky prototypu se provádí za účelem spolehlivého vykonávání požadovaných funkčních vlastností daného výrobku, nebo snížení pracnosti, materiálových nákladů, zvýšení výkonu atd. Z podobných důvodů se prototypování prosazuje i v oblasti tvorby IS.

V čem spočívá rozdíl mezi falešným funkčním SW prototypem?

- funkcionální (prototyp je považován za podpůrný informační model, viz. spirálový životní cyklus, dokud neplní požadované funkce)
- nefunkcionální, tzv. falešné prototypování - podléhá uživatelskému rozhraní, vytváří se vzhled obrazovek a dotváří se formulace uživatelských požadavků

Kapitola 6

V čem spočívají důvody vzniku objektivního přístupu?

V posledním desetiletí (min. století) se začíná stále více prosazovat objektivě orientovaný přístup. Důvody jsou spojeny s řešením následujících problémů

- tvorba nového software je neustále složitější,
- růst požadavků na kvalitu software,
- požadavky na software rostou rychleji než jejich uspokojování,
- rychlá inovace výpočetní techniky,
- vysoký podíl údržby software,
- široké využití ne odborníky vyžaduje jednoduchou obsluhu (její realizace je často složitější než řešený problém),
- selhávání tradičních přístupů tvorby a údržby software (agendový, strukturovaný),
- existencí jisté sémantické mezery, která se projevuje v pohledu projektanta a uživatele na IS.

Jak lze chápat pojem objekt?

Objektem v širším smyslu se rozumí jakýkoli předmět, osoba, abstraktní pojem, činnost, který je předmětem našeho zájmu v problémové oblasti.

Objektem v užším smyslu rozumíme objekty používané v objektových modelech a software. K nim patří:

RWO – tyto v modelu nebo software symbolizují objekty a jejich vazby v reálném světě (Real World Objects),

CSO – tyto v modelu slouží k vytváření implementační struktury a řízení služeb software (Conceptual System Objects, resp. abstract objects),

SWO – Software Objects, resp. Computer Objects.

Jaké jsou základní charakteristiky objektů?

1) Stav objektu je definován hodnotami všech atributů v daný časový moment. Každý atribut může mít pouze hodnotu danou jeho definičním oborem. Tedy stav objektu v daném okamžiku odpovídá určité kombinaci přípustných hodnot všech jeho atributů. Jako obecné pravidlo objektů platí, že stav objektu se mění a lze jej považovat za výsledek předchozího chování objektu.

2) Chování objektu charakterizuje všechny schopnosti objektu něco dělat, popisuje jeho akce i reakce. Každá individuální komponenta chování se nazývá funkce objektu. Tyto reprezentují tzv. *funkční rozměr*. Volba operace je výsledkem vnější stimulace objektu reprezentované *zprávou* zaslanou jiným objektem.

3) Identita

Jak se klasifikují objekty podle chování?

Podle chování, resp. povahy interakce lze objekty rozdělit na následující typy

Klient – aktivní objekt s vazbou na objekty, kterým předává řízení. Typickým příkladem jsou metaobjekty lidí nebo pracovních kolektivů, které modelujeme.

Server – pasivní objekt. Objekty tohoto druhu jsou buď pouze předmětem na nich vykonávaných operací nebo pouze reagují na podněty, události vzniklé uvnitř systému. Je vždy cílovým místem zprávy.

Agent – objekt, který má vlastnosti obou předchozích. Je to objekt podobný lidem, kteří mohou na základě vlastní nebo vnější motivace vyvolat interakci s jinými lidmi. Objekty tohoto typu se mohou chovat jako filtry před jinými objekty.

Vysvětlíte pojem „zapouzdření“ SW objektu a úroveň zapouzdření.

Zapouzdření objektu – objekt se jeví svému okolí jako uzavřená jednotka, protože všechny jeho vlastnosti jsou v něm zapouzdřeny. Data zapouzdřená v objektu jsou chráněná proti nežádoucímu přístupu, což garantuje jejich integritu.

1) Privátní – privátní atribut nebo funkce třídy. Jsou to neviditelné, důsledně zapouzdřené části třídy.

2) Chráněné – charakterizuje prvky, které jsou viditelné (přístupné) pouze z dané třídy odvozeným objektům.

3) Veřejné – charakterizuje nejslabší úroveň zapouzdření. Atributy uložené v této části jsou viditelné všem třídám objektů.

Co je obsahem zpráv mezi objekty?

Obsahem jsou požadavky na provedení nějaké operace nebo odpovědi. Požadavky musí přicházet v souladu s protokolem a mohou ovlivnit provedení zvolené operace podle okamžitého kontextu požadavku. Zpráva je komunikační jednotka mezi dvěma objekty. Prochází komunikační vazbou, která dynamicky propojuje objekty oddělené v procesu dekompozice systému. Umožňuje aby funkce softwarové aplikace byla výsledkem spolupráce skupiny objektů. Zprávy jsou tvořeny kombinací řídicích a datových toků.

Jaké jsou základní typy přenosu zpráv mezi objekty?

Zprávy lze podle jejich funkce rozdělit do pěti následujících kategorií:

- *konstruktor* – pro tvorbu objektů,
- *destruktor* – pro rušení objektů,
- *modifikátor* – umožňují kompletní nebo částečnou modifikaci stavu objektu,
- *selektor* – vrací kompletní nebo částečnou informaci o stavu objektu,
- *iterátor* – pro prohlížení stavu objektu nebo obsahu datové struktury zahrnující několik objektů.

Jakým způsobem se zobrazuje interakce mezi objekty?

Interakce objektů se znázorňuje modely spolupráce. Model spolupráce je vhodný pro analytickou fázi, kdy se identifikují objekty. Objektové diagramy sice odrážejí statickou strukturu systému, ale s ohledem na vazby i jeho chování.

Nedostatkem těchto modelů je to, že lze přehledně zobrazit pouze omezený počet komunikujících objektů. Znázornění většího počtu zpráv u jednotlivých komunikačních vazeb je méně srozumitelným.

Co jsou třídy objektů?

Třídy lze považovat za abstrakce, které charakterizují vymezenou množinu objektů se společnými podstatnými vlastnostmi, což nám dává možnost zařadit každý objekt do nějaké třídy předmětné oblasti.

To ale neznamená, že všechny objekty třídy mají stejné všechny vlastnosti. Obecné vlastnosti jsou dány třídou a zvláštní vlastnosti objektem samotným. Konkretizace objektů z tříd se tvoří procesem, který se nazývá *instancionalizace* a jeho výsledkem je pak nějaká konkrétní *instance* (prvek) třídy.

Co jsou asociace?

Sémantické (významové) souvislosti mezi třídami. Množina všech asociací mezi třídami je daná množinou všech vazeb mezi objekty těchto tříd. Vazba mezi objekty představuje jistou komunikaci mezi těmito objekty zatímco její abstrakce reprezentovaná vztahem tříd osvětluje důvod této komunikace.

[Jak se zobrazuje model logický databáze v objektovém přístupu?.....](#)

Jaké jsou základní typy vazeb mezi třídami objektů? (Relační vazby – určitá souvislost)

- asociace – sémantická souvislost mezi třídami (unární, binární, resp. n-ární). Asociace představuje určitý druh vazby mezi třídami objektů.
- agregace – druh asociace, který představuje silnější vazbu mezi třídami. Jedna třída hraje důležitější roli nežli druhá třída. Agregace je asymetrická asociace. Je to vztah *celek – část, nadmnožina – podmnožina, nadřazený – podřazený*.
- generalizace – proces definování společných prvků, operací a omezení uvnitř množiny tříd a vytvoření nové třídy na základě těchto společných prvků. Nově vytvořená třída se nazývá supertřída.

Co je generalizace (specializace)?

Generalizaci lze chápat jako proces definování společných prvků, operací a omezení uvnitř množiny tříd a vytvoření nové třídy na základě těchto společných prvků. Nově vytvořená třída se obvykle nazývá supertřída.

Specializace je proces tvorby nové třídy (subtřídy), při kterém tato třída přebírá všechny prvky již existující (definované) supertřídy a rozšiřuje se o nové prvky.

Generalizace a specializace jsou dva obrácené pohledy na klasifikaci tříd, představují směry, ve kterých se hierarchie mění.

Uveďte příklad prosté agregační vazby mezi třídami objektů.

vazby mezi objekty typu *Student* a objekty typu *Univerzita*. Tyto vazby napovídají, že každý konkrétní student studuje na konkrétní univerzitě. Vytvoříme-li z dílčích univerzit třídu UNIVERZITA a ze studentů všech univerzit třídu STUDENT, je očividné, že i mezi těmito třídami bude existovat sémantická vazba (asociace).

Uveďte příklad kompoziční vazby mezi třídami objektů.

Kompozice – přísnější forma agregace.

Příklad kompozice s třídami AUTO, MOTOR, KAROSERIE. Odstranění instance třídy AUTO pak předpokládá i odstranění instance třídy MOTOR a KAOSERIE, které toto auto obsahuje. Kardinalita na straně komponent celku může být v případě kompozice libovolná. Další př. Každá budova má minimálně jedno podlaží a každé podlaží minimálně jednu místnost. Každá komponenta vstupuje do agregátu pouze jednou, ale agregát se může skládat z několika komponent. Je zřejmé, že odstranění instance ze třídy BUDOVA vyvolá nutnost odstranění příslušných instancí třídy PODLAŽÍ a následně i instancí třídy MÍSTNOST.

Co je kardinalita vazby a jak se stanovuje?

Kardinalita je mohutnost asociace ve vztahu k dané třídě. V diagramu se uvádí vedle odpovídající role.

Kardinalita charakterizuje výskyt instancí třídy vystupující v dané roli vzhledem k jedné instanci třídy s rolí na opačné straně asociace.

Co je vazební třída, jak vznikne?

Někdy se stane, že je potřebné přidat vazbám mezi objekty zvláštní atributy, které nemohou být obsaženy v množině atributů žádných z asociovaných tříd, protože konkrétní smysl mají pouze v souvislosti s asociací, resp. vazbou mezi konkrétními objekty.

Co je abstraktní třída?

Třída, která nemůže mít žádné konkrétní objekty.

V hierarchické struktuře dopravy je naprosto nemožné si představit dopravní prostředek bez nějaké jeho další konkretizace. Tedy třída DOPRAVNÍ PROSTŘEDKY je abstraktní třída. Abstraktní třídy mají zpravidla i abstraktní operace, tj. takové operace, které v rámci dané třídy nemají specifikovanou implementační metodu. Třída, která má alespoň jednu abstraktní operaci musí být abstraktní třída. Protějškem je *konkrétní třída*.

Vysvětlete pojmy „polymorfismus a dědičnost“.

Polymorfismus – mnohotvárnost, která je dána tím, že stejnou funkci může několik objektů vykonávat každý svým způsobem. Např. funkci "Zobraz" provádí jeden objekt na tiskárnu, druhý na displej a třetí na grafický zapisovač. V případě, že všechny funkce bude vykonávat jeden objekt, je nutné v požadavku zadat zobrazovací zařízení. Pak se bude jednat o řízení činnosti objektu na základě kontextu.

Dědičnost (inheritance) se týká vlastností objektů, která souvisí s pojmem třída. Objekty subtřídy dědí atributy a funkce supertřídy.

Co zobrazuje model jednání (use case) a jaké jsou jeho základní prvky?

Model jednání (Use-Case) je výchozím modelem objektového modelu systému. Popisuje to, co by měl nový systém dělat nebo co dělá existující systém ve vztahu k jeho okolí, resp. prvkům okolí, které stimulují jeho činnost.

Prvky okolí reprezentují aktéři a vlastní systém s jeho funkcemi specifikovaná jednání (angl. Use-Case) vyvolaná aktéry.

Model jednání celého navrhovaného systému může obsahovat několik dílčích diagramů jednání, jejichž základními prvky jsou aktéři, jednání a relační vazby. Dílčí typy jednání jsou popisovány prostým textem. Jako alternativu však zvolit i diagram aktivit, který bude probírán později. Nicméně je vhodné poznamenat, že pro komunikaci s uživatelem je vhodnější diagram jednání.

Aktér je v reálné situaci nějaký objekt (osoba, věc), který je v interakci se systémem. Tj. posílá zprávy systému nebo je ze systému přijímá.

Jednání představují kompletní funkci systému z pohledu aktérů. V UML je model jednání definován jako množina akcí nějakého systému, které poskytují registrovatelné výsledky dílčím aktérům. Za účelem prezentace výsledků mohou vyvolat aktivaci jiných jednání a komunikaci s několika aktéry.

Co obsahuje scénář?

Každé jednání představuje množinu posloupností a každá posloupnost, nazývaná scénář, jeden možný tok událostí vysvětlující chování (činnost) daného. I jednoduchý komplexní systém se může skládat z desítek jednání a každé jednání může představovat značný počet scénářů. V každém jednání lze však nalézt primární scénář definující hlavní posloupnost událostí a vedlejší scénáře popisující alternativní posloupnosti událostí. Vztah scénáře k danému jednání je stejný jako instance ke třídě, tj. scénář lze chápat jako jednu instanci jednání.

Základní typy komunikace (přenosu) mezi objekty

1) Jednoduchý přenos (simple broadcasting) se vyznačuje tím, že aktivní je odesílatel zprávy. Aktivace příjemce, tj. pasivního objektu se provede současně s odesláním zprávy, protože tento je neustále ve stavu očekávání zprávy. Znázorňuje se prostou šipkou orientovanou ve směru od odesílatele k příjemci. V programových systémech se vyskytuje všude tam, kde činnost pasivního objektu navazuje na činnost prováděnou předchozím objektem.

2) Synchronizovaný přenos zpráv se vyznačuje tím, že se spouští až tehdy, když je příjemce připraven zprávu přijmout. V průběhu přenosu je odesílatel blokován až do ukončení přenosu. O tom, zda je příjemce připraven přijat zprávu může být odesílatel informován prostřednictvím nějakého viditelného atributu stavu příjemce. Jako příklad lze uvést spojení mezi dvěma faxy.

3) Asynchronní přenos je charakteristický tím, že odesílatel odesílá zprávu bez toho, aniž by věděl, zda příjemce zprávu zpracuje nebo ne. Z hlediska příjemce i odesílatele může být asynchronní přenos uskutečněn kdykoli. Specifikace druhu synchronizace je obvykle prováděna při modelování přístupu ke kritickým zdrojům.

Kapitola 7

Uveďte hlavní funkce subsystému: INVESTIČNÍ MAJETEK

Podsystém Investiční majetek umožňuje přehlednou evidenci hmotného a nehmotného investičního majetku. Vedeme zde i drobný investiční majetek, o kterém účtujeme v účetové třídě 0.

Údaje o majetku jsou vedeny v podobě karet majetku. Na každé kartě jsou vedeny všechny důležité údaje o majetku, jako je datum pořízení, vstupní cena, daňové a účetní odpisy, odpisová skupina, zůstatkové ceny, postup odpisování, datum ukončení odpisu apod.

Podle stanoveného způsobu odpisování se automaticky vypočítají odpisy. Podle nich se pak automaticky snižují zůstatkové ceny. Mají-li být účetní odpisy stejné jako daňové, je možné na konci roku provést automatické dorovnání.

Všechny provedené pohyby jsou přesně zaevidovány s uchováním všech informací o majetku v době provedení pohybu.

Účetní odpisy a pohyby majetku se mohou automaticky zaúčtovat podle předem stanovených předkontací.

Daňové odpisování je možné přerušit a po novém zahájení odpisování se odpisy opět správně vypočítají podle zákona o dani z příjmů.

U každého majetku je možné podrobně rozepsat jeho příslušenství. Také je možné evidovat opravy a kontroly majetku, s rozlišením, zda se jedná o běžnou opravu nebo o technické zhodnocení, případně o pravidelnou nebo mimořádnou kontrolu.

Provedené daňové a účetní odpisy se každý rok zaevidují ve zvláštním souboru, takže je možné získat přesné údaje o odpisech minulých let. Je-li majetek vyřazen, může být karta se všemi údaji o něm uchována v archívu pro případné kontroly.

Pro každý majetek podsystem umožňuje tisk protokolu o zařazení, vyřazení a převodu majetku. Dále je možné vytisknout i vlastní kartu majetku, a to buď se všemi provedenými pohyby, s posledním provedeným pohybem nebo bez pohybů.

Různé pohledy na kartotéku dávají standardní sestavy, kde je možný výběr z mnoha způsobů třídění majetku.

- soupis měsíčních pohybů provedených v měsíci zpracování,
- evidence INM podle tříd,
- evidence INM podle skupin,
- výpis INM podle osobních čísel správců majetku,
- přehled rozmístění INM,
- přehled pronájmu INM,
- měsíční účetní odpisy,
- očekávané daňové nebo účetní odpisy v následujících letech,
- inventurní sestava,
- daňové odpisy za tento rok; slouží jako podklad k daňovému přiznání,
- roční účetní odpisy,
- evidence podle účtů,
- evidenční štítky,
- sestava o zaúčtování odpisů a pohybů INM pro daný měsíc.

FAKTURACE

Podsystem Fakturace slouží ke zpracování dokladů, které souvisí s dodavatelsko odběratelskými vztahy, a to i v zahraniční měně pro kontrakt s tuzemskem i zahraničím. Jsou to především faktury, dále pak dobropisy, dodací listy a platby.

Jednotlivé doklady je možné zadávat buď přímo v tomto podsystemu, nebo je přenášet z podsystemů jiných.

Faktury a dodací listy, které mají vztah k materiálu na skladě, se mohou zadávat i v podsystemu MTZ a do podsystemu Fakturace přenášet. Z podsystemu Banka a Pokladna je možné přenášet platby, které se po spárování automaticky zapíší k příslušné faktuře.

V podsystemu Fakturace vznikají věty, které obsahují informace pro další podsystemy. Je to především zaúčtování jednotlivých částí fakturačních případů, které se pak přijímá v podsystemu Účetnictví. Dále zde vznikají informace o celkových hodnotách faktur přijatých a vydaných a platbách k nim, které je možné v členění podle zúčtovacích období a partnerů přenášet do podsystemu Partneři (Fakturace, účetnictví, banka). Další informace lze přenášet do podsystemu Zakázka – jsou to přijaté a vydané faktury k jednotlivým zakázkám. Z faktur přijatých a z vystavených dobropisů je možné vytvářet požadavky na platby, které se přenášejí do podsystemu Banka a zde se na jejich základě tiskne příkaz k úhradě nebo složenka.

Možnosti zpracování pro faktury přijaté

- zadávání běžných faktur, dobropisů i proforma faktur,
- zálohové a vyúčtovávací faktury,
- dodatečné rozúčtování faktury do nákladů (likvidace),
- tisk likvidačního protokolu,
- rozpočítávání DPH na jednotlivé položky faktury (např. ze zjednodušeného daňového dokladu),
- rozpočítávání srážek nebo přírážek na jednotlivé položky faktury (např. balné),
- kopie faktury,
- storno faktury,
- stálá informace o stavu faktury,
- odkládání vyřízených faktur.

Možnosti zpracování pro faktury vydané

- \item vystavování běžných faktur, daňových dokladů, dobropisů, prodejek za hotové a proforma faktur,
- zálohové a vyúčtovávací faktury,
- tisk v jazyce českém a slovenském,
- možnost úpravy tvaru formulářů,
- výpočet DPH při dodržení zákonných pravidel pro ČR i SR,
- volitelné zaokrouhlování částky na faktuře,
- generování upomínek, možnost tisku hromadné upomínky pro partnera,
- generování penalizačních faktur jednotlivě nebo hromadně na partnera,
- vytváření sběrných faktur z dodacích listů,
- vzory pro jednotlivé řady faktur,
- kopie dokladu,
- storno dokladu,

- stálá informace o stavu faktury,
- odkládání vyřízených faktur.

Tiskové výstupy – sestavy z podsystemu Fakturace umožňují sledování

- knihy a deníku faktur,
- fakturačního saldokonta,
- nezaplacených a zaplacených faktur,
- očekávaných příjmů a výdajů,
- přijatých a poskytnutých plnění vzhledem k DPH,
- zaúčtování faktur,
- dlužníků včetně přehledu upomínek.

SKLADY

Podsystem Skladové hospodářství slouží k získání přesné evidence stavu a pohybů zásob materiálu, výrobků nebo zboží na skladech a v prodejnách.

Sklady je možné vést buď metodou průměrných cen nebo metodou FIFO. Vystavování dokladů se provádí jednoduchým způsobem, kdy se po určení čísla materiálu většina položek naplní automaticky a stačí je jen upravit.

Kromě příjmek a výdejek materiálu je zde možné vystavovat i přijaté a vydané dodací listy a faktury, prodejky, rezervace materiálu, požadavky na objednání materiálu, případně průběžky pro zaevidování současného příjmu a výdeje.

Všechny příjmy a výdeje jsou uchovávány v podobě skladových karet, takže je možné u každého materiálu získat přesný a podrobný přehled o změnách jeho stavu na skladě. Podle předem připravených předkontací pohybů je možné doklady automaticky zaúčtovat, s volbou, zda se má účtovat každý doklad jednotlivě nebo se stejné typy dokladů sečtou a zaúčtují se součty.

U každého materiálu je možné vést až 6 různých prodejních cen, které se mohou vypočítat automaticky podle předem stanovených koeficientů. Každému odběrateli přitom lze určit, za kterou cenu mu bude zboží prodáváno. Při prodeji je možné využít také množstevních slev materiálu případně procentuálních slev, které lze také nastavit přímo u odběratele.

Při výdeji z jednoho skladu na druhý je možné využít automatického převodu, kdy se podle výdeje z jednoho skladu automaticky vytvoří příjem na druhém skladu. Vystavené faktury a dodací listy lze automaticky přenášet do Fakturace, kde se s nimi dále může pracovat. Údaje o spotřebě materiálu na výrobu zakázek je možné přenášet do podsystemu Zakázka.

Informace o drobném majetku vydaném ze skladu do používání je možné v podobě karty majetku přenést do kartotéky Drobného majetku. Při vedení skladů na několika pobočkách je možné využít přenosy, pomocí nichž je možné přenášet údaje mezi pobočkami, případně z poboček do centra.

Průběžný přehled o stavu materiálu na skladech, umožňuje kdykoliv mít k dispozici informace o tom, co lze nabídnout ze skladu hotových výrobků.

Více než 30 standardních sestav dává různé pohledy na pohyby materiálu a jeho stav. Jsou to například tyto sestavy:

- regleta, v níž ke každému materiálu je uveden jeho počáteční stav na začátku měsíce, celkový měsíční příjem a výdej a konečný stav,
- obrátová soupiska – výpis všech materiálů, u kterých byl v daném měsíci pohyb, s uvedením všech pohybů, počátečního a koncového stavu,
- rekapitulace – hodnoty příjmů a výdeje na jednotlivých účtech skladu,
- seznamy příjmeček, výdeječek, rezervací, požadavků, průběžek
- hromadný tisk dokladů (faktur, prodejek, skladových dokladů),
- sestava opis dávky do účetnictví – způsob zaúčtování jednotlivých dokladů,
- sestava inventurních rozdílů zjištěných po provedení inventury,
- seznam materiálů převedených do DIM,
- stav skladu – seznam všech materiálů na skladě s uvedením cen a zásoby,
- nabídkový seznam materiálu s prodejní cenou pro odběratele,
- seznam materiálů podle účtů, na kterých je veden,
- inventurní seznamy,
- informace o vratných obalech,
- materiál nad a pod limitem,
- materiál bez pohybů,
- a další sestavy.

BANKA

Podsystem *Banka* je určený ke zpracování dokladů, které směřují z podsystemu *Fakturace*, *Účetnictví* nebo *PAM* do peněžního ústavu nebo z peněžního ústavu do podsystemu *Fakturace* a *Účetnictví*.

Ve směru do peněžního ústavu dochází ke zpracování požadavků na platby, které do podsystemu *Banka* přicházejí z podsystemů:

- a) *Fakturace* – požadavky na úhrady faktur přijatých a vydaných dobropisů,
- b) *Účetnictví* – požadavky na úhrady zaúčtovaných závazků,
- c) *PAM* – úhrady vyplývající ze zpracování mezd.

Pro platby, které se mají provést bez výše uvedených závazků (např. zálohy, leasingové splátky), je možné zadat požadavky i přímo do podsystemu *Banka*. Požadavky je možno pozastavit nebo dokonce rozdělit na několik částí, které se dostanou na různé výpisy.

Požadavky se podle data splatnosti zpracují na příkazy k úhradě, popřípadě složenky. Při vystavování příkazů je důsledně dbáno na to, aby v den splatnosti závazku už byly částky připsány na účet příjemce.

V příkazech se též eviduje, kterým výpisem byly uhrazeny, což ulehčuje zpětnou kontrolu. Příkazy je možné vytisknout pro Agrobanku, Komerční, Kreditní, Investiční, Čs.obchodní a Moravia banku, transformovat do souboru ve formátu, který tyto banky přijímají k automatickému zpracování.

Na zakázku je možné doplnit rozhraní i pro další peněžní ústavy. Platby složenek je možné realizovat pomocí složenek typu H. Současně se složenkami typu H se generuje i převodní příkaz na účet pošty s částkou, která obsahuje součet všech částek na složence a všech poplatků.

Ve směru z bankovního ústavu je možné přijímat věty o platbách z Agrobanky, Komerční, Kreditní, Investiční, Čs.obchodní a Moravia banky z běžných i devizových účtů v dávkách, které připravuje peněžní ústav. I zde je možné doplňovat na zakázku další rozhraní.

V případě, že organizace nepoužívá automatický bankovní styk, zadávají se přímo výpisy z účtu. Z těchto výpisů se vytvářejí věty o platbách, které je možné přenášet do podsystemu *Fakturace* a *Účetnictví*. Bankovní výpisy je možné i vytisknout. Po zpracování výpisů v podsystemech *Fakturace* a *Účetnictví* se zpět do podsystemu *Banka* dostane informace o tom, v kterém podsystemu byla platba akceptována. To znovu ulehčuje zpětnou kontrolu.

Při zpracování bankovních informací je uplatněna celá řada kontrol, jejichž smyslem je minimalizovat počet odmítnutých vět při automatickém zpracování. Jde hlavně o kontroly, které

- neumožňují zadat neexistující směrový kód,
- neumožňují zadat neexistující konstantní symbol,
- pomocí speciálního algoritmu kontrolují správnost zadaného účtu pro Komerční banku a Agrobanku.

ÚČETNICTVÍ

Podsystem *Účetnictví* umožňuje komplexní zpracování podvojného účetnictví podnikatelů, rozpočtových a příspěvkových organizací, stran, sdružení a spolků.

Jak již z výše uvedeného schématu vyplývá, je podsystem *Účetnictví* centrálním podsystemem celého systému EKOS/VEMA. Do tohoto podsystemu jsou shromažďovány všechny účetní informace vzniklé v ostatních podsystemech (*Pokladna*, *Fakturace*, *Banka*, *Zakázka*, *Investiční majetek*, *MTZ*). Podsystem tato data zákonem předepsaným způsobem zpracovává a vytváří výstupy ve formě účetních sestav, jichž je standardně dodáváno přes 40 druhů. Podsystem *Účetnictví* nevyžaduje spolupráci s ostatními podsystemy, což znamená, že je možné pořizovat data z jiných podsystemů přímo do podsystemu *Účetnictví*.

Podsystem dává uživateli možnost nadefinovat si prostředí, ve kterém bude pracovat tak, jak mu bude nejlépe vyhovovat. V účetnictví se jedná zejména o nastavení účtového rozvrhu a to jak používaných účtů a analytik k nim, tak i parametrů určujících jak se s daným účtem zachází v různých situacích.

Další oblastí, kde může uživatel výrazně ovlivnit složitost své práce s účetnictvím je používání účetních operací. Účetní operace jsou seznam a tabulka účetních souvztažností, které si může uživatel libovolně měnit a doplňovat. Jde v podstatě o nadefinování nejčastěji používaných účetních souvztažností. Účetní operace jsou popsány jako souvztažnosti, do kterých je možno přenášet hodnoty zapsané v Deníku – referentský pohled nebo dosazovat konstanty zadané přímo v popisu účetních operací. Při definování účetní operace volíme variantu deníku, čímž se rozumí výběr položek nutných pro vyplnění jednotlivé souvztažnosti.

Vlastní pořizování vět se provádí do Deníku, na který existují následující pohledy.

- Referentský pohled – zde se účtuje pomocí předem nadefinovaných účetních operací, což umožňuje účtovat osobě bez znalostí všech účetních souvztažností.
- Účetní pohled – slouží k účtování složitých účetních případů a pro osoby se znalostí účetních souvztažností.
- Úplný pohled – neslouží k pořizování zápisů, ale pouze k prohlížení všech údajů zápisů jak z účetního, tak z referentského pohledu.

Věty z deníku jsou nejméně jedenkrát měsíčně kontovány do Hlavní knihy. Podsystem *Účetnictví* řeší vyčerpávajícím způsobem problematiku účtování o dani z přidané hodnoty.

Je zde možno uživatelsky definovat až 99 druhů plánů a rozpočtů a posléze sledovat vývoj jejich plnění.

Podsystem umožňuje automatické rozpočítávání režii a to rozpočítaných podle procent nebo podle pevných základů nebo podle proměnných koeficientů. Rozpočet režii lze provádět buď na zakázky nebo na jednotlivá pracoviště nebo na zakázky v rámci pracoviště. Při použití projektu Režie je možno rozpočítané režie přeučítovat na jednotlivá pracoviště nebo zakázky.

Pro rozpočtové a příspěvkové organizace podsystem *Účetnictví* nabízí možnost rozšíření účtu o položky nové rozpočtové skladby. Sledování rozpočtu včetně rozpočtových opatření. Zákonem předepsané sestavy pro tyto organizace. Výstupy v elektronické podobě do systémů JASU, AFIS a JASU pro MUZO.

Účetnictví je možné zpracovávat na několika úrovních. To znamená, že např. účetní údaje jednotlivých pracovišť zpracovávané na těchto pracovištích je možno přenosit přenášet do *účetnictví* centrálního za podnik jako celek. Účetní data je možno podrobně zpracovávat v projektu *Finanční analýza*. Spojení s finanční analýzou otevírá nepřehledné množství pohledů na hospodaření organizace.

Jak již bylo uvedeno nabízí podsystem *Účetnictví* více než 40 tiskových sestav. Ke všem těmto sestavám si pak ještě uživatel může nadefinovat řadu variant lišících se úrovněmi třídění, sumarizování a uspořádání údajů v těchto sestavách.

- Základní sestavy – Rozvaha, Výsledovka, Předvaha, Hlavní kniha, Účetní deník.
- Sestavy pro plátce DPH – Daňové přiznání DPH, Seznam daňových dokladů, Seznam chybných vět a Záznamní povinnost.
- Další sestavy – závěrkové sestavy (Konečný účet rozvažný, Účet zisků a ztrát), statistiky (Rozvaha a Výsledovka), Opisy (Plánu a Rozpočtu), Účetní knihu, Rozbory, Režijní sestavy, Sestavy se zakázkami.
- Doplnkové sestavy – Druhové členění (členění nákladů a výnosů), Rozpis zůstatků a Rozpis denních zůstatků, Saldokonto deníku, Doklady a Saldokonto dodavatelů a odběratelů.

Kapitola 8

Základní technologie umělé inteligence pro podporu řízení podniku (řešení rozhodovacích problémů)

- logické programování,
- tvorba expertních systémů,
- fuzzy expertních systémů,
- neuronových sítí,
- hybridních inteligentních systémů (neuronových expertních systémů a neuro-fuzzy expertních systémů)
- využití genetického algoritmu.

Co je předmětem znalostního inženýrství?

Znalostní inženýrství (Knowledge Engineering) představuje obor zabývající se profesionální tvorbou znalostních systémů. Tyto systémy mohou být velmi prospěšnou součástí komponent informačních systémů zaměřených na podporu rozhodovacích procesů ve všech úrovních řízení.

Co je znalostní systém? Popište postup jeho tvorby a jaká jsou kritéria volby předmětné oblasti?

- je to programový systém, který obsahuje znalosti z určité specifické oblasti, a které mohou být využity pro řešení problémů z této specifické oblasti. Znalostní systémy zpřístupňují znalosti, které mohou být využívány na různých úrovních řízení. Do jisté míry nahrazují drahé, vzácné a pracovní vyčerpávající odborníky. Využívají se ve výrobě, obchodě, službách, účetnictví, finančním řízení, diagnostice, poradenství, prognostice a vojenství. Jde o široké zpřístupnění znalostí a jejich neomezené uchování...

- pro realizaci znalostního systému je potřebné

- definovat vhodnou předmětnou oblast a tým expertů, ochotných spolupracovat na tvorbě znalostního systému
- sestavit tým odborníků z oblasti informatiky schopných plnit úkoly "znalostních inženýrů",

- K problematice volby předmětné oblasti lze uvést následující kritéria

- Operace, dotazy (rozhodovací) předmětné oblasti by měly být převoditelné spíše do řady pravidel než-li matematických vzorců nebo rovnic. Znalostní systém nemusí být vhodný tam kde základem řešení jsou náročné numerické výpočty.
- Předmětná oblast musí být dobře prostudována, aby bylo možné poznatky z této oblasti správně formulovat a transformovat do podoby v jaké budou uchovávány v počítači.
- Předmětná oblast by neměla být zaměřena na problémy jejichž řešení je příliš triviální ani příliš náročné na čas.
- Při definování poznatků a pravidel řešení musí experti dospět ke všeobecné shodě názorů. Jinak nebude zajištěna obecnost expertního systému a navíc ztěžší může znalostní inženýr dělat arbitra expertům.
- Znalosti z problémové oblasti musí být dostatečně velké. Je-li báze znalostí a pravidel malá, pak může být manuální řešení problémů efektivnější než automatizované.
- Pro dosažení vyšší kvality znalostního systému by se na jeho tvorbě mělo podílet více ochotných expertů než jeden.

Jaká je souvislost mezi informacemi a znalostmi?

Každá uchovaná informace doplňuje naše znalosti, resp. poznatky o jisté realitě. Přijaté informace se v podobě soudů přetváří na poznatky a znalosti. Podstatou znalostí jsou tedy soudy a vlastnosti s nimi souvisejících objektů. Teorie a praktické zkušenosti tvoří bázi znalostí, která je využívána v rozhodovacím procesu při řešení problémů.

Jaké jsou základní činnosti znalostního inženýra?

- získání znalostí odborníků,
- zaznamenání znalostí ve vhodném tvaru (zkonstruovat model znalostí),
- vložit znalosti do počítačového programu,
- ověřit programový systém.

Vysvětlete základní části znalostního systému.

Jádrum je báze znalostí, která obsahuje znalosti lidí (expertů) v určité komprimované formě a z určité problémové oblasti. Obvykle jsou reprezentovány fakty a pravidly. Fakt je jasné a stručné prohlášení (tvrzení), které je pravdivé uvnitř zkoumané problémové oblasti. Další důležitou součástí je odvozovací mechanismus, který prohlíží fakta a pravidla a zajistí odpověď na požadovaný dotaz.

Jaký je rozdíl mezi znalostním a expertním systémem?

1) Znalostní systém - je programový systém, který obsahuje znalosti z určité specifické oblasti, a které mohou být využity pro řešení problémů z této specifické oblasti.

2) Expertní systémy jsou vybaveny silnějším dedukčním mechanismem. Na rozdíl od znalostních systémů by měly umět navržené řešení zdůvodnit.

Uveďte příklady faktů a pravidel v přirozeném jazyce.

Fakt je jasné a stručné prohlášení (tvrzení), které je pravdivé uvnitř zkoumané problémové oblasti.

Například: Vzdálenost z místa A do místa B je X km. Osoba A má dítě B. Osoba A je muž.

Použití pravidel se zakládá na známém logickém principu modus ponens pro odvozování. Příklad: jestliže osoba A má dítě a je žena, pak je osoba A matka. Jestliže je X banka, pak je to peněžní ústav.

Co je predikát?

Predikát – přísudek. Je to výrok, jehož pravdivost je závislá na hodnotách jeho atributů. Z pohledu výrokové algebry není zajímavý obsah výroků, ale zda je výrok pravdivý nebo lživý. Predikát vypovídá něco o objektu (jako přísudek o podmětu).

Uveďte příklady faktů a pravidel s využitím predikátů....

Fakt – P je vlastnost čísla být sudé

Fakt – číslo a je sudé číslo

pak Pa znamená, že a je objekt a P je predikát

Definujte postup tvorby znalostního/expertního systému.

- definovat vhodnou předmětnou oblast a tým expertů
- sestavit tým odborníků z oblasti informatiky schopných plnit úkoly "znalostních inženýrů"
- vlastní programování

Uveďte nástroje logického programování.

programovací jazyky ADA, LISP, PROLOG a CLIPS

Definujte fuzzy množinu?

Splňuje požadavek hodnocení stupně příslušnosti daného prvku do dané množiny podle které objekt může být prvkem dané množiny s jistým stupněm příslušnosti (jistoty, pravdivosti). Na rozdíl od klasických množin, je tedy fuzzy množina definována nejen prvky této množiny, ale i stupněm příslušnosti prvku k dané množině. Interval hodnot stupně příslušnosti je od nuly do jedné (výsledek je: určitě ano, určitě ne = klasická logika (buď nula nebo jednička) + spíše ano, téměř, spíše ne = fuzzy logika (něco mezi nulou a jedničkou)

Definice.

Nechť X je množina objektů x , pak $A = \{(x, \mu(x)); x \in X\}$ je fuzzy množina, kde $\mu(x)$ je charakteristická funkce (resp. funkce příslušnosti) definující stupeň příslušnosti prvku x do množiny A .

Co je fuzzy logika?

Základní rozdíl mezi klasickým odvozováním a odvozováním ve fuzzy logice spočívá v rozsahu pravdivostních hodnot výroků. Zatímco každý výrok klasické logiky může být pouze pravdivý nebo nepravdivý, ve fuzzy logice je pravda a lež záležitost stupně pravdivosti. Fuzzy logika je založena na možnosti zvažovat hodnotu logického výroku i někde mezi úplnou pravdou a úplnou lží. Může-li být logický výrok například pravdivý, skoro pravdivý, ne zcela lživý nebo lživý, pak tyto lingvistické hodnoty lze interpretovat jako stupně pravdivosti.

Jak se ve fuzzy logice využívají znaky OR, AND, NOT?

negace	–	not $X = 1 - X$,
konjunkce	–	X and $Y = \min(X; Y)$,
disjunkce	–	X or $Y = \max(X; Y)$,

kde x, y jsou stupně pravdivosti (příslušnosti). Například pro $X=0,2$, $Y=0,6$ stupeň pravdivosti výrazu

X and (not Y) or (not X) and $Y = \max[\min(0,2; 1-0,6); \min(1-0,2; 0,6)] = 0,6$.

Uveďte příklady vágních pojmů a jejich reprezentaci pomocí fuzzy množin.

Vágní pojmy – jsou to v běžné řeči používané intervaly hodnot. Jedná se o neurčité pojmy typu: malý, velký, výborný, asi ano, spíše ne.

If rating = výborný and Úvěr = malý THEN Výsledek = poskytnout

If rating = velmi dobrý and Úvěr = velký THEN Výsledek = spíše poskytnout...

Jaká je struktura fuzzy systému?

Základem tvorby fuzzy řídicích systémů jsou fuzzy množiny a fuzzy logika.

V čem spočívá výhoda fuzzy řídicího systému?

Jejich využití je vhodné právě v těch situacích, kde exaktní modelování rozhodovacích procesů je příliš složité nebo přímo nemožné, ale formulace rozhodovacích pravidel s využitím vágních pojmů je podstatně jednodušší.

Co je to fuzzyfikace a defuzzyfikace

Fuzzyfikace je transformace údajů o stavu řízeného objektu do podoby lingvistických hodnot, protože ve fuzzy řízení jsou pravidla řízení formulována v lingvistickém tvaru (tj. v pojmech s určitým stupněm pravdivosti).

Defuzzyfikace je opačný proces

Co je perceptron a jakou funkci realizuje?

Je to umělý neuron, který v umělé neuronové síti vypočítá vážený součet vstupních signálů, výsledek porovná s hraniční hodnotou a podle ní transformuje výstupní hodnotu na nula nebo 1.

V praxi bylo testováno několik druhů těchto funkcí, ale praktické uplatnění našli čtyři základní:

- Jednotkový skok $Y(x) = 1$ pro $x \geq 0$, $Y(x) = 0$ pro $x < 0$
- Funkce signum $Y(x) = \text{sign}(x) = 1$ pro $x \geq 0$, $Y(x) = -1$ pro $x < 0$
- Sigmoidální funkce $Y(x) = \frac{1}{1 + \exp(-x)}$
- Lineární funkce $Y(x) = k \cdot x$

Perceptron slouží k váženému součtu vstupních hodnot do neuronové sítě, jehož hodnota je porovnávána s hraniční hodnotou. Pokud je hodnota váženého součtu větší, než hraniční hodnota, pak je zaktivován výstup, nebo-li $\text{vystup}=1$, pokud je hodnota váženého součtu menší než hraniční hodnota, pak je $\text{vystup}=0$. Ta hraniční hodnota má speciální znak i jméno.

Umělá **neuronová síť** je zjednodušeným modelem biologické neuronové sítě (mozek člověka se považuje za nelineární paralelní systém zpracování informací). Skládá se z konečného počtu propojených jednoduchých procesorů – neuronů. Neurony jsou propojeny pomocí vážených vazeb přenášejících signály z jednoho neuronu na druhé neurony. Každý neuron má určitý počet vstupů a jeden výstup, ze kterého je výstupní signál veden na vstupy dalších neuronů.

Co je vícevrstvá neuronová síť?

- Nejrozšířenější a nejpoužívanější síť
- Jak pro klasifikaci tak pro regresi (a tedy i predikci časových řad)
- Síť s učitelem
- Aktivační funkcí je nejčastěji sigmoidální fce.
- Otázka výběru počtu vrstev a počtu neuronů
- K nevýhodám sítě patří obtížné řešení problému lokálních minim a poměrně dlouhá doba učení
- Zlepšení pomocí momentu, šumu, dalších neuronů

Nejrozšířenější je vícevrstvá neuronová síť s učením podle vzorů (s učitelem) Tj. lze sestavit soubor vstupních a korespondujících výstupních vektorů systému. Učení sítě představuje proces korekce náhodně zvolených parametrů sítě tak, aby se minimalizovala chyba na její výstupech, při požadovaných vstupech.

Co znamená učení vícevrstvé perceptronové neuronové sítě?

Proces učení představuje dynamický proces, při kterém dochází k modifikaci vhodných, tzv. nastavitelných, parametrů příslušné sítě za účelem dosažení požadované shody mezi výstupy z modelované soustavy a výstupy z neuronové sítě. V drtivé většině případů se proces učení soustřeďuje na adaptaci vah spojení mezi neurony. Někdy mohou být nastavitelnými parametry také strmosti aktivačních funkcí nebo postupná změna struktury neuronové sítě. Proces učení bývá charakterizován určitým algoritmem učení. Algoritmus učení obecně určuje, jakým způsobem dochází ke změně nastavitelných parametrů neuronové sítě. Algoritmus tedy představuje určitou strategii, která je kombinována určitými konkrétními matematickými operacemi.

Učení s učitelem – jedna z metod adaptace, při které se neuronová síť trénuje zvnějšku. Učitel zadává vstupní vektor dat, vyhodnocuje výsledek a provádí změny. Postupným snižováním odchylky od požadového výsledku dosáhneme zobrazení vstupů dostatečně blízko požadovaným výstupům.

Za jakých předpokladů lze využít vícevrstvou neuronovou síť?

Neuronová síť se skládá z vstupní, střední a výstupní vrstvy. Pokud středních vrstev je více, jedná se o vícevrstvou neuronovou síť. Vícevrstvou neuronovou síť lze použít pro ty klasifikační úlohy, pro které lze vybrat objekty (vzory) definující dílčí množiny.

V případě, že vzorové objekty podmnožin nejsou známy, řeší se tyto typy úloh pomocí samoučící se neuronové sítě.

Jaké konkrétní úlohy lze řešit neuronovou sítí?

Rozhodování o poskytnutí úvěru; Přidělování odměn na základě kvantifikovatelných charakteristik zam-ců; Klasifikaci výrobků podle kvalitativních charakteristik

Kapitola 9

Jaký je účel autorského zákona?

Účelem je pomocí tohoto zákona zabránit nelegálnímu šíření děl.

Kterým okamžikem je autorské dílo (SW) chráněno zákonem?

Autorské právo k dílu vzniká okamžikem, kdy je dílo vyjádřeno v jakékoli objektivně vnímatelné podobě a vztahuje se to jak na celek, tak i na jednotlivé části díla.

Proč je tvorba počítačových programů chráněna právem autorským zákonem?

Jelikož počítačový program (dle §65 zák. 121/2000 Autorského zákona), bez ohledu na formu jeho vyjádření, včetně přípravných koncepčních materiálů, je chráněn jako dílo literární. Počítačové programy jsou tedy autorská díla (autorské dílo je výsledek duševní činnosti), a to od r. 1996 literární díla. Myšlenky a principy, na nichž je založen jakýkoli prvek počítačového programu, včetně těch, které jsou podkladem jeho propojení s jiným programem, nejsou podle tohoto zákona chráněny.

Základní práva autora:

- **Autor má právo** osobovat si autorství, včetně práva rozhodnout, zda a jakým způsobem má být jeho autorství uvedeno při zveřejnění jeho díla. Právo na ochranu autorství je nepřevoditelné a je časově neomezené.
- na nedotknutelnost svého díla.
- své dílo užít a udělit jiné osobě smlouvou oprávnění k výkonu tohoto práva;
- na odměnu u zveřejněných děl, která lze rozmnožovat pro osobní potřebu. Osobou povinnou platit tuto odměnu je dovozce nebo výrobce zařízení pro pořizování rozmnoženin nebo poskytovatel rozmnožovacích služeb.
- na odměnu za tvůrčí práci. Výše odměny se řídí podle hodnoty díla a jeho významu pro společnost. Odměna náleží autorovi za každé užití díla.

Jaký je rozdíl mezi dílem souborným a spojeným?

Dílo souborné - vznikla tvůrčím zpracováním jiného díla. Zařadit dílo do *souborného díla* lze pouze se svolením autora. Pokud programy nějakého programového balíku budou naprosto samostatně využitelné (např. PC TOOLS), pak se bude jednat souborné dílo.

Dílo spojené – dílo, které pro svoji životaschopnost vyžaduje využití jiného díla. Ke spojení díla (nebo více děl) je nutný souhlas autorů. Programátor vytvořil programový balík „Marketing for Windows“. Tento programový produkt vyžaduje pro svoji činnost prostředí Windows.

Zákon 101/2000 Sb. O ochraně osobních údajů

Smyslem tohoto zákona je dosáhnout toho, aby se s osobními údaji občanů nakládalo jen podle stanovených pravidel. Tento zákon upravuje ochranu osobních údajů o FO, práva a povinnosti při zpracování údajů, stanovuje podmínky za nichž se uskutečňuje jejich předání do jiných států a zřizuje se Úřad pro ochranu osobních údajů se sídlem v Praze (dále jen Úřad). **Zákon se vztahuje** na osobní údaje, které zpracovávají státní orgány, orgány územní samosprávy, jiné orgány veřejné moci, jakož i fyzické a právnické osoby, pokud tento zákon nebo zvláštní zákon nestanoví jinak a na veškeré zpracovávání osobních údajů, ať k němu dochází automatizovaně nebo jinými prostředky. **Zákon se nevztahuje** na zpracování osobních údajů, které provádí fyzická osoba výlučně pro osobní potřebu a na nahodilé shromažďování osobních údajů, pokud tyto údaje nejsou dále zpracovávány.

Kdy se mohou zpracovávat citlivé údaje?

Zpracování citlivých údajů je možné provádět, jestliže

- a) subjekt údajů (FO, k níž se osobní údaje vztahují) dal ke zpracování výslovný souhlas – písemně (podpis a musí být zřejmé, k jakým údajům je dává, jakému správci údajů, k jakému účelu, na jaké období a kdo jej poskytuje). Souhlas může subjekt údajů kdykoliv odvolat.
- b) je to nezbytné v zájmu zachování života nebo zdraví subjektu údajů nebo jiné osoby nebo odvrácení bezprostředního závažného nebezpečí hrozícího jejich majetku, pokud není možno jeho souhlas získat zejména z důvodů fyzické, duševní či právní nezpůsobilosti, v případě, že je nezvěstný nebo z jiných podobných důvodů. Správce musí ukončit zpracování údajů, jakmile pominou uvedené důvody, a údaje musí zlikvidovat, ledaže by subjekt údajů dal k dalšímu zpracování souhlas,
- c) se jedná o poskytování zdravotní péče, jakož i jiné posuzování zdravotního stavu podle zvláštního právního předpisu, zejména pro účely sociálního zabezpečení,
- d) je tak stanoveno zvláštním zákonem.

Náplň práce Úřadu na ochranu osob. údajů je:

- provádění dozoru nad dodržováním povinností stanovených tímto zákonem při zpracování osobních údajů,
- vedení evidence oznámení učiněných podle §16 a registr povolených zpracování osobních údajů (dále jen "registr"),
- přijímání podnětů a stížností občanů na porušení tohoto zákona,
- projednávání přestupků a jiných správních deliktů a udělování pokuty podle tohoto zákona,
- zajišťování požadavků vyplývajících z mezinárodních smluv, jimiž je Česká republika vázána,
- poskytování konzultací v oblasti ochrany osobních údajů,
- spolupráce s obdobnými úřady jiných států.

Jaké jsou povinnosti správce a zpracovatele os. údajů?

Správce a zpracovatel je povinen:

- a) *stanovit účel* k němuž mají být osobní údaje zpracovány,
- b) *stanovit prostředky a způsob zpracování* osobních údajů,
- c) *zpracovávat pouze pravdivé a přesné osobní údaje*, které získal v souladu s tímto zákonem. Je povinen ověřovat, zda jsou pravdivé a přesné. Zjistí-li, že údaje nejsou pravdivé a přesné je povinen je blokovat a bez zbytečného odkladu opravit nebo doplnit. Nelze-li je opravit nebo doplnit, musí je bez zbytečného odkladu zlikvidovat.
- d) *shromažďovat osobní údaje odpovídající pouze stanovenému účelu* a v rozsahu nezbytném pro naplnění stanoveného účelu,
- e) *uchovávat osobní údaje pouze po dobu, která je nezbytná* k účelu jejich zpracování.
- f) *zpracovávat osobní údaje pouze v souladu s účelem*, k němuž byly shromážděny, pokud zvláštní zákon nestanoví jinak. Zpracovávat k jinému účelu lze osobní údaj, jen pokud k tomu dal subjekt údajů souhlas.
- g) *shromažďovat osobní údaje pouze otevřeně*; je vyloučeno shromažďovat údaje pod záminkou jiného účelu nebo jiné činnosti, pokud zvláštní zákon nestanoví jinak,
- h) *nesdružovat osobní údaje*, které byly získány k rozdílným účelům, pokud zvláštní zákon nestanoví jinak.

Druhy ochrany dat

Nejužívanější fyzická ochrana výpočetních systémů a tím i dat a software *spočívá v dokonalém zabezpečení budovy a místností*, ve kterých se výpočetní technika nachází. Důležitá zálohovaná data by měla být uložena v pancéřovaném ohnivzdorném trezoru a případně na dvou vzdálených místech.

Je vhodné, aby si každá firma

- vedla přesnou evidenci sériových čísel počítačů a jejich periférií (tiskáren, monitorů, atd.),
- nesmazatelně vlastním logem označila všechny hardwarové komponenty
- využívala hardwarově řešenou ochranu heslem, aby se ztížilo využití odcizeného počítače.

Také

- důsledně využívat hesla
- používat kódované údaje (včetně textu),
- provádět monitorování činnosti systému.

Ochrana dat proti počítačovým virům:

- provádět antivirovou kontrolu každé diskety, ze které se do počítače kopírují nové programy,
- vytvářet záložní kopie nových programů a pravidelně zálohovat datové soubory,
- provádět pravidelnou antivirovou kontrolu počítače, případně využít rezidentní antivirový program,
- nepoužívat pirátské kopie programů, ale pouze legálně získané programy od autorizovaných distributorů.