

Ing. Ivan Daida

OKR Automatizace řízení, k.ú.o.

## PROBLEMATIKA VZTAHU ANALÝZA - PROGRAMOVÁNÍ

- Ve svém příspěvku se chci zabývat dvěma problémy:
- první je pokus o vymezení vztahů analytik-programátor a cílem nahlédnout v tomto vztahu rozpory
  - a druhý je pokus o zavedení pojmu "programový panel" a výklad tohoto pojmu.

I. Na různých seminářích a poradách, kde převažuje profese programátorů, se obvykle místo jiné diskutuje o vztahu k "analytikům". Obecně vzato jde "o vztah k čemu, kdo mi předává práci". A opět obecně vzato, problém vztahů bude existovat vždy (ať jde o profesi užší nebo širší) a těmi partnery, s kterými jsem nucan přicházet v pracovní problematice do styku.

Aby komplikace porozumění byly co nejnižší, je zapotřebí hlavně dvou věcí:

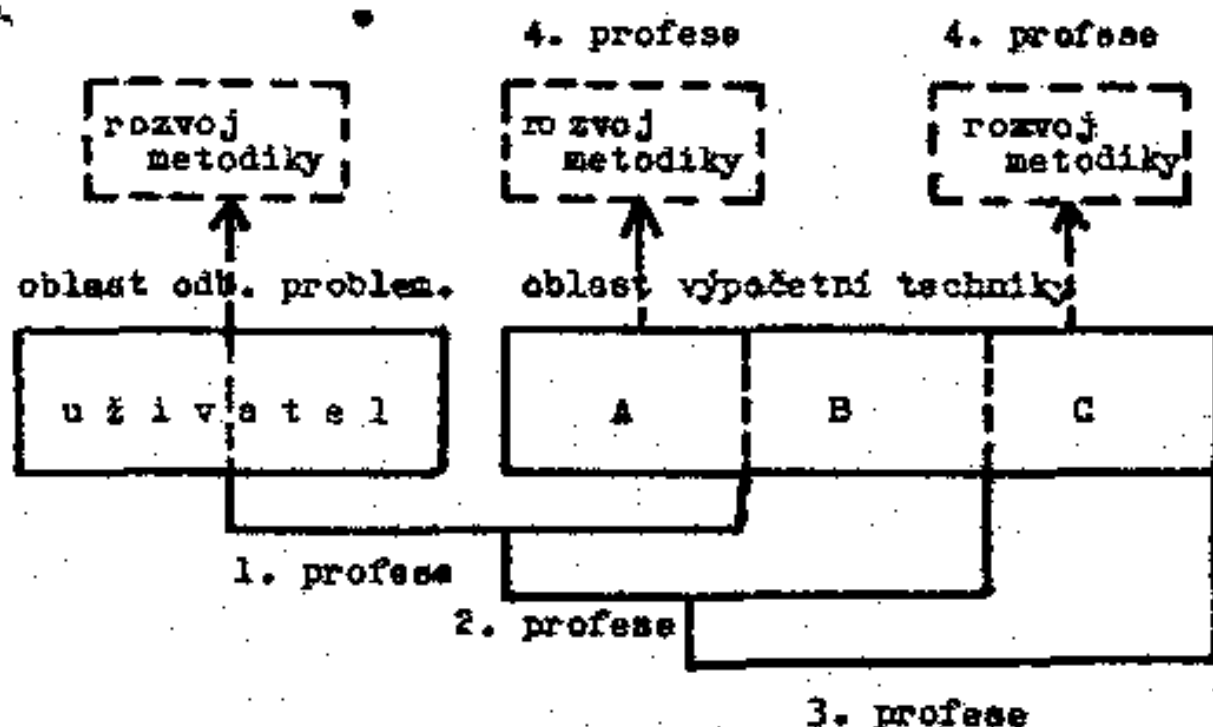
- vhodné a přijatelné normy styku mezi spolupracujícími profesemi
- vhodnou a účinnou organizační strukturu.

Obě by měly směřovat k tomu, aby se co nejrychleji a nej-  
kvalitněji sjistily úkoly, které spolupracující profese  
(útvary, oddělení atd.) musí plnit.

Tolik nezbytný úvod.

Během své praxe ve výpočetní technice jsem dospěl k určitému názoru o tom, jak by průběh zavádění různých aplikací do rutinního zpracování - z hlediska "toků profesí" - měl vypadat.

Průběh je zachycen v následujícím grafu:



Přímý a jednoduchý vztah uživatel řešitel, kde řešitel je prezentován jednou nebo více osobami, kteří "řeší všechno sami" od zadání úlohy až po její realizaci (někdy i výpočet), je běžný v případech, kdy:

- a) dochází v dané jednotce k prvnímu zavádění VT
- b) jde o úlohy malé rozsahem
- c) se otrocky převádí požadavky uživatele na zpracování VT
- d) jde o nasazení relativně jednoduché techniky.

Jakmile alespoň jeden z uvedených bodů se "začíná rozrůstat", musí objektivně dojít dříve nebo později k rozdělování profesí.

Osobně považuji za rozhodující moment okamžik, kdy pracovníci ve VT "odmítnou" převádět na počítač požadavky zákazníka a začnou podrobovat převáděné problémy systematické analýze s cílem vytvořit jinou kvalitu.

Přiznám si, že v současné době máme někde za sebou nebo někde právě probíhá období charakterizované "agendovým přístupem", které se vyznačovalo mimo jiné především tím,

že se současný stav (se všemi nedostatky a problémy) pouze převedl na počítač, pochopitelně s týmiž nedostatky a problémy, zvětšenými o potíže vyplývající z rozvoje VT jako takové. Není se pak čemu divit "úšklebkům" některých uživatelů.

Nastává je toto období se námi, rozvoj VT je toho důkazem, stejně jako "přežívání" zastaralé organizace a některých profesí, které jako by VT nebraly v úvahu.

Za účinné rozdělení "toků profesí" při počítačové zavádění systémů považují následující vymezení rozsahu práce:

### 1. profese -

pracovníci, kteří ovládají především odbornou problematiku a základní problematiku výpočetní techniky (rozsah - viz náplň bodu A). Jde přibližně o profesí systémových inženýrů. Náplň jejich práce by mělo být zvládnutí úvodní analýzy, vypracování ideového a technického projektu. Na prováděcím projektu (někdy i na technickém projektu) by již měli spolupracovat s profesí 2.

### 2. profese -

pracovníci, kteří ovládají důkladně problematiku výpočetní techniky od metodiky syst. inženýrství (pro autnou spolupráci se systémovými inženýry) až po konkrétní programování. Jde přibližně o funkci programátor-analytik, vedoucí programátor, samostatný programátor atp. Náplň jejich práce by měla být spolupráce na prováděcím projektu (tvorba souborů, progr. schémata atd.), řízení programovacích prací, ladění systémů a permanentní dohled nad "programovými balíky" svěřených oblastí.

### 3. profese -

pracovníci, kteří ovládají problematiku běžného programování. Jde o funkci programátor. Náplň jejich práce je především kódování dílčích programů podle přesných a podrobných specifikací a zajištění odladění těchto programů.

#### 4. profese -

jde o profesi, které se vyzývá běžnému nutnému toku. Pracovníci této profese by měli být špičkovými odborníky ve své oblasti a svou práci rozšiřovat metodický záběr (bohatství) dané oblasti (tvorba obecných postupů, zavádění nových aplikací atp.). Výsledky jejich práce by pak promítali do své činnosti řadoví pracovníci přísl. oblastí.

Pokud jde o okruh znalostí v oblastech A, B, C - lze je stručně charakterizovat takto:

oblast A - ovládnutí metod systémového inženýrství

- zásady tvorby systémů
- základní znalosti hardware (co, jak, na čem)
- základní znalosti software (způsob práce počítačů aj.)
- znalost práce a paměti počítače

oblast B - podrobná znalost software (operační systémy atd.)

- důkladná znalost programování
- praktická znalost obecných programových postupů

oblast C - znalost programování

Jistě, že lze jednotlivé pracovní náplně a postup profesí různě charakterizovat i v rámci "nahozené" tendence. Na tomto místě však jde o vysvětlení základního názoru.

Není možné ani třeba připomínat, že čím více daná profese pronikne do ostatních oblastí, je to ku prospěchu věci.

Problém "organizace práce", event. problematiku "časovosti" nepovažují nyní za účelné v této části řešit.

II. Bývá zvykem, že výrobci výpočetní techniky jsou schopni uživateli poskytnout nejen více či méně vyspělý software (z programovacího hlediska), ale v určitých případech i zpracovaný a odleděný určitý praktický subsystém (př.: evi-

dence MTZ, sledování nákladů, technické normování atd.). Skutečností však zůstává, že tyto "hotové" subsystémy v našich poměrech nenalézají uplatnění. Důvodů je mnoho, jde však především o to, že v našich podnicích panují příliš individuální přístupy k řešení jednotlivých úloh a není vůbec zvláštností, že jedna problematika (př.: evidence MTZ) se řeší jinak v jednotlivých podnicích téhož resortu. Druhou závažnou věcí pak je naprostá nechuť (nebo neschopnost?) přizpůsobit své vlastní podmínky požadavkům jiným, byť již systémově a programově vyřešeným. A tak raději si každý podnik znovu a znovu vypracovává všechno sám od začátku až do konce. Můžeme tedy klidně prohlásit, že již zpracováváme třeba mzdy do netta na padesáti počítačích, ale na každém z nich jinak. A tak můžeme pokračovat dále.

Jsem pesimista v tom, že by se situace rychle změnila. Způsob našeho přístupu k řízení to zatím vylučuje. Domnívám se však, že negativní důsledky této "neschopnosti se přizpůsobit" lze zmírnit tzv. "programovými panaly". Tímto pojmem charakterizují následující:

Většina zpracovávaných systémů (subsystémů), především ekonomického charakteru, obsahuje mnoho programů skládajících se ze samostatných dílčích, logických částí (joby, programové balíky atd.), dohromady tvoří jeden logický uzavřený celek. Z jiného pohledu můžeme tentýž systém charakterizovat jako uspořádání programů majících v daném systému zhruba podobný účel. Můžeme charakterizovat systém jako uspořádanou množinu programů:

- nahrávacích
- třídících
- kopírovacích, slučovacích
- aktualizacních
- tiskových
- pracovních
- aj.

A zde již jeme u pojmu "panel". Z mého pohledu je tímto "panelem" již třídění. Třídění je vždy tak softwarově vyřešeno, že je pro běžnou praxi dostačující a co hlavně, nemusí se z hlediska programů specifikovat tak, jako programy ostatní.

Příklad: dejme tomu analytik zadá velmi krátkou informaci, jak si představuje třídít a uvedení popisů souborů a rovněž z programového hlediska se vyplní a vyděruje několik děrných štítků. Zbytek obstará "obecný program".

Táži se. Jsem schopen vytvořit obecný program i v ostatních případech? (Mám na mysli hlavně tisk, aktualizaci, nahrávání). Osobně se domnívám, že ano. K této odpovědi mne vede dlouhodobé zkušenost právě v systémech hromadného zpracování dat. Procento použití těchto panelů by nebylo tak jednoznačné, jako u třídění, ale vyplatilo by se.

Příklad: dejme tomu u tiskových sestav (nikoliv komplikovaných, ale takových, kterých je většina) by analytik vyplnil třeba jen jeden formulář a rozvrh věty. Programátor pak parametruje DČ. Zbytek by zařídil již programový panel pro tisk. Podobně u aktualizčních programů atd.

Není předmětem tohoto příspěvku provést zevrubný popis návrhů na obsah třeba tiskového programu, ale upozornit na výhody a možnost použití programových panelů obecně.

Smyslem tedy je dosáhnout toho, aby již při tvorbě systémových schémat se mohly uplatňovat možnosti vytvořených programových panelů. Pochopitelně, že schéma nebude sestávat jen z obecných programů. Tvorbu systémového schématu lze pak vzdáleně srovnat s postupem práce s dětskou stavebnicí. Výsledek je různý, ale použitý "materiál" je stejný. (Materiálem však již nejsou cihly - instrukce).

Proč tento způsob volby? Jde o kompromis mezi přebíráním celých systémů do používání a mezi obecnými metodami programování. Obojí musí být pochopitelně zachováno, ale:

- neexistuje podnik, který by převzal celý systém, aniž by potřeboval něco změnit
- proč stále programovat stejně různými programátory určitou oblast (třeba aktualizaci), když může být inteligentně zobecněna.

Využití "programových panelů" vidím především ve velkých výpočetních střediscích, kde neustále dochází k tvorbě nových subsystémů, změně stávajících subsystémů a různým jednorázovým požadavkům. Podobně, má-li více podniků stejnou VT, uplatní se tento způsob i zde.

Není třeba zdůrazňovat, že event. vytvoření těchto panelů by znamenalo i jiný přístup v metodice systémové práce, týkající se především architektury systému, nehledě na úsporu práce v programování.