

Ing. Jaroslav HORYNA

TESLA - ZAR, Praha

Eliška HORYNOVÁ, prom. mat.

VÚŽ Praha

## PARAMETRICKY ŘÍZENÉ TYPOVÉ PROGRAMOVÉ PRVKY

### 1. Úvod.

Pracnost a vysoké náklady na řešení ASŘ jsou nespornými argumenty pro racionalizaci tvorby ASŘ. Jednou z nejefektivnějších cest racionalizace je budování ASŘ na bázi typových prvků.

Nejvýznamnější efektem této cesty vytváření ASŘ je:

- podstatné zkrácení vývojového cyklu projektování ASŘ
- úspora pracovních sil
- sjednocení struktury informačního a rozhodovacího systému pro dané třídy uživatelů.

### 2. Definice typového prvku a jeho vlastností.

Typovým prvkem rozumíme funkčně a logicky uzavřenou část ASŘ, kterou je možno vícenásobně využít.

Rozeznáváme dvě třídy typových prvků:

- typový prvek funkční, který vzniká v etapě technického projektu a řeší části ASŘ
- typový prvek programový, který vzniká v etapě prováděcího projektu a je stavebním kamenem pro počítačovou realizaci funkčních typových prvků.

Svými vlastnostmi musí typový programový prvek odpovídat obecné definici typového prvku a musí

- umožňovat vícenásobné využívání typových prvků funkčních

na základě jejich přizpůsobitelnosti na specifické podmínky uživatele

- umožňovat řešení specifických úloh uživatele s využitím typových programových prvků.

Pohled na typový prvek programového řešení jako na nástroj pro tvorbu ASŘ podle uživatelských specifík je určující pro rozvoj systému typových programových prvků a zpracování metodiky jejich tvorby.

Typové prvky, které plní tuto funkci, mohou být totiž použity jak pro typové řešení ASŘ tak i pro specifická řešení. Tím se zvyšuje jejich efektivnost a rozšiřuje oblast jejich využití.

Při konstrukci typových programových prvků je třeba respektovat řadu vlastností, které musí typový prvek mít. Za charakteristické vlastnosti typových programových prvků považujeme zejména:

- opakovatelné využívání
- přizpůsobitelnost
- funkční pružnost
- modularitu
- přehlednost
- spolehlivost
- efektivnost
- jednotnou metodiku tvorby
- další technické atributy

(nezávislost na zařízení, volitelné médium a organizace souborů, volitelná struktura vět, volitelná délka údajů a vět, volitelné měřítko aritmetických hodnot, relativní nezávislost na modelu počítače a konkrétním operačním systému, možnost kombinace s programovými úseky v různých programovacích jazycích)

Jednotlivé vlastnosti typových prvků působí společně a navzájem se ovlivňují.

### 3. Dělení typových programových prvků.

Typové programové prvky v dosavadním řešení dělíme na:

- 1) Typové programové prvky technologické, které tvoří nástroj pro tvorbu prvků na vyšší úrovni. V /1/ jsou tyto typové prvky zahrnuty pod pojmem mikromoduly. V naší pracovní terminologii jsou nazývány normalizovanými úseky.
- 2) Typové programové prvky generovatelné, které se generují podle zadaných parametrů (parametry makroinstrukcí, tabulky popisu souborů, kontrolní tabulky a pod.). Typové programy umožňují vkládání uživatelských úseků ve specifikovaných vstupních bodech. Přizpůsobení umožněné generováním úseků ve fázi kompilace programu je doplněno dynamickou adaptabilitou při provádění programu podle zadaných parametrů.

Pozn.

Generování tu rozumíme především rozvoj makroinstrukcí podle zadaných parametrů popř. úpravu zdrojového textu programu při kompilaci podle podmínkových instrukcí.

- 3) Parametrické programy, které již nepoužívají generovatelných úseků pro přizpůsobení ve fázi kompilace. O generování tu lze hovořit jen ve smyslu dynamické interpretace zadaných parametrů do různých tabulek popř. sekvencí instrukcí, kterou však provádí již zkompileovaný a z knihovny absolutních modulů volaný a spuštěný program.

S hlediska jejich využití můžeme typové prvky rozdělit ještě na:

- typové prvky speciální - pro řešení typových funkčních prvků technického projektu ASŘ a
- typové prvky obecné (univerzální) - pro práci se soubory. Tyto typové prvky tvoří jakousi nadstavbu operačního systému a umožňují pořizování, aktualizaci a zpracování souborů včetně jejich výstupu na tiskárnu nebo magnetické médium.

#### 4. Současný stav řešení typových programových prvků.

V současné době jsou k dispozici a řadou uživatelů využívány typové programové prvky vytvořené na pracovišti TESLA AVOS pod operačním systémem DOS/JSEP, pomocí nichž je zajišťována konverze, transformace, aktualizace souborů a zpracování souborů pro výstup na tiskárnu nebo magnetické médium. Část programů

se generuje, část je řízena parametricky. K nejvyužívanějším programům patří:

- K102 - parametricky řízený program na konverzi černých štítků 80/90 sloupcových
- U102 - generovatelný program pro transformace souborů
- Z102 - generovatelný program pro aktualizaci souborů
- T105 - parametricky řízený program pro výstup na tiskárnu nebo magnetická média
- D102 - parametricky řízený program pro doplňování základního souboru až se dvou doplňujícími soubory

Vedle těchto problémově orientovaných typových programů byly zpracovány typové prvky technologické, které jsou ve všech programech využívány. Mezi nimi zaujímá zvláštní místo generační makrodefinice WPSV, která je součástí všech generovatelných programů a vytváří v programech potřebné tabulky, deklarace vět, souborů a pracovních polí podle uživatelských tabulek popisu souboru - TPS. TPS definují každý zpracovávaný soubor a jsou uloženy jako uživatelské makrodefinice ve zdrojové knihovně.

##### 5. Zkušenosti s využíváním typových programových prvků.

Z uvedených generovatelných a parametrických programů lze vytvořit řešení některých úloh s vysokou mírou adaptability podle specifických podmínek, přičemž efektivnost tvorby takových řešení je vysoká ve srovnání s tradičním postupem (t.j. při vytváření specifických programů).

Ověřování bylo provedeno v řadě aplikací, ve kterých došlo ke zkrácení tvorby z člověkoměsíců na dny, maximálně týdny práce. Řešení vykazuje navíc vysokou dynamickou adepthabilitu a je schopno pružně reagovat na změny podmínek i požadavků. To jsou patrně důvody, proč tato obecná a univerzální část ISRT našla daleko větší uplatnění u uživatelů než specificky řešené subsystémy ASŘP. Aplikace byly provedeny v nejrůznějších oblastech a téměř na všech typech počítačů JSEP. Ve spolupráci s řešiteli uplatnili některé prvky i pracovníci KSNP pro subsystémy MARS a v rámci dále

rozvíjené spolupráce se počítá a využitím inovovaného řešení i pro systém VARS pro počítač EC 1025.

Sami řešitelé ověřovali popsané typové programové prvky v některých vybraných aplikacích jako na př.

- - zpracování statistiky a evidence vybraných ukazatelů při MS v ledním hokeji 1977
- řešení vybraných úloh subsystému personalistika
- evidence zadaných a plněných úkolů ředitele organizace včetně urgenčních dopisů na nositele úkolu při neplnění
- úplné zpracování subsystému účetnictví pro ČSBD a vedení účetnictví pro řadu bytových družstev
- evidence nebydlících členů SBD a výpočtem pořadníků
- evidence bydlících členů, nájemného a předpisů nájemného
- zpracování výkazů pro GŘ TESLA v rámci úloh pro ASŘSČ
- dokumentace programových chodů u úloh řešených typovými programovými prvky
- řešení specifických problémů některých organizací.

Ze všech těchto aplikací byly získány cenné zkušenosti a náměty pro okamžitou i plánovanou inovaci systému typových programových prvků.

Získané zkušenosti současně umožňují provést toto stručné hodnocení systému:

#### Klady:

- systém umožňuje vysokou produktivitu programátorů při řešení úloh. Prakticky bylo možné většinu zadaných úloh s dobře provedenou analýzou uživatele vyřešit ve velmi krátké době.
- systém umožňuje, aby při znalosti jeho funkcí byly některé úlohy formulovány již v technickém projektu jako programové chody skládající se z jednotlivých typových a systémových (SORT) programů.
- systém umožňuje již v dnešní verzi řešit řadu úloh ASŘ pomocí obecných generovatelných a univerzálních parametrických programů.
- systém umožňuje logické výběry informací z uložených souborů podle okamžitých požadavků řídicích pracovníků,

vykazuje vysokou míru přizpůsobitelnosti a může velmi pružně reagovat na změny podmínek a požadavků.

#### Nedostatky:

- systém umí zpracovat jen soubory s větami pevné délky a u indexsekvencčních souborů v některých programech jen neblokované soubory.
- některé komponenty vykazují nižší efektivnost provádění zvláště na pomalých strojích.
- z generovatelných programů U102 a Z102 je třeba vytvořit pro jednotlivé soubory specifické programy. I když jejich vytváření probíhá zcela automaticky, při množství zpracovávaných souborů je zapotřebí značná kapacita v knihovně absolutních programů.
- programy a moduly nejsou většinou samopřemístitelné a neumožňují efektivní multiprogramový režim zpracování.
- program T105 lze využívat jen v prvních 128 KB paměti a v současné verzi nemůže být zlinkován do oblastí nad 128 KB.
- kontrola a zápis některých parametrů v programu T105 jsou málo efektivní.
- operační systém DOS je překážkou dalšího rozšíření systému na větších počítačích JSEP

#### 6. Zásady pro inovaci.

Na základě zkušeností získaných z ověřovacích výpočtů a ze spolupráce s uživateli systému byly koncipovány záměry řešitelů, které směřují k parametrickým programům pro typové funkce ASŘ. Složitost parametrických programů závisí na specifikaci požadovaných funkcí. Kromě jednoduchých zadání, která mohou realizovat i přímí uživatelé systému, budou existovat i relativně složitá, sestavovaná specialisty. To ku př. znamená, že složitě parametrické zadání bude často sestavovat i programátor. Rozhodující je však vysoká efektivnost parametrického řešení.

Pro usnadnění parametrizace mohou být vytvořeny a využívány různé prostředky jako ku př.

- symbolická manipulace s údaji i pro systémové programy (SORT)
- katalogizované parametry
- katalogizované struktury
- katalogizované procedury
- sborníky příkladů
- generátor parametrů zadávaných přes slovník
- implicitní hodnoty parametrů
- dynamické interpretace rozhodovacích tabulek
- standardizované funkce (ku př. pro aktualizací program, formální kontroly a pod.)
- komplexní analýza parametrů s kontrolou formálních i syntaktických chyb
- atd.

Východiskem pro další inovační záměry byla analýza do-  
savadního systému obecných (generovatelných) a univerzálních  
(parametricky řízených) programů a typových programových prvků  
pro řešení úloh ASŘ, studium různých funkcí a algoritmů  
těchto úloh, studium složitých závislostí mezi soubory a  
možností jejich parametrizace, studium I/O systémů v různých  
operačních systémech a možností jejich dynamických úprav  
a pod. Nakonec byly vypracovány technické požadavky na  
systém parametricky řízených typových programových prvků  
(parametrických programů) ve formě hlavních zásad řešení  
a specifikace zadání pro tvorbu systému.

Za hlavní zásady pro tvorbu systému parametrických pro-  
gramů považujeme:

- 1) Programy musí být konstruovány tak, aby mohly pracovat  
pod různými operačními systémy (MOS, DOS, OS, DOS/VS,  
OS/VS).
- 2) Typové prvky musí být konstruovány tak, aby byly rela-  
tivně nezávislé a mohly být využívány i samočtatně, event.  
jako nadstavba operačního systému.
- 3) Typové parametrické prvky pro práci se soubory by měly  
mít interface na různé programové systémy pro organizaci  
a ovládání souborů (databankové systémy).
- 4) Typové prvky musí být otevřené pro inovace v oblasti HW  
(velkokapacitní paměti s přímým přístupem, vstupy

z různých zařízení pro prvotní záznam dat a pod.).

- 5) Pro usnadnění práce s parametrickým systémem budou vytvořeny problémově orientované subsystémy typových prvků pro řešení typových funkcí ASŘ, jako na př.:
- pořízení souborů
  - aktualizace souborů
  - zpracování a výstup souborů
  - specifické subsystémy ASŘ.

Pozn.

Pro specifické subsystémy ASŘ jako na př. TPV a zpracování kusovníků, plánování výroby rozpadem kusovníků, kalkulace a pod. se osvědčilo generování programů s využitím makroinstrukce WPSV, systémového pole YA05xMEZ, makroinstrukce WPU a dalších prvků v dosavadním řešení ISŘT, které umožňují přizpůsobení uživatelským strukturám dat a dalším uživatelským volbám.

- 6) Při tvorbě systému je třeba sledovat původní linii jeho využití pro potřeby řízení a zejména je třeba:
- zajistit vysokou míru pružnosti, možnost libovolných výběrů a zpracování informací z báze dat, rychlou adaptaci
  - sledovat a postupně (s dostupnými prostředky) realizovat i možnost dialogového režimu zpracování
  - sledovat databankový koncept.

Parametrické výběry a parametrické zpracování se blíží dotazovacím jazykům a v některých směrech překračují jejich současnou úroveň.

## 7. Modulární struktura řešení.

Systém parametrických programů je konstruován na bázi relativně nezávislých prvků - modulů, t. j. takových, u kterých se nepředpokládá implicitní znalost systému tabulek a adresářů systému, nýbrž jen znalost struktury konkrétních tabulek potřebných pro funkci modulu. Relativně nezávislé moduly mohou pak být použity v libovolné kombinaci i mimo systém.



V současné době jsou rozpracovány jako relativně nezávislé prvky systému tyto moduly:

(Pozn. symbolické názvy modulů jsou mnemotechnické, přičemž první 4 místa jsou vyhrazena pro označení podle konvence stanovené v interních pracovních normách.)

YxxxREZ - rezidentní řídicí modul  
YxxxPAR - modul pro čtení, analýzu a interpretaci parametrů  
YxxxLOSI - modulu pro logické ovládání souborů na vstupu  
YxxxLOSO - modul pro logické ovládání souborů na výstupu  
YxxxFOS - modul pro fyzické ovládání souborů  
YxxxTRF - transformační modul  
YxxxPRT - tiskový modul  
YxxxARIT - modul pro aritmetické operace  
YxxxKONV - modul pro konverzi prvotních medií (DŠ)  
YxxxAKTZ - modul pro aktualizaci souborů  
YxxxKONT - modul kontrolní  
YxxxPROT - modul pro zpracování protokolu o průběhu výpočtu  
YxxxSTAT - modul pro zpracování statistiky o průběhu výpočtu

Vedle těchto modulů se průběžně zpracovává aparát technologických typových programových prvků - normalizovaných úseků - v návaznosti na dosud vytvořené úseky.

V dalším textu uvádíme stručnou specifikaci funkcí pro jednotlivé moduly.

#### YxxxREZ - rezidentní řídicí modul.

Tento modul se vyskytuje ve více variantách a řídí zpracování problémově orientovaných subsystémů.

Řada funkcí se však stereotypně opakuje:

- zahájení zpracování
- aktivace modulu PAR pro čtení, analýzu a interpretaci parametrů
- rozdělení a kontrola paměti pro subsystém a rezidentní adresáře, tabulky a vygenerované sekvence instrukcí.
- překrytí modulu PAR výkonnými moduly subsystému
- volání uživatelských exitů, t.j. uživatelských úseků v absolutní formě
- zakončení zpracování.

### YxxxPAR - modul pro čtení, analýzu a interpretaci parametrů.

Modul čte a analyzuje parametry, zjišťuje formální i syntaktické chyby a interpretuje bezchybné parametry do soustavy tabulek, adresářů, popř. do sekvencí instrukcí.

Proti dosavadnímu stavu zpracování parametrů jednotlivými programy tu dochází ke sjednocení metodiky zadávání parametrů, k uvolnění formy parametrických štítků, ke komplexnější kontrole zadání a k přesnější lokalizaci chyb. Parametry katalogizované v knihovnách je možno aktualizovat a navíc je možná dynamická aktualizace při volání parametrů z knihovny při spuštění programu. Zjednodušuje se forma výpisu parametrů a identifikace zjištěných chyb.

### YxxxLOSI - modul pro logické ovládání souborů na vstupu.

Úkolem modulu je řadit až 10 vstupních souborů podle parametricky popsaných logických závislostí mezi nimi. Modul zpracovává věty pevné i proměnné délky a všechny organizace souborů.

Soubory mohou být řazeny těmito dvěma základními způsoby:

- 1) Paralelní řazení - funkce zatřídování: k dispozici je dána vždy věta s minimálním argumentem (klíčem).
- 2) Seriové řazení - funkce doplňování: k dispozici jsou dávány větve vět logicky svázaných pomocí klíče nebo adresního směrníku.

Obecně mohou být zatřídováním řazeny soubory, z nichž každý má jinou strukturu věty avšak se srovnatelným klíčem. Funkce zatřídování může předcházet funkci doplňování v případě sjednocení struktury vět. Při doplňování lze pracovat i se soubory s vícenásobným výskytem vět pro daný klíč. Věty těchto souborů mohou být řazeny sekvenčně nebo přes adresní směrník (uvnitř zřetěžené soubory). Vícenásobný výskyt vět u podřízených souborů se smí objevit jen v jedné větvi logicky svázaných souborů. Při doplňování je vždy jeden soubor řídicí, ostatní podřízené. Řídicí soubor se zpra-

covává sekvenčně, podřízené soubory podle definovaných závislostí a organizace. Výstupem z modulu jsou t.zv. aktivní věty, t.j. věty vyhovující zadaným podmínkám řazení a souhrnná informace o těchto větách pro využití v logických testech.

#### YxxxLQSO - modul pro logické ovládní souborů na výstupu.

Úkolem modulu je ředit soubory na výstupu a zapisovat do nich věty ze vstupních oblastí, pracovních oblastí nebo VN - normálních a VS - součtových vět (viz program TL05) vytvářených v modulu YxxxTRF podle parametricky zadaného pořadí. Modul provádí též operace zjišťující a z pracovávající informace poskytované I/O systémem operačního systému (ku př. adresy)

#### YxxxTRF - transformační modul.

Modul provádí transformace mezi vstupními, pracovními a výstupními poli.

Tyto transformace zahrnují:

- přesuny mezi pracovními poli

a) celých vět

b) segmentů a skupinových položek

c) jednotlivých údajů

Přesuny lze uskutečnit jako

a) prosté na úrovni celých vět, segmentů nebo skupinových položek (instr. MVC)

b) kontrolované, t.j. s přihlédnutím k TPS jednotlivých souborů a jako

- úplné s mezerováním výstupní věty

- úplné bez mezerování výstupní věty avšak s přepisem údajů ve výstupní větě

- neúplné - obsazené údaje ve výstupní větě se nemění.

Při těchto přesunech může dojít

- k restrukturalizaci věty

- ke změně formátů, měřítek, délek údajů

- k aktivaci nebo zrušení kontrol předepsaných v TPS

- k provádění parametricky zadaných kontrol

- provádění zadaných výpočtů

a) horizontálních (na aktivních větách v daném okamžiku)

b) vertikálních (na postupně čtených větách souborů) obojí s eventuelním využitím výsledků předchozích aritmetických operací.

- výběry vět vstupních souborů podle logických výrazů
- operace modulu mohou být většinou podmíněny logickými výrazy popř. testem hodnoty systémových přepínačů.
- operace modulu jsou řazeny lineárně podle pořadí zadání s možností skoků podle splněných či nesplněných podmínek.

#### YxxxFOS - modul pro fyzické ovládání souborů.

Modul zajišťuje ve spolupráci s moduly YxxxLOSI a YxxxLOSO interface na I/O systém operačních systémů. Upravuje dynamicky typové tabulky popisu souborů podle parametrického zadání (v DOS 1 a 2), organizuje jejich uložení v paměti a realizuje požadavky na čtení nebo zápis vět. Zjišťuje rovněž a dává k dispozici informace o EOF podmínkách, stavové byty, event. adresy.

#### YxxxPRT - tiskový modul.

Modul zajišťuje výstup na tiskárnu. Jeho funkce odpovídají dosevadní T105 v inovované verzi (přímý výstup na tiskárnu, výstup na MP nebo SD)

#### YxxxARIT - modul pro aritmetické operace.

Modul zpracovává aritmetické výrazy. Vedle základních aritmetických operací sčítání, odčítání, násobení, dělení s výsledky zaokrouhlenými nebo nezaokrouhlenými s přesností podle zadaného měřítka provádí modul i jednoduché operace na skupinových položkách, indexovaných skupinových položkách, elementárních položkách ze skupiny nebo elementárních indexovaných položkách.

#### YxxxKONV - modul pro konverzi prvotních medií.

Modul provádí konverzi dat z DŠ a odpovídá svými funkcemi současnému programu K102 s některými rozšířeními.

#### YxxxAKTZ - modul pro aktualizaci souborů.

Modul zajišťuje svými implicitními funkcemi aktualizaci

uspořádaných souborů podle těchto kódů:

- 0 - vložit větu nebo přepsat již existující větu v základním souboru
- 1 - změnit údaje ve větě
- 2 - vypustit větu ze souboru
- 3 - vypustit věty ze souboru podle generického klíče
- 4 - aditivní změna aritmetických hodnot (přičtení, odečtení)
- 5 - start intervalu pro intervalovou změnu. Měněné údaje jsou uváděny jako u změny 1. Konec měněného intervalu je zadán ve větě s kódem 9.
- 6 - start intervalu pro intervalové vypouštění vět ze souboru
- 7 - indexová změna (mění se údaj určený indexem - pořadovým číslem údaje v TPS. Pro index musí být ve větě s TPS změnového souboru vyhraženo pole)
- 8 - neobsazeno
- 9 - konec intervalu pro intervalové transakce

#### YxxxKONT - modul kontrolní.

Modul zajišťuje kontroly popsané v TPS. Provádí vyhodnocení předepsaných reakcí na zjištěné chyby u jednotlivých údajů a určuje výslednou reakci pro celou větu

#### YxxxPROT - modul pro zpracování protokolu o průběhu výpočtu.

Modul zpracovává protokol o průběhu výpočtu. Obsahuje chybové zprávy, opis chybných vět, opis parametrů a pod. Jeho funkce je analogická s dnešním modulem A06A.

#### YxxxSTAT - modul pro zpracování statistiky o průběhu výpočtu.

Modul zpracovává ve spolupráci s výkonnými moduly na závěr výpočtu statistické údaje jako ná př.

- počet čtených vět ze vstupních souborů
- počet vět vybraných ke zpracování
- počet chyb zjištěných kontrolami údajů
- počet vět zapsaných do výstupních souborů

Modul obsahuje i možnost archivace statistických dat ve specifickém souboru.

## 8. Závěr.

Úplná soustava modulů pro parametrické programy umožní podle našich názorů pokrýt programové řešení velké části úloh ASŘ. Nejvýraznější vlastností parametrického systému je možnost rychlého přizpůsobení na změnu podmínek a změnu požadavků na výstup z informačního systému a rychlého řešení nových úloh. V současné době tyto změny a dodatečné požadavky vyžadovaly vesměs zpracování nových programů. To znamenalo převážně odklad realizace o několik dnů či spíše delší časové období. Využitím parametrických programů se tento odklad snižuje na minimum, většinou lze na změny reagovat v několika hodinách, často okamžitě ("na počkání"). V tom je smysl a efektivnost celého parametrického systému.

## Literatura:

1. Lukášek Jiří Ing. : Příspěvek k metodice typizace v automatizovaných systémech řízení (kandidátská dizertační práce - pracovní verze).
2. ISŘT - normalizované úseky, 1976
3. ISŘT - univerzální a pomocné programy (D102 - 1976, T105 - 1978)
4. ISŘT - SP 1 80/II-III, 1976
5. CULPRIT USERS GUIDE, Release 4.3b, III/78
6. Manuály operačních systémů.