

# AUTOMATIZOVANÉ GENEROVÁNÍ UŽIVATELSKÝCH PROGRAMŮ PRO NAHRÁVÁNÍ, ÚDRŽBU A EDITACI DAT INTERAKTIVNÍM ZPŮSOBEM ZA POUŽITÍ DATABÁZE

Ing. František Raatil

## 1. Úvod

K tvorbě systémů automatizovaného programování (SAP) lze přistoupit po splnění některých nezbytných předpokladů :

- a) musí být dostatečně přesně známa oblast, kde chceme SAP implementovat. Na obecné úrovni je příliš složitý, má vysoké nároky na paměť a lze jej velmi obtížně modifikovat
- b) SAP bude efektivní zejména na takových technických prostředcích, které umožňují interaktivní způsob práce a multiuživatelský přístup
- c) musí existovat výkonný, v ovládnutí jednoduchý operační systém
- d) v operačním systému musí existovat - a pokud ne, tak je nutno jej vytvořit - jednotný typ ukládání dat - databáze a systém řízení databáze
- e) musí vzniknout skupina odborníků "ad hoc" k řešení SAP. Jednotlivci ze skupiny si musí být jisti svou odbornou erudicí a musí být dostatečně důslední při realizaci záměrů a nimiž se stotožnili.

## 2. Oblast v níž byl SAP implementován

V našem konkrétním případě šlo i implementaci SAP v oblasti zpracování hromadných dat s následnou tvorbou informačního systému pro řízení centrálního orgánu infrastrukturního charakteru. Po vytvoření SAP se však ukázalo, že je možno jej použít i v jiných oblastech nasazení. Obecně lze říci, že vytvořený SAP lze nasadit v oblastech kde -

- vznikají data, která musí být interaktivně verifikována na vstupu jak v hodnotách jednotlivých údajů, tak i v logických

vazbách a ukládána do jednotné databáze, která je přístupná pro všechny uživatele,

- z takto uložených dat je nutno provádět generování výstupů v co nejširším propojení všech uložených dat.

### 3. Oblast technických prostředků

SAP s názvem Quick je orientován pro počítače SMEP (SM 3/20, SM 4/20, SM 50/50, SM 52/11). Technický systém umožňuje splnění požadavků v l.b.

Terminálová síť může být lokální nebo vzdálená s možností pevných spojů, nebo po komutovaných spojích. Existuje i možnost propojení počítačů v jednoduché hvězdicové síti.

### 4. Operační systém

SAP Quick byl vypracován pod operačním systémem DIAMS-1. Prostředky pro obhospodařování terminálových sítí byly na našem pracovišti realizovány v rámci státního úkolu P-04.

DIAMS-1 je time-sharingový operační systém. Vnitřní paměť je pevně rozdělena do uživatelských úseků (délka úseku paměti pro uživatele je v našich podmínkách 2K slov). Pod OS existuje pouze jeden interpretační jazyk MUMPS, ovládání OS je jednoduché a provoz s více uživateli dává v době odezvy uspokojivé výsledky.

Velikost uživatelského úseku vnitřní paměti se v počátku řešení jevila jako značné omezení pro tvorbu SAP. donutila však řešitele k vysoké úspornosti při realizaci.

### 5. Databáze a řízení databáze

Databáze v OS DIAMS je konstruována jako kontinuální diskový prostor s normou ukládání dat ve stromové struktuře. Řízení databáze zajišťuje optimalizační metodou fyzické ukládání dat. Odvolání na uložená data se děje pomocí logické databáze. Logická databáze není popisována jako v jiných databankových systémech (např. schema, subschema atd.), je určena pouze vyjmenováním hodnot

klíčových slov, jež udávají logickou strukturu dat. Zajímavým, a řekl bych moderním rysem, je skutečnost, že systém řízení báze dat nevyžaduje specifikaci rozsahu souborů. Podstatným rysem řízení databáze je schopnost využívat ukládání dat v operační paměti i na diskovém médiu ve formě "řádkových polí". Je známo, že údaj je uložen v takové délce jakou fyzicky představuje. Tímto principem a optimalizační metodou ukládání dat se výsoce efektivně využívá diskový prostor databáze.

## 6. Stanovené cíle a řešitelská skupina

Celý SAP Quick byl vypracován skupinou čtyř programátorů za období cca 5 měsíců a nebyl hlavní náplní úkolů těchto pracovníků. Při řešení se ukázala nutnost silné centralizace řízení projektu a neústupnost řídicího pracovníka při realizaci cílů. Řešení probíhalo ve dvou oddělených etapách. Nejprve se definovaly cíle bez ohledu na realizační možnosti. V druhé etapě došlo k vlastním řešení a inspirovaná zdrojem byla konfrontace stanovených cílů a realizačních možností.

Cílový stav byl stanoven v těchto bodech -

- celý systém musí být interaktivní a pokud možno jednoduchý, vše co se dá automatizovat, to provést automatizovaně,
- nevytvářet jeden obecný aparát, použít možnosti generovat výsledný uživatelský program podle požadovaných potřeb,
- výsledný vygenerovaný uživatelský program musí být modifikovatelný, aby programátor mohl použít obraty, které systém neumožňuje,
- výsledný generovaný uživatelský program musí být rychlý ve svém provádění. Musí být, včetně všech uživatelských dat, schopen provozování v uživatelském úseku paměti o délce 2K slov (4kb),
- výsledný vygenerovaný uživatelský systém musí být interaktivní. Dialog uživatele se povede vždy přes formátovanou obrazovku,
- práce se SAP nesmí omezovat osetří uživatele systému,
- při generování výstupních sestav použít jeden univerzální parametrizovatelný program, který bude využívat parametry definované v uživatelské souboru popisu výstupu,
- veškeré informace sloužící SAP a vytvořené před generováním uživatelského programu musí být měnitelné,

- vytvořit mechanismus ochrany dat před zneužitím,
- vytvořit samodokumentující mechanismus o všech datech ukládaných SAPem a tím formalizovat výstupní dokumentaci,
- provádět co nejširší verifikace vstupních hodnot údajů a logických vazeb,
- systém nesmí být závislý na změnách technického vybavení.

Po definování cílů se přistoupilo k realizaci. Většinu cílů se podařilo realizovat. Systém se rutinně využívá od října roku 1982 a projevil se plně životaschopný. Nahtím nových zkušeností a provozem SAP chceme vytvořit novou verzi, která se neliší ve stanovených cílech, ale v realizačních možnostech systému.

Z používání SAP vyplývají závěry -

- úspora pracnosti opakovatelných řešení při manipulaci s daty (nahrávání a údržba dat) činí 30-70%,
- čím je rozsáhlejší uživatelský dialog při nahrávání a údržbě dat, tím větší je úspora programátorských kapacit,
- podobná úměra platí i pro vztah : čím je logická struktura dat složitější, tím je dosažená úspora programátorských kapacit větší.

## 7. Práce se systémem Quick

Práce probíhá v několika oddělených částech

- a) první prací je vytvoření popisu atributů položek. Zde se definují rozsahy, typy dat, mess, atd., tedy atributy, které se budou ve vygenerovaném programu kontrolovat,
- b) definuje se logická databáze, to je klíčová slova a položky příslušející ke klíčovému slovu,
- c) deformace formátů obrazovky. Zde si programátor vykreslí rozmístění řídicího textu dotazů a dat na obrazovce - popřípadě více obrazovkách - pro uživatelský dialog.
- d) pokud je potřeba, definují se sekundární soubory. To v případě, že ve výstupech je požadováno jiné seřídění dat než je uvedeno v definici struktury primárního souboru,

SAP potom sekundární soubory vytváří automatizovaně.

- e) provede se generace uživatelského programu a jeho uložení
- f) generování výstupních sestav se děje podobným způsobem.
- g) podle potřeby je možno vytisknout dokumentaci
- h) modifikace vygenerovaného programu se provádí vkládáním potřebných instrukcí do vysezených částí programu.

Celý systém je interaktivní, programátor je veden kladenými dotazy, jeho odpovědi jsou kontrolovány a přijaty, pokud jsou formálně a logicky správné.

Funkce vygenerovaného uživatelského programu jsou

- vypíše se první stránka navržené obrazovky
- kurzor se nastaví na první dotaz
- po vložení údaje se provede kontrola podle zvolených atributů v popisu položek
- po vložení všech klíčových slov se provede logická kontrola existence
- po vyčerpání dotazů má uživatel volbu
  - a) zápisu vložených dat (Insert)
  - b) změnu údajů (Replace)
  - c) zrušení údajů (Delete)

## 8. Z á v ě r

Domníváme se, že kladen vytvořeného SAP bylo sladění definovaných cílů s realizačními možnostmi systému SMEP. Nechtěli jsme v příspěvku popisovat systém Quick, pouze nastínit cíle a ukázat, že je možno SAP realizovat.

Domníváme se, že právě minipočítače s terminálovými sítěmi umožňují poskytovat řídícím pracovníkům informace v požadované časové odezvě a v místě potřeby. Tím vzniká však větší tlak na řešitele ASŘ co do množství programů, které musí vyprodukovat. Proto jsme se pro usnadnění naší práce pokusili vytvořit v konkrétních podmínkách systém automatizovaného programování. S novou kvalitou minipočítačů je nutno změnit ale i přístupy k projektování a programování ASŘ.

Podle našich zkušeností je možno říci, že pro tvorbu SAP je nutnou podmínkou systém řízení báze dat a svou roli sehrávají i interpretační jazyky. Ladění v interpretačních jazycích je podle složitosti algoritmu až o 40% produktivnější. A to není zanedbatelný fakt.

Domníváme se, že i ostatní účastníci se pokoušejí, nebo budou pokoušet nějaký SAP vytvořit. Uvítáme kontakt s nimi, rádi jim sdělíme své zkušenosti a použijeme se s jejich znalostí při tvorbě vyšší verze našeho systému Quick.



TISK ATRIBUTU POLOZEK POPSANÝCH V SLOŽALU ZAI

I I NAZEV POLOZKY I TYP I ATR. I TYP I MEZE I VYSKYTOUY  
 I POL. I VOLB. I DEL. I DO I REZEZEC

1	PORADOVE CISLO ZAPISU	3	P	U	7	
2	ODESILACI URAD	5	P	F	3	
3	ADRESNI URAD	5	P	F	3	
4	ODLET DNE	1	P	F	4	0101 3112
5	ZEME	1	P	F	3	
6	CISLO ZAUVERU	1	P	U	4	
7	LETOVA LINKA	6	P	F	5	
8	URAD PUVODU	5	P	F	3	
9	URAD URZENI	5	P	F	3	
10	HMOTNOST LC/AD V KG	3	N	U	12	
11	HMOTNOST CP V KG	3	N	U	12	
12	TYP ZAUVERU	1	P	F	1	1 2
13	DRUH ZAUVERU	1	P	F	1	1 3

CELKEM POPSANO 13 POLOZEK I

NAPOVEDA

TYP POLOZKY : 1 - POUZE CISLICE  
 2 - CELE KladNE CI ZAPORNE CISO  
 3 - REALNE KladNE CISO  
 4 - REALNE KladNE CI ZAPORNE CISO  
 5 - POUZE PISMENA  
 6 - PISMENA A CISLICE  
 7 - LIBOVOLNY ZNAK

ATRIBUT VDLITELNOSTI :

P = POUVINY  
 N = NEPOUINY

TYP DELKY :

F = FIXNI  
 U = VARIABILNI