

Ing. Karel Galgonek
FCÚ Praha

SYSTEM PROGRAMU MATICOVÝCH OPERACÍ A ZPŮSOB ULOŽENÍ DATOVÝCH SOUBORŮ

V článku budou shrnuty zkušenosti z práce na vytváření systému maticových operací (dále MATSYS). Snahou bylo vytvořit systém, který by umožňoval řešení takových ekonomických projektů u nichž se pracuje s rozsáhlými datovými soubory navíc značně proměnlivými a u nichž jsou algoritmy řešeny pomocí maticového a vektorového počtu. Vzhledem k tomu, že v řadě případů šlo o zvládnutí úkolů s neopakovatelnými algoritmy, museli jsme MATSYS v maximální míře přizpůsobit variabilitě požadavků zadavatele na jedné straně a nárokům na snohotvárnost a účelnost výstupních informací uživatele na straně druhé. Jelikož rozsah tohoto článku byl z technických důvodů značně omezen jsme nuceni korigovat naše záměry co se týče jeho struktury. Bude proto podána nejdříve stručná charakteristika MATSYSu a jeho funkce jako celku, dále popsána data z věcného a programátorského hlediska, jejich uložení, vstupy a výstupy. Teprve na základě těchto informací se pokusíme vyvodit některé obecné zásady pro vytváření a fungování systému programů v ekonomické sféře.

Stručný popis MATSYSu

MATSYS byl vypracován pro výpočetní systém JEEP a IBM 360/370 a to jak pro operační systém DOS tak i OS. Základním programovacím jazykem se stal FORTRAN IV, jen některé speciální programy byly vypracovány v jazyce ASSEMBLER. Pro používání

MATSYSu musí výpočetní systém umožnit programátorovi využívat minimálně toto zařízení:

- 1 snímač 80 sloup. děrných štítků
- 1 děrovač 80 sloup. děrných štítků
- 1 tiskárnu se 132 znaky na řádek
- 3 magnetopáskové jednotky pro zápis a čtení 9 stopých magnetických pásek
- 3 diskové jednotky pro uložení datových souborů (7.25 MGB)
- 128 KB vnitřní paměti

MATSYS byl konstruován stavebnicově formou podprogramů, které provádějí jednotlivé maticové a vektorové operace, vstup a výstup, přesuny mezi vnější a vnitřní pamětí a řadu speciálních funkcí pomocí nichž se propojila činnost jednotlivých podprogramů a byl vytvořen jednotný systém. Jednotlivé podprogramy MATSYSu podle jejich funkce můžeme rozdělit do následujících čtyř skupin:

oddělena časově i věcně

1. podprogramy vstupu
2. podprogramy maticových operací
3. podprogramy výstupu
4. podprogramy speciální a pomocné

Podprogramy čtvrté skupiny zajišťují propojení a svázání jednotlivých podprogramů a umožňují jejich jednoduché vzájemné používání. Vytváření potřebné vazby, které z jednotlivých podprogramů dělají vlastně teprve systém. Pro vzájemné propojení MATSYS používá čtyři způsoby:

1. vztahu argument - parametr
2. parametrických štítků
3. ukládání mezivýsledků na vnější paměťová média
4. pojmenovaného COMMON

Pomocí argumentů jsou podprogramům dodávány potřebné informace pro jejich činnost. Obdobně fungují i parametrické štítky. Bylo však snahou, aby jejich použití bylo minimální. Pokud při výpočtu je mezivýsledkem vektor nebo matice, je vždy na za-

dávateři zda-li budou na vnější paměťová média mezivýsledky uloženy či nikoliv. Ostatní mezivýsledky, ale i vektory a části matic zůstávají ve společně pojmenované oblasti COMMON. Vzhledem k tomu, že COMMON v propojení systému hraje rozhodující úlohu, poznamenejme, že slouží jak k dodávání potřebných pokynů pro další průběh výpočtu, tak i k předávání informací a mezivýsledků během provádění programu.

Data MATSYSu z věcného hlediska dělíme na: vektory, matice, nomenklatury (popisy prvků vektorů a matic, přičemž každý prvek nomenklatury se skládá z celočíselné a alfanumerické konstanty), agregační klíče (předpisy pro agregaci a dexagregaci vektorů a matic) a blavičky (nadpisy včetně formátů tisku jednotlivých stránek výstupních sestav).

Co se týče faktického uložení dat má MATSYS některá specifika. Z hlediska operačního systému MATSYS pracuje na jednom diskovém svazku vždy s jednou množinou dat (data set), kterou si pak se svými podprogramy již sám obhospodaruje. Tato množina dat je organizována jako DIRECKT s fixní délkou věty. Odpovídající metodou při programování v symbolickém jazyce PL/1 je REGIONAL (1). Navíc MATSYS připouští ukládání matic sekvenčně.

Založená množina dat z hlediska operačního systému je však z hlediska MATSYSu teprve v tzv. nestandardním tvaru to znamená, že MATSYS do takto založené množiny dat ještě nemůže zapisovat žádná data. Zde je nutno vysvětlit pojem soubor, kterým se v MATSYSu označuje jednotlivý vektor, matice, nomenklatura, agregační klíč a blavička. Každý takovýto soubor začíná větou, ve které jsou umístěné informace jednak týkající se přímo daného souboru a jednak umožňující MATSYSu rychlejší orientaci v množině dat. Tyto věty nazývá MATSYS jmenovkami a na rozdíl od jmenovek operačního systému jsou tyto jmenovky součástí množiny dat. Kromě těchto jmenovek se používají v MATSYSu ještě tzv. úvodní jmenovka, která v podstatě charakterizuje celou množinu dat a tzv. koncová jmenovka, která ukončuje aktivní část množiny dat a udává místo odkud je možno připsávat další soubory. Tento

způsob má zejména pro uživatele systému řadu výhod, které uvedeme později. Jak jsme již uvedli, MATSYS byl konstruován tak, aby mohl řešit maticové operace s maticemi do řádu 800. Vzhledem k tomu, že matice s ekonomickým obsahem velkých rozměrů jsou zaplněny maximálně z 25 %, rozhodli jsme se zapisovat, z důvodů šetření místa v množině dat, jen nenulové prvky matic.

MATSYS používá jako média pro primární vstupy dřevných štítků. To samozřejmě nevylučuje pomocí hlavního programu zajistit vstup i z jiných vnějších papěťových médií. Zejména u matic je tento způsob vstupu velice běžný. Naposledy jsme tak např. přebírali z FSÚ statistickou mezioborovou bilanci, tzn. mimo jiné dvě matice 416 x 416.

Programy vstupu jsou vytvořeny tak, aby připouštěly maximální variabilitu, znamená to, že u vstupu vektorů popřípadě matic až na malá omezení není předepsáno pevné obsazení dřevného štítku, ale je zde využíváno volitelného formátu. Před štítky s vektorem je např. umístěn parametrický štítek, který obsahuje řadu informací ať už těch, které se týkají evidence souboru, jako je např. identifikační číslo nebo text popisující tento soubor nebo těch, které nějakým způsobem usnadňují děrování nebo umožňují kontrolu. Mezi tyto informace patří např. konstanta, která je dosazena do všech prvků vektoru, které nebudou děrovány nebo minimální a maximální jejich možná hodnota. Je samozřejmostí, že všechny prvky vektorů nebo matic mohou být děrovány s ekonomickým označením (my používáme číselné nomenklatury), což umožňuje další formální kontrolu.

U výstupu bylo dbáno zejména na to, aby byly vypracovány podprogramy pro standardní výstupy, které v řadě případů stačí uživateli a nepotřebuje pro praktické účely nějaké speciální tisky. Zejména u výstupu velkorozměrných matic bylo dbáno na to, aby výstupní sestava byla pro uživatele co nej-
přehlednější, tzn. jsou tištěny převážně jen nenulové prvky.

Tato skutečnost vede nejen k přehlednosti, ale má i za následek šetření papírem. Rovněž v některých případech je pro uživatele ještě výhodnější, když dostane k dispozici jenom nejvýznamnější hodnoty. Vedle těchto tisků samozřejmě MATSYS obsahuje i podprogramy, které tisknou speciální výstupní sestavy jako např. tisk nomenklatur, agragačních klíčů apod. Každá stránka výstupní sestavy, pokud neobsahuje nic dalšího, má své číslo a datum svého vzniku. I zde je zásadou, že výstupní sestavy obsahují ekonomický popis výsledných hodnot.

Z hlediska fungování MATSYSu jako systému má nejdůležitější význam podprogram DISK, který patří do čtvrté skupiny mezi speciální podprogramy. Tento podprogram zajišťuje v rozhodující míře spojení mezi vnitřní a vnější pamětí. Vytváří na diskovém svazku fiktivní množinu dat s možností přímého zápisu a čtení, kterou opatří úvodní a koncovou jmenovkou (tzv. standartizace disku). Při zápisu nového souboru do množiny dat se koncová jmenovka uloží tak, aby byla její poslední aktivní větou. Za ní jsou dočasně nevyužitá věty. Zapisuje vektory uložené v COMMONu na disk a umožňuje jejich zpětné čtení včetně umístění v COMMONu. Vzhledem k tomu, že pro opakovatelné úlohy je vhodné mít pro stejné veličiny stejné pojmenování (číselné identifikátory) umožňuje podprogram DISK vytvoření tzv. značkového vektoru, tj. vektoru jehož obsah může být MATSYSem měněn. Naproti tomu tzv. kmenový vektor nemůže být MATSYSem změněn a uživatel systému má téměř stoprocentní jistotu, že takovýto vektor nebude nikdy přepsán a zůstane mu po celou dobu užívání k dispozici. Z dalších funkcí podprogramu DISK je třeba jmenovat vyhledání souboru v množině dat a v případě, že se takovýto soubor nenalezne dostane uživatel o tomto informaci a navíc dostane výpis všech jmenovek uložených souborů. Výpis jmenovek souboru můžeme požadovat samozřejmě i samostatně. Při ukládání souboru v množině dat podprogram DISK nalezne koncovou jmenovku, tzn. nalezne místo odkud je možno přepisovat daný soubor. Je možné zrušit libovolný počet naposledy zapsaných souborů a navíc pomocí tzv.

formální jmenovky je možné znepřístupnit MATSYSu původně aktivní věty. Ještě několik poznámek k vyhledávání souboru v množině dat. Soubor můžeme vyhledávat jednak pomocí tzv. identifikačního čísla, což je číslo, které nějakým způsobem charakterizuje daný soubor, dále podle datového čísla, což je v podstatě datum vzniku souboru. Při vyhledávání se rovněž může kontrolovat druh souboru, tzn., zda-li jde o vektor, matici, agregační klíč apod. Pokud známe adresu věty hledaného souboru můžeme ji uvést a tak podstatně zrychlíme práci systému. Podprogram DISK provádí celou řadu dalších kontrol a o všech informuje uživatele, a podle povahy chyby buď je schopen sám ji odstranit a pokračuje ve výpočtu, a nebo není schopen ji odstranit a pak výpočet je ukončen.

Uvedený popis systému je nutno brát jen jako stručnou informaci o jeho práci a možnostech, vůbec nešlo o popis programů. Navíc je si nutno uvědomit, že tím, že tento systém je vypracován stavebnicově umožňuje kdykoliv nějaký podprogram buď zrušit nebo další podprogram vypracovat, aby bylo možno zabezpečit zadavatelem požadované funkce. Přesto, že zde hovoříme o maticovém systému, nechtěl tento systém sloužit jen vysoce komplikovaným matematickým propočtům, ale chtěl být dobrým pomocníkem při běžných hospodářských pracích. Při podrobnější rozboru systému lze souhlasit s tím, že jeho uplatnění má mnohem širší možnosti než by se na první pohled zdálo. Vždyť jakýkoliv sloupec čísel můžeme považovat za vektor a libovolnou tabulku za matici. Pro nematicové operace by zde stačilo vypracovat další podprogram, který by se zařadil do systému. Tuto skutečnost nám umožňuje stavebnicový způsob konstrukce systému.

Obecné zásady konstrukce systému

Při vytváření MATSYSa jsme v první řadě narazili na otázku základního přístupu k jeho konstrukci a ten v podstatě může být dvojit:

- 1) matematický tzn., že systém bude obsahovat v našem případě všechny maticové operace potřebné při strukturální analýze na nejobecnější úrovni, přičemž vstup, výstup, nejrůznější úpravy matic a hlavně komunikace mezi zadavatelem a systémem na jedné straně a systémem a uživatelem na druhé straně budou řešeny nejen v jednoduché formě, ale nebudou, což je nejdůležitější, tvořit se systémem harmonický celek;
- 2) zadavatelsko-uživatelský tzn., při respektování všech matematických zákonů bude hlavní důraz kladen právě na vstup, výstup, předvídatelné věcné úpravy datových souborů, jejich evidenci - jinými slovy - systém musí umět převést požadavky ekonomů do matematických vzorců a zpátky zase výsledky interpretovat tak, aby uživateli nepůsobily co dohromady žádné těžkosti.

Z matematického hlediska jsou doposud zpracovány snad všechny systémy programů, které řeší maticový počet. Tento způsob zpracování má řadu výhod: relativní jednoduchost, větší obecnost a hlavně, že tvůrci se nemusí zabývat vstupem konkrétních dat a jejich praktickou interpretací na výstupu. Takto konstruovaný systém jak vstupy tak výstupy určuje. Tvůrci kladou hlavní důraz na matematickou stránku věci, neboť jejich cílem je pomocí široké škály programů, v daném případě z maticového počtu, zvládnout pokud možno beze zbytku celou jeho problematiku.

Jestliže v prvním případě byl hlavní důraz kladen na matematickou stránku, pak u zadavatelsko-uživatelského hlediska jde v první řadě, při zajištění vysoké matematické úrovně systému (matematické otázky řešené systémem jsou pochopitelně omezeny jen na ty, které vyplývají z požadavků zadavatele a uživatele) o vytvoření takové soustavy programů, která by co nejlépe umožňovala styk zadavatele a uživatele s počítačem.

Je možno konstatovat, že nejvíce práce při programování ekonomických úloh zabírají vstupní a výstupní operace, ale ne vždy uživatel vyžaduje precizně specifikovaný výstup a proto

je výhodné mít u těchto systémů možnost standartních tisků. Mění možno také podceňovat evidenci zpracovaných dat a to zejména z věcného hlediska. Nemáme zde na mysli jen formální informace jako je např. identifikační číslo souboru, datum vzniku, alfanumerický popis, ale i takové, které udávají jeho rozměry, umístění v množině dat, počet zapsaných vět apod. Velmi osvědčenou identifikací souboru je suma všech prvků. Do této oblasti patří i otázky s přesuny, rušení a znovuvytvářením datových souborů apod. V řadě případů vystačíme s operačním systémem počítače, ale u rozsáhlejších systémů a navíc s tak různorodou činností je výhodné tyto funkce začlenit do vlastního systému.

Potřebnost zadavatelsko-uživatelského hlediska při vytváření systému bychom mohli dokumentovat následovně:

- 1) datová základna pro ekonomické úlohy je značně rozsáhlá a a poměrně rychle zastarává, proto je s tím v programech nutno počítat;
- 2) datovou základnu je třeba neustále doplňovat a to z nejrůznějších pramenů; proto vstupy musí být co nejuniverzálnější;
- 3) jelikož je bezpodmínečně nutné zajistit správnost vstupních údajů je třeba mít co nejvyčerpatečnější škálu kontrol a to jak po formální tak i pokud možno po věcné stránce;
- 4) možnost oprav vyžaduje programy na jejich provádění;
- 5) ekonomické data jsou popsány legendami (nomenklaturami) ať už numerickými či alfanumerickými, které je nutno upravit pro výpočty na počítačích;
- 6) výsledky před tiskem je třeba zpátky opatřit legendou;
- 7) výsledky resp. mezivýsledky ve většině případů uchováváme;
- 8) pro praxi je důležité mít možnost data nejrůznějšími způsoby agregovat a desagregovat;
- 9) vedle stabilních tisků, které se programují samostatně, se běžně tisknou různé informativní sestavy a pro tento účel je výhodné mít sice univerzální, ale přece jenom lehce přizpůsobitelný program tisku podle přání zadavatele;
- 10) jakýkoliv systém pro ekonomické výpočty by měl být vybaven

vlastní autoevidenci souborů, jejich třídění, vyhledávání, ukládání, rušení a těmito činnostem odpovídající tisky;

- 11) pro často opakující se algoritmy výpočtu se osvědčuje vytvářet ucelené bloky podprogramů, jako příkz MATSYSu je možno uvést agregaci matice: vzhledem k tomu, že matice jsou ukládány po sloupcích jsou i tak agregovány, tzn. matice musí být nejdříve agregována po řádcích, poté se musí transformovat, agregovat po sloupcích a opět transformovat, aby výsledná matice byla uložena zase po sloupcích.

Závěrem je nutno říci, že autoři jsou si vědomi určitých nedostatků systému, ale řada z nich vyplývá z běžně existujících nešvarů při programování větších ucelených celků: netýmové práce, nedostatečná předběžná analýza, krátkost termínů, nedostatek programátorů a v našem případě k tomu přistupovala ještě skutečnost, že MATSYS sice zjednodušuje práci programátorovi, zkracuje termíny výpočtu a přináší řadu výhod i uživateli, ale není tzv. preferovaným úkolem. Přesto si myslíme, že poukázání na některé charakteristické rysy MATSYSu a zkušenosti při jeho vytváření může být poučkou ostatním, kteří budou stát před podobným úkolem.