

OPTIMALIZACE PŘECHODNOSTI DAT VÝPOČETNÍCH SYSTÉMŮ

V naší republice je dnes již v provozu celá řada větších výpočetních systémů, které jsou podporovány operačními systémy na úrovni OS/MPT nebo OS/MVT (systémy z řady počítačů JSEP 1 - EC 1030, 32,33 a 40 nebo systémy IBM z řady 360) nebo na úrovni OS/VS1 či VM (u počítačů IBM řady 370 - viz /1/) a od loňského roku i první operační systém typu SVS (u počítače z řady JSEP 2 - EC 1055). Všem uživatelům výše zmíněných operačních systémů je věnován tento článek.

Dosavadní přístup výrobců počítačů i programového vybavení, a tedy i operačních systémů k uživatelům výpočetních systémů v otázkách právě programového vybavení je doposud takový, že poskytuje každému uživateli velmi širokou paletu možností. Operační systémy pro konkrétní konfigurace počítačů si pak každý uživatel musí vytvořit tak, že si vybere z celé množiny programových modulů, dodávaných výrobcem pro podporu všech zařízení (jím vyráběných), jen ty moduly, které podporují zařízení instalovaná v jeho konfiguraci. Pro tzv. generování operačních systémů dodává výrobce soubor generačních makroinstrukcí. Seřazením těchto makroinstrukcí do předepsané posloupnosti a zadáním požadovaných operandů makroinstrukcí vznikne předpis pro výběr požadovaných funkcí operačního systému. Funkce operač-

ního systému jsou pak reprezentovány vybranou podmnožinou programových modulů. Jednotlivé programové moduly jsou, jako konečný efekt, uloženy do knihoven operačního systému.

Vzhledem k tomu, že posloupnost generačních makroinstrukcí i jejich parametrů i poloha programových modulů v celé množině, dodané pro generování, je dána výrobcem, je i výsledný operační systém sestaven poněkud "tvrdě".

Programové moduly jsou při generování ukládány do knihoven výsledného operačního systému. Umístění těchto knihoven na diskových paměťových nosičích i jejich vnitřní struktura pak rozhodují nejen o kvalitě operačního systému, ale i o jeho efektivnosti, tj. o rychlosti jeho vlastní činnosti i o množství dat, která výpočetním systémem budou zpracována za jednotku času. Tuto vlastnost operačního systému nazvěme **p r ů c h o d n o s t**.

Je pravdou, že již při prvotním návrhu generace operačního systému může uživatelův systémový programátor do jisté míry rozhodnout o konečné podobě systému, a tím i o jeho průchodnosti pro data. Vzhledem k široké paletě zpracovávaných dat však vnitřní struktura operačního systému a jeho rozmístění po diskových paměťových nosičích nemůže nikdy být optimální pro daný profil zpracování. Je třeba zvolit určitý kompromis a stanovit takovou strukturu operačního systému, která by měla co nejpríznivější vliv na zpracování všech dat.

Hovoříme-li o vnitřní struktuře operačního systému, rozumějme tím umístění jednotlivých programových modulů buďto přímo do jeho rezidentní části (tj. přímo do vnitřní paměti) nebo do knihoven, z nichž je příslušný modul natažen až teprve v okamžiku potřeby. Čím více programových modulů operač-

ního systému bude v jeho rezidentní části, tím bude operační systém rychlejší (není nutno provádět jejich periferní přenos do vnitřní paměti, který je o několik řádů pomalejší než rychlost centrální jednotky), ale na druhé straně roste velikost rezidentní části tohoto operačního systému na úkor paměti, určené pro zpracování uživatelských programů. Operační systém obsahuje také celou řadu tabulek, určující pořadí zavádění jednotlivých modulů z knihoven do vnitřní paměti. Složení těchto tabulek lze rovněž měnit a ovlivnit tak rychlost zpracování, a tím i průchodnost dat výpočetním systémem.

U virtuálních operačních systémů typu OS/VS1, VM či SVS vstupuje do hry i faktor umístění modulů operačního systému do reálné (vnitřní) či virtuální rezidence. Při umístění modulu do reálné paměti rezidentní části operačního systému klesá velikost reálné paměti, určené pro stránkování zbytku operačního systému i virtuální paměti pro stránkování uživatelských programů. Při umístění modulů do virtuální části rezidence pak roste celkový rozsah virtuální paměti, a tím i velikost diskového souboru, určeného pro modelování této virtuální paměti, a potažmo i stránkovací aktivita systému, tj. četnost přenosů mezi reálnou pamětí a pamětí virtuální.

Již v úvodu jsme se zmínili i o vlivu rozmístění knihoven operačního systému na diskových nosičích. Již zmíněné operační systémy se zpravidla po vygenerování vejdou na jeden diskový nosič s kapacitou 29 MB (u OS typu MPT či MVT) či na jeden nosič s kapacitou 100 MB (u OS typu OS/VS1). Z výše zmíněných důvodů průchodnosti systému je však nutné rozmístit všechny knihovny na více diskových jednotek tak, aby mohl být vyřešen současný požadavek na přenos více programových modulů z různých knihoven do rezidentní části operačního systému. Velmi důležitou roli hraje i umístění jednotlivých kni-

hoven vůči tzv. VTOC (Volume Table of Contents) - tj. tabulce obsahu diskového nosiče. Tuto tabulku lze při inicializaci diskového nosiče v zásadě umístit kamkoliv. Ty knihovny operačního systému, k nimž je nejčastější přístup, by měly být z důvodu rychlosti přenosu programových modulů v nich umístěných situovány tak, aby byly co nejbližší VTOC. Pro účely operačního systému se tedy doporučuje umístit VTOC mezi dva nejfrekventovanější soubory, které jsou pak k tomuto VTOC uloženy symetricky. To má za následek zkrácení vzdálenosti, kterou musí překonat čtecí/zapisovací hlavy od VTOC k začátku souboru a o to rychleji může být zahájen vlastní přenos modulu do vnitřní paměti.

Z výše popsaných problémů se sestavením dobře pracujícího operačního systému je zřejmé, že této problematice se jednou začne nutně věnovat každý uživatel. V souladu s terminologií (viz /2/) si nazvěme práci na zlepšení činnosti operačního systému jeho **l a d ě n í m**.

Uživatelé operačních systémů nižší úrovně (typu MFT či MVT) mají pro jejich ladění k dispozici programové prostředky jako GTF (General Trace Facility), respektive SMF (System Management Facility), což jsou programové součásti operačních systémů, zapisující údaje o činnosti systému do zvláštních souborů operačního systému. Tyto údaje lze pak následovně rozebírat z různých hledisek, pochopitelně již pomocí programů pro tyto účely napsaných samotným uživatelem. Pro modernější operační systémy, pracující s virtuální pamětí, jsou kromě již uvedených programových prostředků výrobcem poskytovány prostředky podstatně silnější. Mezi tyto prostředky patří např. programový produkt firmy IBM pronajímáný pod komerčním názvem VS1PT (Program for Tuning of VS1). Pro zajímavost uvádíme cenu, za niž je program VS1PT pronajímán.

Měsíční nájem činí cca 300 US \$.

Žádný ze zde uvedených prostředků pro ladění operačních systémů však není všemocný. Celá řada optimalizačních opatření (v souladu s /1/ a /2/) je však dělána jen na základě dlouholetých zkušeností a "od oka" uživatele a systémového programátora.

Při ladění operačního systému se vykytují tři kategorie problémů:

- řízení výkonnosti operačního systému
- nasycená centrální jednotka, tj. případ, kdy centrální jednotka je vytižena na téměř 100% a přitom požadovaná práce nemůže být udělána
- přetížené diskové jednotky, kanály a řídicí jednotky.

Je třeba zdůraznit skutečnost, že za žádných okolností by nemělo dojít ke sledování a tím i řízení výkonnosti operačního systému až v okamžiku, kdy se situace ve zpracování dat stává kritickou, tj. v době, kdy je celý výpočetní systém přetížen. Doporučuje se proto sledovat průběžně a dlouhodobě celou řadu veličin o výkonnosti operačního systému a vyhodnocovat pak:

- zatížení centrální jednotky
- využití kanálů
- využití diskových jednotek
- množství zpracovaných dat
- dobu odezvy u teleprocessingových aplikací
- počet transakcí
- I/O aktivita ostatních periferních zařízení

a u virtuálních operačních systémů pak navíc i

- stránkovací aktivitu, tj. počet přenosů mezi reálnou a virtuální pamětí
- zatížení diskových jednotek, kde jsou umístěny soubory určené pro stránkování virtuální paměti.

Na celkovou výkonnost výpočetního systému má velký vliv zejména rychlost přístupu k systémovým datům, tedy k jeho programovým modulům, ale stejný význam má i rychlost přístupu k uživatelským datům. Podle literatury /2/ je třeba věnovat v souvislosti s touto rychlostí pozornost celé řadě aspektů. Uvedme si některé z nich i s doporučenými hodnotami, jejichž přesáhnutí znamená degradaci výkonnosti výpočetního systému a tedy i jeho průchodnosti pro data. Na tomto místě je vhodné se zmínit o skutečnosti, že na přetížení výpočetního systému jsou daleko citlivější on-line aplikace a interaktivní subsystémy jako IMS, CICS apod.

Jedním z aspektů, ovlivňujících činnost operačního systému, je tzv. **o b s l u ž n ý č a s**, což je čas potřebný na provedení systémové makroinstrukce EXCP (Execute Channel Program - Zpracuj kanálový program) a čas na provedení instrukce SIO (Start I/O - zahájení přenosu). Obslužný čas je ovlivňován celou řadou faktorů, mezi něž patří:

- využití přenosového kanálu
- využití řídicí jednotky
- využití vlastního zařízení a přímým přístupem - nevhodné SEEK časy, tj. časy pro nastavení čtecích/zapisovacích hlav do pozice, v níž má být přenos zahájen (viz pozice VTOC).

Podle literatury /2/ se doporučuje, aby obslužný čas (v podstatě průměrná doba vyhledávání) např. u diskových jednotek IBM 3330 (100 MB disky) nepřesáhl hodnotu 40 msec. Pro dosažení této hodnoty je pak nutné, aby využití přenosového kanálu všemi ostatními přenosy nebylo vyšší než 40% pro dávkový způsob zpracování a pro dialogový způsob zpracování dokonce ještě nižší. Pro informační systém IMS/VS se uvádí dokonce jenom 30% využití.

Na obslužný čas kanálu má velký vliv délka přenášeného bloku. Uvádí se, že při délce bloku 6 kB u zařízení IBM 3330, což je zhruba blok odpovídající polovině délky stopy, je tento obslužný čas roven hodnotě 8-9 msec. Současně se uvádí, aby pro data, ukládaná sekvenčním způsobem, byl volen blok delší. Toto však neplatí pro data ukládaná do databázových souborů, která jsou většinou prohledávána nahodile. O délce datových bloků bude ještě zmínka v souvislosti s daty ukládanými na magnetické pásky. Jak jsme již uvedli, má na obslužný čas vliv využití kanálu a řídicí jednotky. Stává-li se toto využití natolik velkým, že je zpracování degraďováno, je vhodné uvažovat o uložení datových nebo systémových souborů na diskové jednotky, připojené k centrální jednotce přes různé řídicí jednotky a přenosové kanály. Vzhledem k poměrně vysokým nákladům na tato zařízení je však zatím v našich podmínkách většinou běžné, že všechny diskové jednotky jsou připojovány přes jednu řídicí jednotku a přes jeden přenosový kanál. Pokud je autorovi známo, jediné u počítačů JSEP a některých počítačů IBM 370, které jsou v republice instalovány, jsou diskové jednotky připojeny přes dvě řídicí jednotky a dva přenosové kanály. U velkých výpočetních systémů, které mají velké množství připojených pamětí s přímým přístupem, je běžné, že je využíváno nejméně 4 přenosových kanálů současně.

Na dobu odezvy, zejména u dialogově orientovaných aplikací, má vliv i současné umístění nahodile a sekvenčně organizovaných dat na jednom přenosovém kanálu. K velkým problémům, a o nich jsme se již zmínili v souvislosti s umístěním některých souborů operačního systému, dochází v tom případě, že na jedné diskové jednotce jsou umístěny dva i více vysoce aktivní soubory, které jsou případně více vzdáleny od VTOC. V tomto případě dochází k nutnosti častého a neefektivního pohybu čtecích/zapísovacích hlav, nutného pro zabájení požadovaného periferního přenosu. Velmi nevýhodné je rovněž umísťovat na jednu diskovou jednotku velké množství malých souborů. To totiž způsobí vzrůst rozsahu VTOC a následně i dlouhé časy jeho prohledávání při otevírání těchto malých souborů. Obdobná situace nastává i u velkých knihoven s velkým počtem programů v nich umístěných. Velké množství programů totiž vede k růstu velikosti seznamu (directory), obsahujícího názvy programových modulů, a tím i k nárůstu času, potřebných na prohledávání tohoto seznamu. U operačních systémů se u některých systémových knihoven, k nimž je častý přístup a které jsou hodně velké (např. knihovna SVC - Supervisor Call), řeší tato otázka tak, že seznam všech jejich modulů i s odvolávkami na fyzické adresy poloh na disku je umístěn přímo do rezidentní paměti operačního systému, aby pak přístup k těmto programovým modulům byl co nejrychlejší.

Závěrem uvedme některá doporučení, která mají kladný vliv na průchodnost dat výpočetními systémy, a to nejen systémy řízenými vyššími operačními systémy a které může ovlivnit každý programátor přímo při návrhu svého programového systému.

V e l i k o s t b l o k u

Podle firemní literatury /3/ i /2/ se uvádí a doporučuje

volit pro zařízení IBM 3330 bloky dlouhé maximálně 6 kB u datových souborů, kdežto pro knihovny volit blok rovný délce jedné stopy. V literatuře se uvádí, že např. zvětšením velikosti bloku z 200 B na 6 kB u disků IBM 3330 vzrostlo využití času centrální jednotky o 87%, elapsed time klesl o 90% a využití kanálu kleslo o 65%.

Promluvme si nyní o problematice velikosti bloků na magnetických páskách. Na základě známého vzorce

$$l_{\text{bloku}} = \frac{B_{\text{kazda}} + 82}{64} + 15 \text{ (min) } \quad \text{pro hustotu } 1600 \text{ Bpi}$$

si můžeme odvodit graf /1/, udávající využití magnetické pásky v závislosti na délce bloku. Podle tohoto grafu je vidět, že 100% využití MP dosáhneme při maximální délce bloku, který můžeme na magnetickou pásku napsat, tj. 32 kB. Zmenšíme-li však délku bloku při hustotě záznamu 1600 Bpi o více než polovinu, tj. na zhruba 12 kB, a při hustotě 800 Bpi dokonce o 3/4, tj. zhruba na 8 kB, poklesne využití magnetické pásky jen asi o 5%. Současně však klesne velikost

v y r o v n á v a c í p a m ě t í

potřebné pro zajištění periferního přenosu do vnitřní paměti. Tato velikost vyrovnávací paměti i jejich počet, který je současně použit pro jeden soubor, má vliv na průchodnost dat výpočetním systémem. U nevirtuálních operačních systémů má velikost a počet vyrovnávacích pamětí podstatně menší vliv než u systémů s virtuální pamětí.

Podle obecně známého tvrzení má zvyšování velikosti délky bloku a zvyšování počtu vyrovnávacích pamětí kladný vliv na rychlost přenosu dat do vnitřní paměti a na rychlost jejich

následovného zpracování. U nevirtuálních operačních systémů je programátor omezen pouze velikostí paměti, která je přidělena pro zpracování jeho programu. Jestliže jsou ale všechny magnetické pásky připojeny k centrální jednotce pomocí jednoho přenosového kanálu, má růst délky bloku dat za následek blokování tohoto kanálu přenosem dlouhého bloku po relativně dlouhou dobu a s ohledem na multizpracování pak i k čekání ostatních programů na ukončení dlouhého přenosu v případě, že samy požadují podobný periferní přenos.

U virtuálních operačních systémů aplikovaných v našich podmínkách, tj. pracujících na výpočetních systémech s maximální velikostí reálné paměti 1 MB, ale ve většině případů jen 512 kB a méně, je však problematika velikosti a počtu vyrovnávacích pamětí podstatně důležitější. Kromě toho, že platí stejné aspekty, o kterých jsme se zmínili u nevirtuálních operačních systémů, platí i omezení velikosti vnitřní paměti, určené pro stránkování uživatelských programů. Jestliže si např. uvědomíme skutečnost, že rezidentní část operačního systému spotřebuje z 512 kB zhruba 120-150 kB, zbývá nám pro stránkování celého zbytku operačního systému o rozsahu několika MB jen něco přes 350 kB. Jestliže pak dále vezmeme v úvahu potřebu držet vyrovnávací paměti pro přenos dat do paměti programu uvnitř reálné paměti po celou dobu jeho činnosti, zjistíme, že velikost bloku a počet vyrovnávacích pamětí začíná hrát velkou roli. Běžný program, napsaný ve vyšším programovacím jazyce, zabere okolo 100 kB. Jestliže si představíme, že každý aktualizací program používá nejméně tři vnější soubory a kdyby každý z nich měl blok dlouhý 32 kB a používal dvě vyrovnávací paměti, tak jenom na tyto vyrovnávací paměti je zapotřebí 192 kB, které musí být drženy v reálné paměti po celou dobu činnosti programu, což může být řádově i desítky či stovky minut. Začneme-li v tomto okamžiku uvažovat

o nutnosti multiprogramování, s nímž musíme u těchto výkonných výpočetních systémech počítat, vidíme, že nám pro stránkování ostatních programů nezbyvá k dispozici prakticky žádná reálná paměť. U všech virtuálních operačních systémech, které jsou u nás doposud používány, jsou stránkovací algoritmy udělány tak, že operační systém se snaží pozastavit ty programy, které degradují průchodnost dat a tato skutečnost způsobí deaktivování (pozastavení) takového programu až do doby, kdy jsou ostatní programy s malými nároky zpracovány. Může tak poměrně snadno dojít k pozastavení již zmíněné úlohy na velmi dlouhou dobu.

S ohledem na hustotu záznamu se tedy doporučuje používat délku bloku maximálně 8-12 kB a u virtuálních systémů pak v případě nutnosti zvolit větší délku bloku, použít pro takovéto soubory jen jednu vyrovnávací paměť.

Použitá literatura

1. Některé zkušenosti se sledováním činnosti systému OS/VS1 a VM/370 - Ing. J. Nedoma (Programování 79)
2. An MVS tuning approach, IBM Systems Journal (Volume 19, Nr 1, 1980)
3. Direct Access Storage Devices and Organization Methods (GC 20 - 1649)
4. SMP Manual (GC 35 - 0004)
5. GTP Guide (GC 24 - 5092)

**GRAF 1 : VYUŽITÍ MP O DÉLCE 2400 STOP
V ZÁVISLOSTI NA DÉLCE BLOKU.**

