

NĚKTERÉ ZKUŠENOSTI SE SLEDOVÁNÍM ČINNOSTI SYSTÉMU OS/VSI A VM/370

Jednou ze základních vlastností moderních výpočetních systémů je jejich obecnost. Dá se ukázat, že většina běžně dostupných počítačů je navzájem logicky ekvivalentní, tj. tyto počítače jsou schopny řešit totéž širokou třídu úloh. Protože výrobci operačních systémů obecně nemohou předpokládat podmínky zpracování v konkrétních instalacích, dodávají operační systémy s "otevřenými" parametry. Každá instalace, respektive její systémoví programátoři pak provádějí důležitá rozhodnutí, jejichž cílem je maximální přizpůsobení dodaného operačního systému charakteru úloh zpracovávaných v dané instalaci.

Neméně důležitým pokračováním tohoto procesu je pak ověření, zda-li takto vytvořený operační systém splňuje požadavky instalací stanovené. Iste ověření se provádí na základě dat získaných z činnosti operačního systému. Pak existují dvě možnosti: konstatujeme, že běžící operační systém pracuje plně k naší spokojenosti, nebo zjistíme, že některá jeho část pracuje neuspokojivě. V posledním případě musíme provést modifikaci operačního systému, která vede k odstranění zjištěných nedostatků a znovu ověřit vliv provedených změn. Tyto kroky opakujeme tak dlouho, pokud lze za daného stavu (např. konfigurace) dosáhnout jistého zlepšení práce operačního systému.

V následující části svého příspěvku se budu z výše uvedených hledisek zabývat dvěma operačními systémy, které provozujeme

v ÚVT SPK na počítači IBM systém/370 model 148, a to OS/VSI a VM/370.

OS/VSI:

OS/VSI je operační systém, který je vybudován na operačním systému OS/MFT, určeném pro řadu S/300. Základní rozdíl proti OS/MFT je realizace virtuální paměti. U OS/MFT je operační paměť rozdělena na dvě části. V jedné je rezidentní supervisor systému, který obsahuje nukleus a některé funkce, které je vhodné mít rezidentní. Druhá část je rozdělena na sekce, maximálně 15, ve kterých je možno nouběžně zpracovávat úlohy uživatelů. Zde je nutno si povšimnout, že rezidentní funkce zabírají reálnou paměť stále, ať jsou momentálně potřebné či nikoliv. Počet rezidentních funkcí je pak velmi omezený, zpravidla na pouze nejn nutně-jší a kritické funkce, bez kterých by systém pracoval velmi neuspokojivě.

Situace v OS/VSI je analogická až na to, že celá záležitost se odehrává ve virtuální paměti. Všechny rezidentní funkce je možno umístit do logicky významné části virtuální paměti, tzv. stránkovatelného supervisuera. Zde můžeme umístit daleko větší počet těchto funkcí, protože máme k dispozici např. IBM máme 50Kb, tedy dvacetkrát více paměti. V reálné paměti pak mají tendenci netrvávat jen ty funkce stránkovatelného supervisuera, které jsou relativně často volány.

Tyto funkce, o kterých hovoříme, jsou na externích médiích uloženy v knihovnách SYS1.SVCLIB a SYS1.LINKLIB, stejně jako u OS/MFT. které z nich budou při zahájení operačního systému zavedeny do paměti, je určeno některými členy knihovny systémových parametrů SYS1.PARMLIB, ani v systému OS/VSI není vhodné (z hlediska velikosti virtuální paměti, frekvence stránkování), aby všechny moduly ze SYS1.SVCLIB a SYS1.LINKLIB byly částí stránkovatelného supervisuera; proto má umístění těchto knihoven značný význam na chod celého operačního systému. Kromě nich existují ještě další dva soubory zásadního významu a to SYS1.PAGE a SYS1.SYSPool. SYS1.PAGE je soubor, ve kterém jsou všechny stránky virtuální paměti operačního systému.

SYS1.SYSP00L obsahuje vstupny a výstupny dílnu uživatelů.

Nejjednodušší způsob sledování činnosti systému OS/VSI nabízí podsystém SMF, který automaticky hromadí údaje o využití CPU, stránkování, počtu provedených periferních operací, spotřebované virtuální paměti, atd. Je pak jen otázkou času napsání programů, potřebných pro získání potřebné informace z dat, která SMF poskytuje. Co se týče frekvence použití jednotlivých modulů ve stránkovatelném supervisoru, je nejjednodušší aplikace systémového programu GIF známého i v OS/MFT. Tento program je schopen sledovat všechna SVC 6, 7, 8, kterými jsou realizována makra LINK, LOAD, XCTL a ATTACH. V záznamech, které GIF ukládá např. na pásku, se pak vyskytují jména volaných modulů. Relativně triviálním úkolem je pak vytvoření tabulky obsahující jména všech programů na pásece a počet jejich výskytu. Kromě těchto programů má firma IBM v zásobě řadu daleko speciálnějších prostředků, poskytujících detailnější pohled na činnost systému OS/VSI.

Další možností, vyžadující již netriviální úsilí je vytvoření programu, který by v pravidelných intervalech testoval obsah významných polí všech UCB bloků v systému (pro každé periferní zařízení konfigurace existuje jeden UCB blok). Tak je možno zjistit zatížení jednotlivých disků a pásek na selektorových nebo blok-multiplexních kanálech.

Všechny popsané prostředky poskytují detailní informace, které je potřebné zpracovat dále určitým způsobem, např. zpřůměrovat, vyjádřit jako frekvenci, v procentech, atd., tedy ve formě vhodné pro pochopení získaného údaje.

V ÚVT SPK byl operační systém OS/VSI nainstalován na podzim roku 1977. Celý systém byl nagenеровán na jediném 100MB disku. Pro vytvoření stránkovatelného supervisoru byly použity seznamy, které byly do SYS1.PYRMLIB uloženy při generování. S takto vytvořeným systémem jsme dosáhli ukazatelů, které jsou v tabulce 1 uvedeny ve sloupci 1977. První nagenеровaná verze byla OS/VSI Rel 06.08.

V roce 1978 jsme nagenerovali Rel. 06.001, kde jsme se již zaměřili na účelné rozmístění výše zmíněných kritických souborů na disku. Tyto soubory jsme se snažili umístit co nejbliže fyzického středu disku symetricky kolem VTOC. Kromě toho jsme provedli rozsáhlé modifikace většiny seznamů rezidentních funkcí v SYSL. PARM LIB podle údajů, které jsme získali již výše popsanou cestou. Výsledek se dostavil. Je uveden v tabulce 1 ve sloupci 1978.

Na začátku roku 1979, po rozšíření diskové kapacity konfigurace jsme nagenerovali Rel. 06.0F. Systém byl nagenerován na dvě diskové sady tak, že SYSL. SVCLIB a SYSL. LINKLIB jsou umístěny uprostřed jednoho disku a SYSL.PAGE a SYSL.SYSPool uprostřed druhého disku. Kromě toho byly provedeny další změny v seznamech v SYSL. PARM LIB a některé menší změny v JES (zmenšili jsme počet vyrovnávacích pamětí). Naměřené hodnoty jsou uvedeny v tabulce 1 ve sloupci 1979. Z tabulky je vidět výrazně rostoucí využití stroje při práci s operačním systémem OS/VS1.

Je vidět, že i po relativně mírném vyladění bez zásahů do interní logiky systému, lze dosáhnout překvapivě uspokojivých výsledků.

Tabulka 1

	1977	1978	1979
1. využití CPU /%/	38,9	49,7	87,6
2. stránkování /počet/s/	1,4	2,8	2,9
3. kanál 1 obsazen, CPU čeká /%/	32,3	10,5	2,4
4. SYSRES obsazen a CPU čeká /%/	46,0	22,0	4,2
5. SYSLIB obsazen a CPU čeká /%/	-	-	2,2
6. stránkovatelná paměť /KB/	791,2	802,0	800,9

Reálná paměť byla ve všech případech 1024KB, velikost nuklea 120KB, zpracování probíhalo současně v osmi lekcích paměti. Uvedená data kolísají v rozmezí $\pm 10\%$.

VM/370:

VM/370 je operační systém naprosto odlišný od ostatních operačních systémů firmy IBM, protože jeho určení je také naprosto odlišné. Zatím co systémy jako OS/MFT, OS/VSI a další jsou určeny pro dávkové zpracování, eventuelně s možností interaktivního přístupu do některých jejich podsystémů (CICS, IMS apod.), VM/370 neposkytuje uživateli žádné podobné funkce, ale pouze rozděluje prostředky reálné konfigurace (CPU, paměť, periferní zařízení) jednotlivým uživatelům tzv. virtuálních strojů. Virtuální stroj je simulovaná konfigurace počítače řady S/360 nebo S/370. Uživatel tohoto virtuálního stroje se tak zároveň stává jeho operátorem a je na jeho libovůli, jaký operační systém bude ve svém stroji provozovat, zda-li OS/MFT, OS/VSI, DOS/VS nebo s VM/370 speciální dodávaný systém CMS. Tak VM/370 umožňuje provozovat na jediné reálné konfiguraci počítače řady S/370 více operačních systémů souběžně.

Konfigurace virtuálního stroje se skládá z virtuální CPU jednotky, simulované přidělováním času reálné CPU jednotky, z virtuální paměti realizované stejně jako v OS/VSI jen s tím rozdílem, že stránka zde má 4KB místo 2KB, z virtuálních vstupů a výstupů (snímačů, děrovačů, tiskáren) simulovaných pomocí disku a z virtuálních disků (minidisků) realizovaných jako část reálných disků. Pouze pásky tvoří výjimku a jeden virtuální páskový stojan musí odpovídat jednomu páskovému stojanu reálné konfigurace stroje. Virtuální konzole, ze které uživatel řídí chod virtuálního stroje, se stává kterýkoliv terminál reálné konfigurace po provedení určité zahajovací posloupnosti. Definice všech virtuálních strojů, které lze provozovat pod daným VM/370 jsou uloženy ve speciálním souboru zvaném VM/370 directory.

Jedním ze základních systémových programů je VM/370 plánovač. Stručně popíše způsob jeho práce.

Uživatel virtuálního stroje, respektive virtuální stroj, je v klidovém stavu, nemá-li nevyřešen žádný požadavek. Jakmile uživatel zadá požadavek, vstupuje do množiny kan-

didátů na zpracování a pak, když to dovolí dostupnost prostředků reálného systému, vstupuje do multiprogramované množiny jedoucích uživatelů. Pouze členové této množiny skutečně dostávají čas CPU, prostor paměti a přístup k periferním zařízením. Jsou rozdělení na dvě podmnožiny, Q1 a Q2.

Q1 obsahuje uživatele, kteří spotřebovali méně než určité množství času CPU (kolem 100ms) od okamžiku, kdy opustili klidový stav. Q2 obsahuje všechny ostatní. Podobně množina kandidátů je rozdělena na dvě podmnožiny E1 a E2. Členové E1 čekají na vstup do Q1, členové E2 na vstup do Q2.

Systém odhaduje, resp. předpovídá velikost paměti, kterou bude uživatel potřebovat pro účinné zpracování. Uživatel je zařazen do Q1 nebo Q2 pouze tehdy, je-li požadované množství paměti dostupné, přičemž členové z E1 mají přednost před členy z E2.

VM/370 plánovač zařazuje členy z E2 do zpracování podle disciplíny FIFO. Tato disciplína není vždy nejvýhodnější, protože velcí uživatelé systému pak mohou omezovat úspěšnou práci malých uživatelů.

IBM nabízí jiný plánovač, s tzv. strategií slušného podílu, tj. všem CPU orientovaných uživatelům se dostane stejné množství času CPU za jednotku času reálného. Práci plánovače můžeme ovlivnit také změnou čtyř jeho konstant, které zastávají funkci určitých vah při výběru kandidáta do zpracování.

Kandidát je na zpracování vybirán zhruba podle čtyř kritérií: podle priority, která je uvedena v definici příslušného virtuálního stroje, podle jeho požadavku na reálnou paměť, podle okamžiku jeho příchodu do některé z podmnožin kandidátů a podle toho, zda-li byl naposledy hodnocen jako interaktivní uživatel či nikoliv. Změnou výše uvedených konstant se dá ovlivnit důležitost popsaných kritérií.

Priorita virtuálního stroje, uvedená v jeho definici, se prosadí velmi nevýrazně, jsou-li požadavky tohoto stroje nějakým způsobem mimořádné a to i tehdy, jestliže jí při-

soudíme relativně velkou váhu.

Požadavek na reálnou paměť, jehož relativní důležitost můžeme v plánovači také modifikovat, nepřímo určuje frekvenci, se kterou se virtuální stroj s tímto požadavkem dostává k reálným prostředkům systému.

Další faktor v plánovači může ovlivnit pořadí virtuálního stroje právě zařazovaného do některé z množin kandidátů E1 nebo E2.

Posledním faktorem modifikovatelným v plánovači můžeme zvýhodnit nebo potlačit interaktivní uživatele systému vzhledem k ostatním.

Zásadní problém související s modifikací plánovače výše popsaným způsobem tkví ve skutečnosti, že zatížení celého systému se neustále mění. Tedy hodnoty vhodné včera nejsou již nejlepší dnes. Bylo by nutno zabudovat monitor zatížení systému, který by podle okamžité situace nastavil optimální strategii plánovače virtuálních strojů. Avšak to je pro uživatele problém příliš složitý, kromě toho i firma IBM jej již vyřešila a nabízí nový plánovač jako podstatné zdokonalení systému VM/370.

Instalace, která provozuje VM/370, však může přispět k jeho vyššímu výkonu i jinak. Stejně jako u OS/VSI i zde hraje důležitou roli rozvážení logicky spolu souvisejících oblastí na disku. Z hlediska stránkovacího algoritmu VM/370 je nejvýhodnější umístění oblastí pro stránkování uprostřed reálného média (pro IBM 3330 je střed 202. válec ze 404 možných). Je výhodné definovat oblasti pro stránkování na více médiích a to vždy uprostřed. Stejně tak je důležité umístění vysoce aktivních miniisků (CMS systém, programovací jazyky, kompilátory apod.) pohromadě, čímž dosáhneme minimalizace pohybu výběrového mechanismu na disku.

Každý virtuální kanál a virtuální řídicí jednotka vyžadují pro svůj popis 40 bytů, proto je nutné omezit definici virtuálního stroje na co nejmenší počet kanálů a řídicích jednotek.

Na adrese DMKSYSMA v řídicí sekci DMKSYS v nukleu VM/370 můžeme uvést maximální počet uživatelů, kterým dovolíme přístup k systému a takto omezíme počet současně pracujících virtuálních strojů.

Časový úsek, po který má virtuální stroj přidělenou reálnou CPU jednotku, je definován na adrese DMKDSPQS v řídicí sekci DMKESP takéž v nukleu systému VM/370. Jeho zvětšením se zredukuje čas, který CPU potřebuje pro systémové funkce.

Kromě uvedených návodů k potenciálnímu zvýšení výkonu instalace pracující s VM/370 existuje řada dalších jednoduchých modifikací. Všechny tyto modifikace lze na rozdíl od modifikací OS/VSI provést velmi jednoduše. Otázkou však je vyhodnocení vlivu provedených změn na celkový chod systému.

VM/370 poskytuje silný nástroj pro sledování své práce a to příkaz MONITOR. Veškerá důležitá data o práci reálného systému jsou automaticky ukládána na páskové médium nebo jako diskový soubor. Instalace tak dostane do rukou ohromné množství údajů o chodu celého systému. Nastává nejobtížnější fáze a to je nezbytná redukce těchto dat do podoby vhodné pro rozhodování, protože vyžaduje znalost interní logiky VM/370 a je časově dosti náročná.

V ÚVT SPK jsme zatím ve fázi pochopení činnosti a struktury systému VM/370, v některých případech jsme se pokusili o redukci dat z pásky, vytvořené příkazem MONITOR, např. pro analýzu SEEK operací na discích a z toho vyplývající reorganizaci minidisků na reálných médiích.

Závěrem lze říci, že tato práce je značně náročná jak časově tak odborně, avšak přináší dané instalaci vyšší využití výpočetní techniky a větší spokojenost uživatelů.