# Střediskové počítače a technologie klient-server

Už v roce 1985 dochází v bývalém Československu k revoluci. Nejednalo se však o revoluci politickou, ale o revoluci ve světě počítačů. Do té doby se drtivá většina informací o počítačích a jejich aplikacích diskutovala pouze v odborných kruzích. Informační projekty typu evidenčních a řídicích aplikací se provozovaly na střediskových počítačích a minipočítačích. Většina z nás pamatuje z té doby rozšířenou řadu IBM 360/370 známější pod zkratkou JSEP. Mnoho z těchto sálových strojů například EC 10xx pracuje dodnes a slouží pro centrální provoz výrobních i nevýrobních projektů zpracování dat v různých organizacích. Vzpomínám-li rok 1985, pak právě v tomto roce byla uvolněna informační a obchodní bariéra a umožněn průnik novodobé počítačové technice. Masový rozvoj osobních počítačů PC s procesorem Intel, které jsou vybaveny jednoduchým DOSem se projevil i u nás. Nebylo z počátku laciné ani jednoduché pécéčka získat, ale postupně se ovládání tohoto počítače stává druhou gramotností. Celkem bezproblémová byla i volba vývojového a provozního prostředí. DOS i programátorské prostředky na něm jsou relativně laciné a hlavně dají se jednoduše kopírovat ze stroje na stroj.

Na tomto místě bych chtěl zdůraznit dva faktory, které u nás na dlouho ovlivnily pohled na dění kolem výpočetní techniky. V prvé řadě je to úloha software. Vznikají desítky firem, které tvoří poměrně jednoduché a samostatné aplikace v různorodých softwarových prostředcích. Ne vždy jde o nástroje odpovídající závažnosti použití aplikací. Je potřebné si uvědomit k čemu dochází. Například aplikace řízení základních prostředků, jejíž zpracování na středním počítači obsluhuje tým profesionálních pracovníků výpočetního střediska, je nyní svěřena plně do péče účetní ZP. Tato aplikace na PC proto musí obsahovat organizační, kontrolní i technické zabezpečení a navíc by měla být to, čemu říkáme "user friendly". Při tvorbě moderních aplikací tedy hrozí určité nebezpečí dané překotným rozšířením novodobých programovacích prostředků spolu s praktickou nezkušeností nové generace analytiků a programátorů. Poměrně často se setkávám s následujícím názorem : " Programování není náročnou kvalifikací, vždyť jej zvládají i děti na základní škole". Tento výrok může být zcela v pořádku, když ho pronese člověk, který si může dovolit být laikem. Osobní počítače zcela jistě mají svou vzdělávací a produkční roli v jednoduchých izolovaných úlohách. Takto lehkomyslně by však neměl uvažovat manažer úseku informatiky organizace, který zodpovídá za vytvoření integrovaného informačního systému řízení. Domnívám se, že se tak dnes velmi často děje.

Druhým aspektem pohledu na funkce výpočetní techniky je úroveň technického vybavení pro zpracování informačních projektů. Plánujemeli vytvoření rozsáhlého automatizovaného systému řízení podniku (dříve se používala zkratka ASŘP), zcela jistě nevystačíme se sítí počítačů PC a relační databází typu FOXPRO.

Dnes vedle sebe existují v počítačové oblasti dvě zdánlivě nesouvisející profese. Jednak je to relativně neveřejný svět aplikací a informačních systémů střediskových počítačů a na druhé straně na veřejnosti populární a velmi rozšířené programování v systémech typu DOS, v menší míře UNIX.

Smyslem článku je ukázat specifika a možnosti symbiózy obou těchto prostředí. Opět použiji k ilustraci tohoto vztahu příhodu, která se mi opakovaně stala na různých výstavách výpočetní techniky. Pokládal jsem následující otázku pracovníkům firem, které v minulosti významně ovlivňovaly dění ve zpracování rozsáhlých aplikací na střediskových počítačích. Jsou to známé podniky jako PVT, Datasystém, Kancelářské stroje. Na otázku "Je možno provést transformaci projektu řízení výroby na EC 1032, zpracovaném technologií hierarchické databáze IMS (IBM) v hostitelském prostředí jazyka COBOL ?" jsem dostával vzácně jednotnou odpověď. " K čemu by vám ta aplikace byla, přejděte na náš moderní projekt ve FOXPRO". Tato odpověď, podle mého názoru postrádá primární logiku. Totiž jenom dlouhodobý uživatel jakéhokoli výrobku může zodpovědně prohlásit zda je nebo není nahraditelný. Proto mě překvapuje názor bývalých

praktiků sálových počítačů, že je možno snadno nahradit rozsáhlý sálový projekt odpovídajícím řešením na PC.

**Technologie "CROSS-PLATFORM"**

V odborné terminologii se dnes prolínání různých výpočetních platforem označuje pojmem "Cross-platform". O co jde ?
S masovým rozšířením počítačů ve všech oborech vzniká potřeba výběru hardware, operačních systémů a systémového software vzhledem k požadavkům aplikace. Provedená volba musí předjímat i předpoklady budoucnosti. Může dojít k opodstatněným požadavkům na zvýšení objemu zpracovávaných dat, počtu uživatelů v síti, změnu komunikačního prostředí, hardware a podobně. Různá prostředí jako DOS, Windows (NT), OS/2, UNIX, AIX, síťové systémy i systémy počítačů typu HOST umožňují dosáhnout různé úrovně parametru cena/výkon a různou úroveň produktivity využívání aplikací koncovými uživateli. Chybný výběr a investice do nesprávného operačního prostředí nebo technologie mohou znamenat roky zničené práce a někdy i ohrožení chodu celé organizace. Strategie "cross-platform" je nutná pro izolování aplikací od změn výpočetní základny. Musí umožnit programátorům a analytikům co nejvhodněji využívat výhody specifických vlastností každé HW platformy a každého operačního systému na nichž pracují. Odpovídající aplikační řešení musí splňovat tato specifická kriteria :

FLEXIBILITA
aplikaci lze provozovat na každé platformě bez tvorby a aktualizace různých verzí pro danou platformu

INTEROPERABILITA
možnost doplnění nových funkcí a technologií software do systému aplikace s její minimální přestavbou

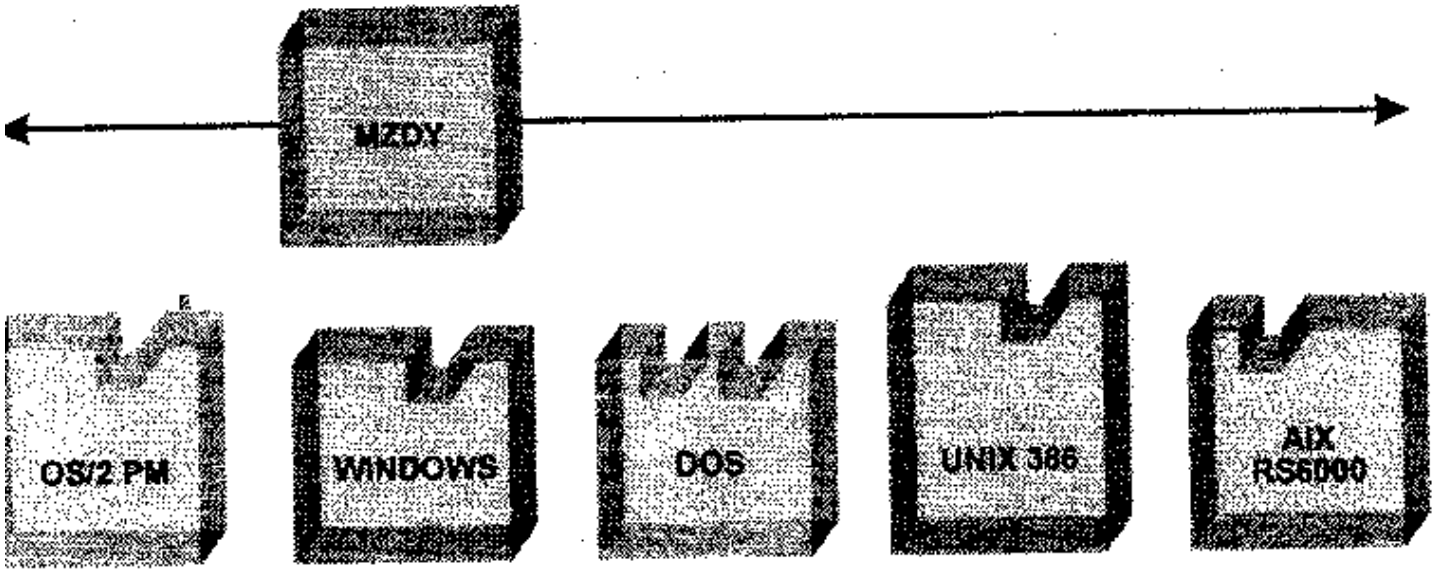
REUSABILITA
ovlivňování nákladů na provoz aplikace a zvyšování její produktivity portací na výhodnější platformu

RELIABILITA
možnost volby stabilního provozního prostředí odpovídající výkonnosti

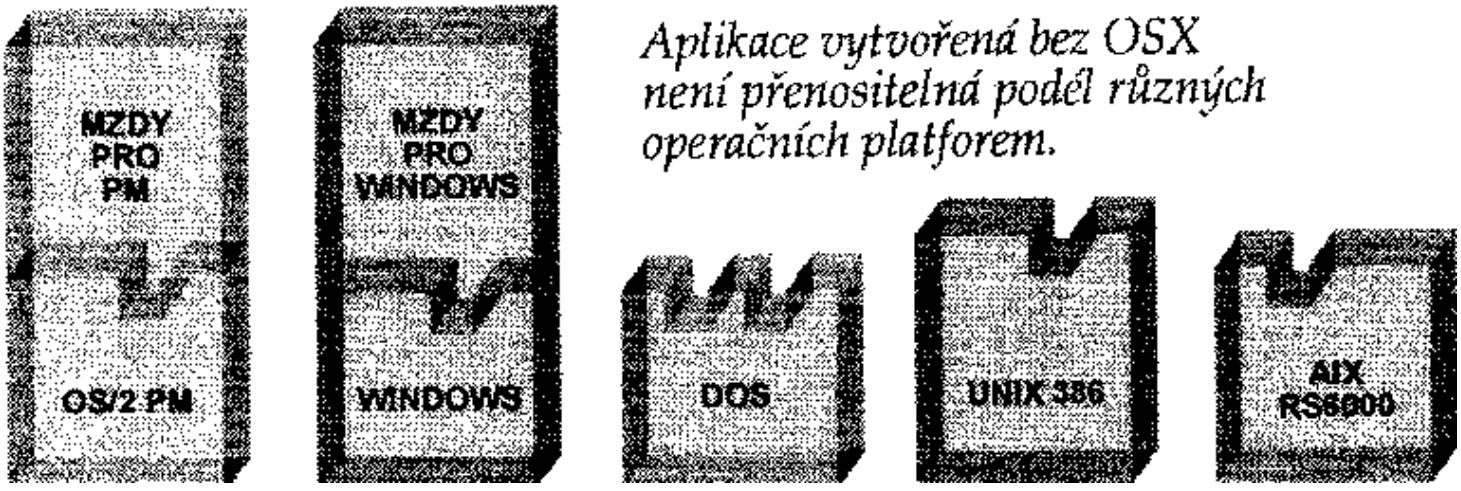
PORTABILITA
přesun aplikace z platformy na jinou platformu s minimálním úsilím a finančními náklady

Nyní je vhodné uvést příklad technologie, která uvedená kriteria splňuje. Začnu možná poměrně šokujícím způsobem. K demonstraci příkladu použiji COBOL. Ano, troufám si tvrdit, že COBOL je stále ještě nejvhodnější jazyk pro reprezentaci modelu datového zpracování. Bohužel, toto tvrzení je v České republice poměrně ojedinělým názorem. Dost odlišná je situace v západních zemích, kde COBOL patří standardně do rodiny programátorských a provozních prostředků. Jako důkaz uvádím pravidelné listování renomovaným časopisem DATAMATION, kde například v rozsáhlé inzertní části patří "cobolská" kvalifikace k nejčastějším. COBOL splňuje dva základní požadavky modelu datového zpracování. Má standardní podporu elementárního programového rozhraní API pro programování na nejnižší úrovni a zároveň dostatek prostředků pro vysokou produktivitu vývoje a provozu aplikací. V provedení Micro Focus Cobol se tento prostředek stal průmyslovým standardem, protože kombinuje robustnost s vlastnostmi potřebnými pro migraci aplikací mezi HW a SW platformami a operačními

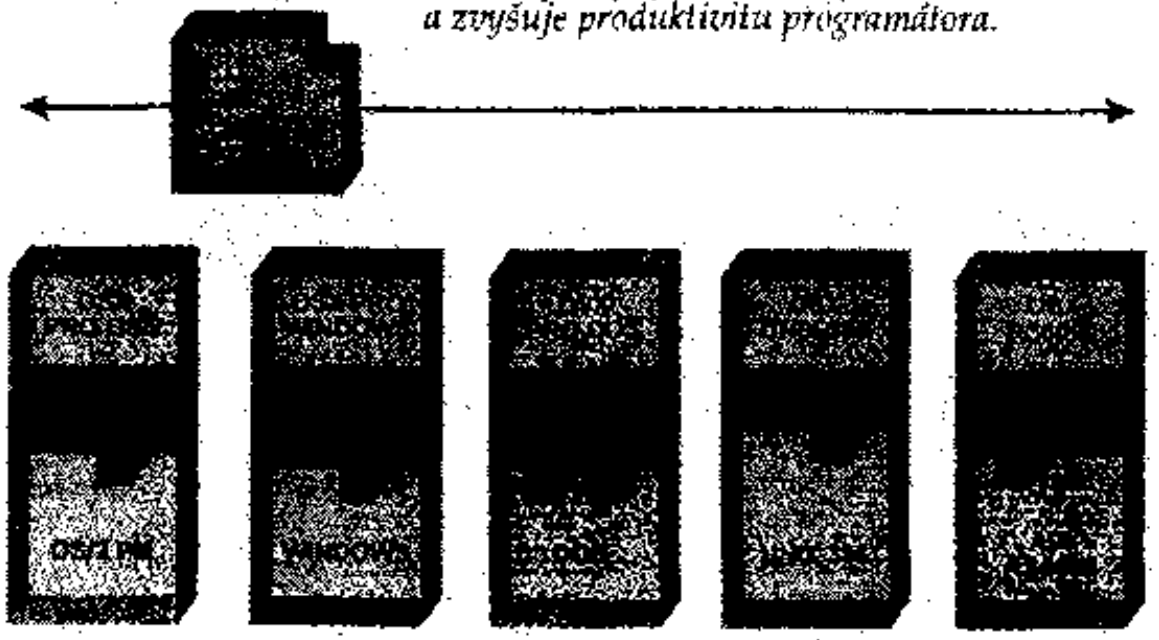
systémy. COBOL byl od počátku devadesátých let zcela rekonstruován. Podporuje standardní operační systémy DOS, Windows (NT), OS/2, UNIX, AIX i MVS. Aplikace lze vyvíjet v 16 ti i 32 bitovém kódu a provozovat na stovkách různých platforem hardware od střediskových přes mini až k osobním počítačům. Podporuje tvorbu grafických aplikací. Obsahuje vývojové a provozní 32 bitové objektově orientované grafické prostředí pro tvorbu uživatelského rozhraní srovnatelné s produkty jako Visual Basic nebo PowerBuilder. Cobolská aplikace včetně grafického rozhraní může vzniknout v libovolném operačním systému a být portována na DOS, Windows, OS/2(PM) a Motif. Aplikace COBOLu provozované na počítačích typu HOST (mainframe, midrange, mini) lze přenést do prostředí těchto systémů, heterogenních datových sítí a technologií klient-server. Server může být zastoupen systémy typu MVS, UNIX, AIX, Windows NT nebo OS/2, na pracovních stanicích můžeme kromě toho používat i DOS. Výše uvedená flexibilita je umožněna technologií OSX (Operating System Extension), která se stane součástí aplikace. OSX poskytuje aplikaci standardní rozhraní a tedy i nezávislost vůči specifickým operačním systémům. Stačí pouze doplnit programový balík knihovnami OSX pro požadovaný operační systém




*Ideální aplikace by měla být přenositelná podél různých operačních systémů.*




*Aplikace vytvořená bez OSX není přenositelná podél různých operačních platforem.*




Pojem portabilita neznamená pouze možnost přenosu aplikace ve formě zdrojového kodu. Pro tvorbu aplikace je zvolen optimální vývojový operační systém a provozní forma aplikace se pouze kopíruje na ostatní operační systémy.

V oblasti organizací dat jsou standardně podporovány přirozené sekvenční i indexové organizace (QSAM, ISAM) COBOL je však schopen ovládat hierarchickou databázi IMS i relační databáze různých výrobců. S požadavky kladenými na datové zpracování souvisí pojem interoperabilita. Jde o klíčový požadavek řešení technologie "cross-platform". Aplikace často vyžadují při inovacích vkládání nových technologií. V případě cobolské souborové orientace je vhodné použít produkt FILESHARE pro sdílení datových souborů v sítích. FILESHARE poskytuje rovněž prostředky k zajištění datové integrity jako je zálohování změn, obnovování kolekcí souborů a transakční zpracování s dvoufázovým režimem COMMIT. Výsledkem je nejen zajištění bezpečnosti dat, ale také snížení objemu přenášených dat v síti a tím i zvýšení výkonu aplikací. Pro aplikace typu IMS jsou k dispozici emulace jak databázových, tak i komunikačních služeb spolu s výhodami použití FILESHARE.

Přístup k relačním databázím je zajištěn v řadě databázových produktů různých operačních systémů. Lze provádět migraci existujících systémů, které užívají souborové organizace VSAM, ISAM do relačního modelu. Použitím Micro Focus produktu COBOL/SQL Transparency System můžeme dosáhnout následujícího efektu. Zdrojový program COBOLu neměníme, výsledný spustitelný modul aplikace však ovládá místo souborové organizace prostřednictvím cobolských vstupně výstupních příkazů databázovou relační strukturu. Při běhu aplikace je každý cobolský příklad transformován do příkazu SQL.

Podporované datové struktury

| | |
|---|---|
| . Gupta | . Sybase |
| . Informix | . DB2/2 |
| . Microsoft SQL Server | . DL/I |
| . Ingres | . DB2 |
| . Oracle | . ISAM/VSAM |
| . OS/2 DB Manager | . Btrieve |
| . Unify | . SQL/DS |
| . XDB | . OS/400 |

Svoboda volby a nezávislost na provozním prostředí jsou základní výhody řešení "cross-platform". COBOL a další Micro Focus produkty umožňují kooperovat aplikacím s aplikacemi v jiných jazycích, s jinými komunikačními protokoly a s jinými databázovými systémy. Tyto technologie chrání investice uživatelů tím, že umožňují maximální možnou "znovupoužitelnost" existujících aplikací. S vysokou podporou syntaxe jazyka včetně emulace kompilátoru mainframe IBM mohou být střediskové aplikace transformovány bez značných investic do rekvalifikace vývojových pracovníků respektive bez nutnosti přeprogramování.

**Redukce nákladů na vývoj a zvyšování produktivity**

Vedoucí pracovníci informačních systémů jsou nuceni prosazovat, snižování nákladů a zvyšování produktivity vývoje a údržby aplikací. Současně je nutno zvažovat integraci nových technologií v hardware i software do existujících informačních a řídících systémů. Odpovídajícím řešením tohoto problému může být změna role střediskového počítače na produktivní aplikační server. Veškerý vývoj aplikací a jejich údržbu je nutno přesunout na rychlé, efektivní pracovní stanice. Získáme tím snížení nákladů na provoz počítačů HOST a zvýšíme produktivitu práce programátora. Pro maximalizaci finančního přínosu vyžaduje nové vývojové prostředí produktivní stanice PC, které svou konfigurací odpovídají náročnosti vyvíjené aplikace. Programátor pro zvýšení efektivity své práce však musí mít k dispozici nejen portabilní prostředí, ale především produktivní nástroje software odpovídající technologii osobních počítačů. Klíč v realizaci těchto cílů je ve využití existujících znalostí a schopností. Kompatibilita s provozním prostředím mainframe je nutností, a to jak na zdrojové, tak i na datové úrovni. Tento proces přenesení vývoje aplikací z počítačů typu HOST na pracovní stanice označujeme termínem "Offloading".

**Micro Focus technologie OFFLOADING**

Dosud uvedené technologie transformace aplikací střediskových počítačů jsou hlavní aktivitou firmy Micro Focus. Řešení Offloading této firmy umožňuje programátorům používat pracovní stanice PC pro editování, kompilaci, testování a ladění jejich programů, které potom budou na mainframe pouze provozovány. Micro Focus nabízí škálu produktivních nástrojů neexistujících na střediskových počítačích, které podstatně zkvalitňují a urychlují vývoj a údržbu aplikací. Toto řešení spojuje kompatibilitu mainframe, produktivitu PC a požadavek na integraci vývojového prostředí. Programy vytvořené na mainframe v Cobolu, PL/I, CICS, IMS, DB2, SQL/DS, JCL, Assembleru 370 a dalších prostředcích mohou být přeneseny na PC nebo na něm vyvinuty a dále aktualizovány před opětným návratem k provozu na HOST počítači.

**Micro Focus Workbench**

Micro Focus Workbench je ústředním produktem řešení Offloading. Spolu s dalšími produkty podporuje všechny výše uvedené technologie mainframe. Patří sem i nástroje pro přenos souborů, ladicí nástroje úrovně klient-server, prostředky statické i dynamické analýzy programů aovlivňování výkonu aplikace.Workbench poskytuje uživatelské rozhraní, jak ve znakové, tak i v grafické formě a dovoluje používání dat v kódu ASCII nebo EBCDIC.

**ADMVS**

Micro Focus ADMVS je prostředí pro vývoj aplikací, které jsou určeny k provozování pod systémy MVS. Používá nástroje pracující v operačních systémech DOS a OS/2, které jsou důvěrně známy vývojářům MVS. Jsou zde integrovány možnosti používání kompilátorů a animátorů spolu s SPF editorem, provozní funkce MVS JCL a PROC, běžné utility MVS a emulátor ISPF Dialog Manager.

**Aplikace technologií klient-server**

Offloading směrem k pracovním stanicím může být prvním krokem k migraci programátorů a aplikací do nových technologií. Vývojáři poznávají a začínají používat charakteristiky nových prostředí - grafického uživatelského rozhraní (GUI), lokálních sítí nebo kooperativního zpracování. Kvalita aplikací je spojenou nádobou se schopností programátora porozumět programu a důsledkům navrhovaných změn. Micro Focus prostředky dávají programátorům možnosti více pohledů na program nebo aplikaci a vizuálního porozumění probíhajícího algoritmu.

**Online Transaction Processing OLTP**

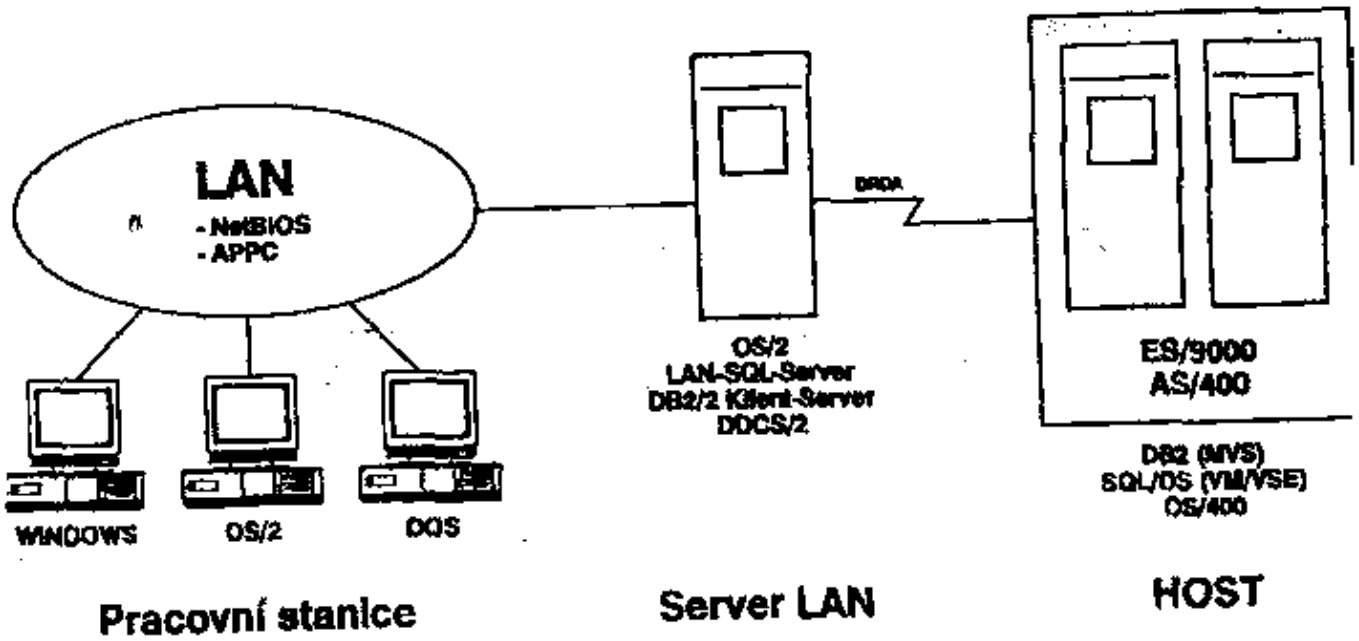
Micro Focus OLTP je základnou pro mnoho druhů aplikací klient-server. Centrálním produktem je MF CICS Option včetně Transakčního systému. Jde o víceuživatelský a víceúlohový OLTP systém, který je k dispozici na AIX, SCO, DOS, Windows, OS/2, Windows NT a jiných unixových platformách. Pomocí tohoto systému je možno provádět downsizing aplikací CICS, přenos provozu těchto aplikací do heterogenních datových sítí. Transakční systém zaručuje kompatibilitu s IBM CICS tak, že existující host aplikace mohou být produktivněji vyvíjeny na optimální platformě. Tento systém integruje prostředky GUI a externí aplikace, které zpřístupňují funkce a data OLTP. Komunikace s mainframe systémy CICS je v současnosti podporována za použití IBM OS/2 Communication Manager v OS/2 nebo IBM Personal Communications/3270 v DOS nebo Windows. Micro Focus řešení poskytuje snadné rozhraní s OLTP standardy jako Encina, Tuxedo a průmyslovými standardy API jako DCE (Distributed Computing Environment).

**Transformace aplikací 3270 do klient-server**

Micro Focus produkt Dialog System 3270 umožňuje import CICS (BMS) nebo IMS (MFS) obrazovek a vytvoření grafických obrazovek v systémech Windows nebo OS/2. Vývojář potom může tyto obrazovky zdokonalovat ve vývojovém prostředí produktu Dialog System Profesional, grafickém 32 bitovém objektově orientovaném prostředí. Lze přitom doplňovat další objekty jako jsou událostmi řízené procesy a datové přístupy. V provozu DS3270 poskytuje podporu komunikace vytvořeného front-end rozhraní s původní aplikací HOST počítače.

**Micro Focus DB2 Option**

MF DB2 Option je rozsáhlá kolekce nástrojů pro vývoj a testování aplikací SQL na platformě PC a jejich provozování na DB2 a SQL/DS platformách PC, AS/400 nebo střediskových počítačů. Jedná se o produkt vytvořený spoluprací firem Micro Focus a IBM. Je dodáván v provedení single-user pro jednotlivé pracovní stanice nebo multi-user pro pracovní stanice OS/2 nebo Windows včetně funkce brány (gateway) k databázím DB2, SQL/DS nebo DRDA na IBM/390 a AS/400. MF DB2 Option je tvořen relační databází IBM DB2/2 a produkty HCO (Host Compatibility Option) a CSDO (Cobol SQL Development Option). HCO zajišťuje kompatibilitu Host verzí DB2 a DB2/2. CSDO poskytuje možnost tvorby SQL dotazů na PC a on-line provozování na Host počítači.



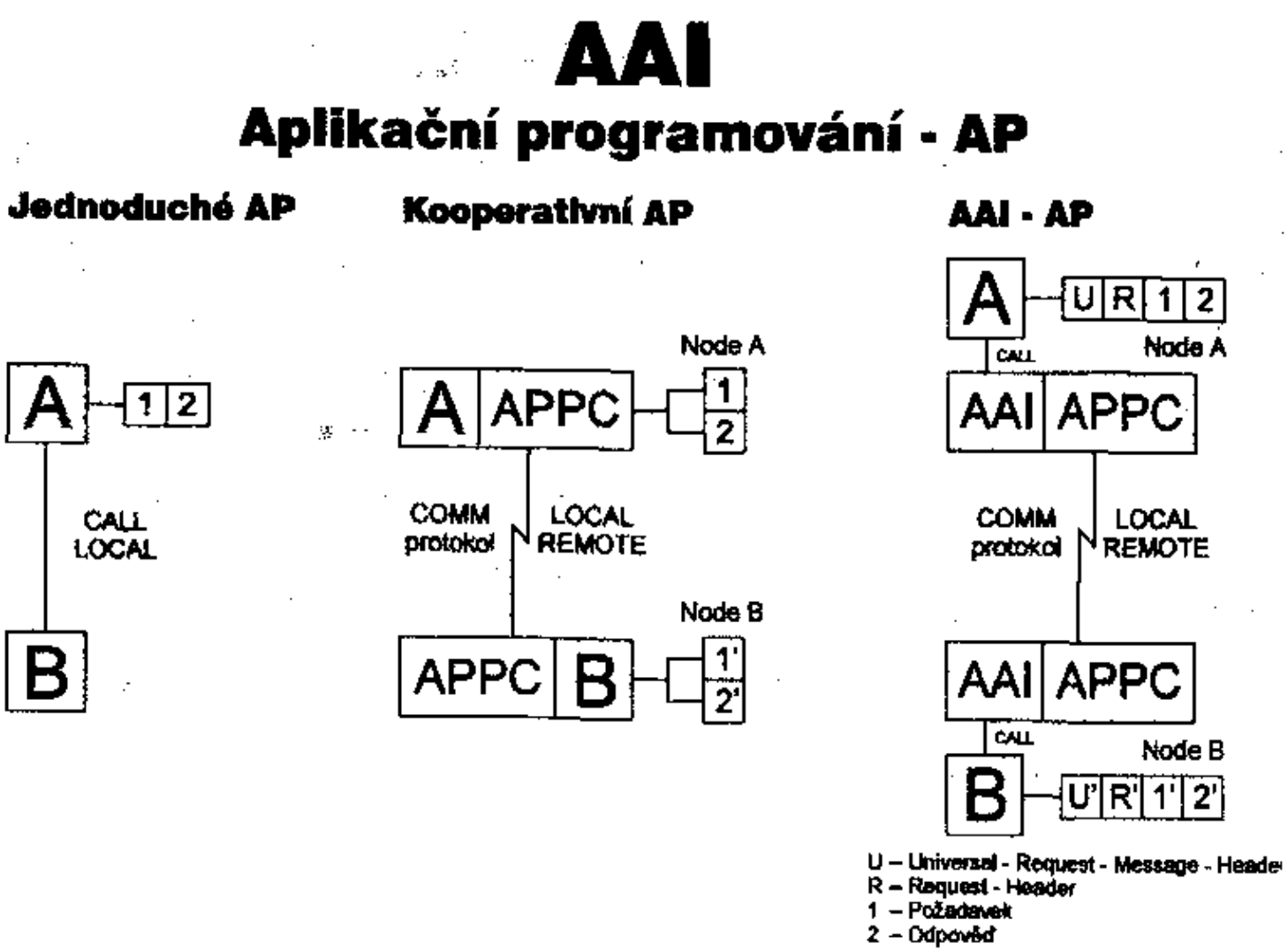

**Dialog System Profesional**

Dialog System Profesional je produktivní vývojový prostředek aplikací určený především pro vývojáře v Cobolu. Kombinuje v sobě výhody produktivity nástrojů 4GL a portability cobolských aplikací. Obsahuje designer grafického uživatelského rozhraní a interaktivní prostředek datového přístupu pro tvorbu dotazů buď do relační databáze nebo do standardních cobolských souborů. Na tomto nástroji je však unikátní skutečnost, že na místo použití ve specifické (proprietary) technologii generuje cobolskou aplikaci, která může být snadno

aktualizována a portována do různých provozních operačních systémů včetně jejich grafických prostředí Windows, OS/2, Motif.

**Aplication to Aplication Interface AAI**

Podstatnou vlastností správného řešení klient-server je schopnost efektivně propojovat různé platformy. Komunikační vlastnosti aplikace by měly být nezávislé na komunikačním protokolu a snadno konfigurovatelné. MF AAI poskytuje pravou peer-to-peer komunikaci (program-program) použitím standardní funkce "call". AAI má schopnost propojovat širokou škálu platforem včetně MVS, OS/400, AIX, UNIX, DOS, Windows a OS/2 buď lokálně nebo způsobem remote.




AAI umožňuje aplikačním programům rezidovat na kterémkoli z uvedených systémů, volat a provozovat aplikace na kterémkoli z ostatních. Libovolný systém může být klientem vyžadujícím služby a libovolný systém (mimo DOS) může být serverem, který požadované služby zajišťuje. Jednoduché programové rozhraní dovoluje psát aplikace klient / server, kdy programátor nepotřebuje mít speciální znalosti komunikačního prostředí. AAI podporuje více typů konverzace včetně rozhraní Remote Procedure Call (RPC), Store & Forward a Peer-to-Peer a umožňuje vývoj smíšených jazykových aplikací.