



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

ARCHITEKTURA SOUČASNÝCH JADER KONVENČNÍCH OS

CURRENT CONVENTIONAL OPERATING SYSTEMS ARCHITECTURE

PŘÍPADOVÁ STUDIE

CASE STUDY

AUTOR PRÁCE

AUTHOR

Bc. JAROSLAV BARTOŇ

BRNO 2008

Obsah

1	Současné konvenční operační systémy	2
2	GNU/Linux	3
2.1	Stručná historie	3
2.2	Vývoj jednotlivých řad jádra Linuxu	3
2.3	Architektura Linuxového jádra	5
2.4	RT-Linux strom (patch)	6
3	FreeBSD	8
3.1	Stručná historie	8
3.2	Vývoj jednotlivých verzí FreeBSD	8
3.3	Architektura jádra FreeBSD	10
4	Microsoft Windows	12
4.1	Veze Microsoft Windows	12
4.2	Historie Windows NT	12
4.3	Architektura jádra Windows NT	14
4.4	Real time rozšíření Windows NT	15
4.4.1	Komentář	15
4.4.2	Možná řešení	16

Kapitola 1

Současné konvenční operační systémy

Konvenční operační systémy je třeba rozdělit podle architektury procesoru na které běží. Mezi nejběžnější architekturu patří procesory odvozené od Intel x86, dále pak procesory ARM, PowerPC, MIPS či například procesor Cell (rozšířen v herní konzoli Playstation 3). V zabudovaných zařízeních pak mohou být rozšířené zcela jiné procesory a na nich běžící operační systémy.

Na procesorech Intel x86 v současné době mezi konvenční systémy patří převážně operační systémy firmy Microsoft. Microsoft, ale není jediný dodavatel operačních systémů pro počítače založené na těchto procesorech. Mezi další operační systémy, které lze označit jako konvenční lze zařadit GNU/Linux a různé deriváty BSD (převážně FreeBSD, MacOS X).

Pro jiné architektury procesorů je zastoupení operačních systémů různé, Microsoft zástupce nemusí mít a místo něj je na dané třídě nejběžnější například VxWorks či Symbian. Rozšíření těchto odlišných architektur procesorů a operačních systému si uživatelé ani neuvědomují, jsou skryty například v telefonu, televizoru, multimediálním centru.

V dalším textu se zaměřím na operační systémy běžící na počítačích s architekturou Intel x86. Konkrétně se podívám GNU/Linux, FreeBSD a Windows NT.

Kapitola 2

GNU/Linux

2.1 Stručná historie

Historie operačního systému GNU/Linux začíná přibližně v roce 1992, kdy Linus Torvalds poslal do konference zabývající se operačním systémem Minix [15] informaci o tom, že pracuje na svobodném operačním systému pro počítače AT 386, 486 a jejich klony. Také se zmiňuje o tom, že systém není zcela portabilní, protože využívá specifické vlastnosti architektury procesorů třídy x86.

Linus se základní principy operačních systémů naučil na systému Minix, který byl dostupný i se zdrojovými kódy pro každého kdo si koupil knihu *Operating Systems: Design and Implementation* od Andrew S. Tanenbaum [12].

Postupně vyvinul emulátor terminálu, ovladač pevného disku, a souborový systém podporující ukládání souborů. První dostupné verze vzbudily velké vzrušení u internetové komunity. Hlavním důvodem tohoto vzrušení z jinak nepoužitelného systému byla jeho nezávislost na kódu Minixu.

Postupně se přidávali vývojáři a díky nim se Linux rychle dostal do stadia, kdy byl provozuschopný. K jádru se přidaly nástroje z projektu GNU Richarda Stallmana [11], jako například shell bash, překladač gcc, či proudový editor sed. Bash a gcc na Linux přeportoval sám Linus s využitím předchozích portů na Minix. Pro překlad systému byl třeba systém Minix.

Od verze 0.11 je pak Linux nezávislý na Minixu, obsahuje ovladače nejrůznějších zařízení a podporuje virtuální paměť, sdílení paměti procesy a další. Po spuštění ale nabíhal rovnou do shellu, žádné přihlášení do systému zatím nebylo možné.

Ve verzi 0.12 byla změněna licence na GNU GPL, jež bývá někdy označována jako copyleftová licence [6]. Linux tím zaujal další lidi, kteří se původně k systému stavěli skeptičtěji.

V dalších verzích se podařilo Linux naportovat na další architektury procesorů a tak k rodné architektuře Intel x86 přibily architektury jako Alpha, PowePC, MIPS, Blackfin, ARC, ARM, Atmel AVR32, PA-RISC, IA-64, SPARC a jiné [19].

Více o dalším vývoji Linuxu se můžete dočíst na stránce *Historie Linuxu pěkně od začátku* [23], odkud jsem čerpal.

2.2 Vývoj jednotlivých řad jádra Linuxu

V roce 1994 vyšla první stabilní verze 1.0 s následujícími vlastnostmi:

- podpora TCP/IP sítí,

- síťové programové rozhraní kompatibilní s BSD sokety,
- ovladače síťových karet,
- podpora SCSI rozhraní,
- přepracovaný souborový systém.

V roce 1996 pak následovala verze 2.0, která přinesla podporu dalších platforem, víceprocesorových strojů a další:

- vylepšená správa paměti,
- výkonnější implementace síťových protokolů TCP/IP,
- vlákna na úrovni jádra,
- podpora víceprocesorových strojů (SMP),
- podpora modulů,
- standardizované konfigurační nástroje,
- firewall ipf (a nástroje ipfwadm).

Následující stabilní verze 2.2 vyšla v roce 1999 po nedodržení původního termínu prázdniny roku 1998:

- plnou podporu pro architektury x86 (Intel), Sparc, Ultra, Alpha, m68k, PowerPC,
- vylepšenou podporu CD-ROM a výměnných médií, zvukových a grafických karet,
- podporu tzv. frame bufferu,
- bohatší zastoupení souborových systémů, např. Amiga FFS, Macintosh HFS, UFS, CODA, NTFS (pouze čtení) a jiné,
- dokonalejší návrh podle standardu Unix98,
- podporu názvů v různých kódováních UNICODE,
- firewall ipchains.

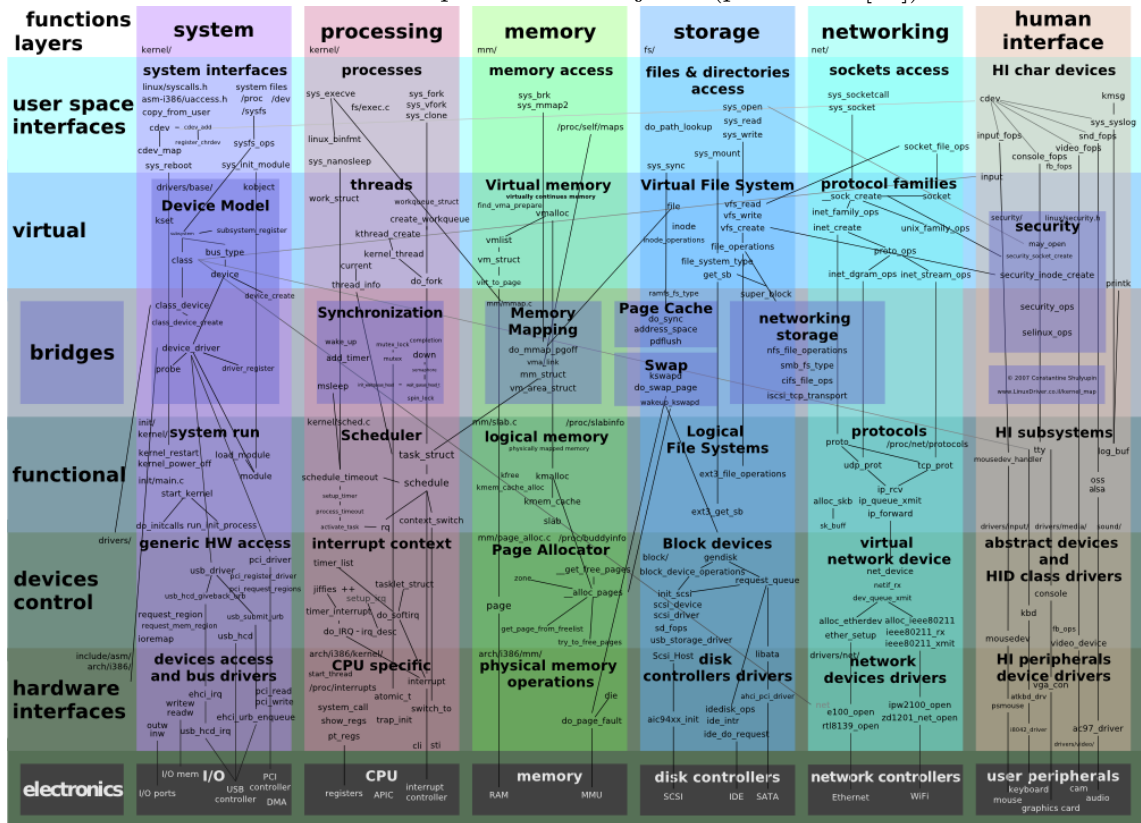
Předposlední stabilní verze 2.4 vyšla v roce 2001:

- podpora 64GB operační paměti na intelovských strojích,
- efektivnější multiprocessing a multithreading,
- lepší podpora USB a IEEE 1394 (FireWire),
- souborový systém Reiser FS s podporou žurnálování,
- firewall iptables.

U poslední stabilní řady jádra 2.6, která vyšla v roce 2003, byl změněn způsob vývoje a nové vlastnosti jsou do jádra přidávány s každou podverzí namísto vývoje ve vývojových řadách jádra (vývojové řady mají liché číslo podverze – 2.1, 2.3, 2.5) a tak vypsát všechny přidávané vlastnosti by bylo nad rámec tohoto textu.

2.3 Architektura Linuxového jádra

Obrázek 2.1: Mapa Linuxového jádra (převzato z [18])



Linuxové jádro je monolitické s možností nahrávat moduly do jádra. Ovladače mohou být součástí jádra nebo jako moduly. Jádro a ovladače na většině architektur běží v takzvaném ring 0 – mají plný přístup k zařízením. Existují ale i ovladače v uživatelském prostoru (například tiskárny, skenery a jiné). Grafický uživatelský systém není součástí jádra, běží jako proces v uživatelském prostoru.

Je podporována preempece v režimu jádra aby ovladače mohly být při určitých podmínkách preemtovány. To je potřeba ke zpracování přerušení a lepší podpoře víceprocesorových systémů. Preempece také zlepšuje latence, odezvy a tím umožňuje nasazení Linuxu v real time aplikacích [18].

I přes tyto vlastnosti je zde problém s tím, že jádro Linuxu nebylo původně navrženo pro víceprocesorové systémy. Při úpravách pro podporu více procesorů byl do jádra zaveden velký jaderný zámek (big kernel lock). Ten zajišťoval, že kód kernelu poběží současně pouze na jednom procesoru [3]. Ostatní procesory, které chtějí vykonávat kód jádra tak musí počkat, až jiný procesor kód jádra opustí.

Probíhají rozsáhlé práce na odstranění tohoto velkého jaderného zámku a jeho nahrazení zámky subsystémovými, takže kód jádra se bude moci vykonávat na více procesorech pokud se jednotlivé části nebudou ovlivňovat. Velký jaderný zámek se již pomalu stává minulostí, přesto však v jádru jádra (core kernel) existuje přibližně 400 míst, kde lze nalézt volání `lock_kernel()` a dalších cca 800 míst, kde se používá `ioctl`. Tyto volání jsou často umístěna v 15 let starém kódu a je třeba špičkových odborníků k jejich zmapování a odstranění [2].

Dalším milníkem k real time Linuxu je obsluha přerušení ve vláknech, která v současnosti žije v real time stromě (mimo hlavní vývojové místo jádra). Real time vývojáři se vrací k přemýšlení o začlenění zbývajících real time kódu. Real time strom je nastaven tak, aby byla téměř všechna přerušení obsluhována ve vláknech, ale to by v hlavní řadě nefungovalo. Některá zařízení dále poběží se synchronním zpracováváním přerušení a nápad řešit softwarová přerušení ve vláknech se nelíbí vývojářům síťování. Bylo tedy navrženo poskytnout novou verzi `request_irq()` případně nové volání místo modifikace stávajícího, která by ovladači umožnila nastavit obsluhu přerušení ve vláknech. Pokud by správce ovladače žádnou změnu neprovedl, obsluha přerušení by nadále běžela synchronně [1].

Další velkou změnou, která zlepšuje real time vlastnosti je přechod na plánovač procesů CFS. CFS má několik zajímavých aspektů. Především zcela odstraňuje pole `front`. Místo toho CFS pracuje s jediným red-black stromem, pomocí kterého sleduje všechny procesy ve spustitelném stavu. Proces, který se objeví na uzlu stromu nejvíce vlevo, je ten, který má největší šanci běžet v kteroukoliv dobu. Takže základem k porozumění tomuto plánovači je získat představu o tom, jak vypočítává klíčovou hodnotu používanou pro vložení procesu do stromu. CFS navíc přináší možnost modulárních plánovačů. Jde o jednoduchý spojový seznam modulů seřazených podle priority; první plánovací modul, který dokáže najít spustitelný proces, se může rozhodnout, kdo přijde na řadu jako další. V současné době jsou k dispozici dva moduly: CFS plánovač a zjednodušená verze real time plánovače. Real-time plánovač je v seznamu na prvním místě, takže všechny real time úlohy poběží před ostatními procesy [7].

Nastavení jádra je možné při startu systému předáním parametrů jádru. To je jedna z úloh zavaděče systému. V Linuxu je jako zavaděč používán program LILO nebo také GRUB (na procesorech Intel x86 kompatibilních). Jádro lze také nastavovat pomocí souborů umístěných v adresářích `/sys` a `/proc`. Tyto adresáře jsou virtuální, plně spravované jádrem. Odpovídají aktuální konfiguraci jádra, zavedeným modulům, spuštěným procesům. Dále lze jádro nastavit pomocí `sysctl()` operací.

2.4 RT-Linux strom (patch)

Real time vlastnosti, které nejsou součástí jádra je možné získat jako patch či v real time stromě jádra.

Mezi vlastnosti, které tento patch (strom jádra) přidává patří [10]:

- převádí jaderné zámky (spinlock) na preemptivní pomocí jejich reimplementace na `rtmutexy`,
- kritické sekce chráněné `spinlock_t` a `rwlock_t` jsou převedeny na preemptivní,
- nepreemptivní kritické sekce jsou možné za pomoci `raw_spinlock_t`,
- implementuje dědění priorit pro jaderné zámky a semaforey,
- převádí zpracování přerušení z handlerů na jaderné vlákna,

- odděluje staré API časovačů od časovačů s vysokým rozlišením, což vede k uživatelskému prostoru s POSIX časovači s vysokým rozlišením

Kapitola 3

FreeBSD

3.1 Stručná historie

Ve srovnání s moderními operačními systémy nebyl původní UNIX příliš dobrý. Ale protože spousta studentů měla přístup k jeho zdrojovým kódům a učitelé potřebovali pro své studenty projekty, byl jejich společným úsilím UNIX vylepšován. Postupně vznikaly užitečné programy, byla přidána schopnost ovládat běžící programy (jobs control), objevil se souborový systém s podporou funkcí, které dnes považujeme za samozřejmé. Po mnoho let byly kusy operačního systému UNIX vylepšovány a nahrazovány.

Různé univerzity, které na UNIXu pracovaly, sdílely svá vylepšení a zdokonalení se skupinou Computer System Research Group (CSRG) pracující na Kalifornské univerzitě v Berkeley. Ta se stala střediskem úprav kódu původního systému UNIX. Jejich kód byl distribuován zdarma pro všechny s platnou licencí AT&T.

Výsledná kolekce záplat UNIXu se stala známá pod názvem Berkeley Software Distribution či také BSD UNIX. Vývojový proces pokračoval dlouho podle tohoto scénáře.

Protože se finanční prostředky CSRG začaly tenčit, došlo po jistých tahanicích uvnitř Kalifornské univerzity v roce 1992 k uvolnění kódu pod licencí, která se stala známou jako BSD licence. Tato licence bývá označována jako nejliberálnější softwarová licence a místo slova copyright se ve spojení s licencí BSD používá spíše copycenter.

Uvolnění kódu BSD se však setkalo s velikou nevolí ze strany UNIX System Laboratories (USL), organizační složky AT&T, jejíž zástupci okamžitě zažalovali některé uživatele a univerzity. Spor spočíval v autorských právech, na která si dělaly nárok obě strany. USL i Kalifornská univerzita. Mezitím vzali lidé kód uvolněný v Berkeley a začali na jeho základě stavět komerční i volně dostupné produkty. Jedním z nich byl i systém 386BSD, který se později stal základnou pro FreeBSD 1.0.

Po dvouletém sporu se prokázalo, že značná část kódu pochází zcela z BDS a ne naopak. Zbývalo pouze pár souborů, které byly zdrojem sváru. Některé z nich byly uvolněny a jiné nahrazeny. Tato nová, vyčištěná, verze byla uvolněna veřejnosti jako BSD4.4-Lite. Následující update, BSD4.4-Lite2, je praotcem současných zdrojových textů FreeBSD, ale i jiných jako NetBSD, OpenBSD či MacOS X.

Historie FreeBSD je převzata z [8].

3.2 Vývoj jednotlivých verzí FreeBSD

Pravděpodobně nejvýznamnějším zlepšením **FreeBSD 2.0** bylo:

- obnovení originálního Mach systému virtuální paměti, který byl optimalizován pro práci pod vysokým zatížením,
- vytvoření systému FreeBSD portů, který učinil stahování, sestavení a instalaci softwaru třetích stran velmi jednoduché.

Také FreeBSD 3.0 přineslo mnoho změn:

- přechod na ELF formát binárek,
- započítí podpory SMP systémů,
- podpora 64bitové Alpha platformy.

Toho času byla větev 3.X několikrát kritizována, protože mnoho změn nebylo evidentně prospěšných a postihly výkon, nicméně byl to potřebný krok pro vyvinutí toho, co se mělo stát velmi úspěšnou větví 4.X [13]:

- podpora více než 32 signálů, SA_SIGINFO odpovídající normě POSIX 1003.1,
- jail a sysctl podporováno i pro Linuxový mód,
- spousta vylepšení virtuální paměti,
- podpora pro přístup na disky se souborovým systémem NTFS,
- nové ATA/ATAPI ovladače,
- ovladače pro síťové karty Tigon 1 a 2, SysKonnect, Lucent Hermes IEEE 802.11,
- podpora IPv6 ve firewallu IPFW, podpora firewallu IPFilter,
- lepší podpora pro IPv6.

Velkým rozdílem ve FreeBSD 5 byla:

- změna v nízkoúrovňovém mechanismu jaderných zámků pro zajištění lepší podpory symetrického multiprocessingu (SMP), uvolňující kernel z MP zámků, někdy označovaného jako „Giant Lock“. Nyní je možné spouštět současně více než jeden proces v režimu jádra,
- změny zahrnují m:n řešení vláken zvané KSE (Kernel Scheduled Entities), které je nyní defaultní knihovnou vláken (pthreads) počínaje verzí 5.3,
- rozšiřitelný framework mandatorní kontroly přístup,
- Access Control Listy (ACL),
- nový souborový systém UFS2,
- významná změna vrstvy blokového vstupu/výstupu představením GEOM modulárního transformačního frameworku diskových I/O požadavků.

Mezi největší přednosti FreeBSD patří binární kompatibilita s jinými UNIXovými operačními systémy. Nejzajímavější je asi podpora GNU/Linuxu. Důvodem k tomu je, že spousta aplikací existuje pouze v binární podobě pro GNU/Linux a autoři nejsou schopni či ochotni naportovat aplikaci na FreeBSD – nejčastěji jde o komerční aplikace. Tato vrstva kompatibility byla ale vůči GNU/Linuxu nekompletní a hodně pozadu. V rámci akce Google Summer of Code 2006 došlo k aktualizaci.

Jádro systému FreeBSD také poskytuje spoustu mechanismů pro zabezpečení systému. Při startu lze nastavit úroveň zabezpečení od nejnižší, která je vhodná pro konfiguraci systému po nejvyšší, kdy případný útočník nemůže změnit konfiguraci systému a tak si zajistit zadní vrátka pro další návštěvy. Je jen na správci systému, jaké zabezpečení si zvolí [8].

Kapitola 4

Microsoft Windows

4.1 Veze Microsoft Windows

Microsoft Windows se dají rozdělit do několika nezávislých řad. Windows 3.X, 9X, ME mohou být příkladem jedné řady. Jsou to Windows, která ke svému spuštění vyžadují DOS, jsou jeho nadstavbou. V případě Windows ME šikově maskovanou. Jde o hybridní 16/32-bitové systémy. Dnes se již nevyvíjí.

Další řada Windows je založena na jádře NT. Toto jádro je vytvořeno zcela od začátku s nezávislostí na předchozích verzích Windows. Od začátku je toto jádro zcela 32-bitové. Windows NT je částečně kompatibilní s normou POSIX. Nejčastější platformou pro provoz Windows NT je Intel x86, případně AMD64 (64 bitové rozšíření Intel x86, také označováno jako x86_64) [21].

Také je třeba zmínit Windows CE určené pro vestavěná zařízení. Windows CE jsou optimalizovány pro zařízení s minimálním diskovým prostorem a minimální velikostí paměti. V současnosti jsou zařízení s Windows Mobile nejčastěji osazeny 64MB RAM a 128MB ROM. Windows CE vyhovují definici real time operačního systému – mají deterministické zpoždění přerušování, 256 úrovní priority, používají zvýšení priority pro předcházení inverzi priority [20].

V dalším textu se budu zabývat Windows NT.

4.2 Historie Windows NT

Historie Windows NT začíná kolem roku 1988. Šlo o projekt napsání nového, pokročilého operačního systému úplně od začátku. Tento projekt se podařilo dokončit v roce 1993, kdy vyšel operační systém Windows NT 3.1. Windows NT neznamenal nic menšího než fundamentální změnu v tom jak společnosti mohou adresovat svoje obchodní požadavky na počítače.

Tato změna je reprezentována v názvu produktu – NT, new technology (nová technologie). Pro zachování konzistence s Windows 3.1, tehdejším prostředím známým v domácnostech i v obchodní sféře, byl nový systém pojmenován Windows NT 3.1. Na rozdíl od Windows 3.1 byly Windows NT plně 32 bitové.

Systém Windows NT byl určen pro high-end, klient/server aplikace, byl dostupný ve verzi klient i server, přičemž serverová verze se jmenovala Windows NT Advanced Server. Klientská verze byla dobře přijata vývojáři pro její stabilitu, bezpečnost a aplikační rozhraní Win32, které umožnilo snadný a rychlý vývoj aplikací [9].

Windows NT Workstation 3.5 vyšly v roce 1994 a přinesly následující vylepšení:

- podpora grafického rozhraní OpenGL,
- vysoký stupeň zabezpečení pro kritická data a aplikace,
- vylepšení výkonu 32-bitových aplikací,
- podpora pro souborové a tiskové servery Novel NetWare,
- podpora pro dlouhé názvy souborů.

V roce 1996 pak následovala verze Windows NT Workstation 4.0:

- zjednodušeno uživatelské rozhraní (stejně jako Windows 95),
- zjednodušena správa systému,
- zvýšena síťová propustnost,
- nástroje pro správu a vývoj intranetu.

Další verze Windows NT se již prodávala pod jiným obchodním názvem – Windows 2000:

- určeny k nahrazení Windows 9X, Windows ME,
- zvýšena spolehlivost systému, použitelnost,
- vylepšena spolupráce se sítí internet,
- podpora mobilních počítačů,
- mohutně vylepšena podpora Plug-and-Play zařízení, snadnější instalace ovladačů (Windows Driver Model).

Krátce na to vyšly Windows XP, které měly několik řad – Professional, Home, Media Center, Tablet PC, 64bit:

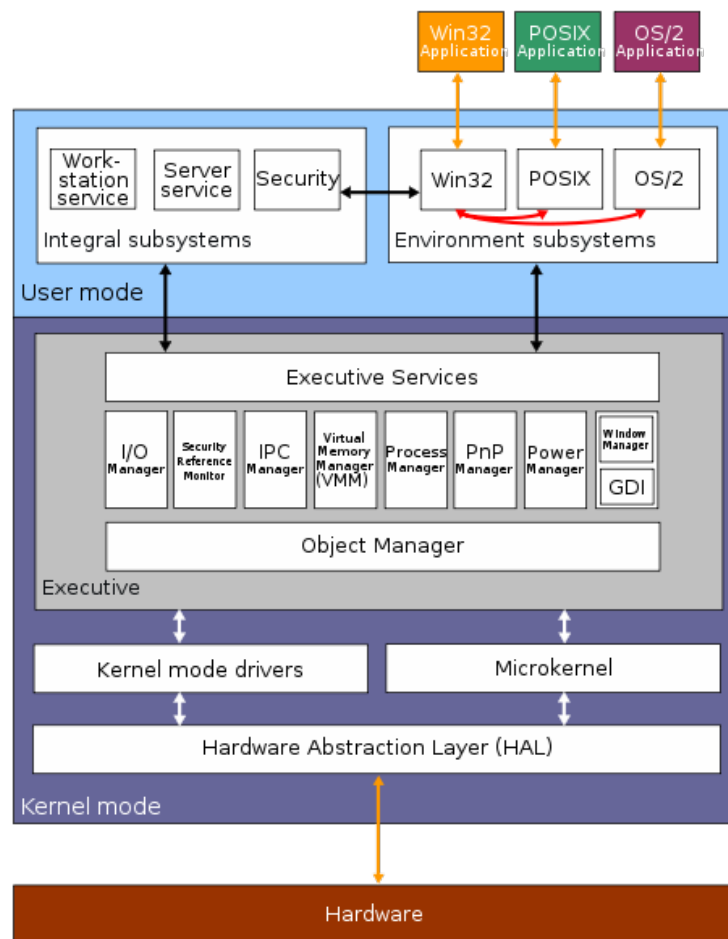
- spojení řady NT s řadou Windows 9x,
- zlepšují výkon, bezpečnost a spolehlivost Windows 2000,
- nový vzhled uživatelského rozhraní,
- vlastnosti vhodné pro firemní prostředí
 - vzdálený přístup,
 - šifrovaný souborový systém,
 - obnova systému,
 - pokročilé síťování
- podpora pro 802.1x

- 64bit verze přinesla podporu pro procesory Intel Itanium, je to první 64bitový operační systém firmy Microsoft určený pro pracovní stanice s velkým množstvím paměti používaných k vědeckým výpočtům, animacím, speciálním efektům,
- Media Center Edition je určena pro počítače nasazené jako domácí zábavní centra
 - přidává podporu pro digitální zábavní média,
 - podporu pro sledování a nahrávání televize, DVD, hudební kolekce,
 - snadné propojení s televizí,
- Tablet PC přidává podporu pro rozpoznávání rukou psaného písma

Dalšími zástupci řady Windows NT jsou Windows Server 2003, Windows Vista, Windows Server 2008. Čerpáno z [9].

4.3 Architektura jádra Windows NT

Obrázek 4.1: Architektura jádra Windows 2000 (převzato z [16])



Architektura Windows NT je vysoce modulární a skládá se ze 2 hlavních vrstev: komponenty mohou běžet v uživatelském režimu nebo v režimu jádra. Programy a podsystémy v uživatelském režimu jsou omezeny v přístupu k systémovým zdrojům, naopak režim jádra má neomezený přístup k systémové paměti i zařízením. Takto pracující operační systémy bývají označovány jako hybridní, přestože je toto označení zavádějící – všechny důležité části jádra běží v režimu jádra a tvoří tak monolitický kernel, který je strukturován podobně jako mikrokernel. Architektura zahrnuje jednoduché jádro, vrstvu tvořící abstrakci nad zařízeními (Hardware Abstraction Layer, HAL), ovladači a rozsahem služeb, které běží v režimu jádra [4].

Uživatelský režim je tvořený ze subsystémů schopných předávat vstupně výstupní požadavky odpovídajícímu ovladači v režimu jádra přes správce I/O. Uživatelský režim je tvořen dvěma komponentami: tzv. „Environment subsystem“, který slouží k emulaci různých systémů (podpora pro POSIX), a „Integral subsystem“, který řídí systémově specifické funkce pro podporu environment subsystem viz obrázek 4.1. Jádro neumožňuje uživatelskému prostoru přístup ke kritickým oblastem operačního systému.

Grafické uživatelské rozhraní je těsně spjaté s jádrem systému (viz GDI a Window Manager v obrázku 4.1).

4.4 Real time rozšíření Windows NT

Pro Windows NT existují real time rozšíření od firem VenturCom (RTX), Radisys (INTime) a Imagination Systems (Hyperkernel) [14]. Aby mohly být Windows NT označovány za real time systém, musí být splněno několik podmínek:

- musí být vícevláknový,
- musí být preemptivní,
- musí mít priority,
- musí mít předvídatelné synchronizační mechanismy,
- dostupný mechanismus pro zahrnutí priority,
- chování operačního systému musí být předvídatelné – zpoždění přerušování, přepínání úloh, ovladače...

4.4.1 Komentář

První tři body Windows NT částečně splňují, ale mají pouze málo úrovní priority použitelných pro real time.

Pro snížení zpoždění způsobených zpracováním přerušování Windows NT zavedly odložené volání procedur. Priorita těchto odložených procedur je vyšší než priorita ostatních procesů, ale všechny odložené procedury běží na stejné úrovni priority – jsou zpracovávány v pořadí FIFO. Zpracování přerušování s vysokou prioritou ale bude až poté co jsou všechny předchozí procedury zpracovány, což vede k neodhadnutelným zpožděním.

Správa paměti je ve Windows NT založena na mechanismu virtuální paměti. To znamená, že obsahuje ochranu adresního prostoru, překlad adres, používání swapu. Použití swapu je neakceptovatelné pro real time aplikace. Stránky paměti mohou být zamčeny – nedochází pak k jejich odswapování.

Windows NT nelze využít v real time aplikacích

4.4.2 Možná řešení

- Windows NT mohou být použity tak jak jsou – existuje několik metod jak jejich omezení obejít (použití nemaskovatelných přerušení),
- implementace Win32 API na komerčním real time operačním systému,
- spoluexistence real time operačního systému a Windows NT,
- rozšíření předchozí myšlenky pro víceprocesorové prostředí (každý operační systém běží na svém procesoru).

Nás ale zajímají řešení postavená pouze na Windows NT:

- modifikace HAL vrstvy přijímáním přerušení a zahrnutím malého real time plánovače nebo real time operačního systému,
- spuštění Windows NT jako jednu z úloh v real time operačním systému.

Tyto 2 možnosti jsou uplatněny v již výše popsáných real time rozšířeních.

Literatura

- [1] Bourek, J.: Jaderné noviny - 17. 9. 2008 (Kernel Summit 2008) — ABCLinuxu. [on-line], 31.10.2008 [cit. 2008-11-08]. Dostupný z WWW <<http://www.abclinuxu.cz/clanky/jaderne-noviny/jaderne-noviny-17.-9.-2008-kernel-summit-2008-2>>.
- [2] Bourek, J.: Jaderné noviny - 19 a 20/2008 — ABCLinuxu. [on-line], 12.6.2008 [cit. 2008-11-08]. Dostupný z WWW <<http://www.abclinuxu.cz/clanky/jaderne-noviny/jaderne-noviny-19-a-20-2008>>.
- [3] Corbet, J.: The Big Kernel Lock lives on. [on-line], may 26, 2004 [cit. 2008-11-07]. Dostupný z WWW <<http://lwn.net/Articles/86859/>>.
- [4] Finnel, L.: MCSE Windows 2000 server study guide (exam 70-215). 2000, ISBN 1-57231-903-8.
- [5] FreeBSD kernel: FreeBSD kernel kern code Documentation. [on-line], jul 11 2007 [cit. 2008-11-08]. Dostupný z WWW <http://www.delphij.net/fbdoxy/kern/html/d5/d2e/kern__thread_8c.html>.
- [6] FSF: GNU General Public License, version 2. [on-line], 2008/09/12 [cit. 2008-11-07]. Dostupný z WWW <<http://www.gnu.org/licenses/old-licenses/gpl-2.0.html>>.
- [7] Krátký, R.: Jaderné noviny - 18. 4. 2007 — ABCLinuxu. [on-line], 10. 5. 2007 [cit. 2008-11-08]. Dostupný z WWW <<http://www.abclinuxu.cz/clanky/jaderne-noviny/jaderne-noviny-18.-4.-2007>>.
- [8] Lucas, M.: *FreeBSD: Síťový operační systém*. Nám. 28. dubna 48, 635 00 Brna: Computer Press, 2003, ISBN 80-7226-795-7.
- [9] Microsoft Corporation: Windows History: Windows Desktop Products History. [on-line], march 7, 2006 [cit. 2008-11-08]. Dostupný z WWW <<http://www.microsoft.com/windows/WinHistoryDesktop.aspx>>.
- [10] Singh, J.; Fu, L.; Schwebel, R.; aj.: How does the CONFIG_PREEMPT_RT patch work? [on-line], 17 September 2008 [cit. 2008-11-08]. Dostupný z WWW <http://rt.wiki.kernel.org/index.php?title=Frequently_Asked_Questions&oldid=2239#How_does_the_CONFIG_PREEMPT_RT_patch_work.3F>.
- [11] Stallman, R.: Linux and the GNU Project. [on-line], 2008/10/07 [cit 2008-11-07]. Dostupný z WWW <<http://www.gnu.org/gnu/linux-and-gnu.html>>.

- [12] Tanenbaum, A. S.; Woodhull, A. S.: *Operating Systems: Design and Implementation*. Prentice Hall, 2006, ISBN 9780131429383.
- [13] The FreeBSD Project: RELEASE NOTES: FreeBSD Release 4.0-RELEASE. [on-line], 21.3.2000 [cit. 2008-11-08]. Dostupný z WWW <<http://www.freebsd.org/releases/4.0R/notes.html>>.
- [14] Timmerman, M.; Beneden, B. V.; Uhres, L.: Windows NT Real-Time Extensions better or worse? *Real-Time Magazine*, ročník 99-3, 1998: s. 11–19.
- [15] Torvalds, L.: comp.os.minix. [on-line], [cit. 2008-11-07]. Dostupný z WWW <<http://groups.google.com/group/comp.os.minix/msg/b813d52cbc5a044b>>.
- [16] Wikipedia: Architecture of Windows NT — Wikipedia, The Free Encyclopedia. [on-line], 6 October 2008 [cit. 2008-11-08]. Dostupný z WWW <http://en.wikipedia.org/w/index.php?title=Architecture_of_Windows_NT&oldid=243371428>.
- [17] Wikipedia: DTrace — Wikipedia, The Free Encyclopedia. [on-line], 5 November 2008 [cit. 2008-11-08]. Dostupný z WWW <<http://en.wikipedia.org/w/index.php?title=DTrace&oldid=249869659>>.
- [18] Wikipedia: Linux kernel — Wikipedia, The Free Encyclopedia. [on-line], november 3, 2008 [cit. 2008-11-08]. Dostupný z WWW <http://en.wikipedia.org/w/index.php?title=Linux_kernel&oldid=249366299>.
- [19] Wikipedia: Linux kernel portability and supported architectures — Wikipedia, The Free Encyclopedia. [on-line], 20 October 2008 [cit. 2008-11-07]. Dostupný z WWW <http://en.wikipedia.org/w/index.php?title=Linux_kernel_portability_and_supported_architectures&oldid=246592996>.
- [20] Wikipedia: Windows CE — Wikipedia, The Free Encyclopedia. [on-line], 7 November 2008 [cit. 2008-11-08]. Dostupný z WWW <http://en.wikipedia.org/w/index.php?title=Windows_CE&oldid=250338411>.
- [21] Wikipedia: Windows NT — Wikipedia, The Free Encyclopedia. [on-line], 3 November 2008 [cit. 2008-11-08]. Dostupný z WWW <http://en.wikipedia.org/w/index.php?title=Windows_NT&oldid=249460509>.
- [22] Wikipedie: FreeBSD — Wikipedie: Otevřená encyklopedie. [on-line], 12. 10. 2008 [cit. 2008-11-08]. Dostupný z WWW <<http://cs.wikipedia.org/w/index.php?title=FreeBSD&oldid=3154493>>.
- [23] Šumský, D.: Historie Linuxu pěkně od začátku. [on-line], 18-Apr-2003 [cit. 2008-11-07]. Dostupný z WWW <<http://www.fi.muni.cz/usr/jkucera/pv109/2003/xsumsky.htm>>.