

Operačné systémy a počítačové siete
RNDr. Jaroslav Janáček, PhD.

Organizačné informácie

- prednášajúci: RNDr. Jaroslav Janáček, PhD.
 - M253, jaroslav.janacek@uniba.sk
- cvičenia (len DAV+BIN): Mgr. Marek Šuppa
- rozvrh:
 - prednášky: štvrtok 11:30 v F1, 2h
 - cvičenia (DAV+BIN): streda 16:30 v H6, 2h
- web:
 - prednášky: <http://www.dcs.fmph.uniba.sk/osps/>

Organizačné informácie

- hodnotenie
 - DAV + BIN: 50% tvorí hodnotenie z cvičení, 50% tvorí test z tém z prednášok
 - MAT: 50% tvorí aktívna účasť na prednáškach, 50% test
 - $\geq 60\%$ E, $\geq 68\%$ D, $\geq 76\%$ C, $\geq 84\%$ B, $\geq 92\%$ A

Cieľ predmetu

- získať prehľad o:
 - princípoch fungovania počítačov
 - úlohách a princípoch fungovania operačných systémov
 - princípoch fungovania a bezpečného používania počítačových sietí a ich služieb
- cvičenia (DAV+BIN):
 - získať základné zručnosti v používaní Linux-u potrebné aj pre ďalšie predmety

Čo sa napríklad dozviete

- Z čoho sa skladá počítač a čo to tam robí?
- Prečo občas počítač pracuje pomalšie než inokedy?
- Prečo má 200GB disk iba 190GB?
- Čo sa deje pri zapnutí počítača?
- Ako OS rozhodne, ktorá úloha má prednosť?
- Aký je rozdiel medzi http:// a https:// ?
- Môže niekto čítať vaše emaily?
- Čo robiť, keď vám nefunguje sieť?

Obsah – počítače a OS

- zloženie počítača
- vykonávanie inštrukcií
- typy pamätí a ich využitie
- vstupno-výstupné zariadenia
-
- úloha OS
 - správa procesov, správa pamäte, súborové systémy, bootovanie, ...
- virtualizácia

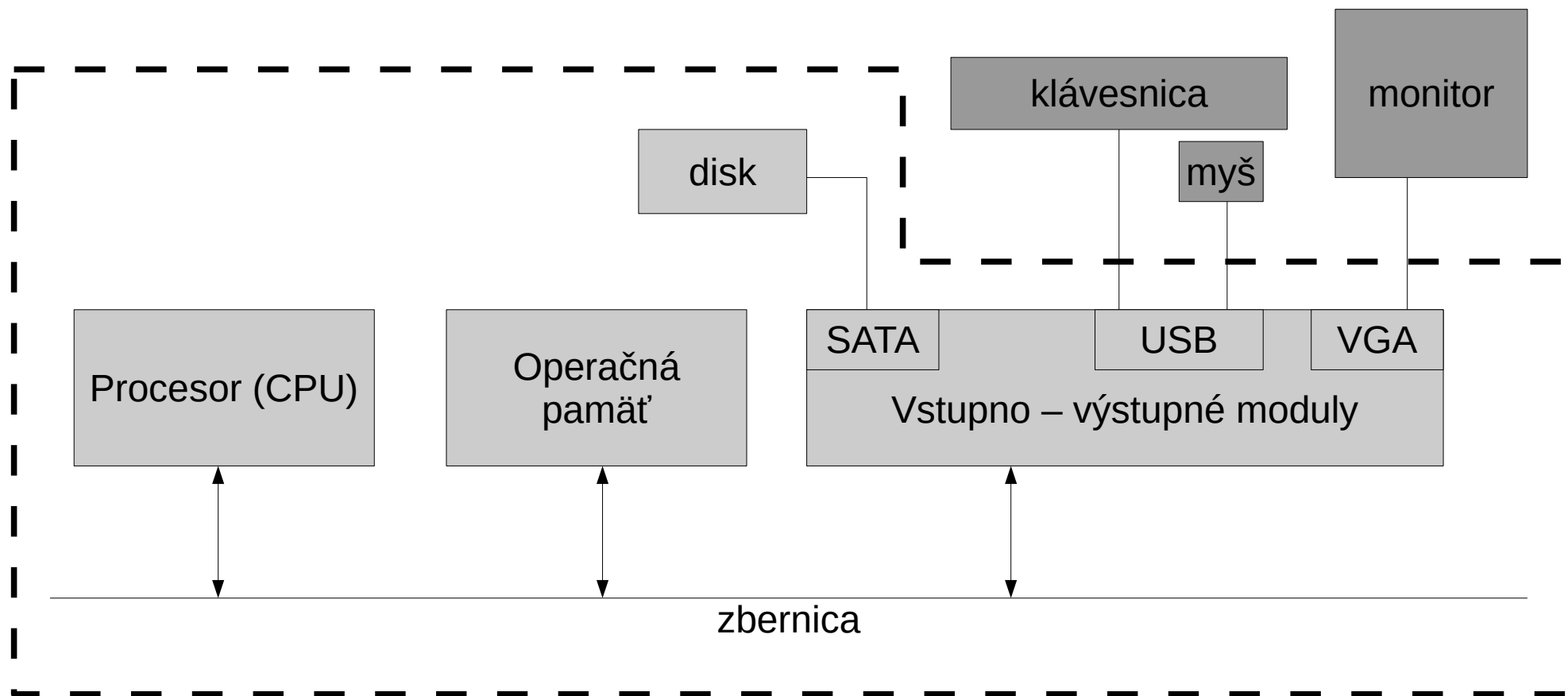
Obsah – počítačové siete

- čo je to počítačová sieť
- hardwarové aspekty
- softwarové aspekty
- ako fungujú siete (ISO OSI, TCP/IP)
- ako funguje Internet
- základné služby Internetu
 - elektronická pošta
 - WWW, DNS
- bezpečnosť

Prečo hovoriť o hardvéri?

- softvér vs. hardvér
- používateľ chce, aby mu fungovali **aplikačné programy**
- tie využívajú OS na prístup k hardvéru
- ak máme rozumieť úlohám OS, potrebujeme niečo vedieť hardvéri (HW)

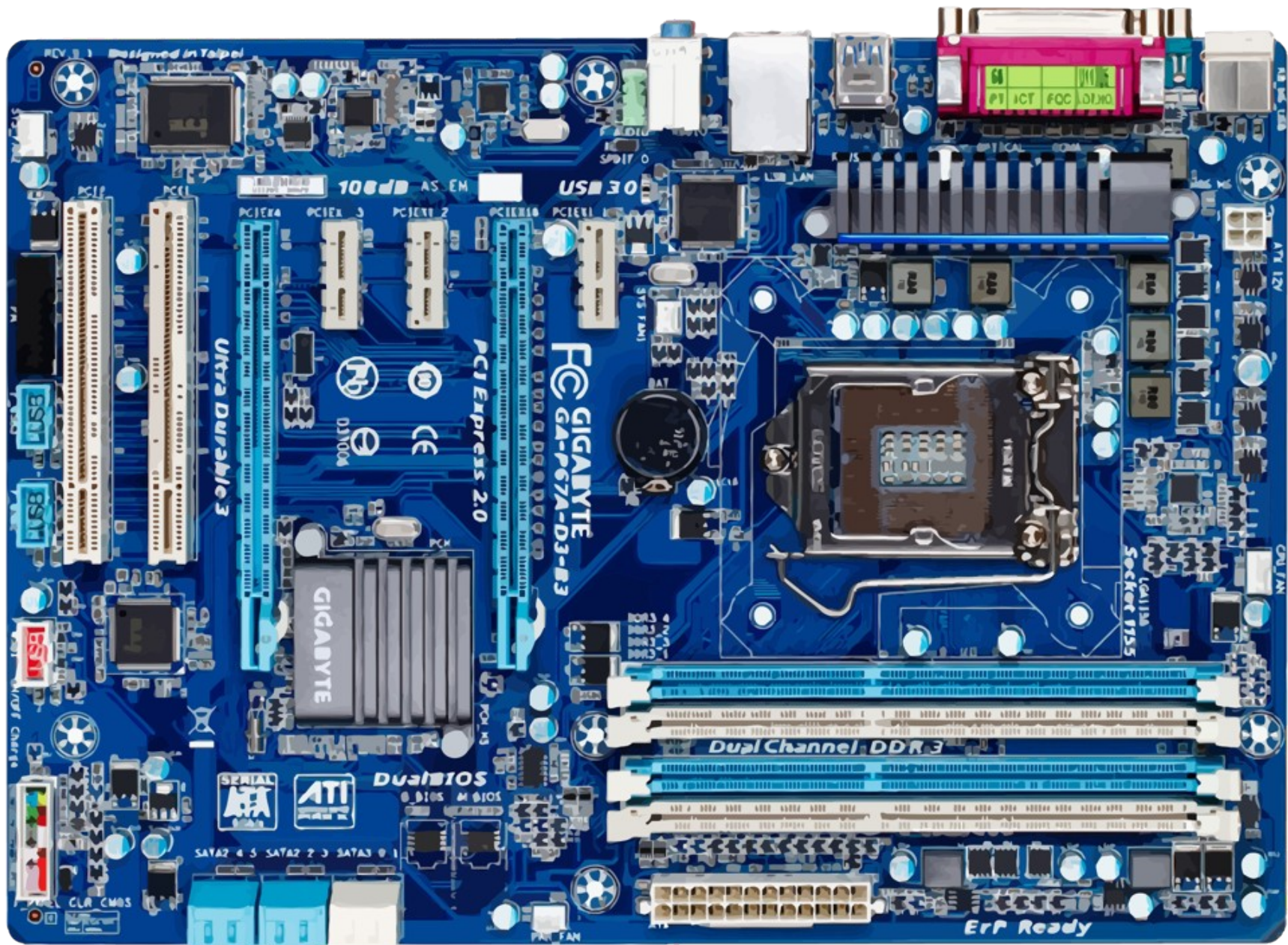
Zjednodušený model počítača



Základná doska

- zbernica na prepojenie jednotlivých komponentov
 - zbernica medzi CPU a pamäťou
 - PCI, PCI Express
- často integrované základné vstupno-výstupné moduly
 - radič diskov (IDE, SATA)
 - radič USB, PS/2
 - grafická karta

Základná doska



Processor

- môže ich byť aj viac
- *jadrá* – v podstate samostatné procesory
- riadi činnosť počítača a vykonáva väčšinu operácií s údajmi
- základné časti
 - riadiaca jednotka
 - aritmeticko-logická jednotka
 - registre

Processor



Processor

- riadiaca jednotka
 - načítava a dekóduje *inštrukcie programu*
 - aktivuje ďalšie obvody na vykonanie inštrukcie
- aritmeticko-logická jednotka
 - vykonáva základné aritmetické a logické operácie
- registre
 - slúžia na uloženie práve spracovávaných údajov
 - malá, rýchla pamäť priamo v procesore

Processor

- inštrukcia
 - elementárny príkaz vykonateľný procesorom
 - načítanie údajov z operačnej pamäte do registra
 - zápis údajov z registra do operačnej pamäte
 - sčítanie/odčítanie/porovnanie/násobenie/delenie 2 čísel
 - „skok“ - presun na inú časť programu
 - ...

Processor

- program
 - zdrojový text vo *vyššom programovacom jazyku*
 - vyššia úroveň abstrakcie, nezávislá od procesora
 - C/C++, Java, ...
 - preklad do *strojového kódu*
 - postupnosť inštrukcií konkrétneho procesora
 - alebo virtuálneho procesora (napr. Java Virtual Machine)
 - vykonáva sa pomocou *interpretera*

Processor

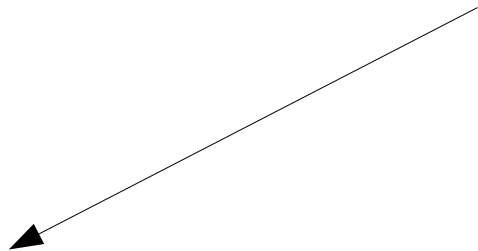
```
int f(int a, int b)
{
    return a+b;
}
```

```
c=f(1,2);
```

```
8b 44 24 04
03 44 24 08
c3 ...
```

```
f: mov 4(%esp), %eax
    add 8(%esp), %eax
    ret
```

```
push $2
push $1
call f
add $8, %esp
mov %eax, c
```



Operačná pamäť

- dočasná pamäť s náhodným prístupom (RAM)
 - procesor môže pristupovať k ľubovoľnému miestu v pamäti
- uloženie vykonávaného programu a spracovávaných údajov
- veľkosti v rozsahu niekoľkých gigabyte-ov (GB)
 - v serveroch aj desiatky GB

Operační paměť



Operačná pamäť

- po vypnutí napájania stráca obsah
- ale nejaký čas to trvá
 - pri izbovej teplote sekundy až minúty
 - pri nízkych teplotách aj hodiny
- preto vypnutie nie je z bezpečnostného hľadiska postačujúce na okamžité zničenie obsahu pamäte

Operačná pamäť

- čítanie údajov z operačnej pamäti
 - procesor pošle cez zbernicu pamäti **adresu** miesta, ktoré chce čítať
 - pamäť pošle cez zbernicu procesoru príslušnú hodnotu
- zápis údajov do operačnej pamäte
 - procesor pošle cez zbernicu pamäti adresu a hodnotu
 - pamäť hodnotu uloží na príslušné miesto

Ďalšie typy pamäti

- „nezabúdajúce“ pamäte
 - EEPROM, Flash, ...
 - obsahujú program pre inicializáciu hardvéru (BIOS) a na načítanie a spustenie operačného systému
 - ich obsah je možné vymazať a nanovo nahrat'
 - riziko zmeny škodlivým kódom
- cache pamäť
 - obsahuje kópiu používaných údajov z oper. pamäte
 - na preklopenie rozdielu v rýchlosti oper. pamäte a CPU

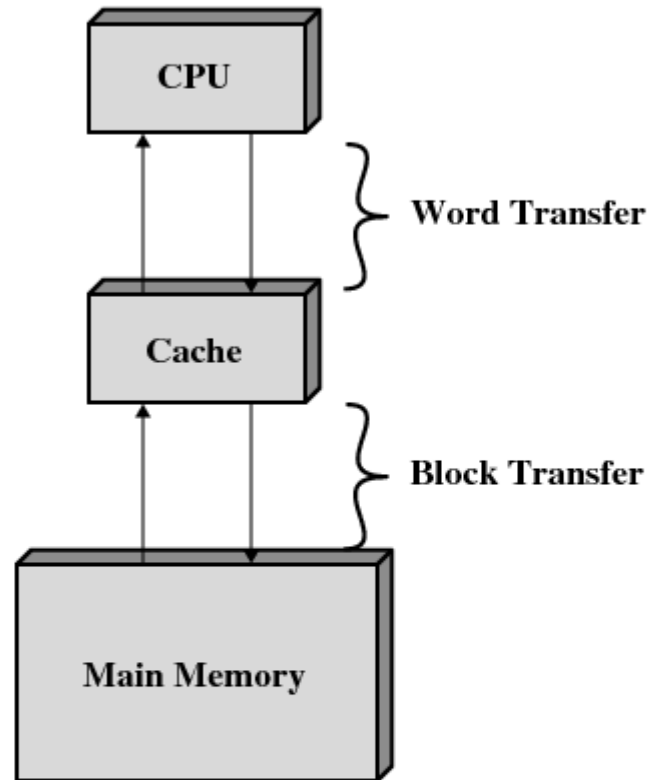


Figure 1.16 Cache and Main Memory

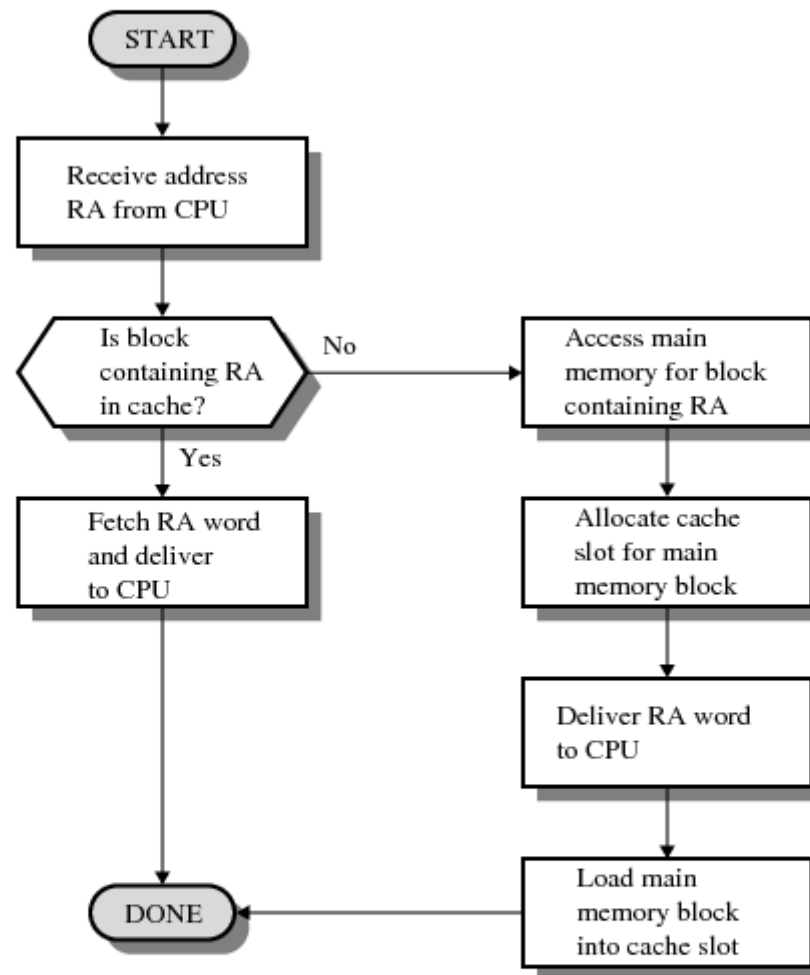


Figure 1.18 Cache Read Operation

Charakteristika pamätí

- RAM = Random Access Memory
- SAM = Sequential Access Memory
- DAM = Direct Access Memory - „hybrid“ medzi RAM a SAM
- ROM = Read Only Memory
- RWM = Read-Write Memory
- bežné kombinácie
 - RAM+RWM – napr. operačná pamäť, cache
 - RAM+ROM – napr. BIOS
 - SAM+RWM – napr. magnetická páska
 - DAM+RWM – napr. pevný disk
 - DAM+ROM – napr. CD

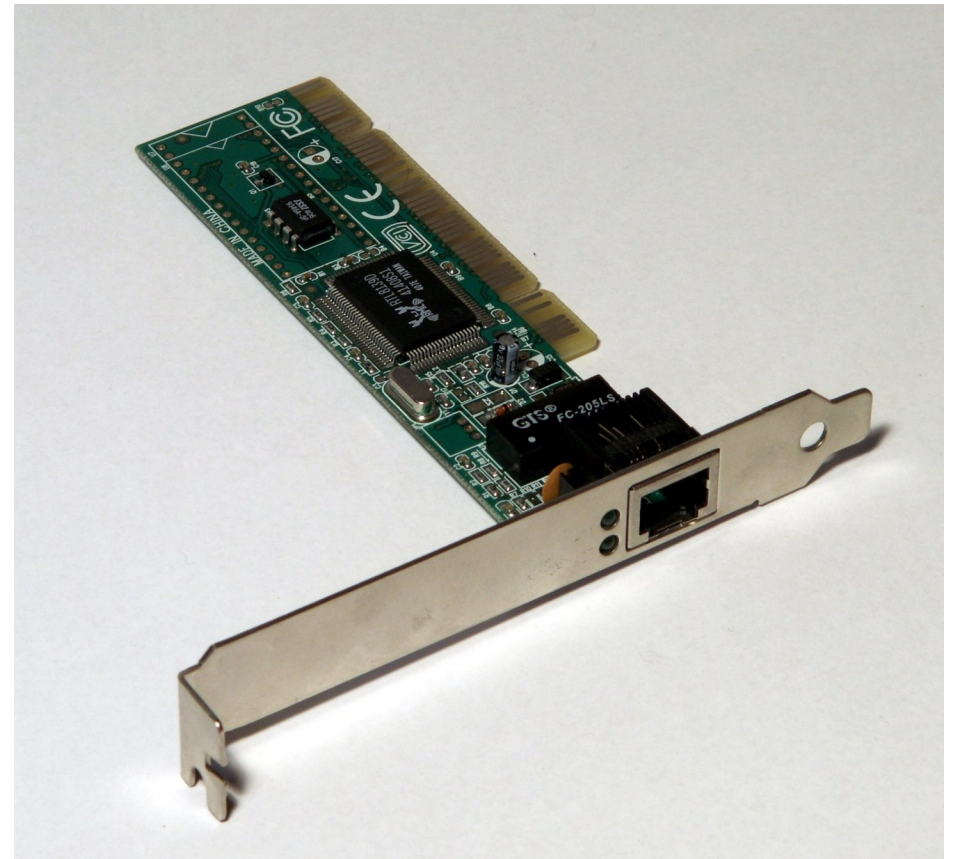
Operačná pamäť v bežných počítačoch

- Rôzne typy
 - SDRAM
 - DDR
 - DDR2
 - DDR3
- navzájom nekompatibilné

Vstupno-výstupné moduly

- rozširujúce karty pripojené na zbernicu
 - prípadne integrované priamo na základnej doske
- príklady:
 - grafická karta – zobrazovanie na monitore
 - VGA, DVI, HDMI, DP
 - sieťová karta – pre pripojenie k poč. sieti
 - radič diskov SCSI/SAS

Vstupno-výstupné moduly



Pevné disky

- trvalejšie ukladanie údajov a programov
- veľkosti v rozsahu desiatok GB až niekoľko TB
- obsahujú mechanicky citlivé prvky
 - čítacie a zapisovacie hlavy, platne (5-15 tisíc ot./min.)
 - časom sa opotrebovujú a pokazia – strata údajov

Pevné disky



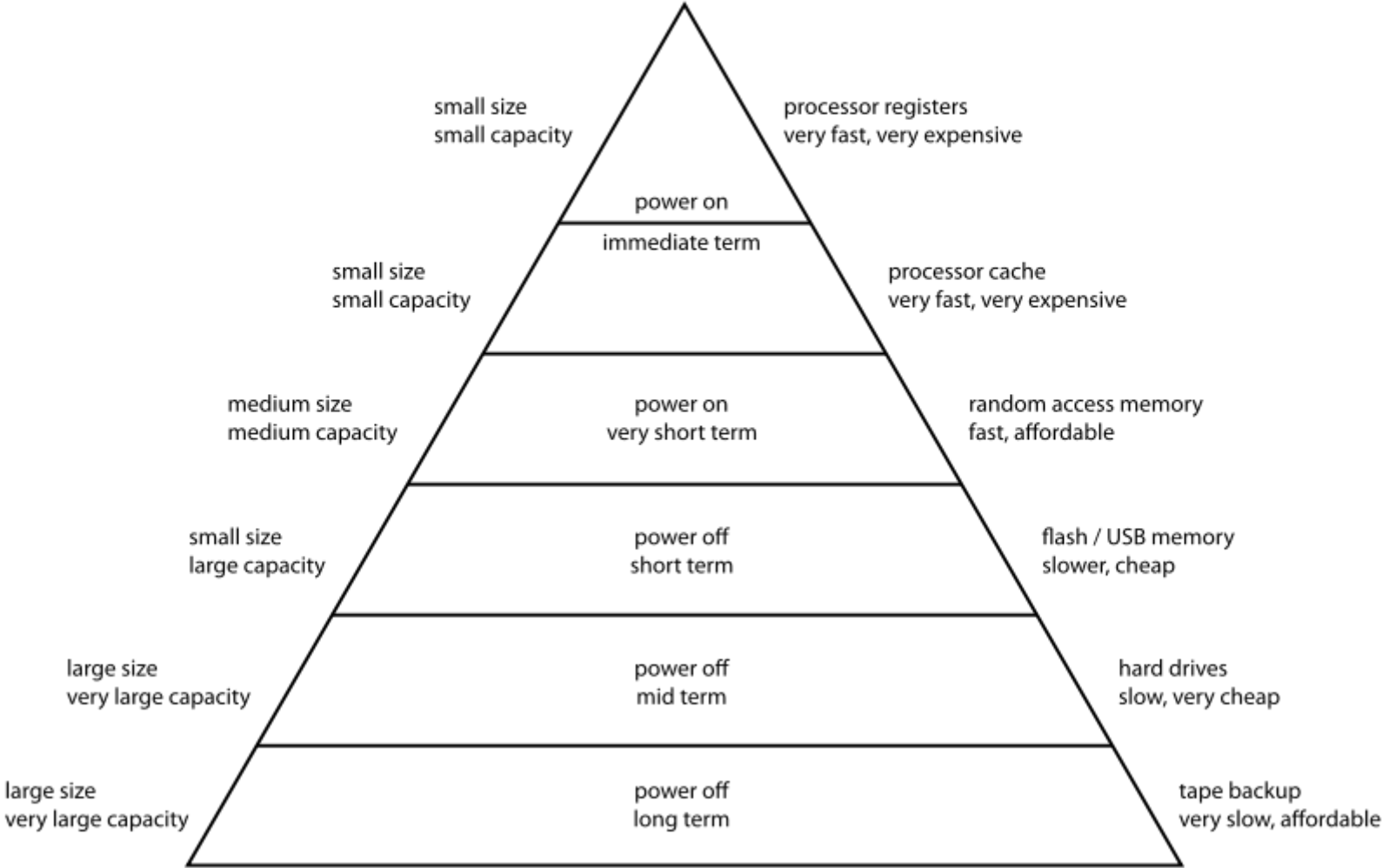
Pevné disky

- rozhrania
 - IDE, SATA – v stolných počítačoch a notebook-och
 - SCSI, SAS, FC – v serveroch
- rozhrania pre externé disky
 - USB, eSATA
- SSD
 - náhrada za pevné disky bez pohyblivých častí
 - rýchle, ale malá kapacita, obmedzený počet zápisov

Nie je GB ako GB

- pri pamätiach
 - 1KB = 1024B (2^{10})
 - 1MB = 1024KB = 1048576B (2^{20})
 - 1GB = 1024MB = 1048576KB = 1073741824B (2^{30})
 - 1TB = 1024GB = 1048576MB = 1073741824KB = 2^{40} B
- pri diskoch
 - 1kB = 1000B (10^3)
 - 1MB = 1000kB = 1000000B (10^6)
 - 1GB = 1000MB = 1000000kB = 1000000000B (10^9)
 - 1TB = 1000GB = 1000000MB = 1000000000kB = 10^{12} B

Computer Memory Hierarchy



Registre procesora

- špeciálny register obsahuje adresu aktuálnej inštrukcie, ktorá sa má vykonať
 - PC – Program Counter, alebo
 - IP – Instruction Pointer
- stavový register (PSW)
 - obsahuje rôzne stavové informácie
- všeobecné registre
 - údaje, adresy

Zjednodušený inštrukčný cyklus

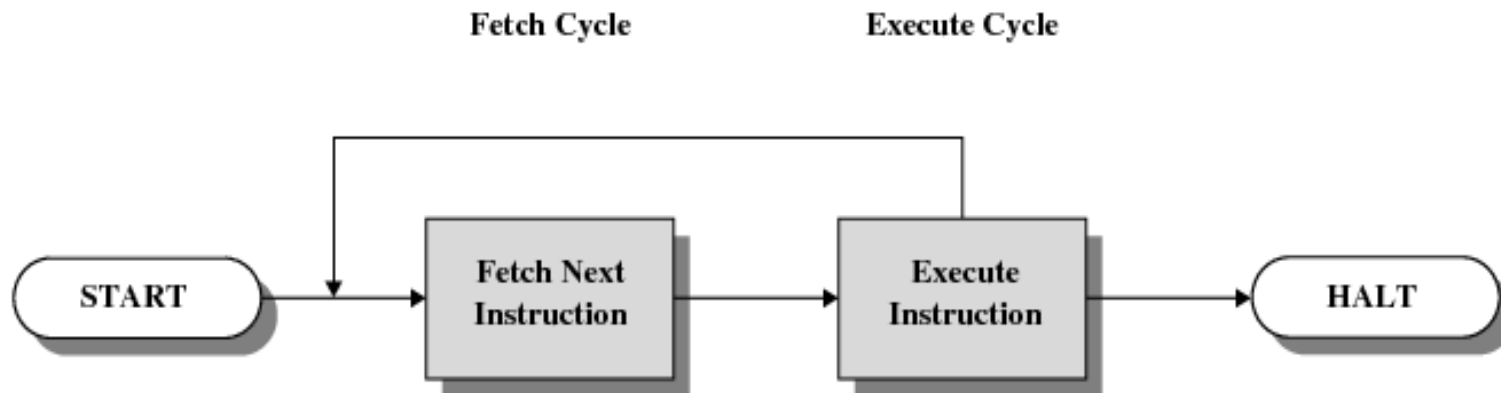


Figure 1.2 Basic Instruction Cycle

Zjednodušený inštrukčný cyklus

- Fetch cyklus
 - načíta inštrukciu
 - zvýši PC
- Execute cyklus
 - vykoná inštrukciu
- Problém
 - spracovanie vstupno-výstupných operácií
 - procesor musí dlho čakať nevyužitý

Realizácia vstupno-výstupnej operácie

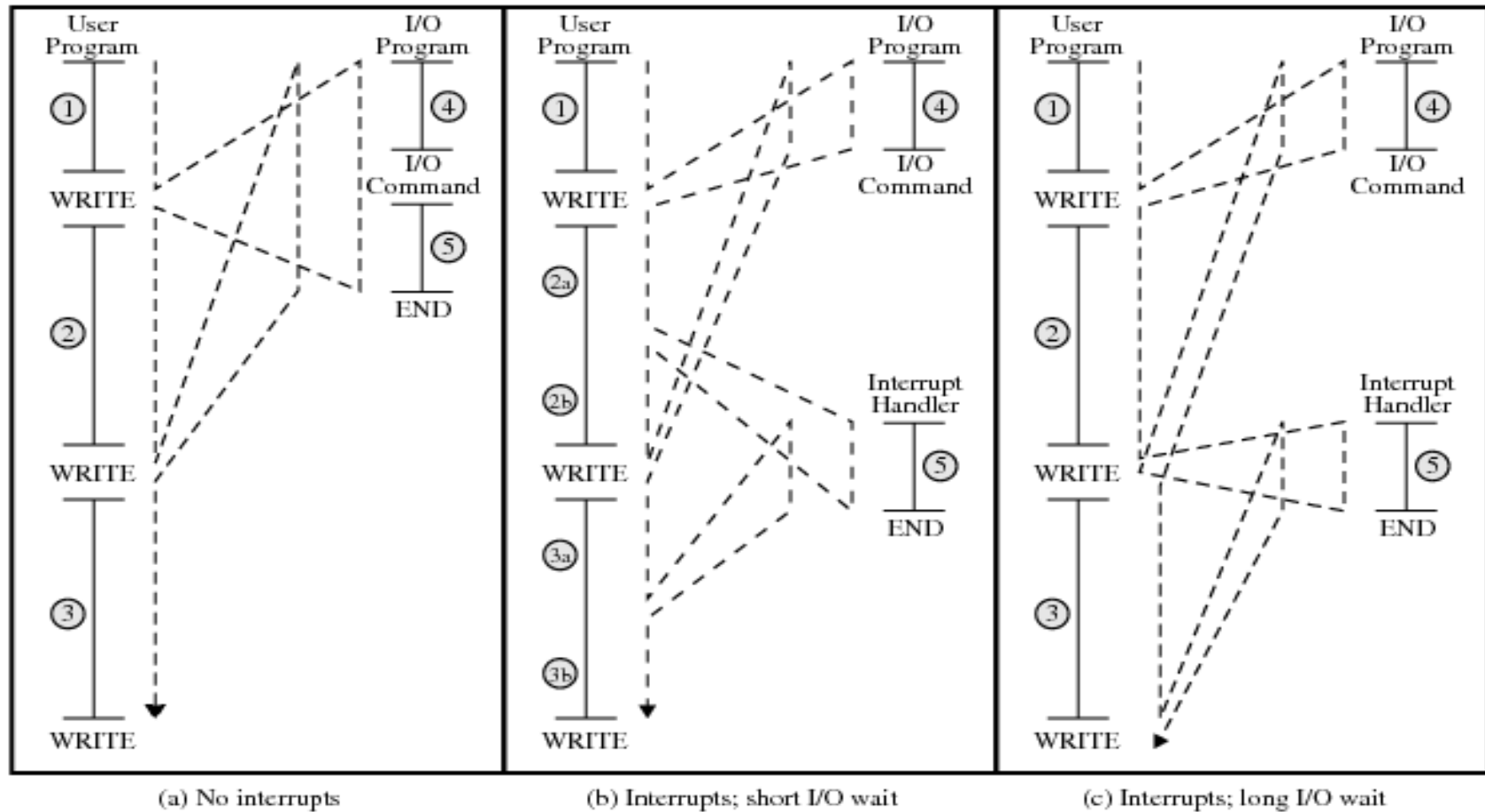


Figure 1.5 Program Flow of Control Without and With Interrupts

Realizácia vstupno-výstupnej operácie

- procesor odošle požiadavku V/V zariadeniu (4)
- V/V zariadenie požiadavku spracováva, procesor zatiaľ čaká na jej dokončenie
- procesor získa výsledok operácie (5)

Prerušená

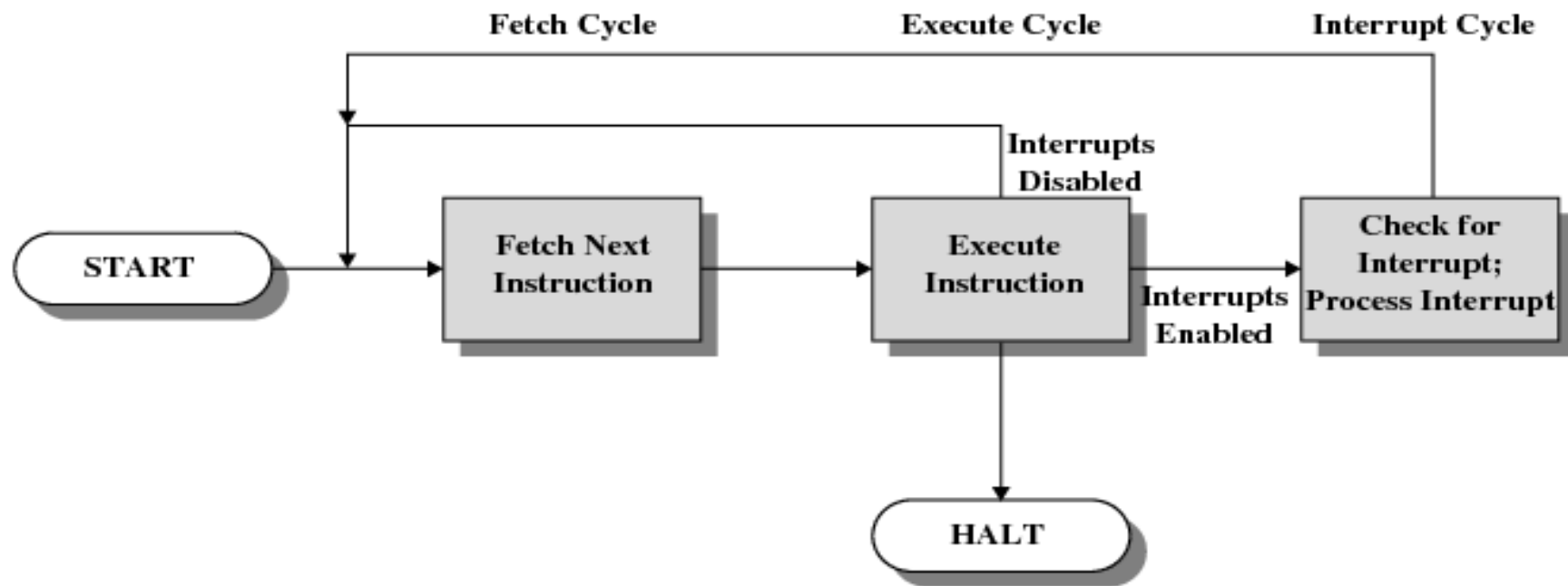


Figure 1.7 Instruction Cycle with Interrupts

Prerušená

- Po vykonaní inštrukcie procesor kontroluje, či nevznikla žiadosť o prerušenie.
- Ak áno, uloží stav (PC a PSW) a začne vykonávať funkciu na obsluhu prerušenia,
- po jej skončení obnoví stav a pokračuje v pôvodnej činnosti.

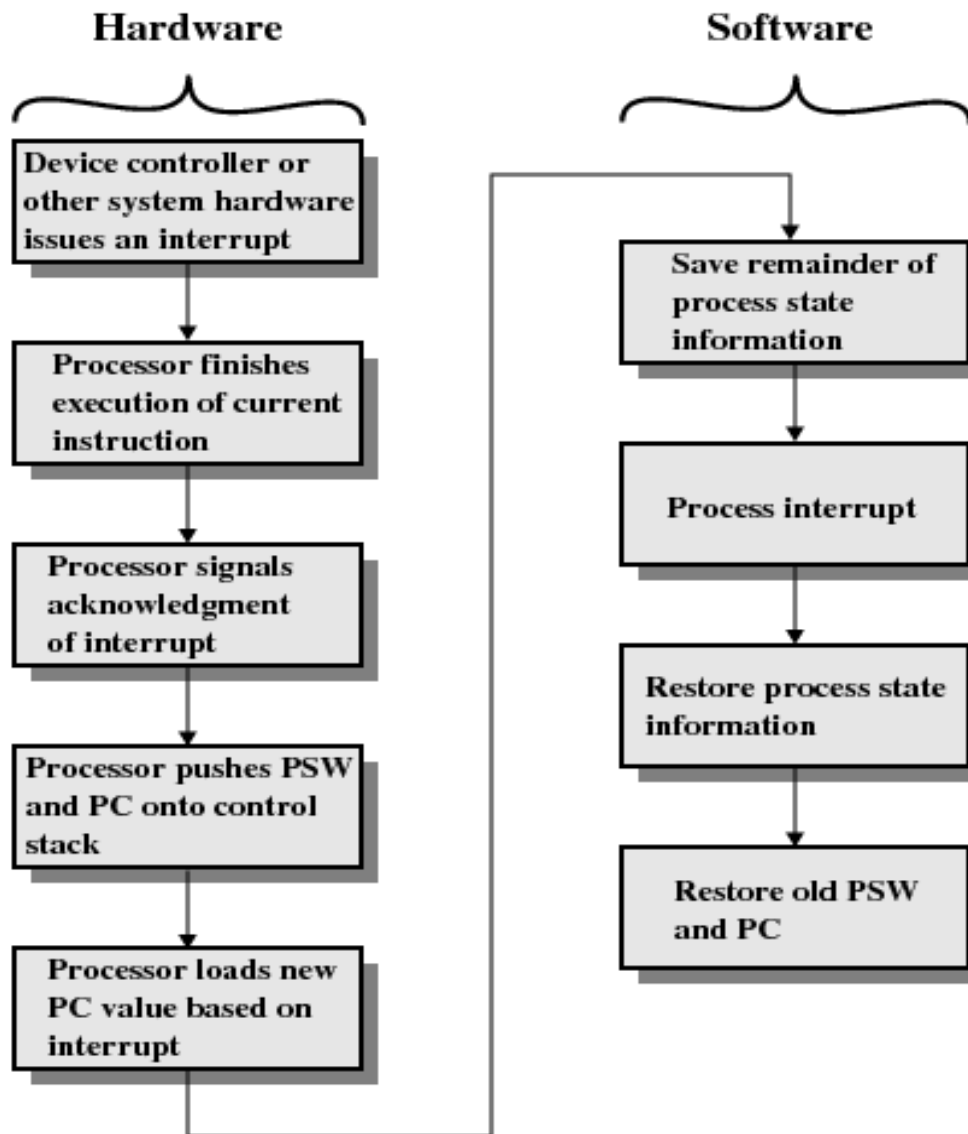


Figure 1.10 Simple Interrupt Processing

Realizácia vstupno-výstupnej operácie

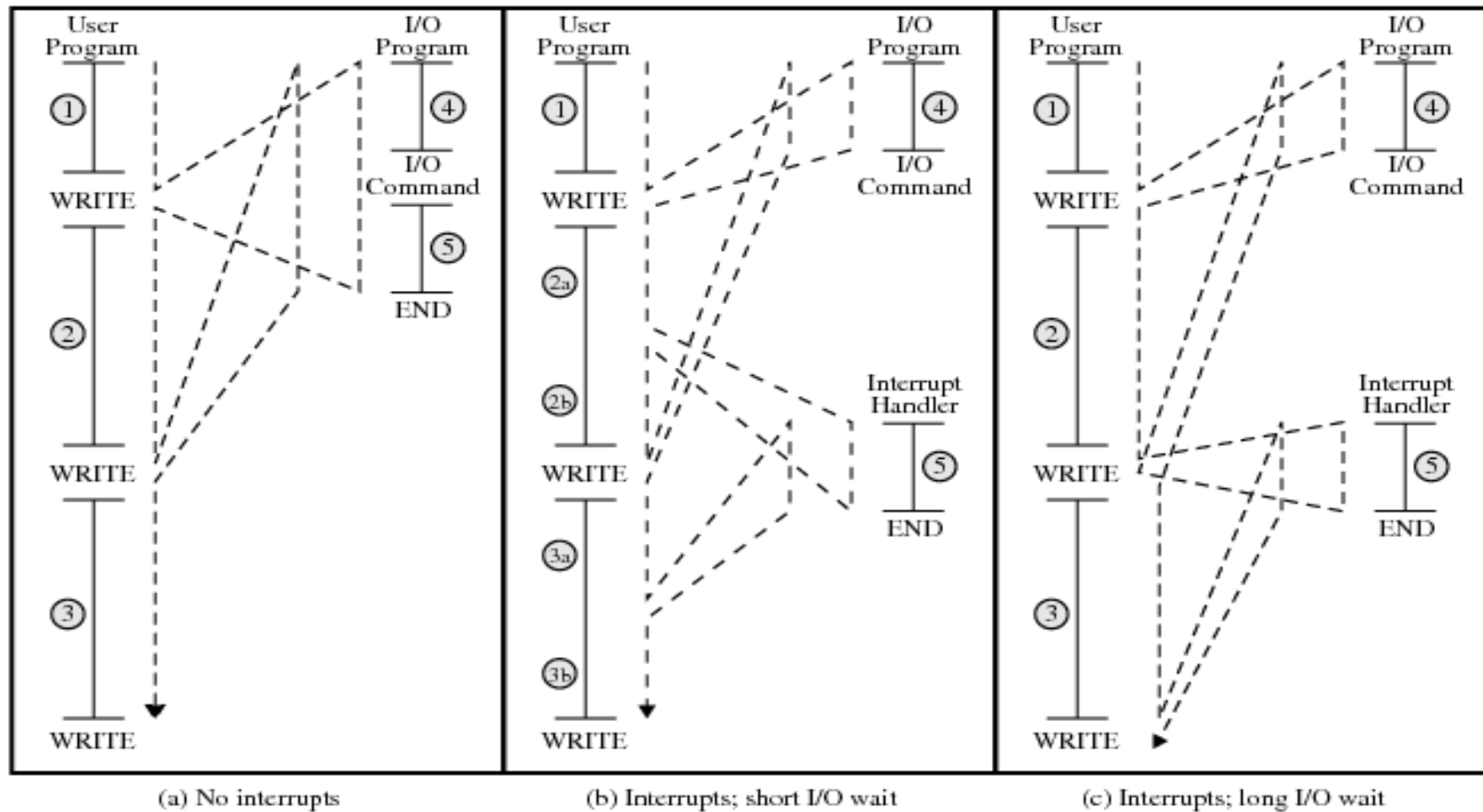


Figure 1.5 Program Flow of Control Without and With Interrupts

Realizácia vstupno-výstupnej operácie pomocou prerušení

- procesor odošle požiadavku V/V zariadeniu (4)
- V/V zariadenie požiadavku spracováva, procesor zatiaľ pokračuje s vykonávaním programu (2a)
- V/V zariadenie vyvolá prerušenie
- procesor obslúži prerušenie - získa výsledok operácie (5)
- procesor pokračuje v pôvodnej činnosti (2b)

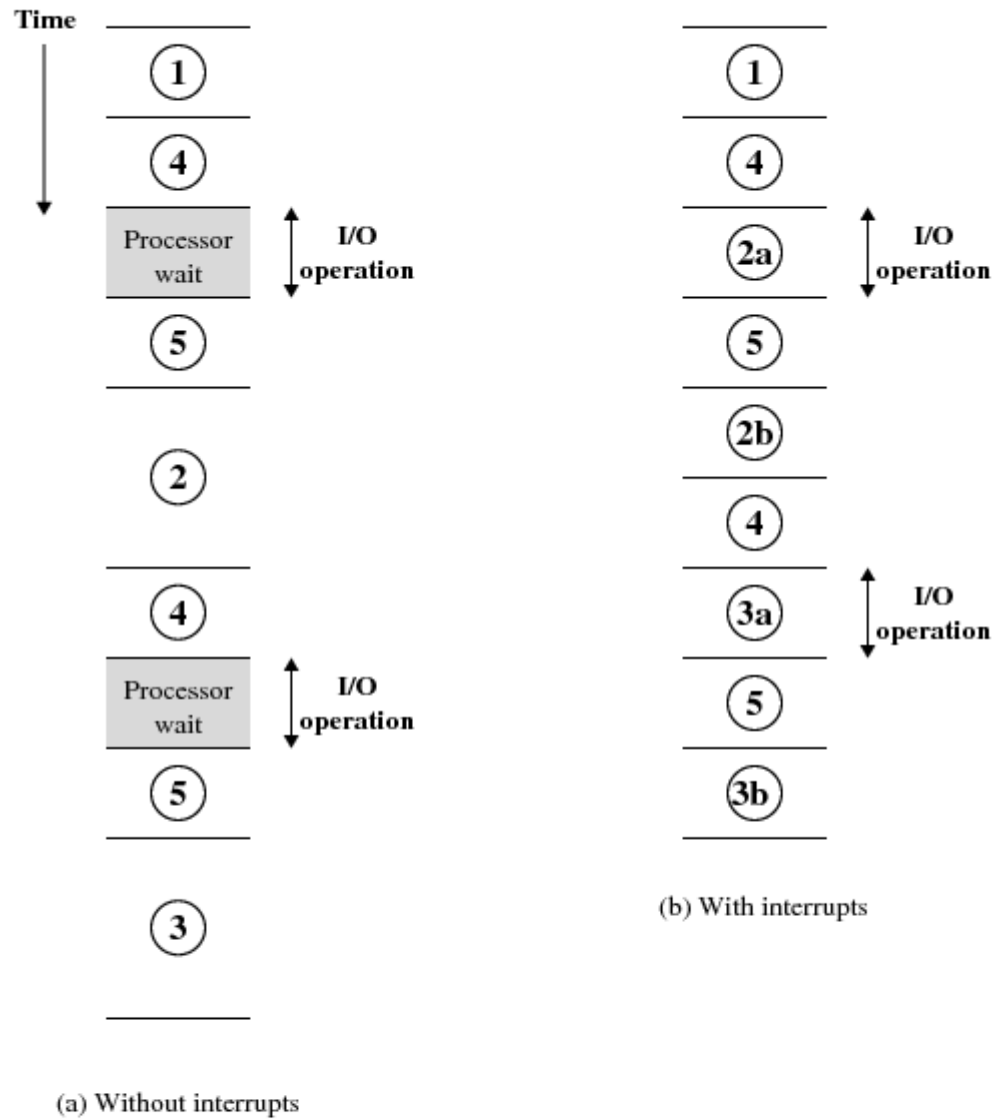
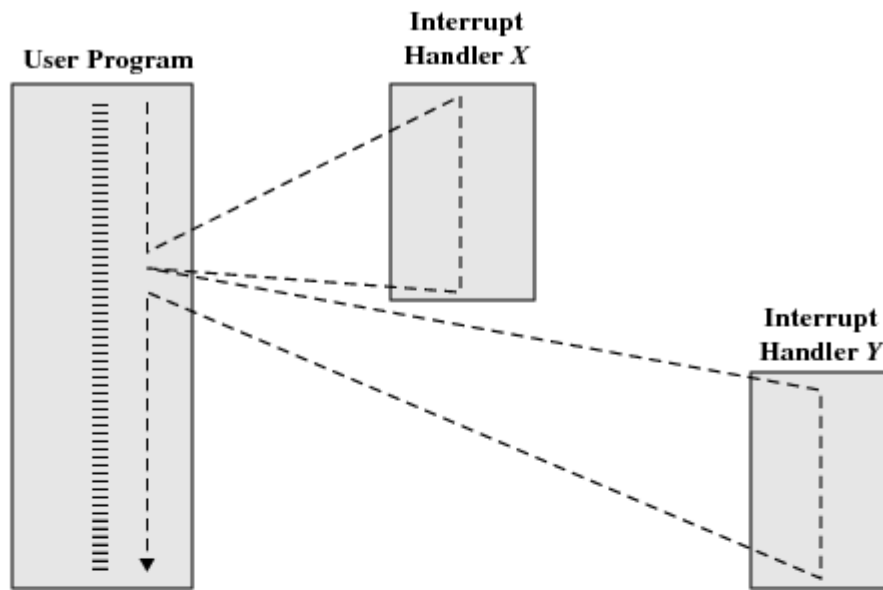


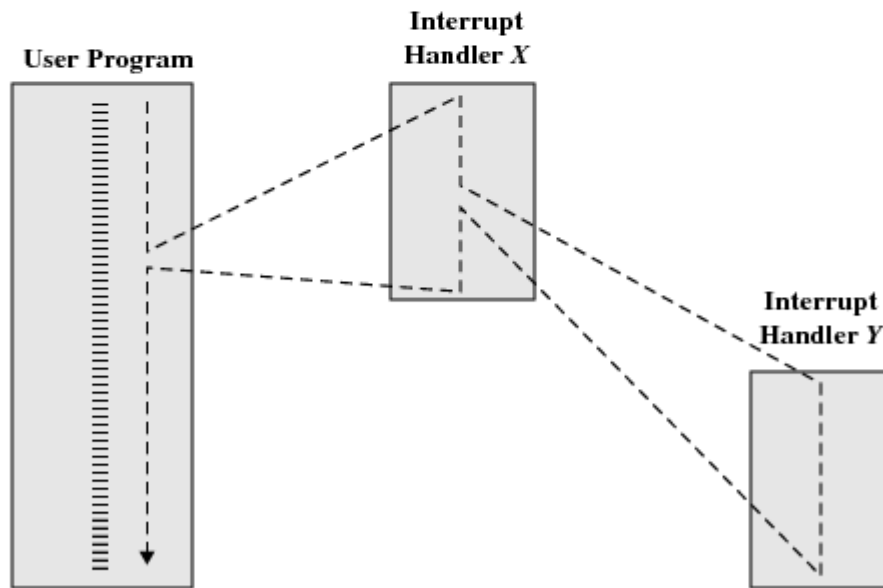
Figure 1.8 Program Timing: Short I/O Wait

Čo s viacerými prerušeniami?

- Keď počas obsluhy prerušenia vznikne ďalšie prerušenie
 - sekvenčná obsluha
 - obslúžia sa postupne
 - vnorené prerušenia
 - nové prerušenie preruší obsluhu staršieho
 - priority prerušení



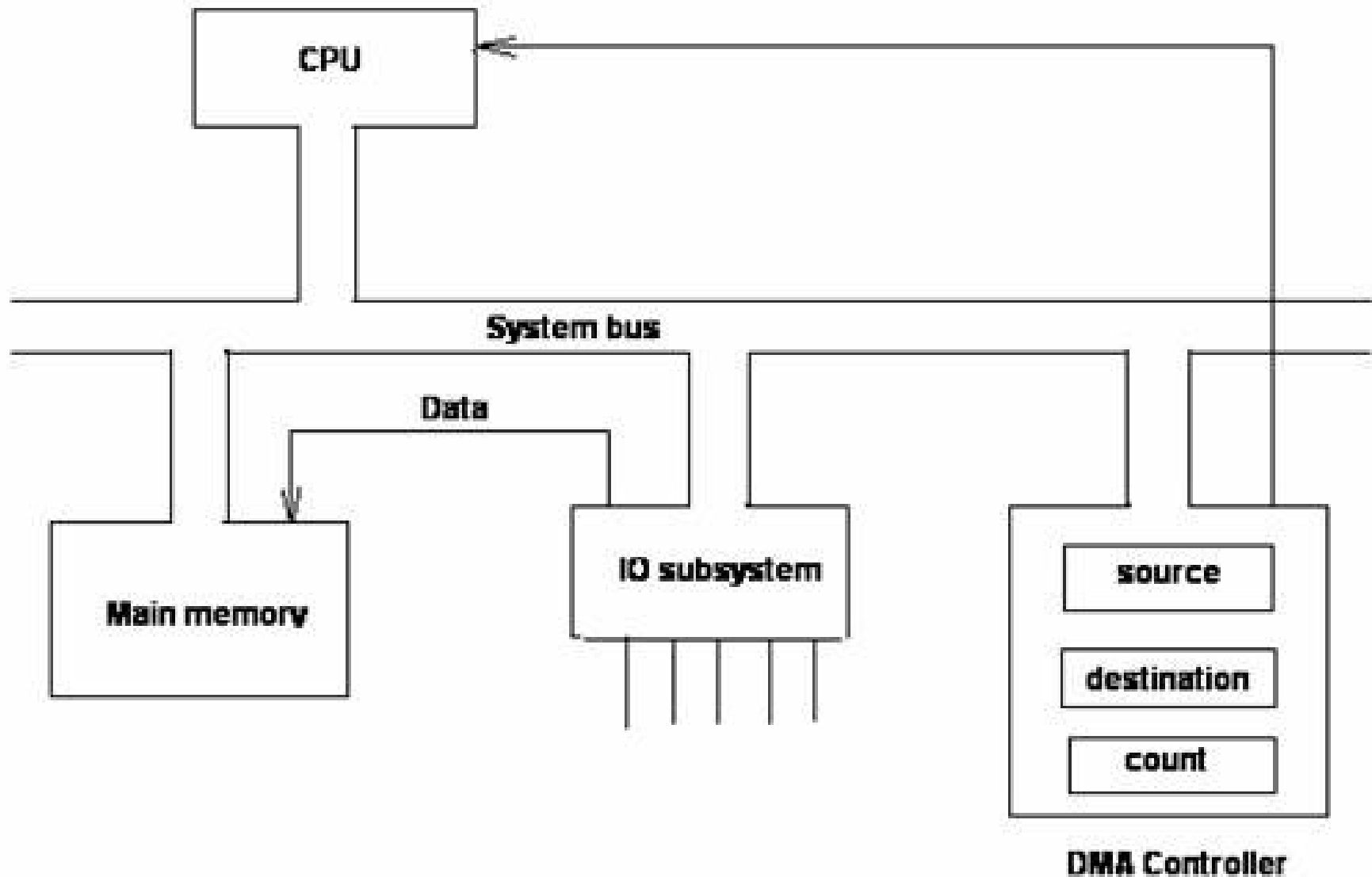
(a) Sequential interrupt processing



(b) Nested interrupt processing

Figure 1.12 Transfer of Control with Multiple Interrupts

Priamy prístup do pamäte (DMA)



Priamy prístup do pamäte (DMA)

- procesor len inicializuje informácie o prenose
 - odkiaľ, kam, koľko
 - a robí iné
- DMA radič zabezpečuje prenos po zbernici priamo medzi V/V zariadením a pamäťou
 - bez účasti procesora
- DMA radič generuje prerušenie po ukončení prenosu
 - procesor prečíta stavovú informáciu

Porovnanie

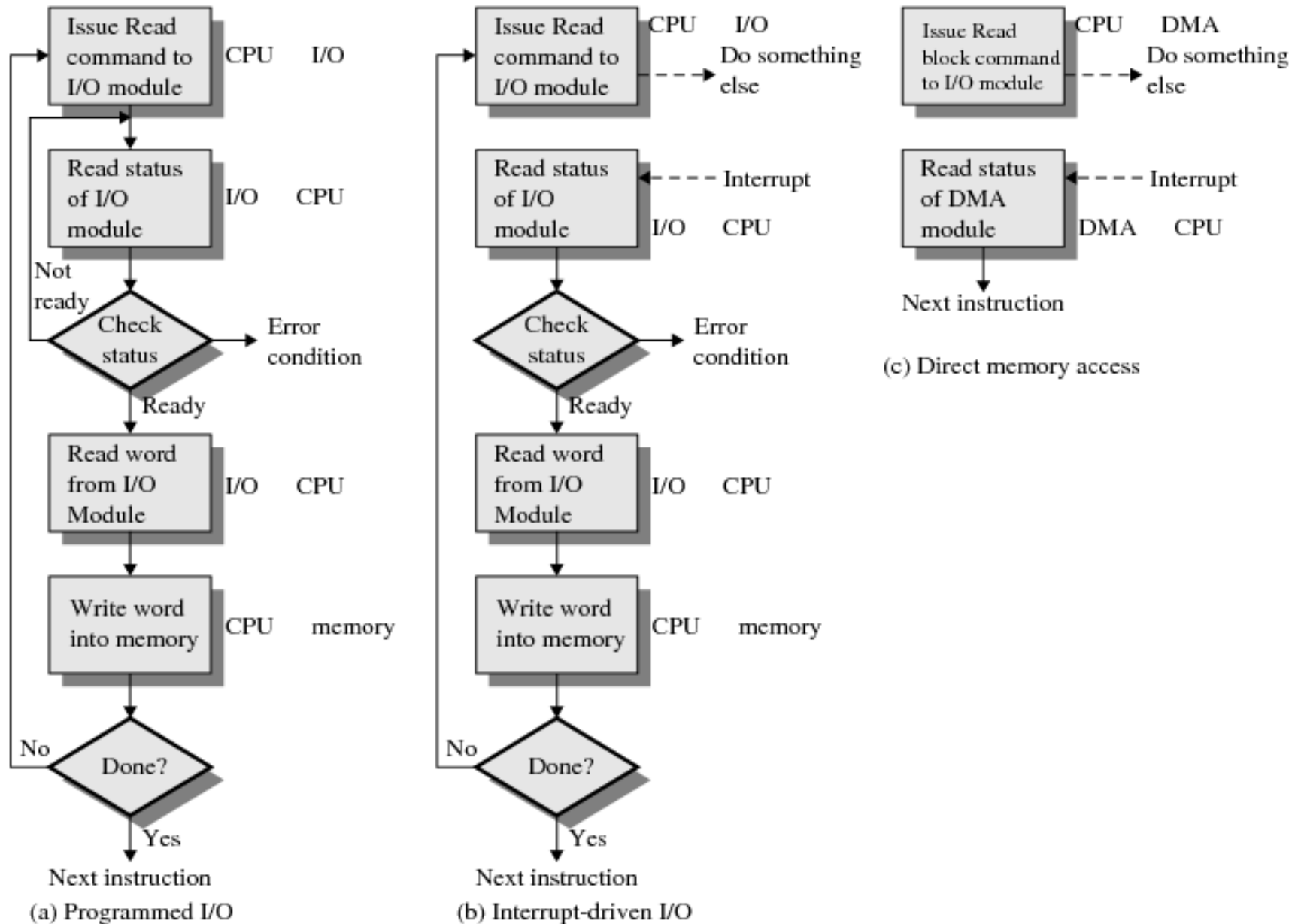


Figure 1.19 Three Techniques for Input of a Block of Data

Operační systémy

OS ako rozhranie medzi používateľom a hardvérom

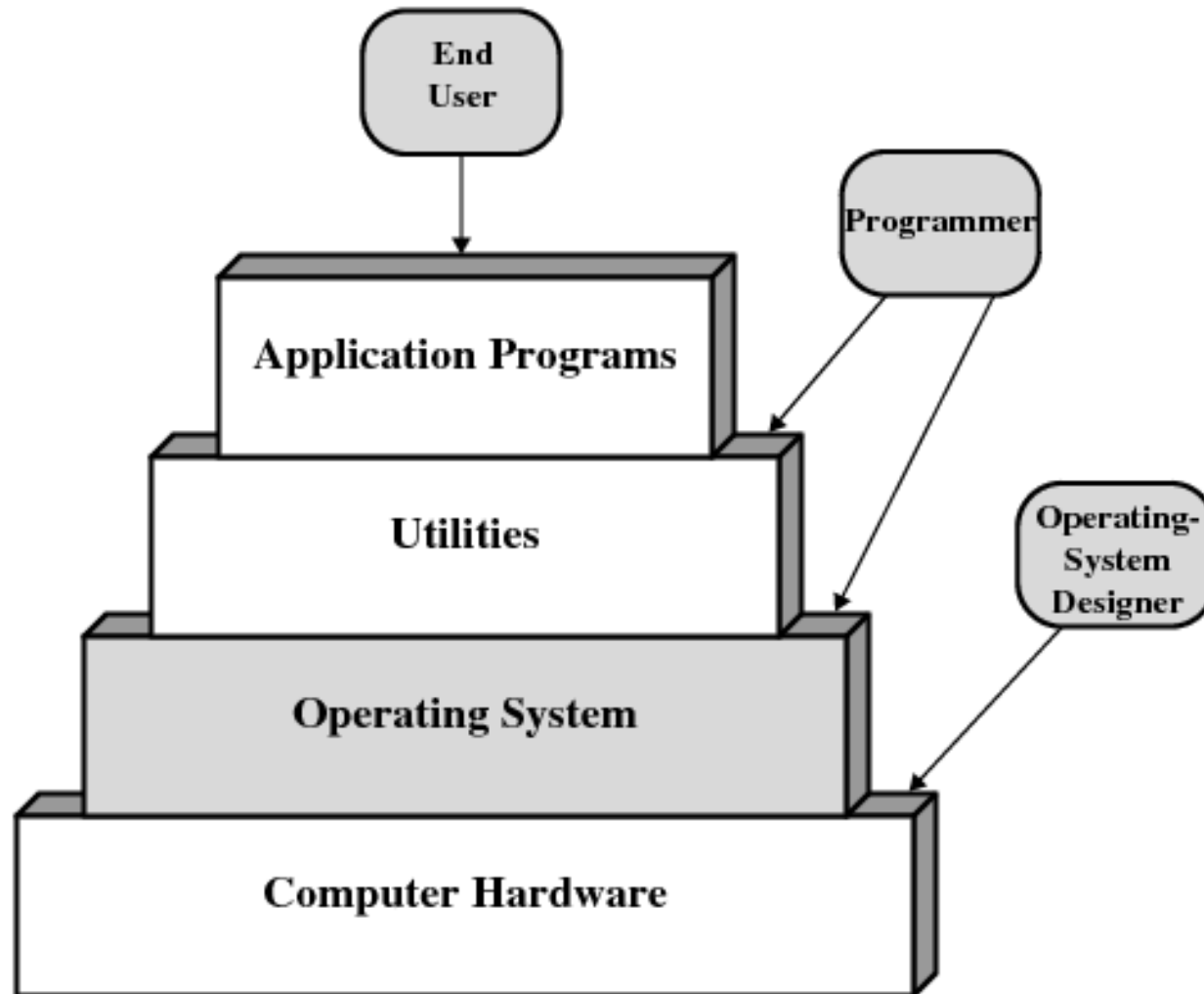


Figure 2.1 Layers and Views of a Computer System

OS ako rozhranie medzi používateľom a hardvérom

- OS
 - poskytuje (čiastočnú) abstrakciu hardvéru
 - pre aplikačné programy a pre používateľov
 - rieši komunikáciu s rôznymi HW zariadeniami
 - spravuje prostriedky (pamäť, disky, procesor(y), ...)
 - pridelovanie priestoru
 - striedanie programov
 - zabezpečuje riadenie prístupu a ochranu údajov
 - prístupové práva, používateľ vs. administrátor (správca)

Služby OS

- vykonávanie programov
- prístup k V/V zariadeniam
- prístup k súborom – súborové systémy
- bezpečnosť
 - identifikácia, autentifikácia, autorizácia
- detekcia a reakcia na chyby
- accounting (účtovanie)
 - štatistiky využitia zdrojov procesmi/používateľmi

Príklad štruktúry OS

- členenie
 - jadro systému (kernel)
 - základné funkcie
 - ovládače zariadení
 - komunikácia s rôznymi periférnymi zariadeniami
 - ďalšie služby
 - napr. grafické používateľské rozhranie, ...
- modularita
 - nutný predpoklad prispôsobiteľnosti novému hardvéru a vývoja

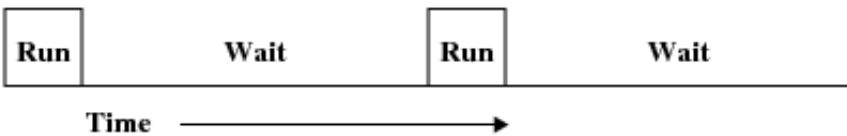
OS ako správca prostriedkov

- pamäť
 - pridelovanie fyzickej pamäte
 - virtuálna pamäť
- procesor
 - striedanie procesov
- V/V zariadenia
-
- OS je ale tiež len program :-)

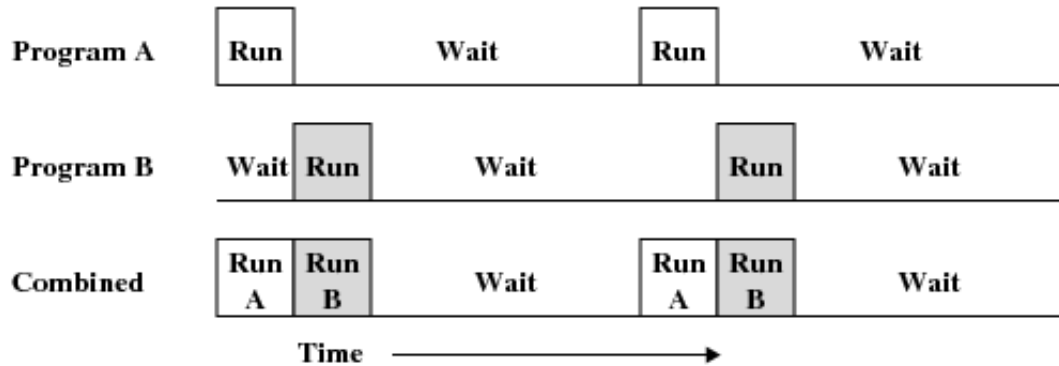
História OS

- systémy so sériovým spracovaním
 - len 1 program (job)
- dávkové (batch) systémy
 - pripravené programy (job-y) postupne za sebou
- multiprogramové batch systémy
- systémy so zdieľaním času
 - striedanie vykonávaných programov

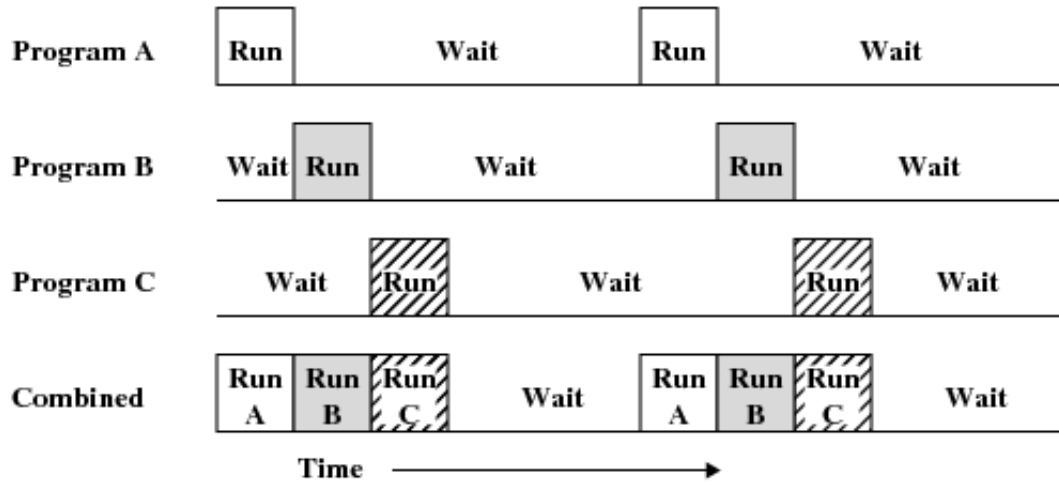
Multiprogramovanie



(a) Uniprogramming



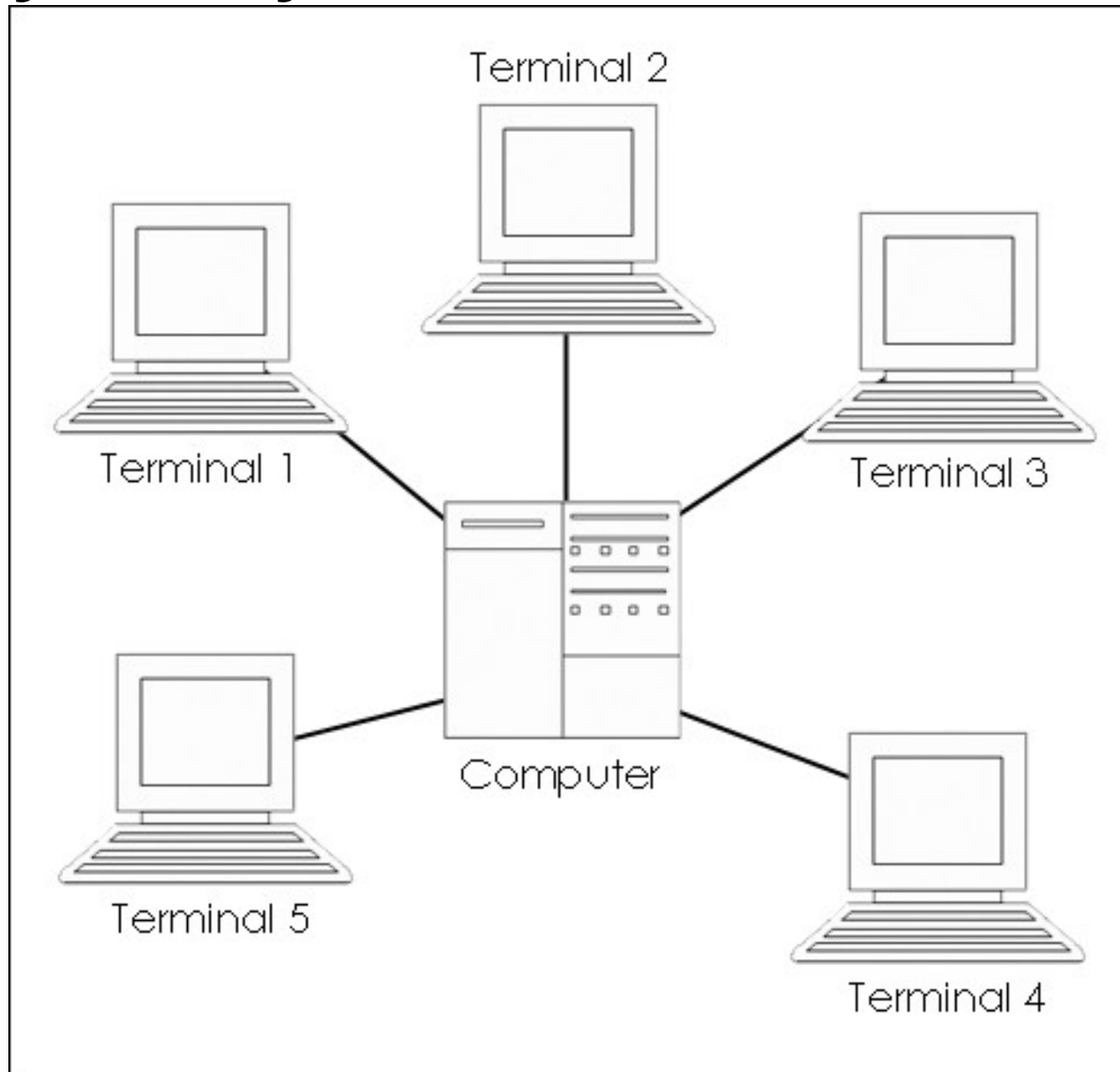
(b) Multiprogramming with two programs



(c) Multiprogramming with three programs

Figure 2.5 Multiprogramming Example

Systemy so zdieľaním času



Niektoré pojmy

- multitasking
 - schopnosť vykonávať viac úloh (programov) „naraz“
 - kooperatívny vs. preemptívny
- multithreading
 - jedna úloha môže mať viac „**vlákien**“, ktoré sa môžu vykonávať naraz
- multiprocessing (SMP)
 - viac procesorov
 - môžu byť aj v jednom čipe – dual core (2), quad core (4)

Príklady moderných OS

- domáce/kancelárske počítače
 - Windows XP, 7
 - Linux
 - Mac OS X
- servery
 - Windows Server
 - Linux
 - Solaris, HP-UX, AIX, ...
- tablety, mobily
 - Android, Windows Phone, iOS, Symbian, WebOS, ...

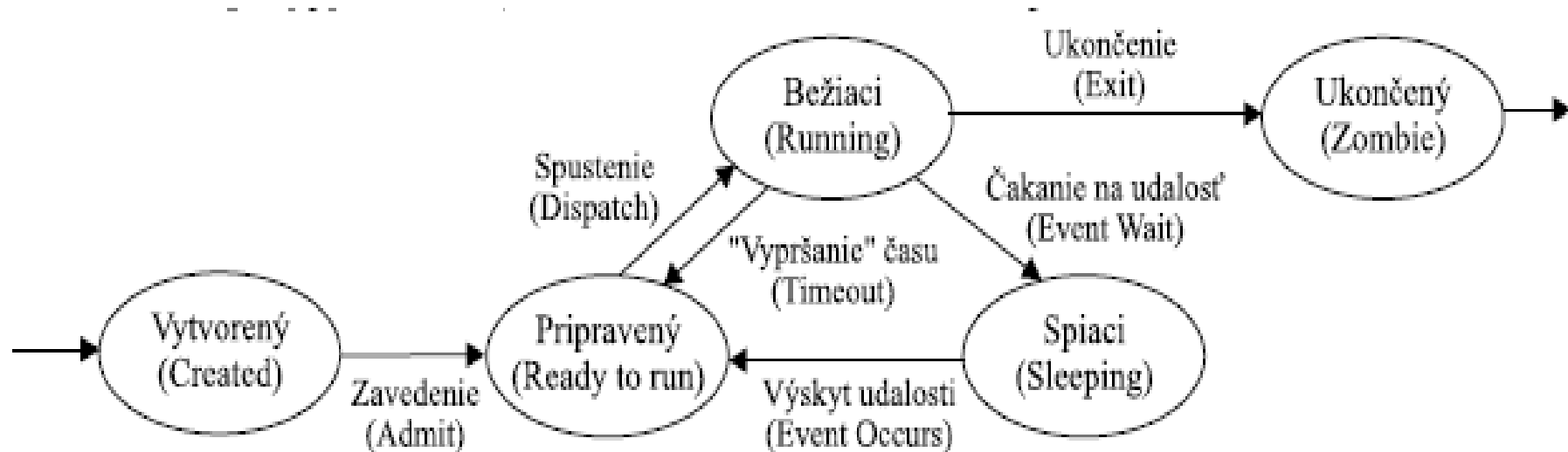
Operační systémy

Procesy

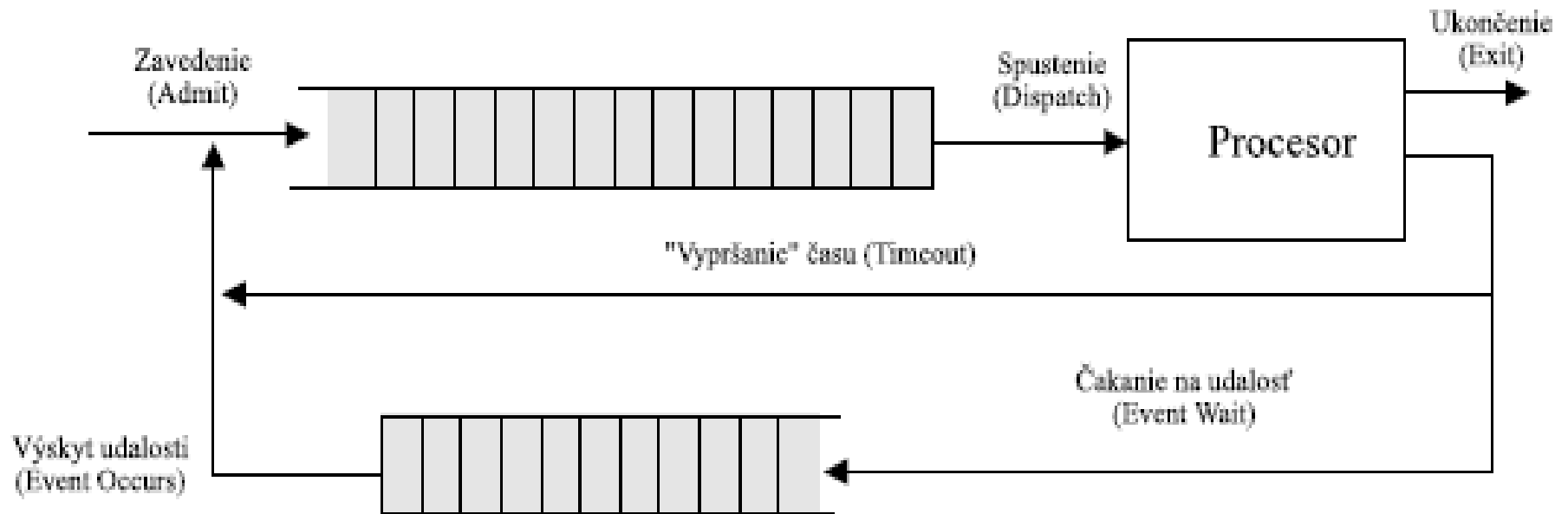
Čo je to proces

- ad-hoc riešenia striedania úloh => problémy
 - zlá synchronizácia, neželané ovplyvňovanie sa, uviaznutie, snaha o súčasný prístup k zariadeniam
- proces
 - vykonávaný program
 - dáta
 - kontext
 - registre, informácie o pridelených zdrojoch, ...

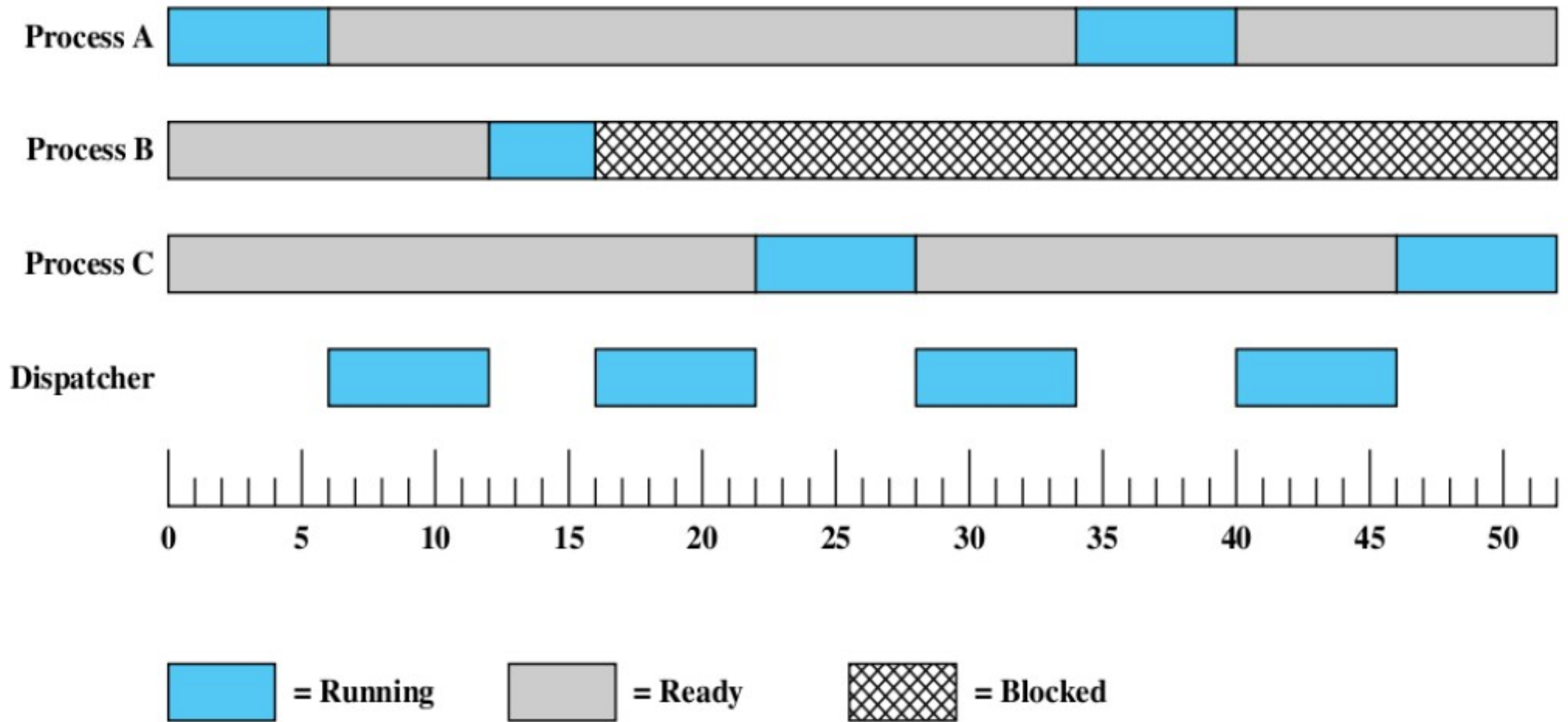
Stavy procesu a prechody medzi nimi



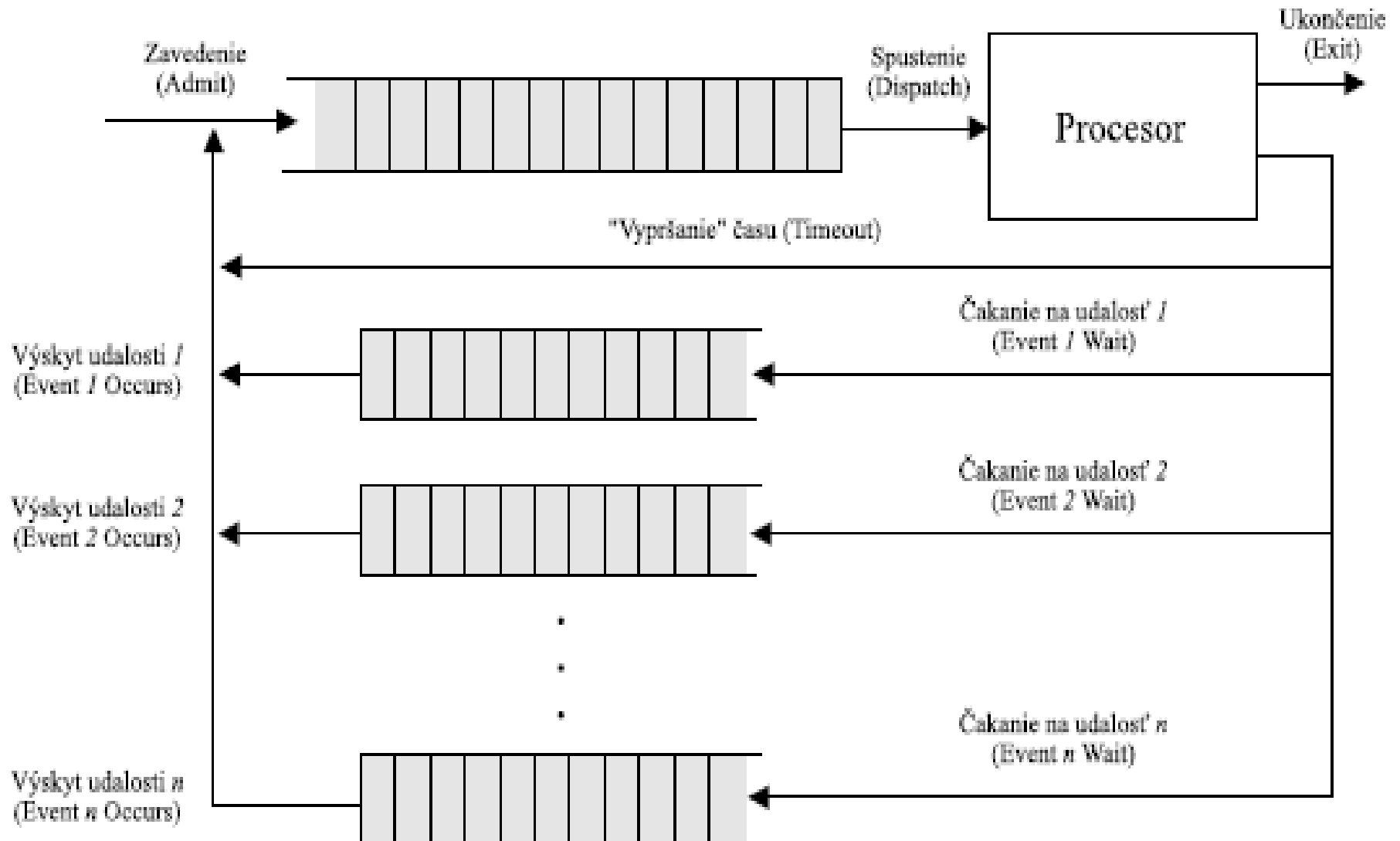
Zarad'ovanie procesov



Príklad striedania



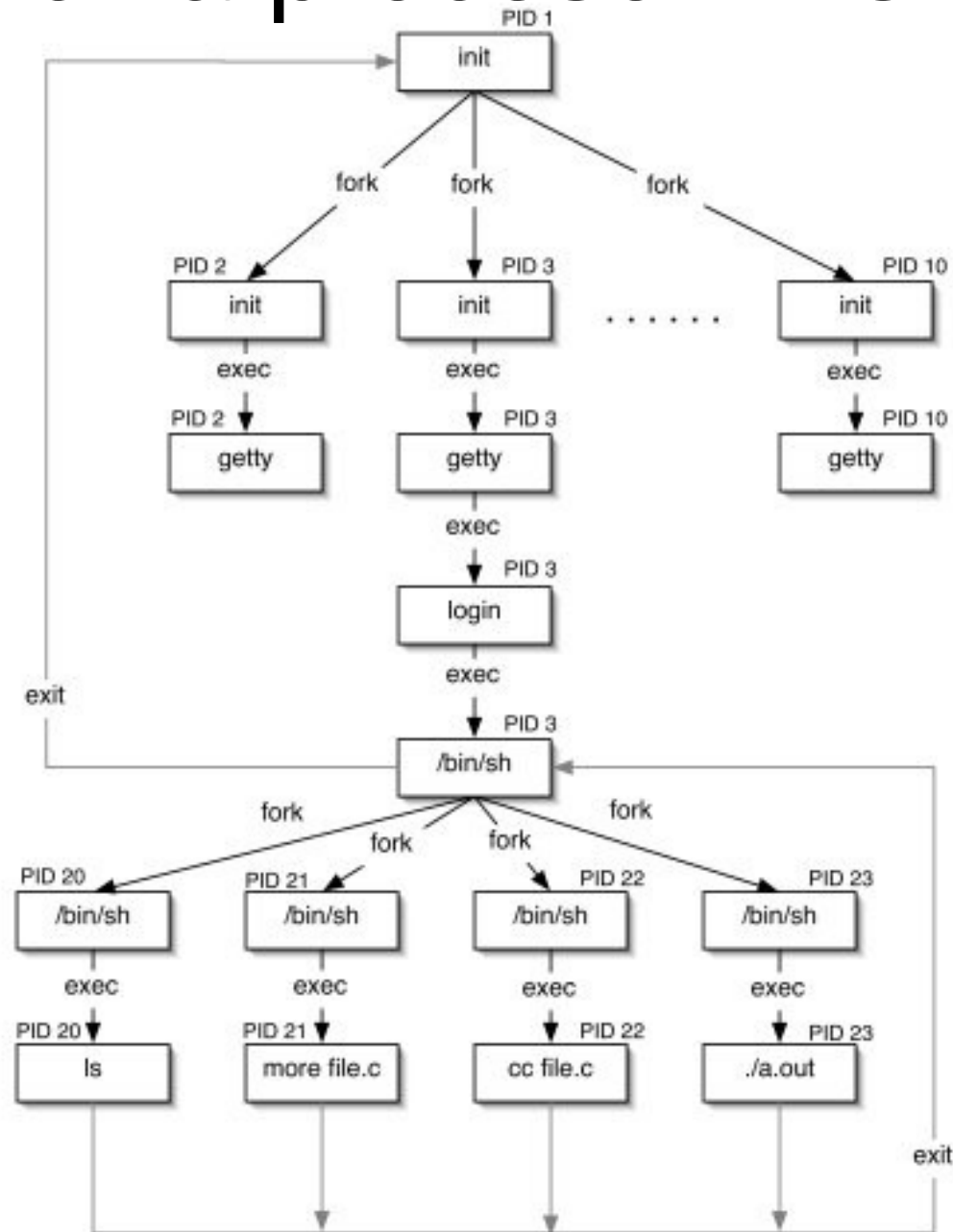
Zarad'ovanie procesov



Prechody medzi stavmi so swapovaním



Hierarchia procesov v UNIXe



Praktický pohľad

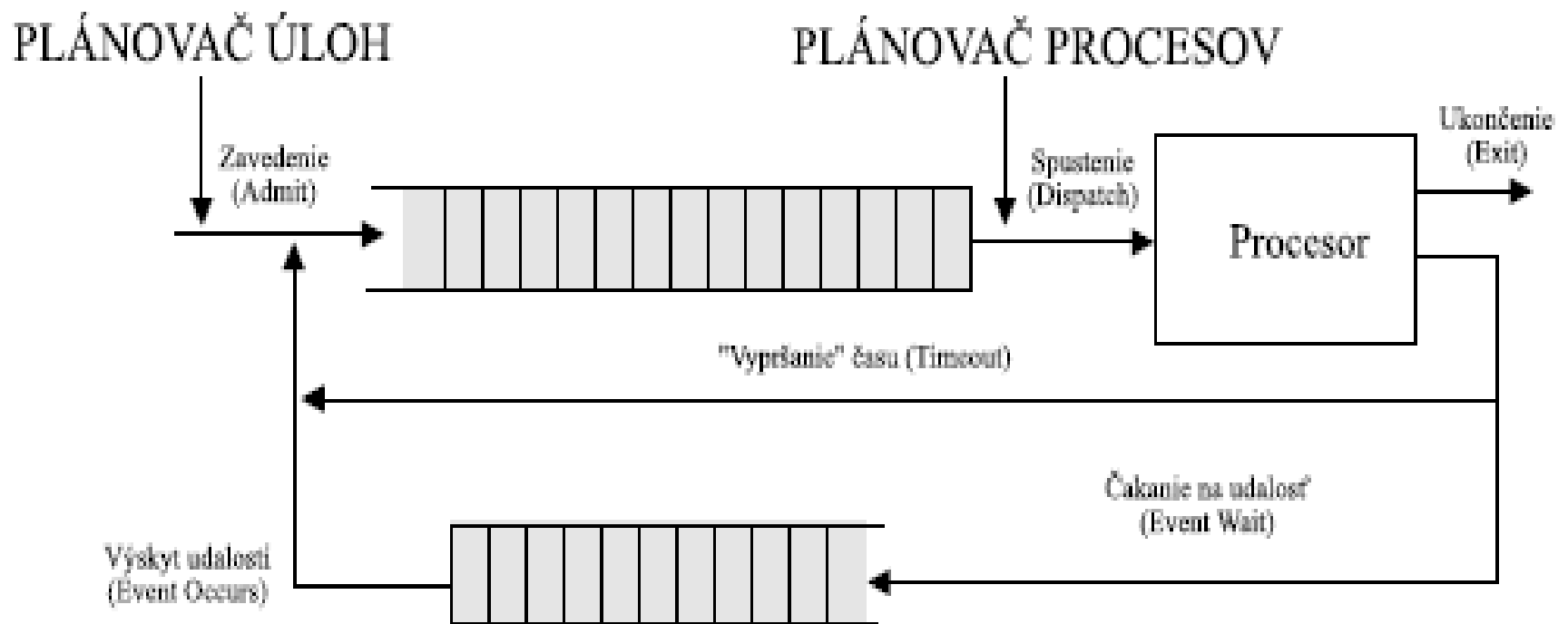
- Windows
- Task Manager
 - zobrazenie aplikácií a procesov
 - možnosť ukončiť proces
- UNIX/Linux
 - top
 - ps
 - kill

Plánovanie procesov

Plánovače

- Plánovač úloh
 - rozhoduje o zaradení úlohy na spracovanie
 - dôležitý v dávkových systémoch
 - v dnešných bežných systémoch ho sčasti nahrádza používateľ
 - sčasti stále existuje na spúšťanie dávkových úloh
- Plánovač procesov
 - rozhoduje o pridelovaní procesoru jednotlivým procesom

Plánovače



Plánovače

- nepreemptívne
 - nemôžu odobrať procesor procesu/úlohe, kým sa ho sám nevzdá alebo nezostane čakať na V/V operáciu
- preemptívne
 - môžu procesu/úlohe odobrať procesor

First Come First Served (FCFS)

Proces	Čas zadania	Čas spracovania (T_s)	Čas spustenia	Čas ukončenia	Doba prechodu (T_q)	$\frac{T_q}{T_s}$
1	0	3	0	3	3	1.00
2	2	6	3	9	7	1.17
3	4	4	9	13	9	2.25
4	6	5	13	18	12	2.40
5	8	2	18	20	12	6.00
Priemer					8.60	2.56

First Come First Served (FCFS)

Proces	Čas zadania	Čas spracovania (T_s)	Čas spustenia	Čas ukončenia	Doba prechodu (T_q)	$\frac{T_q}{T_s}$
1	0	1	0	1	1	1
2	1	100	1	101	100	1
3	2	1	101	102	100	100
4	3	100	102	202	199	1.99
Priemer					100	26

Shortest Job First (SJF)

Proces	Čas zadania	Čas spracovania (T_s)	Čas spustenia	Čas ukončenia	Doba prechodu (T_q)	$\frac{T_q}{T_s}$
1	0	3	0	3	3	1.00
2	2	6	3	9	7	1.17
3	4	4	11	15	11	2.75
4	6	5	15	20	14	2.80
5	8	2	9	11	3	1.50
Priemer					7.60	1.84

Highest Response Ratio Next (HRN)

priorita=(čas čakania + čas spracovania) / čas spracovania

Proces	Čas zadania	Čas spracovania (T_s)	Čas spustenia	Čas ukončenia	Doba prechodu (T_d)	$\frac{T_d}{T_s}$
1	0	3	0	3	3	1.00
2	2	6	3	9	7	1.17
3	4	4	9	13	9	2.25
4	6	5	15	20	14	2.80
5	8	2	13	15	7	3.50
Priemer					8.00	2.14

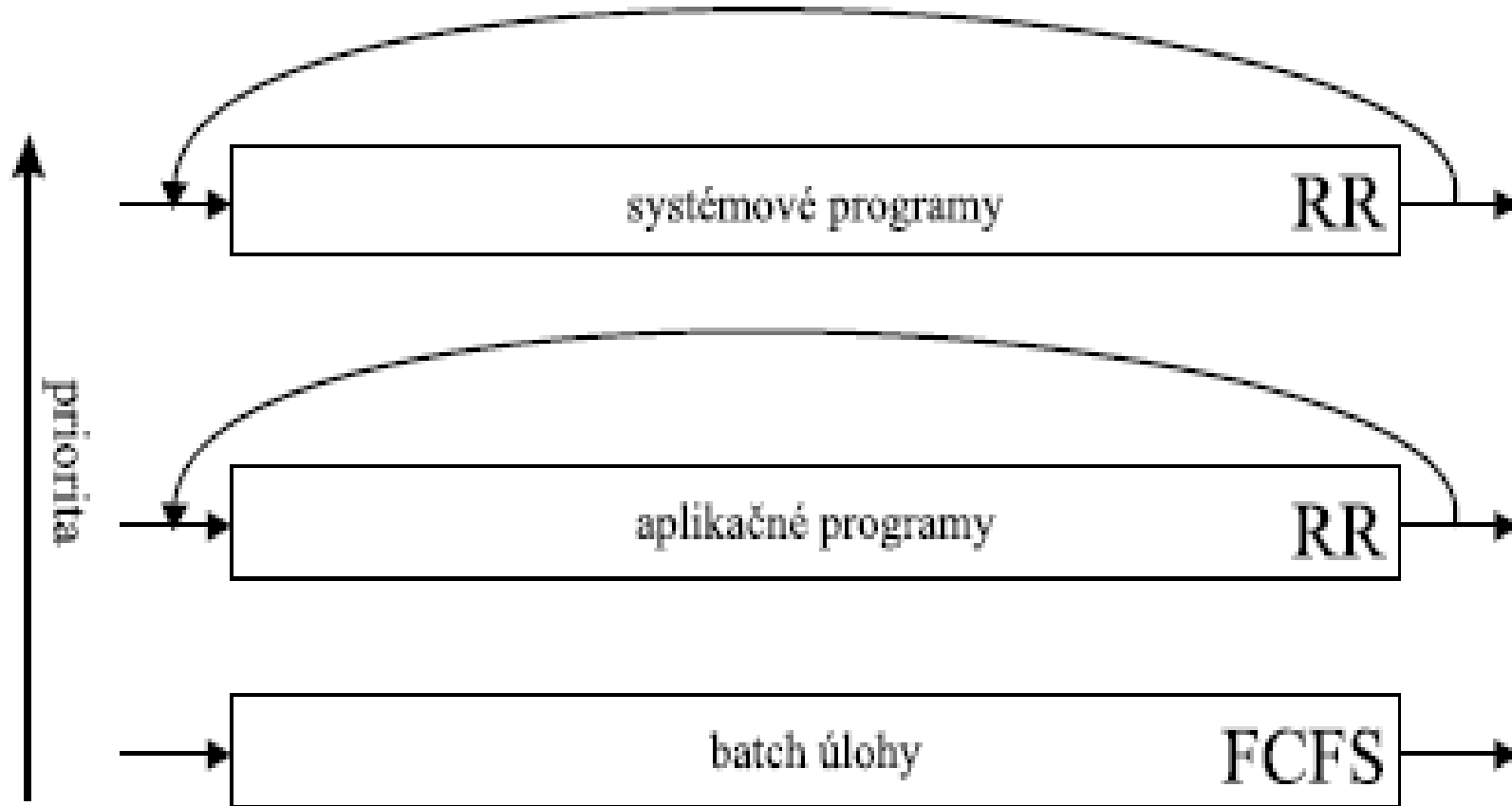
Shortest Remaining Time (SRT)

Proces	Čas zadania	Čas spracovania (T_s)	Čas ukončenia	Doba prechodu (T_T)	$\frac{T_d}{T_e}$
1	0	3	3	3	1.00
2	2	6	15	13	2.17
3	4	4	8	4	1.00
4	6	5	20	14	2.80
5	8	2	10	2	1.00
Priemer				7.20	1.59

Round Robin (RR)

- cyklický zoznam pripravených procesov
- každý proces dostane maximálne určený čas
 - ak sa dovtedy nevzdá procesoru, je prerušený
 - zaradí sa na koniec zoznamu

Stratégia s viacerými zoznamami



Viac zoznamov s premiastňovaním

- niekoľko zoznamov s RR
 - vyššia priorita, nižšie kvantum
 - nižšia priorita, vyššie kvantum
 - ak proces vyčerpá malú časť svojho času, presunie sa do vyššieho zoznamu
 - ak proces vyčerpá celé kvantum, presunie sa do nižšieho zoznamu
- automatické preradovanie procesov podľa charakteru

Synchronizácia procesov

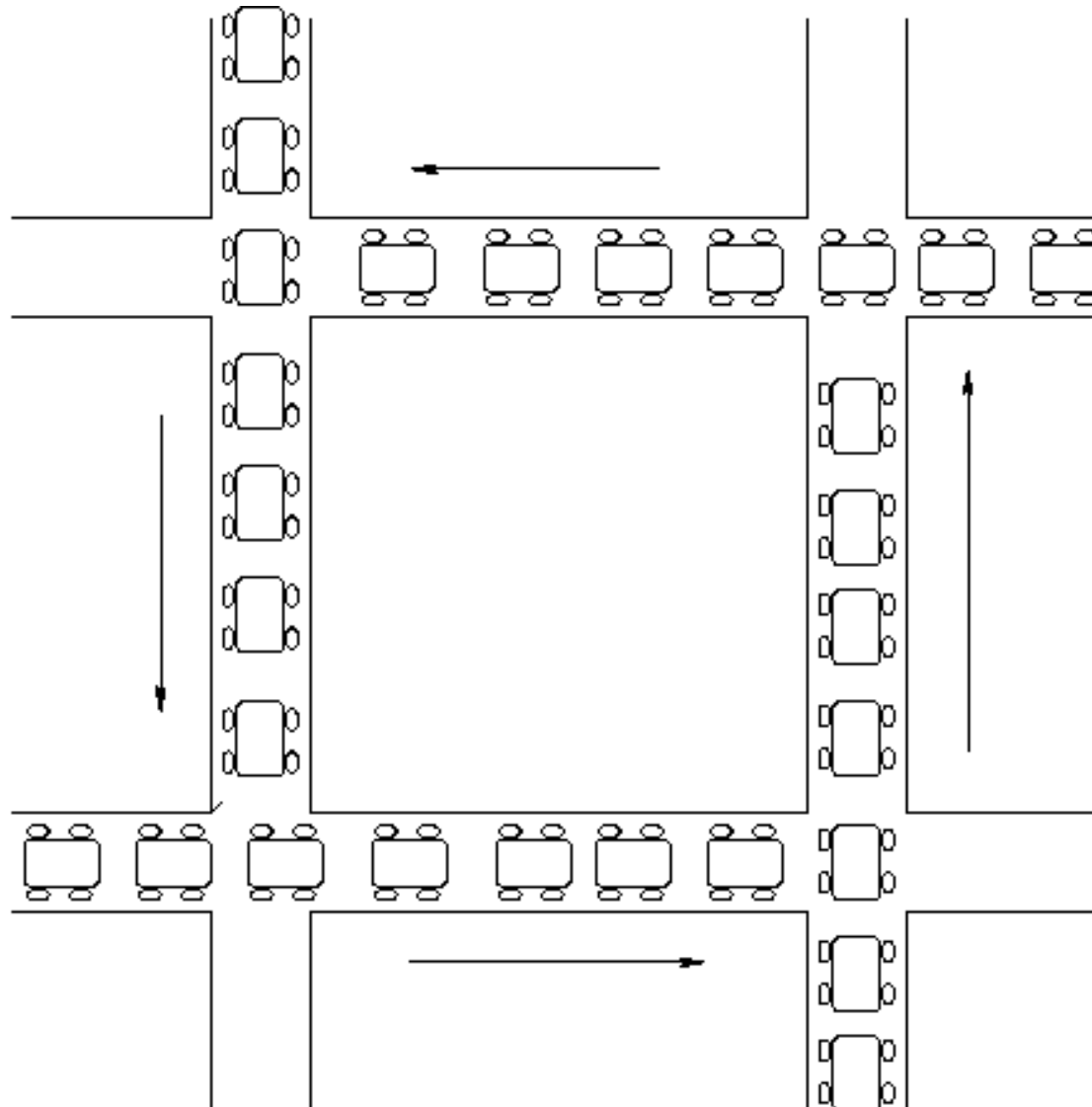
Vzájomné vylúčenie

- prostriedky často neumožňujú „súčasné“ používanie viacerými procesmi
 - napríklad: tlačiareň
- race condition
 - výsledok situácie závisí od „načasovania“ striedania procesov
- vzájomné vylúčenie (mutual exclusion)
 - zakážeme viacerým procesom pracovať so zdieľaným prostriedkom naraz

Deadlock – uviaznutie

- Podmienky:
 - vzájomné vylúčenie
 - postupné získavanie prostriedkov s čakaním
 - nemožnosť prerozdelenia prostriedkov
 - cyklické čakanie

Uviaznutie v praxi :-)



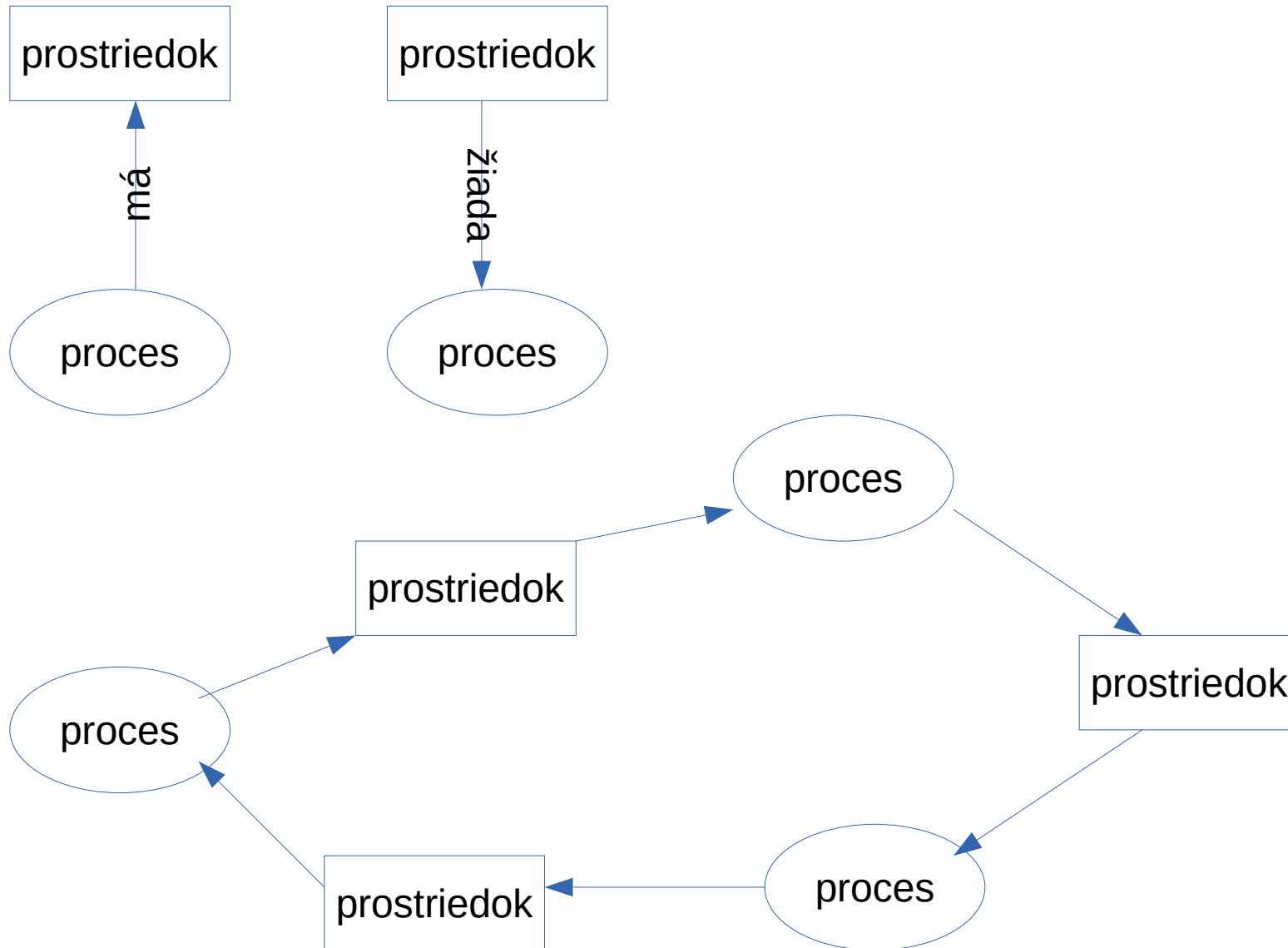
Prístupy k riešeniu uviaznutia

- „pštroší algoritmus“ - ignorovanie problému
- detekcia a vyvedenie
- prevencia
- vyhýbanie sa

Pštrosí algoritmus



Detekcia a vyvedenie



Detekcia a vyvedenie

- Ak graf čakania obsahuje cyklus, nastalo uviaznutie
 - riešením je odstrániť niektorý z procesov
- Ak neobsahuje cyklus, uviaznutie (zatiaľ) nenastalo
 - existuje aspoň jeden proces, ktorý môže pokračovať
 - nečaká na žiadny obsadený prostriedok

Prevencia

- znemožníme splniteľnosť niektorej z podmienok
 - vzájomné vylúčenie
 - napr. spooling pre tlačiareň
 - postupné získavanie prostriedkov
 - proces bude musieť všetky prostriedky získať naraz
 - nemožnosť prerozdelenia prostriedkov
 - umožníme systému odoberať prostriedky procesom
 - napr. procesor, pamäť
 - cyklické čakanie
 - očíslovanie prostriedkov a obmedzenie poradia ich získavania

Vyhýbanie sa

- zabránenie súčasnému splneniu všetkých podmienok uviaznutia
- príklad – bankárov algoritmus
 - predpoklady:
 - poznáme max. množstvo požadovaných prostriedkov jednotlivých typov
 - ak proces dostane všetky požadované prostriedky, v konečnom čase skončí a prostriedky vráti
 - systém garantuje pridelenie prostriedkov v konečnom čase

Bankárov algoritmus

- pred každým pridelením prostriedku skúmame, či výsledný stav by bol bezpečný
 - ak áno, pridelenie schválime
 - ak nie, pridelenie zamietneme (odložíme na neskôr)
- bezpečný stav
 - existuje usporiadanie procesov, ktoré umožní všetkým procesom skončiť

Bankárov algoritmus

- $M[i]$ = max. prostriedky pre proces i
- $A[i]$ = aktuálne pridelené pre proces i
- V = voľné prostriedky
- algoritmus:
 - $W=V$, nič neoznačené
 - opakuj kým existujú neoznačené i :
 - nájdí i , kde $M[i]-A[i] \leq V$
 - i označ a $W=W+A[i]$
 - ak také i neexistuje, stav nie je bezpečný
 - ak je všetko označené, stav je bezpečný, inak nie je

Správa pamäte

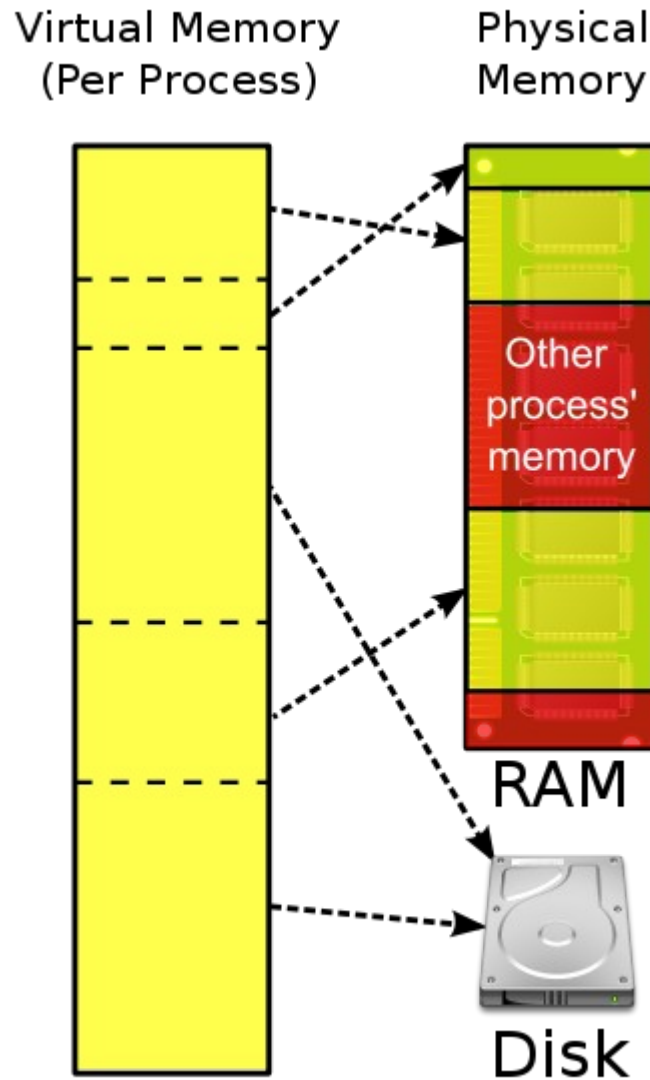
Úlohy správy pamäte

- evidencia voľnej pamäte
- pridelovanie a uvoľňovanie pamäte procesom
- ochrana prístupu do pamäte

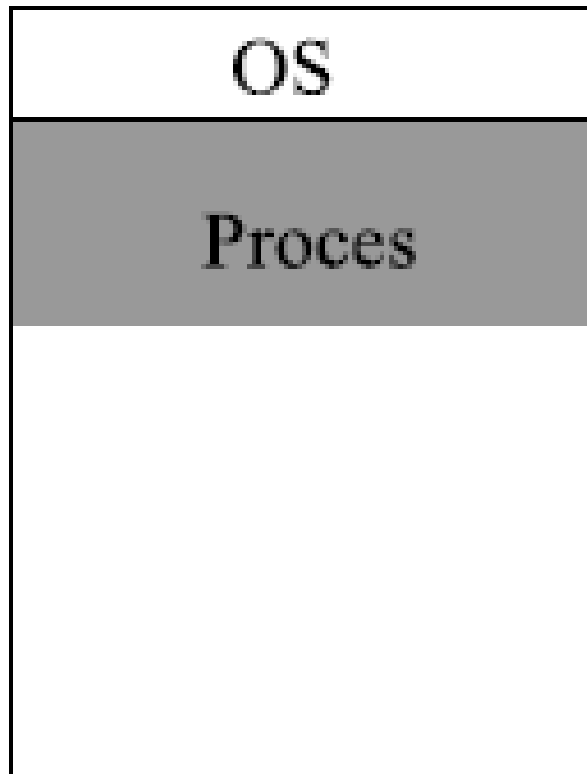
Logické vs. fyzické adresy

- Logické adresy
 - z pohľadu procesu
- Fyzické adresy
 - z pohľadu pamäte
- Vzťah môže byť identita, ale aj zložitejšia funkcia
- umožňujú abstrahovať od fyzického umiestnenia (častí) procesu vo fyzickej pamäti

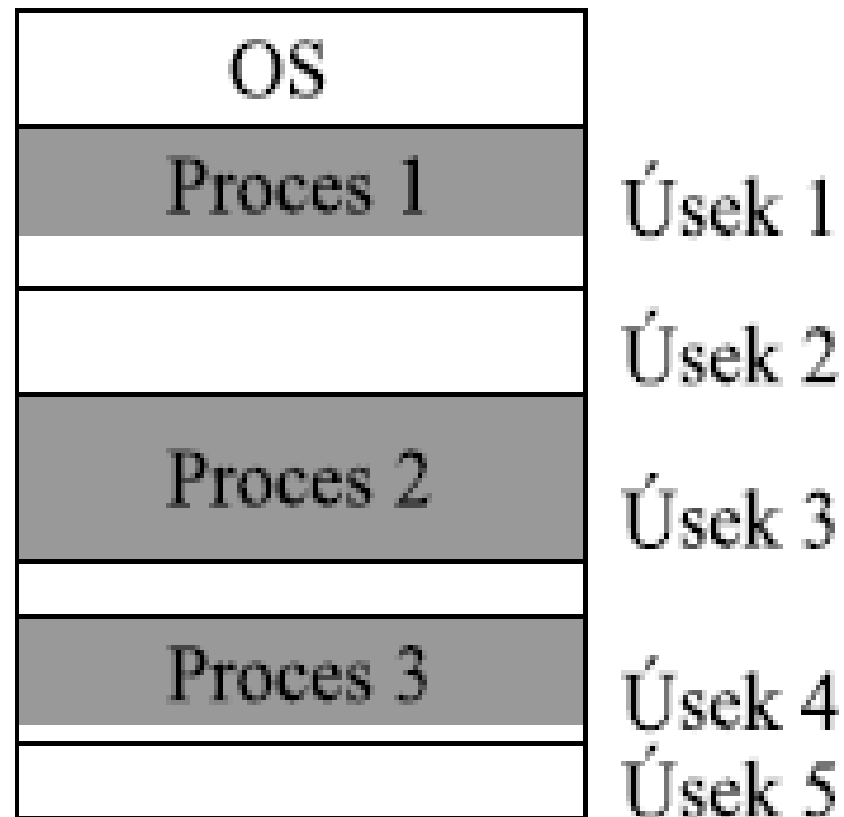
Logické vs. fyzické adresy



Jeden súvislý úsek (monoprogramovanie)



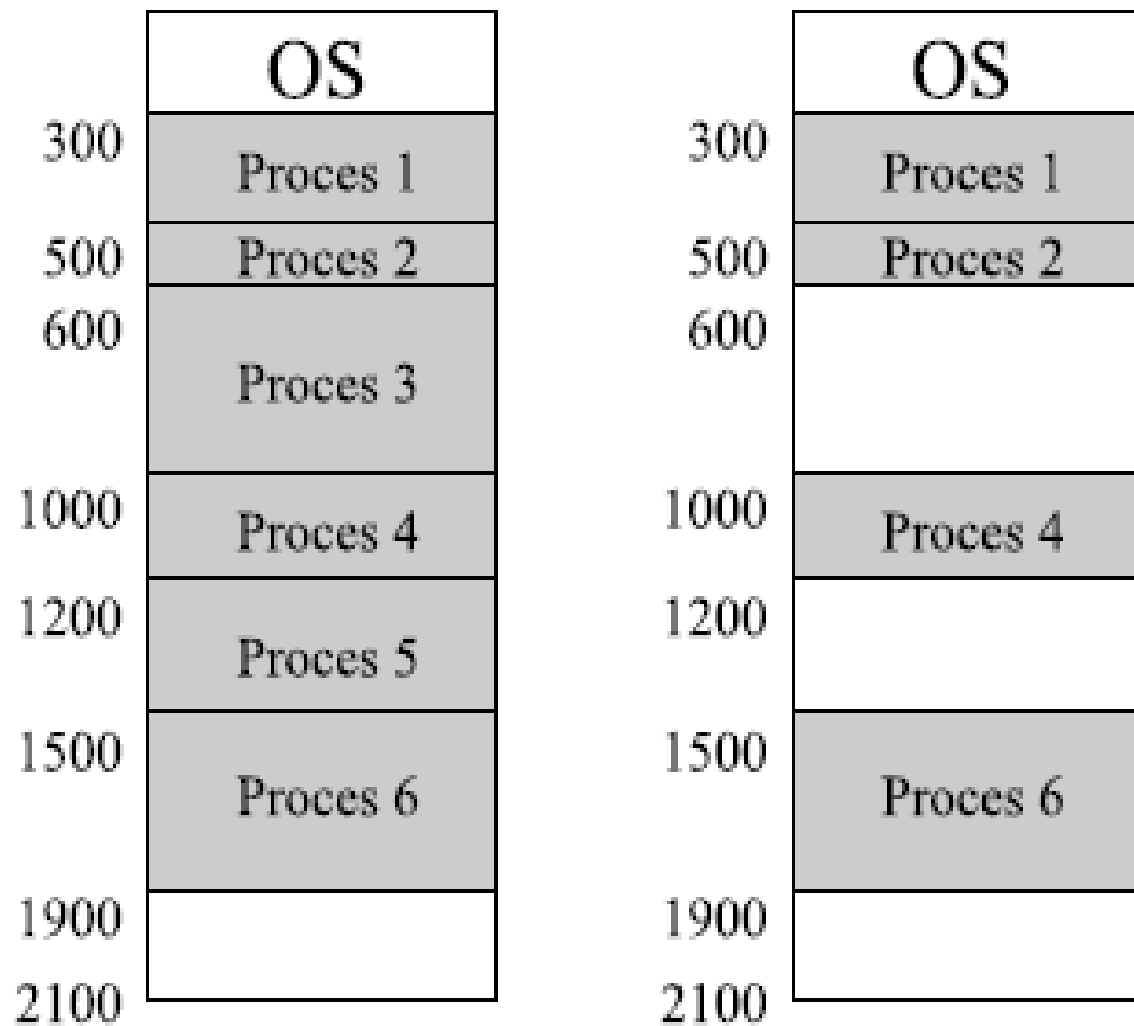
Statické súvislé úseky



Statické súvislé úseky

- vonkajšia fragmentácia
 - celkový objem pamäte je dostatočný, no neexistuje voľný vhodný úsek pamäte pre proces
- vnútorná fragmentácia
 - proces má pridelený úsek, ktorý nevyužíva celý
- výber vhodného úseku
 - first fit
 - best fit

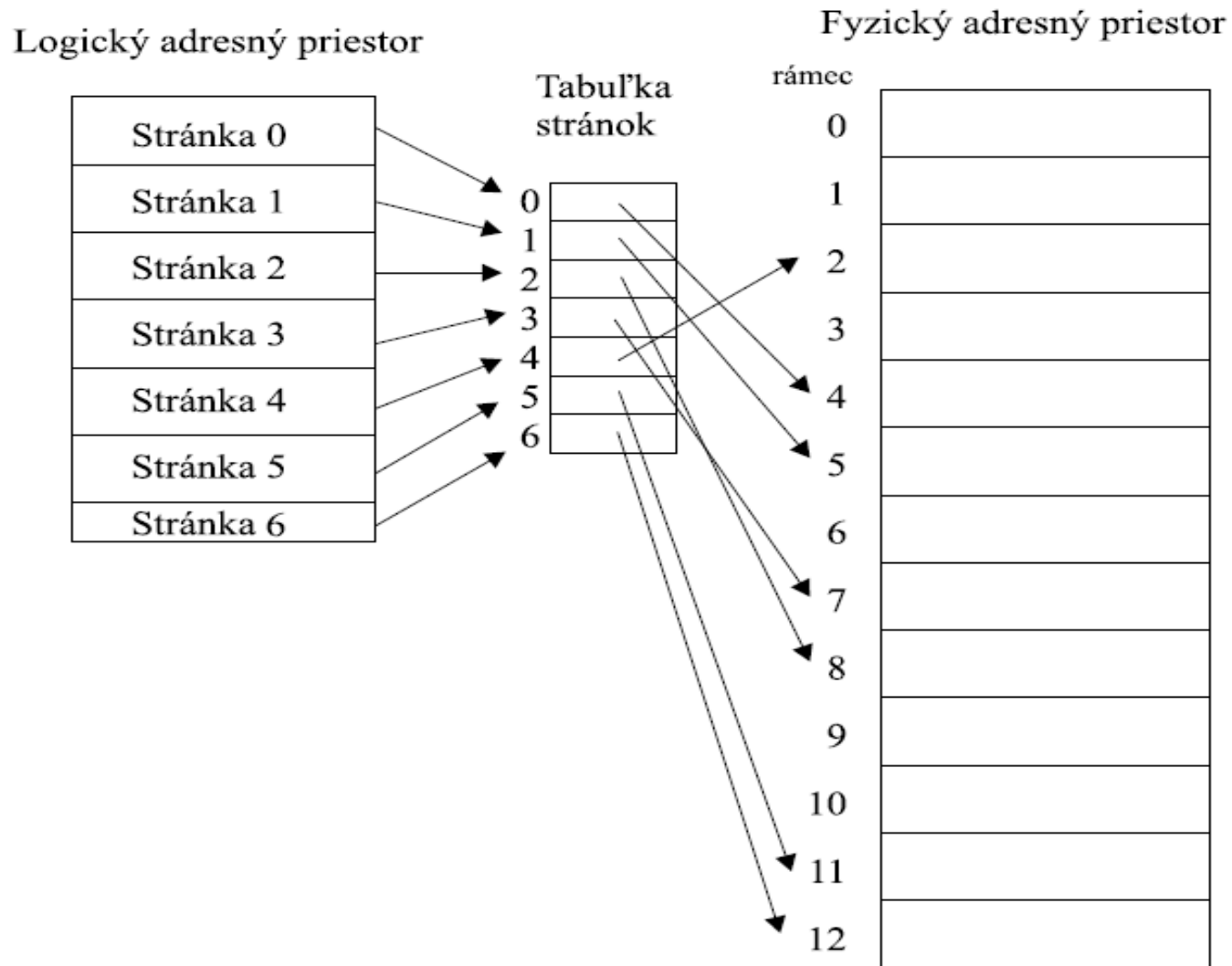
Dynamické súvislé úseky



Dynamické súvislé úseky

- odstraňuje problém s vnútornou fragmentáciou
 - úsek sa veľkosťou prispôsobí na mieru
- vonkajšia fragmentácia zostáva
 - teoreticky sa dá robiť defragmentácia, ale je to veľmi náročné
- výber vhodného úseku
 - first fit
 - best fit
 - worst fit

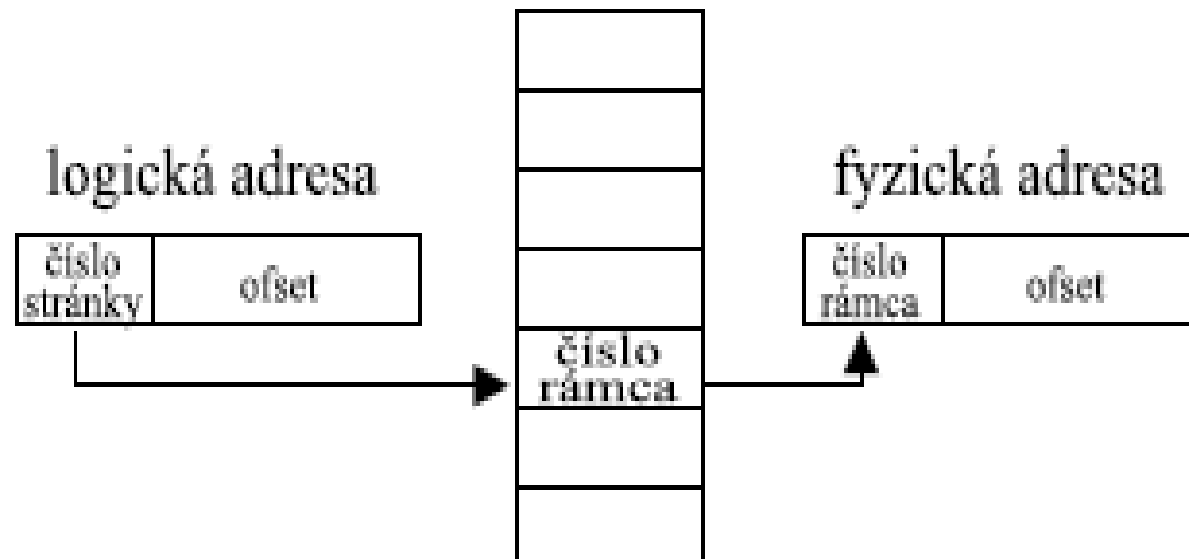
Stránkovanie



Stránkovanie

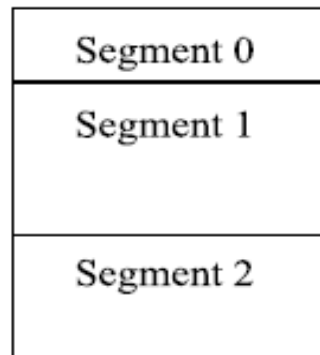
- logický adresný priestor sa delí na stránky rovnakej veľkosti
- fyzický adresný priestor sa delí na rámce rovnakej veľkosti
- logická adresa sa pomocou tabuľky stránok mapuje na fyzickú
 - číslo stránky na číslo rámca
- odstraňuje problém s vonkajšou fragmentáciou

Stránkovanie



Segmentácia

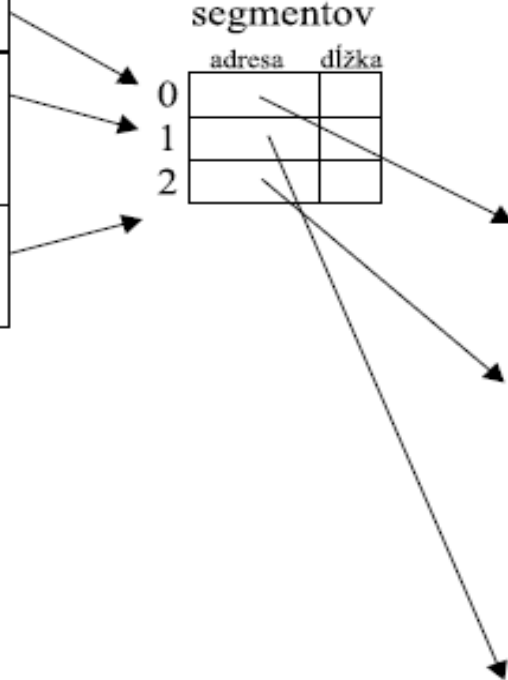
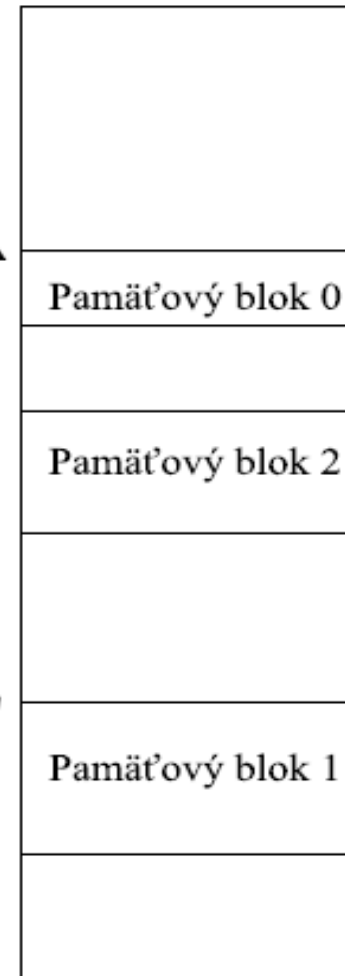
Logický adresný priestor



Tabuľka segmentov

	adresa	dĺžka
0		
1		
2		

Fyzický adresný priestor



Segmentácia

- Logický adresný priestor je rozdelený na segmenty (rôznej veľkosti)
- Logické adresy sa mapujú na fyzické pomocou tabuľky segmentov
- Rozdiely oproti stránkovaniu
 - rôzna veľkosť segmentov
 - segmenty sú programátorovi viditeľné
 - stránkovanie funguje transparentne

Virtuálna pamäť

- najčastejšie pomocou stránkovania
- niektoré stránky nemusia byť v RAM, ale sú na disku
 - pokus o prístup k takej stránke = **výpadok stránky**
 - OS nájde (alebo vyrobí) voľný rámec, načíta do neho obsah stránky z disku, upraví tabuľku stránok a nechá procesor zopakovať inštrukciu
- swapovanie procesu vs. stránkovanie

Súborové systémy

Bootovanie

Súbor

- slúži na „dlhodobé“ ukladanie informácií
 - dlhodobé = aj po vypnutí napájania
- od hardvéru nezávislá abstrakcia
- textové vs. binárne súbory
- súborový systém
 - dátové štruktúry
 - operácie
 - slúži na ukladanie súborov

Názvy súborov

- často sa skladajú z hlavného mena a prípony
 - prípona zvyčajne hovorí o type súboru
 - na niektorých systémoch je dobrovoľná (UNIX/Linux)
 - inde (takmer) povinná (Windows)
- rôzne obmedzenia
 - povolené znaky
 - najlepšie je vyhnúť sa diakritike, medzerám, ...
 - dĺžka (napr. DOS, CD: 8+3)
 - (ne)rozlišovanie veľkých a malých písmen
 - UNIX/Linux áno, Windows nie

Textový súbor

- riadky
 - postupnosť „tlačiteľných“ znakov ukončená oddeľovačom riadkov
 - UNIX: LF 10 0xA \n
 - DOS/Windows: CR,LF 13,10 0xD,0xA \r\n
 - Mac: CR 13 0xD \r
- zrozumiteľný človeku
- upravovateľný textovým editorom

Textový súbor

- kódovanie znakov
 - zobrazenie medzi „znakmi“ a ich „kódmi“
 - ASCII
 - 128 znakov
 - ISO 8859-x
 - 1 – západoeurópske jazyky
 - 2 – stredoeurópske jazyky
 - CP 1252 vs. ISO 8859-1 (podobné)
 - CP 1250 vs. ISO 8859-2 (nesedia ľ,š,t,ž,Ľ,Š,Ť,Ž, ...)

Textový súbor

- UNICODE

- univerzálne kódovanie pre „všetky“ jazyky
- 17x65536 znakov
- UTF-32
- UTF-16
 - bežné znaky v 16 bitoch, iné dvojicou 16 bitových hodnôt
- UTF-8
 - 1 až 4 B na jeden znak
 - prvých 128 znakov identických s ASCII

Binárny súbor

- človeku nezrozumiteľný
- často priama reprezentácia údajov v podobe ako v pamäti
- štruktúra zvyčajne špecifická pre konkrétnu aplikáciu

Binárny súbor

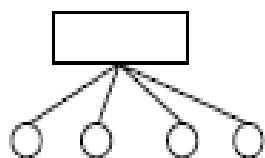
```
View: Audi VDO Crypto 93C86.BIN
Audi VDO Crypt> ↓FRO ----- 00000000 | View 7.28 (c)SEN
00000000: 1D 50 AA 37-D5 80 9B 51-46 30 C1 34-B8 75 CE 1E
00000010: BD 03 9B 29-99 44 60 A3-18 06 43 48-8B 53 03 B0
00000020: 68 50 47 62-EA E5 F3 B1-74 C0 23 9E-09 B0 A4 93
00000030: 88 E4 27 7F-62 B9 BA FC-67 8C 66 7B-95 C4 AB E1
00000040: F6 11 29 46-49 4C EE D3-58 D3 28 6C-5F B0 8A 65
00000050: 82 20 99 20-67 43 A6 0C-50 24 99 24-27 9B D1 47
00000060: AD 1E 17 30-A5 D1 9B E0-43 68 DE D0-7F CB 9B 85
00000070: 0A 37 6E A0-97 20 D2 E2-9B 4C 86 81-9B A4 D9 E4
00000080: 06 2E DA 57-C5 D4 8B 42-1A 76 A2 3E-A5 2D A3 0A
00000090: 48 01 44 87-AA 70 B6 EE-68 06 42 DA-BE 73 D9 A2
000000A0: A0 46 50 61-56 D7 14 5A-6C FD 77 A4-20 A3 01 76
000000B0: 9C 81 87 3D-9F B9 BC E7-1E C8 99 DE-97 D8 AB F4
000000C0: A6 87 30 38-C7 D2 00 00-F3 DC 97 2B-9B DC 99 62
000000D0: AE 22 99 22-67 66 4F 90-3C 27 99 26-67 99 18 EC
000000E0: B5 F7 19 C6-F3 54 9B E2-BA 97 1A B8-5F F4 CA CB
000000F0: 2E C6 67 B0-8D E2 B9 E2-8B AC D9 9B-91 66 65 E6
00000100: 4B 91 AB 20-68 ED 82 F8-02 7A A3 58-5F F0 82 FC
00000110: EB 06 E3 11-AB 50 4D 99-03 1B 4B 14-03 55 C0 15
00000120: 42 64 6F A1-E0 DE D4 9F-AD 7F 8B 2D-AA BF CB 84
00000130: 98 90 9C 90-98 D0 F8 4B-98 94 26 48-88 D5 A5 E6
00000140: 15 D6 D6 A9-64 61 C2 29-32 94 D7 83-80 24 C2 2D
00000150: 9A C5 F9 30-B0 64 72 C0-B8 74 58 4B-9A 72 3B 7B
00000160: 86 95 C2 19-6F 4D 3F 49-58 9A C9 82-32 8D 83 D4
1Global 2FilBlk 3CryBlk 4ReLoad 5 6String 7Direct 8Table 9 10Leave
```

Príklady súborov

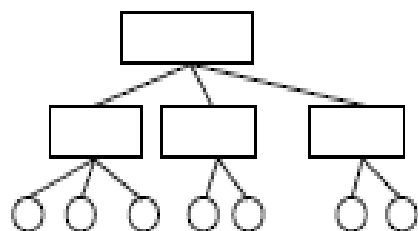
- textové
 - .c, .h, .cpp, .html, .php, .xml, .txt, .csv, .bat, .java
 - .ps
- binárne
 - .doc, .docx, .xls, .xlsx, .ppt, .pptx
 - .odt, .ods, .odp
 - .exe, .dll, .sys, .class
 - .zip, .rar, .tgz, .tar.gz, .tar.bz2
 - .jpeg, .gif, .png, .bmp
 - .pdf

Adresáre

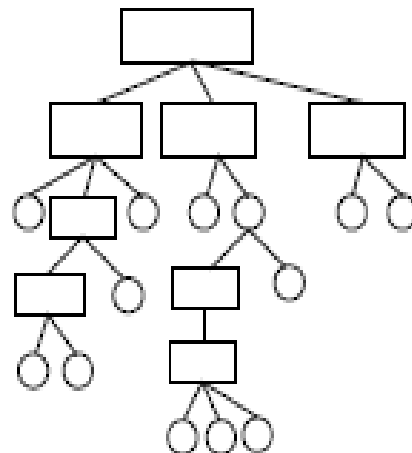
- organizácia súborov v súborovom systéme



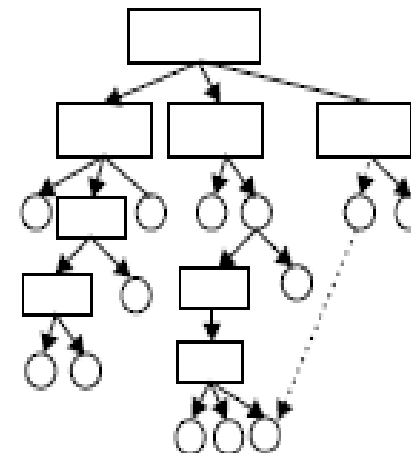
jednourovňová



dvojúrovňová



stromová



orientovaný acyklický graf

Adresáre

- UNIX/Linux
 - acyklický orientovaný graf
- Windows
 - les – množina stromov
 - jeden strom pre jeden súborový systém
 - každé pamäťové médium tvorí samostatný strom označený písmenom
 - A:, B: - diskety
 - C:, D:, ... - disky, CD, USB, sieťové disky, ...

Cesta k súboru

- určuje umiestnenie súboru v adresárovej štruktúre
 - absolútna
 - C:\XY\ABC\SUBOR.TXT
 - /home/user/dir/file.txt
 - relatívna
 - vzhľadom k aktuálnemu adresáru
 - abc.txt
 - ..\XY\FFF.TXT
 - ../xy/fff.txt

Linky, zástupcovia

- UNIX/Linux
 - hard link
 - viacero názvov reprezentuje ten istý súbor
 - symbolický link
 - špeciálny súbor obsahujúci cestu k inému súboru
- Windows
 - „zástupca“ (shortcut)
 - špeciálny súbor obsahujúci cestu, ikonu, ...
 - nefunguje transparentne ako symbolický link

Príklady súborových systémov

- FAT
 - podporovaný väčšinou OS
 - krátke vs. dlhé mená
 - FAT16 (<2GB), FAT32 (>512MB)
- NTFS
 - Windows NT, 2000, XP, Vista, 7, Linux
 - prístupové práva
- ext2/3/4, xfs, ReiserFS, jfs, ...
 - Linux
 - prístupové práva, linky, ...

Príklady súborových systémov

- ISO 9660
 - CD
 - mená 8+3, len veľké písmená, číslice a pár znakov
 - rozšírenia
 - RockRidge
 - MS Joliet
 - dlhšie mená, väčšia voľnosť, alternatívna štruktúra
- UDF
 - DVD

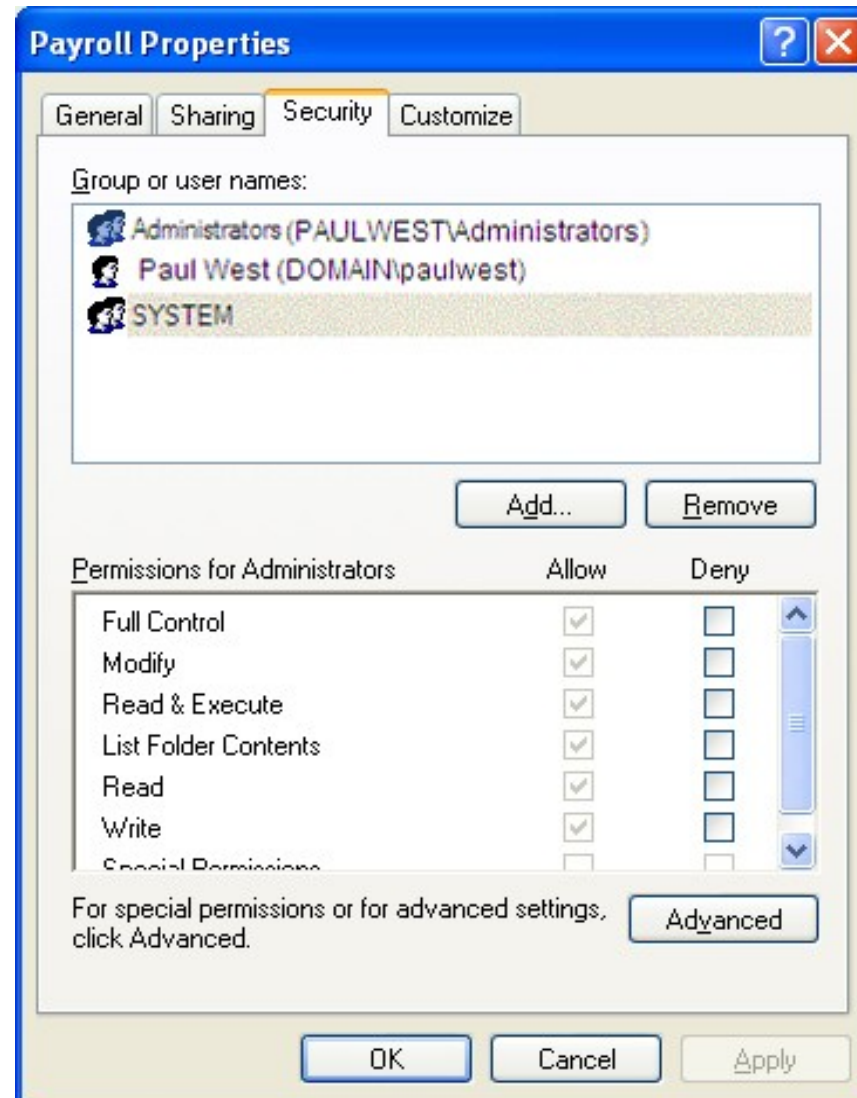
Vnútorosti súborových systémov

- priestor pridelovaný po blokoch
 - 512B – 64 KB
- malé súbory často zaberú oveľa viac miesta ako reálne využijú
- súbory nemusia byť na disku súvislé
 - fragmentácia a defragmentácia
- konzistencia dát súborového systému

Atribúty súborov

- dátum a čas
 - modifikácie, vytvorenie, prístupu
- veľkosť
- vlastník, prístupové práva
 - DOS/Windows atribúty
 - read only
 - system
 - hidden
 - archive

Prístupové práva NTFS



Prístupové práva NTFS

- pomerne podrobné
- dedenie z adresára na podadresáre/súbory
- práva pre skupiny, používateľov
- zakazovacie vs. povoľovacie práva

Prístupové práva UNIX/Linux

- čítanie, zápis, spúšťanie/použitie adresára
- pre vlastníka, skupinu, ostatných
- žiadne dedenie z adresárov

Delenie diskov v PC

- disk rozdelený na max. 4 partície
 - samostatné súborové systémy
- extended partition
 - umožňuje vyrábať ďalšie „logické“ disky
- jedna partícia je zvyčajne označená ako aktívna
 - obsahuje operačný systém

Bootovanie – štart systému

- po zapnutí počítača prázdna operačná pamäť
- vykonáva program z ROM – BIOS
 - test hardvéru
 - načítanie „boot loader“-a z určeného média
 - disk, CD, USB
- boot loader
 - načítanie jadra OS
 - zvyčajne z aktívnej partície disku

BIOS

• Award Modular BIOS v6.00PG, An Energy Star Ally
• Copyright (C) 1984-2007, Award Software, Inc.



Intel P35 BIOS for P35C-DS3R F2o

Memory Runs at Dual Channel Interleaved

IDE Channel 1 Master : TSSTcorpDVD-ROM SH-D163A SB00

IDE Channel 2 Master : WDC WD1500ADFD-00NLR1 20.07P20

IDE Channel 3 Master : WDC WD1500ADFD-00NLR1 20.07P20

IDE Channel 4 Master : None

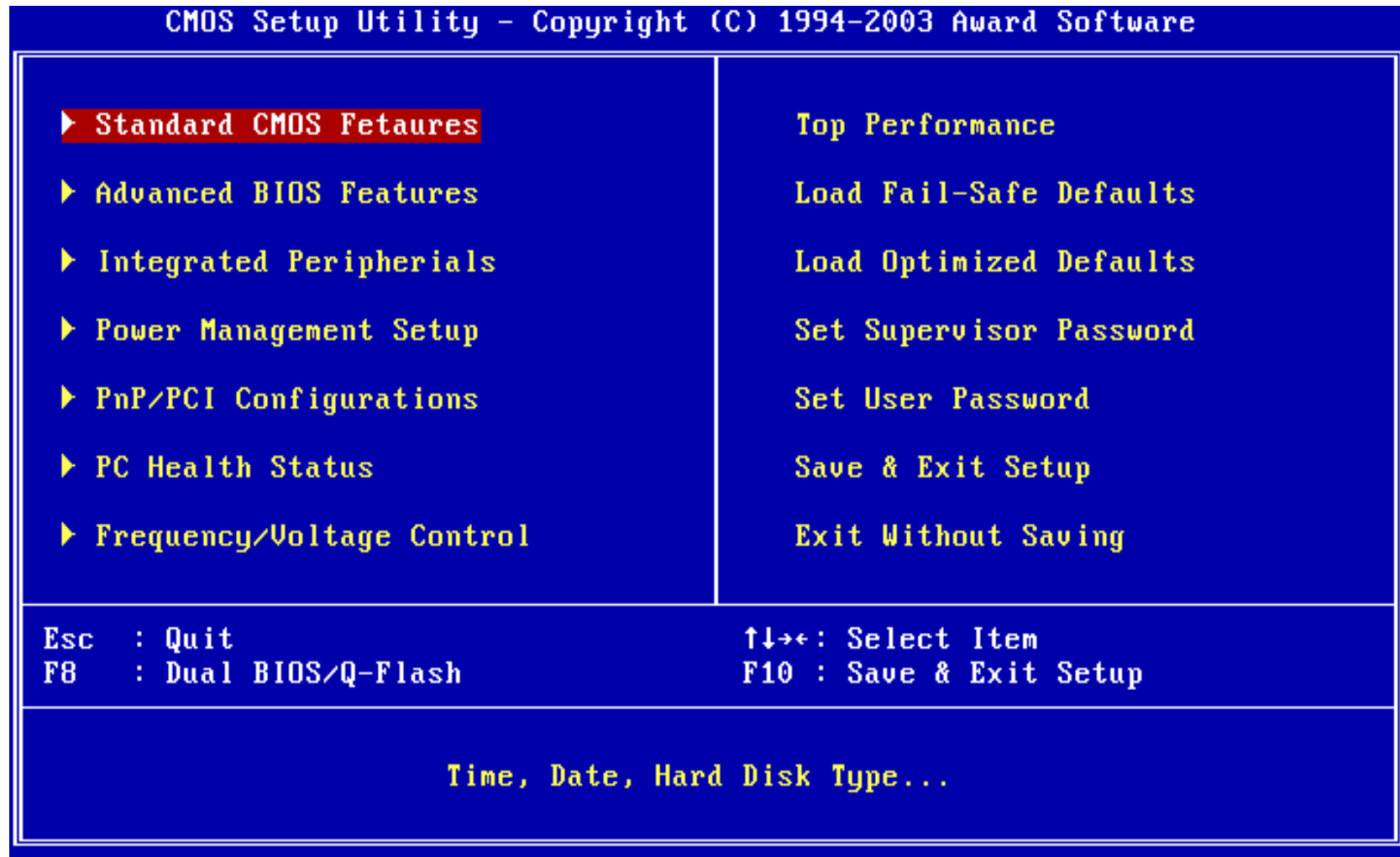
IDE Channel 4 Slave : None

IDE Channel 5 Master : None

IDE Channel 5 Slave : None

:BIOS Setup/Q-Flash <F9>:XpressRecovery2 <F12>:Boot Menu <End>:Qflash
05/11/2007-P35-ICH9-6A790G0BC-00

BIOS



„Lepšie“ boot loader-y

- umožňujú výber OS
 - prípadne nastavenie parametrov pre OS
- v rôznych partíciách disku môžu byť nainštalované rôzne OS
- napr. GRUB
 - poznáte z učební