Analýza hlavního zaváděcího sektoru a bootovacího procesu Windows 8.1

Martin Dráb
martin.drab@email.cz
Http://www.jadro-windows.cz

Obsah přednášky

- **Základní pojmy**
- **Struktura MBR**
- **Kód MBR**
- Průběh bootování
- …

Platné převážně od Windows Vista

Struktura (peného) disku

Master Boot Record (hlavní zaváděcí sektor)

Volume Boot Record (boot sektor)

Data (primárního) oddílu

Neobsazený prostor

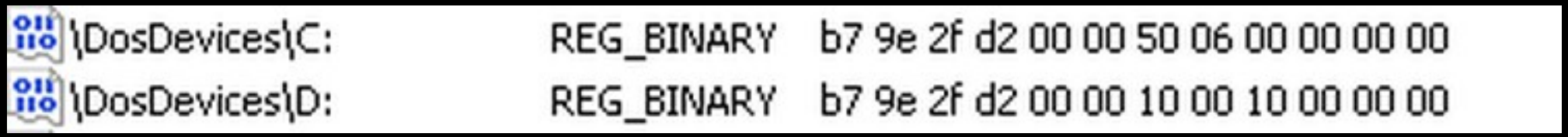
- Rozdělen na sektory (obvykle 512 B)
- FS většinou pracují s clustery (4 KB) (alokační jednotky)

Struktura MBR

- 512 bajtů celkem
- 355 bajtů užitečného kódu (70 %)
- 85 bajtů chybových hlášek (15 %)
- 4 bajty na ID disku (1 %)
- 2 bajty neznámé funkce (0,5 %)
- 4*16 bajtů na tabulku oddílů (13 %)
- 2 bajty na značku konce (0x55 0xAA) (0,5 %)

ID disku

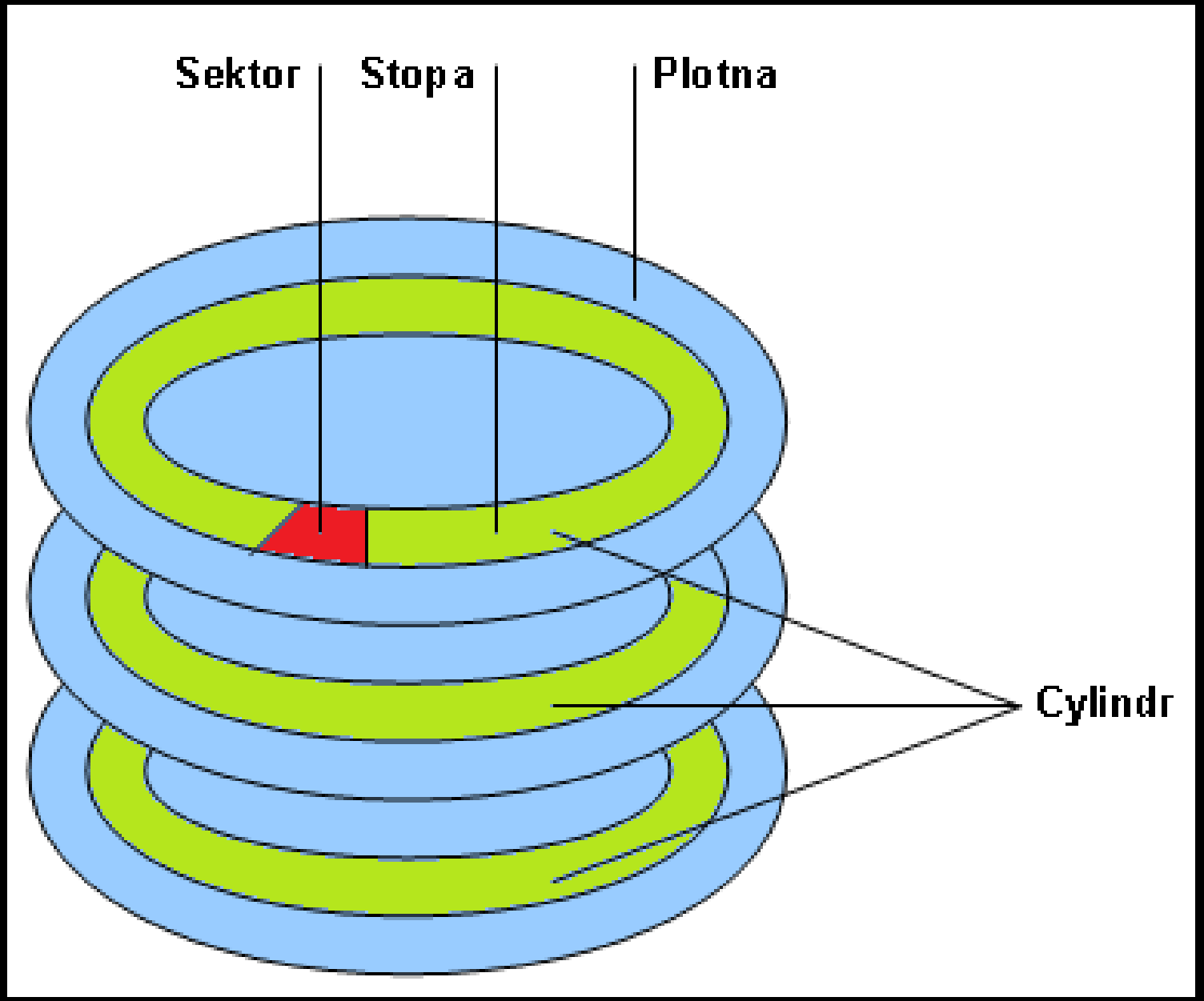
- 4bajtové číslo jedinečně identifikující disk v rámci systému.
- **HKLM\SYSTEM\MountedDevices**, hodnoty **\DosDevices\X:**
  - 12bajtová data
  - 4 bajty identifikují disk (ID)
  - 8 bajtů udává pozici prvního bajtu oddílu mapovaného pod písmenem X



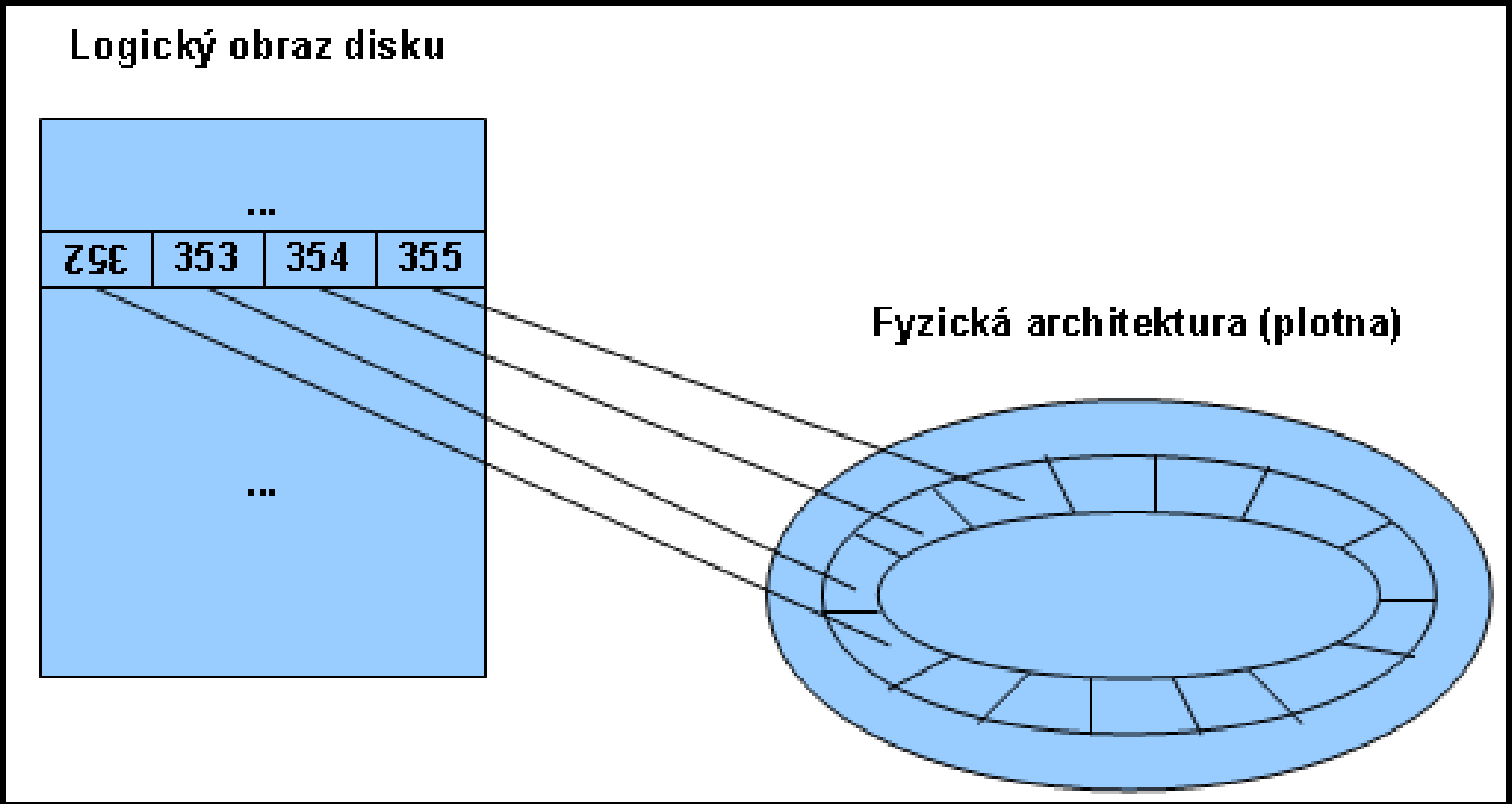
Záznam o oddílu

Offset	Velikost	Popis
0	1	Příznaky 0x80 = aktivní (bootovatelný) 0x00 = neaktivní
1	3	CHS adresa prvního sektoru
4	1	Typ/souborový systém (hint) 0x07 = NTFS 0x06 = FAT 0x0C = FAT32 0x07 = exFAT 0x0F = extended
5	3	CHS adresa posledního sektoru
8	4	LBA prvního sektoru
12	4	Délka oddílu, v sektorech

CHS



LBA


Logický obraz disku
...
352
353
354
355
...
Fyzická architektura (plotna)

Tabulka oddílů a vlastnosti disku

Partition #1		Partition #2		Partition #3		Partition #4	
Flags	0x80	Flags	0x0	Flags	0x0	Flags	0x0
CHS	0x0:0x20:0x21	CHS	0xC:0xDF:0x14	CHS	0x3FF:0xFE:0x	CHS	0x0:0x0:0x0
Type	7	Type	7	Type	7	Type	0
Start	0x800	Start	0x32800	Start	0x8000800	Start	0x0
Length	0x32000	Length	0x7FCE000	Length	0x1D42D800	Length	0x0

Properties

Cylinders	243197 (0x3B5FD)	Sectors per track	63 (0x3F)
Media type	12 (0xC)	Bytes per sector	512 (0x200)
Tracks per cylinder	255 (0xFF)	Total sectors	3906963456 (0xE8DF8800)

Běhové prostředí

4bajtové adresy

- **Segment** (2 bajty) – obvykle zadán implicitně
- **Offset** (2 bajty)
- 16 * Segment + Offset = Fyzická adresa

- Příklad shodných adres:
  - 0x0000:0x7C00
  - 0x07C0:0x0000
  - 0x0700:0x0C00
- „služby OS“ přes přerušení
  - INT 0x10 – obrazovka
  - INT 0x13 – práce s diskem
  - INT 0x18 – restart počítače
  - …

Struktura kódu MBR

- Kopírování MBR (relokace)
- Hledání aktivního oddílu
- Čtení VBR
- 8042 controller
- TPM
- Předání řízení na první bajt VBR

Kopírování

- MBR načten na adresu 0x0000:0x7C00
- Na stejné adrese by měl být spuštěn i VBR
  - Disk nemusí vůbec obsahovat MBR
  - Transparentnost
- Proto:
  - celý obsah MBR překopírován na adresu 0x0000:0x0600
  - a spuštěn.

Hledání aktivního oddílu

- Postupné zkoumání (primárních) oddílů
- Rozhoduje se podle hodnoty prvního bajtu (příznaky):
  - 0x00 = neaktivní oddíl, hledání pokračuje,
  - 0x80 = aktivní oddíl, hledání končí, začíná práce s diskem,
  - Jiná hodnota = neplatný oddíl, chyba „Invalid partition table“,
  - žádný oddíl není aktivní = předání řízení firmwaru (reboot, nebo nic)

Čtení VBR

- Detekce, zda disk podporuje LBA
- ANO = pokus o čtení VBR přes LBA, v případě chyby se pokračuje dalším bodem
- NE = pokus o čtení přes CHS
- Pokud chyba, reset disku a nový pokus
- Maximálně pět pokusů
- Chyba „Error loading operating system“
- INT 0x13

8042 Controller

- Nepovinné
- Ovládá
  - Klávesnici (PS/2)
  - Myš (PS/2)
  - Pin A20, restart stroje
- MBR
  - Povolí A20
  - Povolí přerušení generovaná klávesnicí (stisk či uvolnění klávesy)

TPM

- Nepovinné
- Speciální HW modul dovolující provádět kryptografické operace
- Přístupný přes přerušení 0x1A
- MBR jej používá k výpočtu hashe obsahu VBR
  - Z výsledku nedělá žádné závěry

Volume Boot Record (VBR)

- **Cíl:** najít, načíst a spustit **bootmgr**
- **Velikost:** 512 B – 4 KB
- **Prostředí:** 16bitové (reálný režim procesoru)
- **Obsah:** kód, důležité informace pro souborový systém
- **Umístění:** první sektory oddílu
- Konkrétní podoba VBR závisí na souborovém systému, kterým je oddíl formátován

Boot loader (Bootmgr)

- **Cíl:** dostat se do 64bitového režimu a předat štafetu programu **winload.exe**
- **Velikost:** stovky KB
- **Prostředí:** 16bitové, 32bitové, 64bitové
- **Obsah:** kód a data (**startup.com** + **bootmgr.exe**)
- **Umístění:** kořenový adresář aktivního oddílu. Nesmí být komprimován či šifrován

Winload a ntoskrnl

- **Winload.exe**
  - **Cíl:** načíst hlavní modul jádra a ovladače, které budou v rámci startu OS potřeba
  - **Obsah:** standardní PE soubor
  - Stejně jako **bootmgr** čte a chová se podle konfigurace uložené v BCD.
- **Ntoskrnl.exe**
  - Implementuje většinu mechanismů, bez kterých se žádný větší OS neobejde (VM, plánování, komunikace s HW, synchronizace, obsluha přerušení)

???

Martin Dráb

Email: martin.drab@email.cz

- **Jabber:** vrtule@jabber.cz

ICQ: 332970040

WWW: http://www.jadro-windows.cz