

# ZÁKLADNÍ POJMY, OVLADAČE

Správa periferních zařízení a použití ovladačů představují základní součásti operačního systému. Tato prezentace se zabývá strukturou I/O systému, různými typy periferií, funkcemi ovladačů a způsobem, jakým komunikují s operačním systémem. Prozkoumáme správu zařízení ve Windows a Linuxu, rozdíly mezi ovladači v jádře a uživatelském prostoru, a ukážeme, jak se používají ovladače Plug-and-Play i non-Plug-and-Play.

# STRUKTURA I/O SYSTÉMU

## I/O systém

- Sada rutin a objektů pro přístup k periferním zařízením.

## Hierarchie I/O systému

- Rozdělená do uživatelského a privilegovaného režimu.
- I/O API, správci I/O a ovladače komunikují s Hardware Abstraction Layer (HAL) a hardwarem.

## Role HAL

- Vrstva abstrakce hardware, která zajišťuje přístup ke konkrétním periferiím.

# TYPY PERIFERÍÍ

## Vstupní a výstupní zařízení

- **Vstupní:** klávesnice, myš.
- **Výstupní:** monitor, tiskárna.
- **Vstupně-výstupní:** pevné disky, dotykové obrazovky.

## Rozdělení dle sdílení

- **Vyhrazená:** např. tiskárny, přístup mají postupně jednotlivé procesy.
- **Sdílená:** paměť nebo disky, každý proces má vlastní část.
- **Společná:** mikrofony, teploměry, které mohou být použity současně.

## Rozdělení dle datové komunikace

- **Znaková:** pracuje s jednotlivými oktety.
- **Bloková:** pracuje s bloky dat, např. pevné disky.
- **Speciální zařízení:** časovače nebo virtuální zařízení.

# OVLADAČE ZAŘÍZENÍ

## Úloha ovladačů

- Zajišťují komunikaci mezi zařízeními a procesy.

## Standardizované funkce ovladačů

- `Init()`, `Open()`, `Close()`, `Read()`, `Write()`, `Seek()` a další.

## Horní a dolní část ovladačů

- **Horní část:** Přijímá data od procesů (producent).
- **Dolní část:** Předává data zařízení (konzument), přímá komunikace s hardwarem.

## Oddělení ovladačů na hardwarově závislé a nezávislé

- Dolní část je specifická pro typ zařízení, horní část zůstává konzistentní.

# STRUKTURA OVLADAČŮ VE WINDOWS

## Typy ovladačů

- **Windows Driver Model (WDM):** Klávesnice, zvuková karta.
- **Windows Display Driver Model (WDDM):** Pro grafické karty od Windows Vista.

## Ovladače v režimu jádra vs. uživatelském prostoru

- **Kernel-Mode Drivers (KMDF):** Přímo v jádře, rychlejší komunikace, ale větší riziko.
- **User-Mode Drivers (UMDF):** Bezpečnější, běží v uživatelském režimu.

## Rozdělení podle Plug-and-Play

- **Plug-and-Play ovladače:** Automatické rozpoznání a komunikace.
- **Non-Plug-and-Play:** Rozšiřují jádro, méně často spojené s konkrétním hardwarem.

# TYPY OVLADAČŮ V LINUXU

## Kernel Drivers

- Moduly jádra komunikující přímo s hardwarem.

## User-Space Drivers

- Běží mimo jádro, zajišťuje bezpečnost, např. pomocí FUSE (FileSystem in User Space).

## Moduly jádra (Kernel Modules)

- Načítají se jako dynamicky linkované knihovny (.KO soubory) pro specifická zařízení.

## Výhody a nevýhody ovladačů v uživatelském prostoru

- **Výhody:** Zvýšená bezpečnost, nižší pravděpodobnost pádu systému.
- **Nevýhody:** Pomalejší komunikace s hardwarem, vhodné pro nižší nároky na výkon.

# TYPY OVLADAČŮ PODLE FUNKCE (WINDOWS)

## Funkční ovladače

- Přímá komunikace s konkrétním zařízením, např. tiskárny.

## Sběrníkové ovladače

- Spravují komunikaci na sběrnici (PCI, USB), detekují zařízení Plug-and-Play.

## Filtrační ovladače

- Modifikují komunikaci s jinými ovladači, např. šifrování nebo sledování.

## Vrstvové ovladače

- Komunikují přes správce I/O a KMDF/UMDF, umožňují lepší správu a optimalizaci komunikace.

# SPRÁVA I/O A HAL (LINUX)

## Role HAL (Hardware Abstraction Layer)

- Vrstva zajišťující jednotný přístup ke všem typům zařízení.

## Udev

- V moderních systémech nahradil HAL, spravuje připojení zařízení a strukturu /sys.

## Struktura zařízení

- Stromová struktura, která mapuje zařízení a přidělená práva přístupu.

## Objekt zařízení

- Každé zařízení má vlastní objekt se specifickými informacemi a přístupovými oprávněními.



# VÝVOJ A NASAZENÍ OVLADAČŮ

## Unifikace ovladačů

- Použití standardizovaných API a funkcí (např. Init(), Open()) usnadňuje vývoj.

## Kernel-Mode Driver Framework (KMDF) a User-Mode Driver Framework (UMDF)

- KMDF pro jádrové ovladače, UMDF pro ovladače v uživatelském prostoru.

## Bezpečnostní mechanismy

- Použití mutexů, spinlocků a dalších synchronizačních technik pro stabilitu systému.

## Plug-and-Play a možnosti správy zařízení

- Moderní OS používají Plug-and-Play ovladače, které usnadňují správu hardwaru.

# SHRNUTÍ

- I/O systém je klíčovým prvkem komunikace mezi operačním systémem a periferiemi.
- Různé typy periferií vyžadují specializované ovladače pro správu přístupu.
- Windows a Linux poskytují strukturované přístupy k rozdělení ovladačů podle funkce a bezpečnosti.
- HAL a Udev v Linuxu a KMDF/UMDF ve Windows umožňují efektivní správu a abstrakci hardwaru.

# KONTROLNÍ OTÁZKY

1. Jaká je role HAL v architektuře operačního systému?
2. Jaké jsou rozdíly mezi ovladači v jádře a v uživatelském prostoru?
3. Jaký je účel funkcí Init(), Open() a Close() v kontextu ovladačů?
4. K čemu slouží KMDF a UMDF v systému Windows?
5. Jaký je rozdíl mezi sběrnicevým ovladačem a funkčním ovladačem?

# DOPORUČENÁ LITERATURA

1. **Silberschatz, A., Galvin, P. B., & Gagne, G.** - *Operating System Concepts* - Základy správy periferií a ovladačů.
2. **Tanenbaum, A. S., & Bos, H.** - *Modern Operating Systems* - Pokročilé metody správy I/O a ovladačů.
3. **Stallings, W.** - *Operating Systems: Internals and Design Principles* - Kapitoly o hardwarové abstrakci a správě ovladačů.
4. **Microsoft Documentation on Windows Driver Frameworks** - Dokumentace KMDF a UMDF.
5. **Linux Programmer's Manual** - Informace o struktuře a správě ovladačů v Linuxu.