1. Quản lý bộ nhớ (Memory management)

Mô tả chi tiết

Vai trò: cấp phát, theo dõi, bảo vệ và tối ưu hóa việc sử dụng bộ nhớ (RAM) cho hệ thống và tiến trình.

Các cơ chế chính: không gian địa chỉ ảo (virtual memory), ánh xạ trang (paging), phân vùng (segmentation), bảng trang (page tables), trao đổi (swap / paging out), và thuật toán thay thế trang (LRU, FIFO, Clock...).

Khi một tiến trình truy cập một địa chỉ ảo, phần cứng (MMU) dịch sang địa chỉ vật lý theo bảng trang; nếu trang chưa có trong RAM → page fault → kernel sẽ nạp trang từ ổ đĩa (swap/file) vào khung (frame) và cập nhật bảng trang. Hệ điều hành cũng theo dõi phân quyền (read/write/execute) để bảo vệ bộ nhớ giữa các tiến trình.

repository.unikom.ac.id

Ví dụ thực tế

Khi bạn mở nhiều tab Chrome: mỗi tab là tiến trình/luồng có vùng nhớ riêng. Nếu RAM đầy, OS sẽ chuyển (swap) một số trang ít được dùng ra ổ cứng — gây chậm (disk thrashing nếu quá nhiều).

Trên Linux: malloc() trong chương trình cấp phát địa chỉ ảo; kernel + MMU xử lý mapping, và khi thiếu sẽ xảy ra page fault rồi kernel dùng swap.

Tóm tắt ngắn

Quản lý bộ nhớ giữ cho nhiều tiến trình chạy an toàn, hiệu quả bằng cách cung cấp cho mỗi tiến trình không gian địa chỉ ảo, cấp phát/thu hồi bộ nhớ, xử lý page fault và chạy thuật toán thay thế trang để tối ưu hoá sử dụng RAM.

repository.unikom.ac.id

2. Quản lý thiết bị nhập / xuất (I/O & Device management)

Mô tả chi tiết

Vai trò: làm trung gian giữa phần mềm và phần cứng I/O (đĩa, mạng, bàn phím, máy in, USB...), cung cấp API thống nhất cho ứng dụng, điều phối truy cập, xử lý ngắt (interrupts), DMA, buffering, và quản lý driver.

Device driver (trình điều khiển) là mô-đun đặc thù từng thiết bị, chạy trong kernel hoặc ở lớp thấp, nhận/gửi lệnh tới controller, đóng gói/giải mã dữ liệu, và báo lỗi. I/O manager thường dùng các cấu trúc như IRP (I/O Request Packet) để truyền yêu cầu tới driver (mô hình Windows).

Microsoft Learn

+1

Ví dụ thực tế

Cắm USB: kernel phát hiện thiết bị, tải driver tương ứng, gắn (mount) thiết bị lưu trữ — người dùng thấy ổ mới xuất hiện trong File Explorer.

In ấn: ứng dụng gửi job in tới subsystem in; driver điều khiển hàng đợi, gửi dữ liệu tới máy in bằng lệnh cụ thể; nếu máy in bận, job nằm trong spooler.

DMA: khi đọc/ghi đĩa lớn, controller dùng DMA để chuyển dữ liệu trực tiếp giữa ổ và RAM, giảm tải CPU.

cs.fsu.edu

+1

Tóm tắt ngắn

Subsystem I/O + device drivers che dấu sự khác biệt giữa các phần cứng, cung cấp giao diện thống nhất cho ứng dụng, xử lý ngắt/DMA/buffering và quản lý đăng ký/giải phóng thiết bị để I/O hoạt động trơn tru.

Microsoft Learn

+1

3. Cung cấp giao diện người dùng (User Interface — CLI / GUI)

Mô tả chi tiết

Vai trò: cho phép người dùng tương tác với hệ điều hành và ứng dụng. Hai loại chính: Command-Line Interface (CLI) (shell, terminal) và Graphical User Interface (GUI) (desktop, cửa sổ, menu).

GUI được xây dựng trên các thành phần window manager, compositor, toolkit (ví dụ: Win32 trên Windows, Quartz/WindowServer trên macOS, X11/Wayland trên Linux). CLI (shell như bash, PowerShell) cung cấp dòng lệnh, scripting và quyền tự động hoá. Hệ điều hành cung cấp API, process và quyền để các thành phần UI hoạt động an toàn và hiệu quả.

blogs.helsinki.fi

+1

Ví dụ thực tế

GUI: Windows Explorer / Finder — người dùng nhấp chuột để mở file, OS xử lý sự kiện chuột, gọi hệ thống file để truy xuất nội dung, vẽ cửa sổ.

CLI: mở Terminal, gõ ls/dir — shell gửi system call (read dir) tới kernel, kernel trả danh sách, shell in ra màn hình. Scripting (bash script) tự động hoá nhiệm vụ quản trị.

Tóm tắt ngắn

Giao diện người dùng cung cấp cách người dùng điều khiển và nhận thông tin: GUI thân thiện cho thao tác trực quan; CLI mạnh mẽ cho tự động hóa và tác vụ chuyên sâu. Hệ điều hành quản lý sự kiện, quyền và tài nguyên để UI hoạt động mượt.

blogs.helsinki.fi

+1

4. Quản lý hệ thống tệp (File system management)

Mô tả chi tiết

Vai trò: tổ chức, lưu trữ, truy xuất và bảo vệ dữ liệu trên thiết bị lưu trữ. Thành phần bao gồm: cấu trúc metadata (siêu dữ liệu — inode/MFT), thư mục, bảng phân bổ (FAT, B-tree, extent), bộ nhớ đệm (cache), journaling (ghi nhật ký để phục hồi), và lớp trừu tượng VFS (Virtual File System) để hỗ trợ nhiều FS khác nhau. Hệ điều hành cung cấp các system call như open/read/write/close, cùng cơ chế quyền truy cập (owner/group/others).

Microsoft Learn

+1

Ví dụ thực tế

Lưu file Word: ứng dụng gọi system call write(); kernel viết dữ liệu xuống buffer cache trước, rồi flush vào đĩa; nếu file system hỗ trợ journaling (NTFS, ext4 với journal), thì metadata được ghi vào journal trước để tránh hỏng khi mất điện.

Định dạng ổ cứng: chọn NTFS (Windows) hay ext4 (Linux) quyết định cách dữ liệu được tổ chức và các tính năng như journaling, bảo mật, kích thước file tối đa.

Tóm tắt ngắn

Quản lý hệ thống tệp là lớp chịu trách nhiệm lưu trữ và bảo vệ dữ liệu vật lý, cung cấp API file cho ứng dụng và đảm bảo tính toàn vẹn thông qua cơ chế như journaling và cache.

Microsoft Learn

+1

5. Quản lý tiến trình (Process management)

Mô tả chi tiết

Vai trò: tạo/huỷ tiến trình, lập lịch CPU, chuyển ngữ cảnh (context switch), đồng bộ/đồng thời (locks, semaphores, mutexes), IPC (pipes, sockets, shared memory), quản lý trạng thái tiến trình (running, ready, blocked), và xử lý tín hiệu.

Scheduler chọn tiến trình nào được cấp CPU dựa trên chính sách (Round-Robin, Priority, Multilevel Feedback Queue...). Context switch lưu trạng thái CPU (registers, PC, stack) của tiến trình hiện tại rồi nạp trạng thái của tiến trình kế tiếp. Kernel cung cấp system calls (fork/exec/wait/exit) để ứng dụng tạo và quản lý tiến trình.

CS Princeton

+1

Ví dụ thực tế

Khởi chạy chương trình trên Linux: shell gọi fork() để sao chép tiến trình, tiến trình con gọi exec() để thay thế bằng chương trình mới. Kernel đặt tiến trình vào hàng đợi sẵn sàng, scheduler cấp CPU theo lượt.

Khi một tiến trình chờ I/O (ví dụ: đọc đĩa), tiến trình chuyển sang trạng thái blocked; CPU được cấp cho một tiến trình khác — điều này tối ưu hoá sử dụng CPU.

Trên Windows, Task Manager hiển thị các tiến trình, cho phép kill process (gửi tín hiệu terminate hoặc gọi API tương ứng).

Tóm tắt ngắn

Quản lý tiến trình điều phối việc thực thi các chương trình, cấp phát thời gian CPU, xử lý đồng bộ và truyền thông giữa tiến trình, đảm bảo hệ thống đa nhiệm hoạt động trơn tru