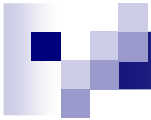




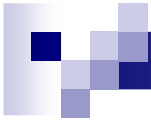
I/O: Von der Platte zur Anwendung

Von Igor Engel



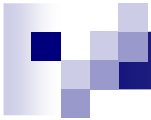
Gliederung

- 1 Einleitung
- 2 Übersicht
- 3 Systemaufrufe
 - Beispiel in Unix
- 4 Dateien
 - 4.1 Dateisysteme
 - 4.2 Transport der Daten
- 5 Festplattentreiber
- 6 Festplattenkontroller
- 7 Festplatte
- 8 Beispiel der I/O Programmierung
- 9 Zusammenfassung
- 10 Quellen



1 Einleitung

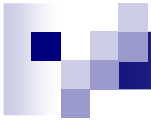
- Betriebssystem repräsentiert ein abstraktes Modell geräteunabhängiger Dateien
- Anwendungen greifen auf die Dateien zu
- Festplatte liefert aber einen sequenziellen Bitstrom
- Betriebssystem spielt hier die Rolle des Vermittlers



2 Übersicht

Der Weg von oben nach unten:

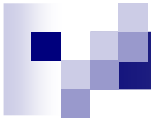
- Anwendungen greifen auf die Dateien zu
- Dateien bestehen aus Blöcken,
 - um Dateien zu lesen bzw. zu schreiben werden Systemaufrufe aufgerufen
- das Dateisystem gibt Lese- und Schreibbefehle an den Plattentreiber weiter
- der Plattentreiber schreibt hardwarenahe Anweisungen in die Steuerregister des Festplattencontrollers
- der Controller führt dann die Anweisungen aus seinen Registern aus



2 Übersicht

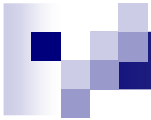
Der Weg von unten nach oben:

- Der Controller liest Bit für Bit von der Platte und konvertiert den Bitstrom in Byte-Blöcke
- danach werden diese Blöcke Schicht für Schicht nach oben weitergegeben.



2 Systemaufrufe

- Schnittstelle zwischen dem Betriebssystem und der Anwendung durch Systemaufrufe
- Werden vom Betriebssystem zur Verfügung gestellt
- Werden benötigt um mit Dateien zu Arbeiten

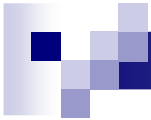


Beispiel in Unix

Prozess im Benutzermodus will einen Systemaufruf nutzen (wie etwa lesen einer Datei)

Er muss eine Unterbrechung ausführen und dadurch die Kontrolle an das Betriebssystem übergeben

Systemaufruf wird ausgeführt und die Kontrolle kehrt zum Benutzerprogramm zurück

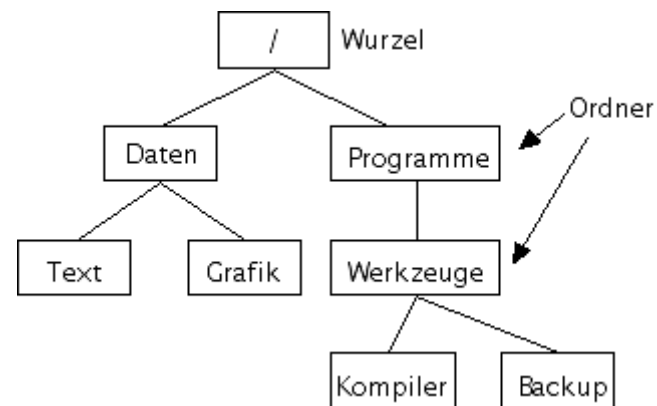


3 Dateien >> 3.1 Dateisysteme

- Dateien sind in einem Dateisystem organisiert
- Betriebssysteme arbeiten mit verschiedenen Dateisystemen
- Anwendungen greifen auf Dateien über das Dateisystem
- Beispiele für Dateisysteme: FAT,NTFS,EXT,...

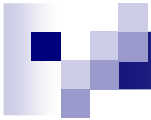
Beispiel

Hierarchisches Dateisystem von Unix



Das Dateisystem, in dem hierarchisch alle Dateien, Ordner und Geräte verzeichnet sind.

Alle Einträge werden über die gleiche Schnittstelle angesprochen.



3.2 Transport der Daten

- Aufteilung der Dateien in Blöcke
- Blöcke sind kleinste übertragende Einheit
- Blöcke sind üblich 512 Byte groß

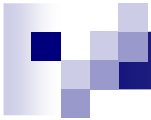


4 Festplattentreiber

- Allgemein als Vermittler zwischen zwei Schichten

z.B. Anwendung >> Treiber >> Gerät

- Wandelt geräteunabhängige Anweisungen der Anwendung in konkrete Befehle an das Gerät um
- Stellt das Ergebnis der Aktion der Anwendung zur Verfügung



5 Festplattencontroller

- Liest Befehle aus seinen Registern und führt sie aus
- Konvertiert beim lesen den seriellen Bitstrom in Byte-Blöcke
- Führt Fehlerkorrekturen durch
- Beim Abschluss der Operation wird eine Unterbrechung initiiert



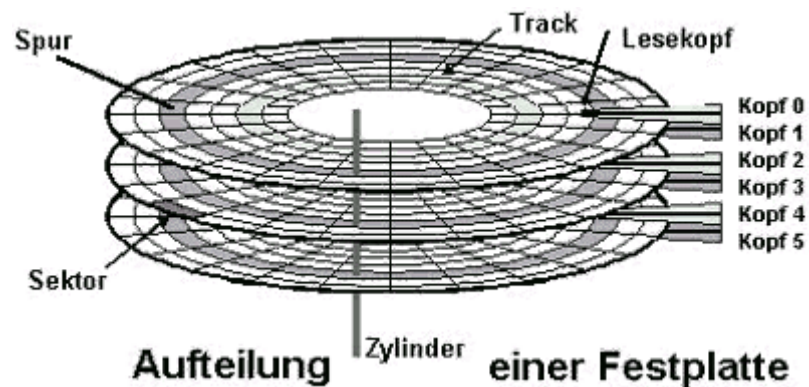
5 Festplattencontroller

- Drei Arten der E/A
 - programmierte E/A(PIO)
 - unterbrechungsgesteuerte E/A
 - DMA

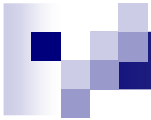
- Verschiedene Scheduling – Verfahren des Zugriffs auf die Festplatte
 - FCFS
 - SSTF
 - SCAN,C-SCAN
 - LOOK,C-LOOK

6 Festplatte

- Aufbau



- Festplatten bestehen aus einer oder mehreren Metallplatten, beweglichen Schreib/Leseköpfen. Adresse des physikalischen Blocks besteht aus: Zylinder-, Track- und Sektornummer.

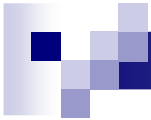


7 Beispiel der I/O Programmierung

```
int main(int argc, char *argv[]){
    char c;
    FILE *fp;
    char *point = argv[1];
    fp = fopen(point, "r");

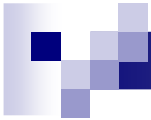
    while((c = getc(fp)) != EOF){
        putc(c, stdout);
    }
    fclose(fp);

    return(0);
}
```



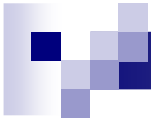
8 Zusammenfassung

- Anwendungen greifen auf die Dateien über Dateisystem
- Systemaufrufe werden benötigt um mit Dateien zu arbeiten
- Festplattentreiber wandelt Anweisungen der Anwendung in konkrete Befehle an das Gerät um, schreibt Befehle in Register des Controllers
- Festplattencontroller führt die Anweisungen aus seinen Register aus



9 Quellen

- Andrew S. Tannenbaum, Moderne Betriebssysteme 2., überarbeitet Auflage 2002
- Abraham Silberschatz , Operating System Concepts, sixth Edition 2002



Danke für's Zuhören!!!