

Proseminar „Speicher-und Dateisysteme“

Betriebssystemschichten

Dominik Downarowicz

Gliederung

- Einführung Betriebssystem
- Aufgaben eines Betriebssystems
- Der Kernel
 - Kernel-und Benutzermodus
- Kernelarten
 - Monolithischer Kernel
 - Mikrokernel
 - Hybridkernel
 - Vergleich der Kernelarten
- Beispiel: Linux
- Beispiel(kurz) : Minix
- Beispiel(kurz) : Mac OS X
- Überblick Dateisysteme
- Zusammenfassung

Einführung Betriebssystem

Was ist ein Betriebssystem?

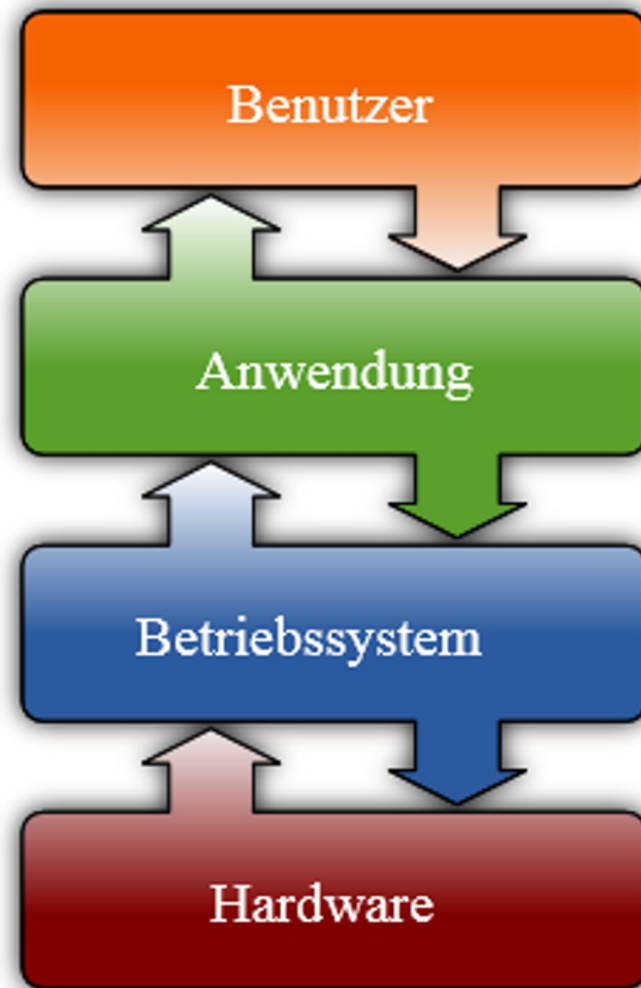
- Nach Din44300 :

„Die Programme eines digitalen Rechensystems, die zusammen mit den Eigenschaften dieser Rechenanlage die Basis der möglichen Betriebsarten des digitalen Rechensystems bilden und die insbesondere die Abwicklung von Programmen steuern und überwachen“

Einführung

- Verwaltung der Systemressourcen
 - Arbeitsspeicher, Festplatte, Ein-und Ausgabegeräte
- Stellt Systemressourcen den Anwendungsprogrammen zur Verfügung
- Bildet dadurch die Schnittstelle zwischen Hardware und Anwendungsprogrammen

Einführung Betriebssystem



Bildquelle: <http://de.wikipedia.org/wiki/Betriebssystem>

Einführung Betriebssystem

Anmerkung:

- Es gibt kein allgemeines Schema wie ein Betriebssystem aufgebaut ist

Grund:

- Verschiedene Systemfamilien

Aufgaben eines Betriebssystems

Prozessverwaltung

- Verschiedene Aufgaben gleichzeitig ausführen (Multitasking)
- Verschiedene Aufgaben im Hintergrund ausführen

Aufgaben eines Betriebssystems

Speicherverwaltung

- Verwaltung des Arbeitsspeichers
- Verteilung des Arbeitsspeichers an die einzelnen Prozesse
- Speicherraum wird in Segmente unterteilt (Segmentierung)

Aufgaben eines Betriebssystems

Dateisysteme

- Verwaltung von Dateien
- Unterteilung des Datenträgers in logische Zuordnungseinheiten
- Verwendung von unterschiedlichen Modellen, um Dateien abzulegen

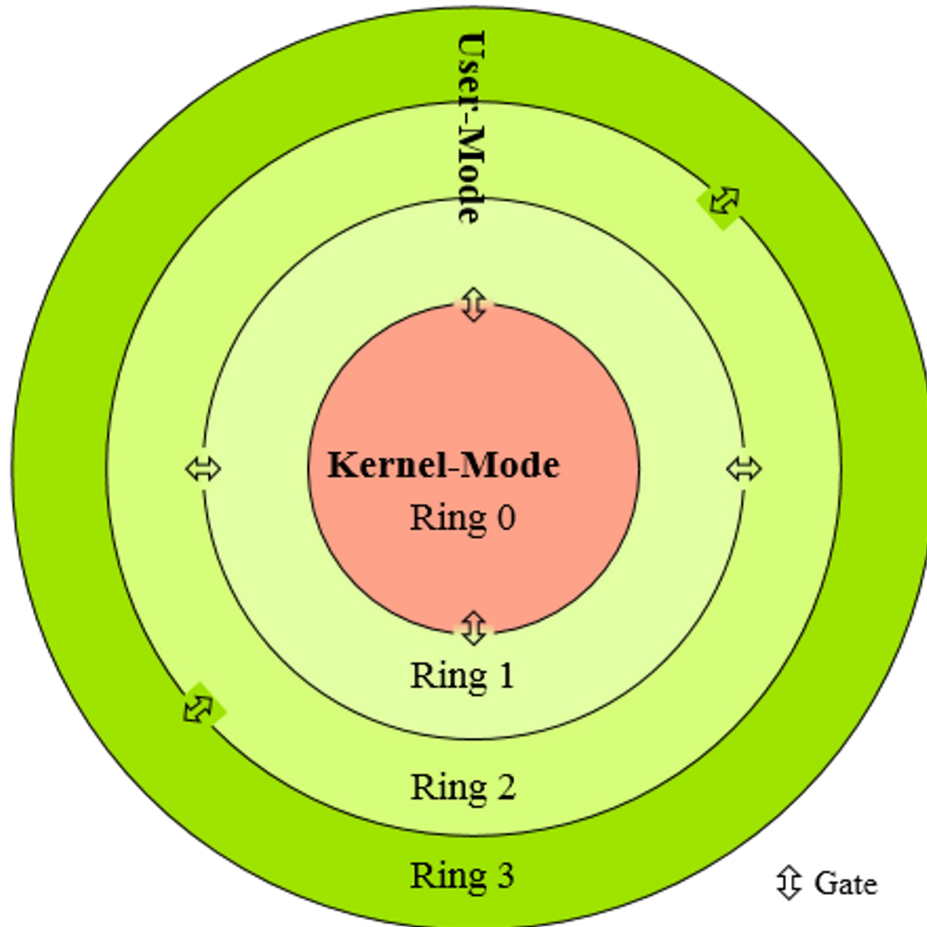
Der Kernel

- Der zentrale Bestandteil des Betriebssystems
- Wird im engeren Sinne als das eigentliche Betriebssystem gesehen
 - „Editoren, Compiler, Assembler, Binder und Kommandointerpreter sind definitiv nicht Teil des Betriebssystems, auch wenn sie bedeutsam und nützlich sind.“
 - Andrew S. Tanenbaum: [2]

Der Kernel

- Ist aufgebaut in folgende Schichten
 - Schnittstelle zur Hardware
 - Speicherverwaltung
 - Prozessverwaltung
 - Geräteverwaltung
 - Dateisysteme
- Es gibt verschiedene Kernel Arten

Kernel-und Benutzermodus



- 4 Privilegienstufen
- Prozesse bleiben innerhalb ihres Ringes
- Ring 0 hat vollen Zugriff
- Kommunikation erfolgt über Gates
- Selten werden alle Ringe involviert

Kernel-und Benutzermodus

Gründe für diese Unterscheidung

- Abschottung einzelner Prozesse
- Zugriffserlaubnis für unterprivilegierte Prozesse
- Trennung der Speicherbereiche
 - Sicherheit und Stabilität
- Fehlerhafte Prozesse erhalten keinen Zugriff

Kernelarten

auf monolithischem Kernel
basierte Betriebssysteme

Mikrokernel-basierte
Betriebssysteme

Hybridkernel-basierte
Betriebssysteme

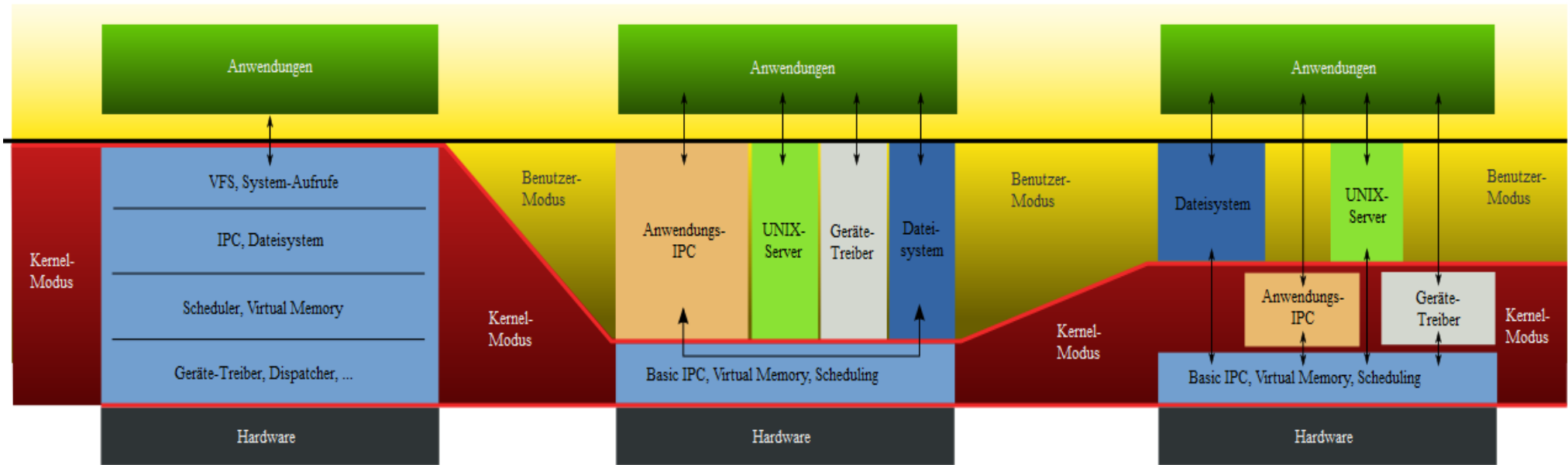
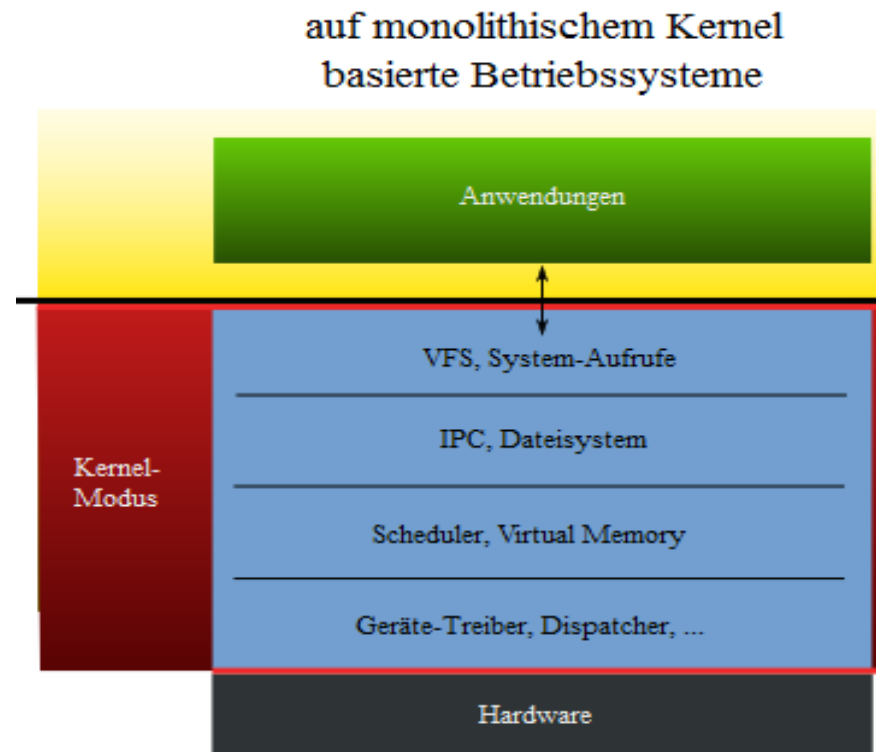


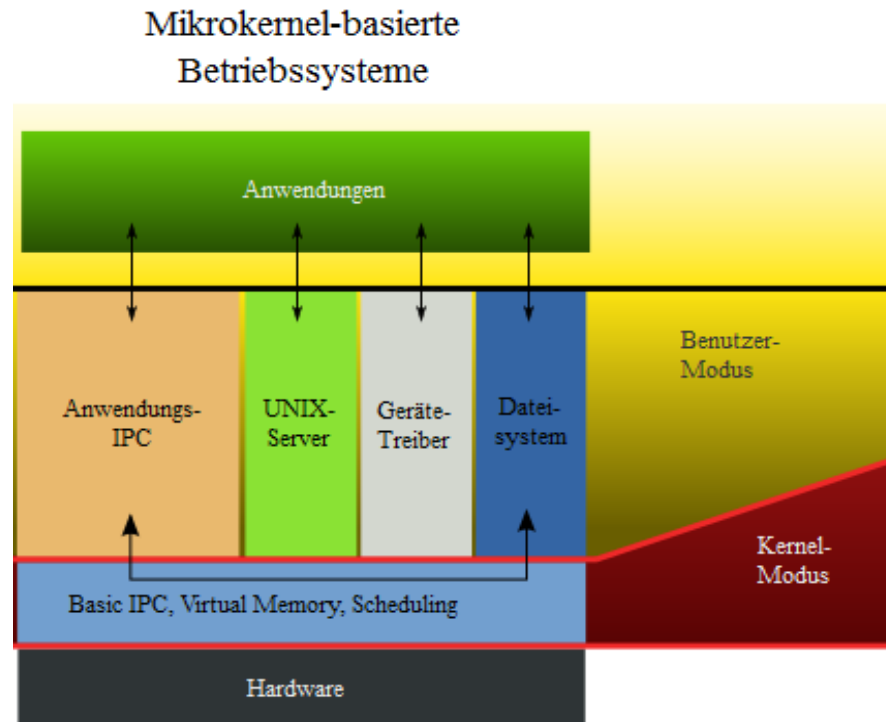
Bild geändert: http://de.wikipedia.org/wiki/Monolithischer_Kernel

Monolithischer Kernel



- Alle Betriebssystemkomponente im Kernelmodus
- Speicher- und Prozessverwaltungsfunktionen, Kommunikation zwischen Prozessen, Gerätetreiber und weitere Funktionen
- Beispiel: Linux

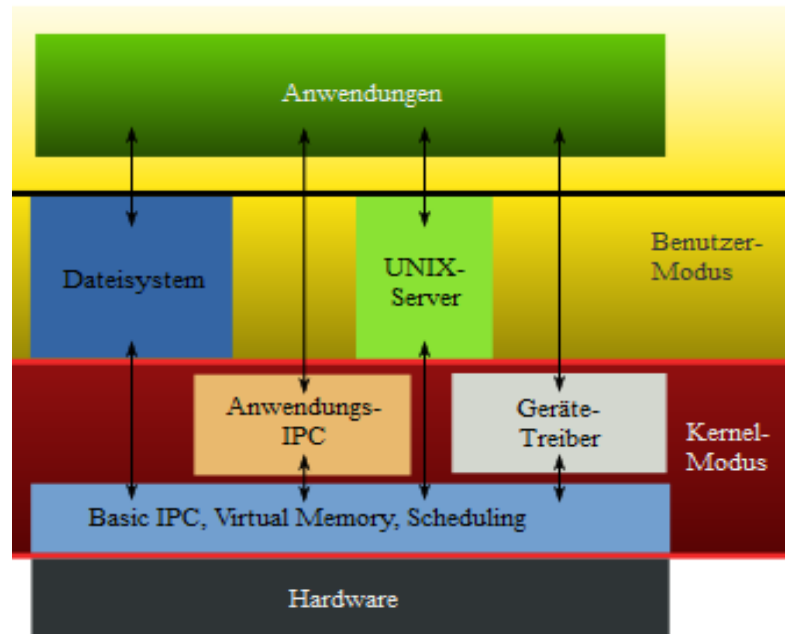
Mikrokernel



- Nur wenige Betriebssystemkomponente im Kernelmodus
- Grundfunktionen, Speicher-und Prozessverwaltung
- Beispiel: Minix

Hybridkernel

Hybridkernel-basierte
Betriebssysteme



- Kompromiss zwischen Mikro-und monolithischen Kernel
- Versuch, die Vorteile beider Kernelarten zu vereinen
- Beispiel: Mac OS X, Windows

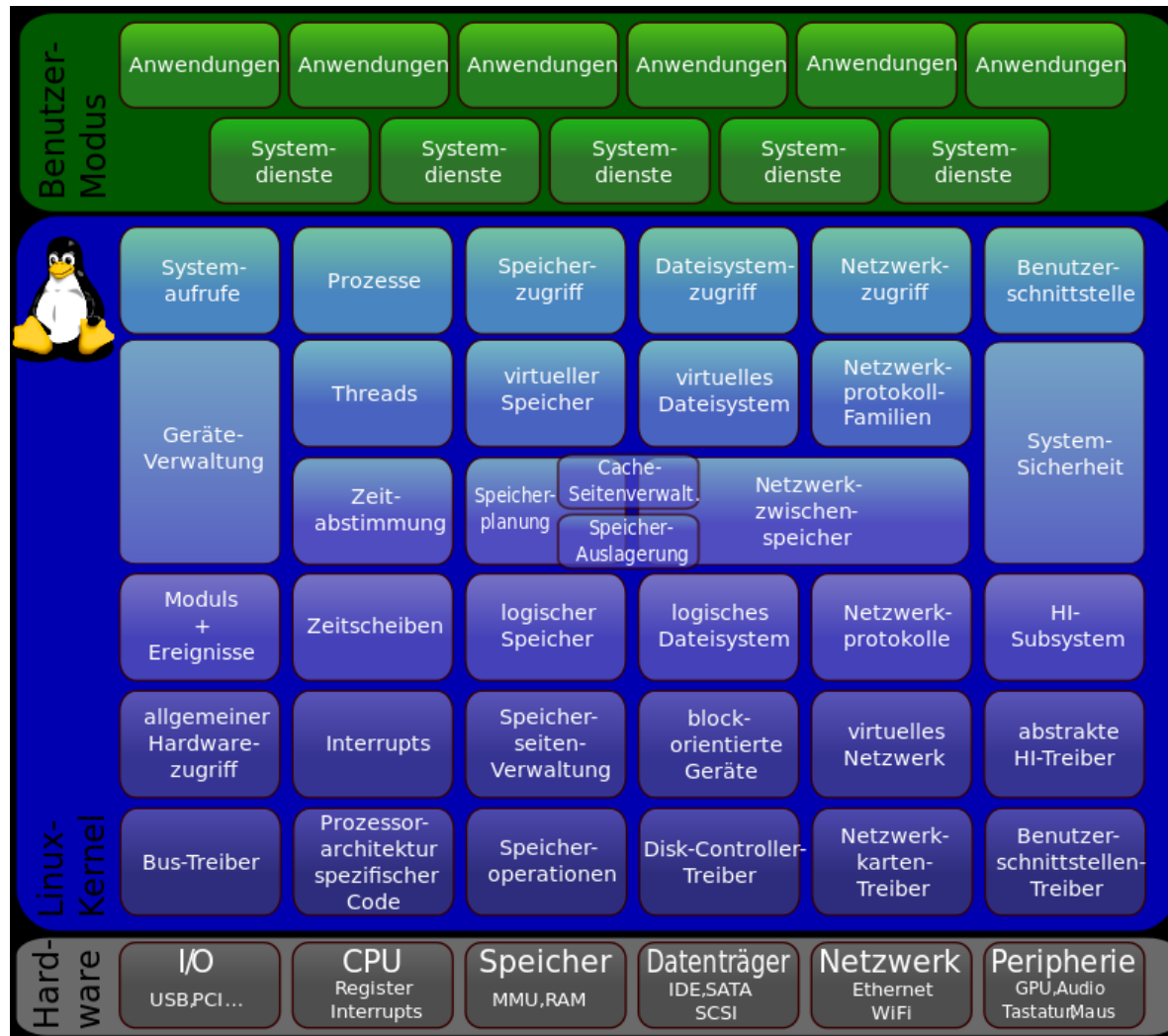
Vergleich der Kernelarten

	Monolithischer Kernel	Mikrokernel
Vorteile	<ul style="list-style-type: none">• Gute Performance• Keine Kommunikation notwendig• Nur minimaler Aufwand beim Wechsel der Prioritätsstufen	<ul style="list-style-type: none">• Hoher Ausfallschutz• Durch separierte Komponente Austausch ohne Beeinträchtigung möglich• Treiber im Benutzermodus
Nachteile	<ul style="list-style-type: none">• Aufwendiges Auswechseln von Komponenten• Komplettausfall durch fehlerhafte Systemkomponente• Funktionsänderung führt zur Kernel-Neuübersetzung	<ul style="list-style-type: none">• Geschwindigkeitsverlust durch Wechsel der Modi• Schwierige Synchronisation zwischen den Nutzer-Prozessen

Beispiel: Linux

- Linux enthält einen monolithischen Kernel
- Linux selbst ist der Kernel
- Der Kernel ist zuständig für:
 - Speicherverwaltung
 - Prozessverwaltung
 - Multitasking
 - Lastverteilung
 - Eingabe/Ausgabe-Operationen auf verschiedenen Geräten

Beispiel: Linux



Bildquelle: http://de.wikipedia.org/wiki/Linux_%28Kernel%29

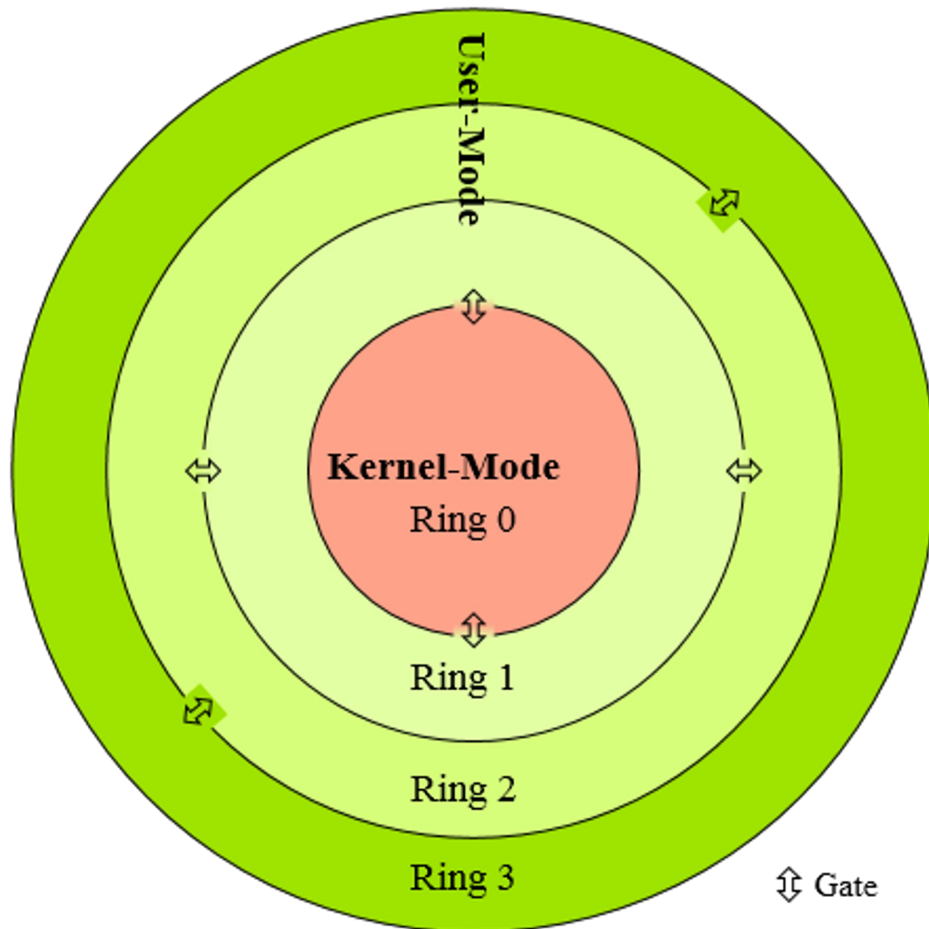
Beispiel: Linux

Unterschied zu einem strikt monolithischen Kernel:

- Linux nutzt Module
- Erhöhte Flexibilität, um unterschiedlichste Hardware anzusprechen
- Durch diese Flexibilität, keine Notwendigkeit, alle Treiber im Speicher zu halten

Zur Erinnerung

Kernel-und Benutzermodus



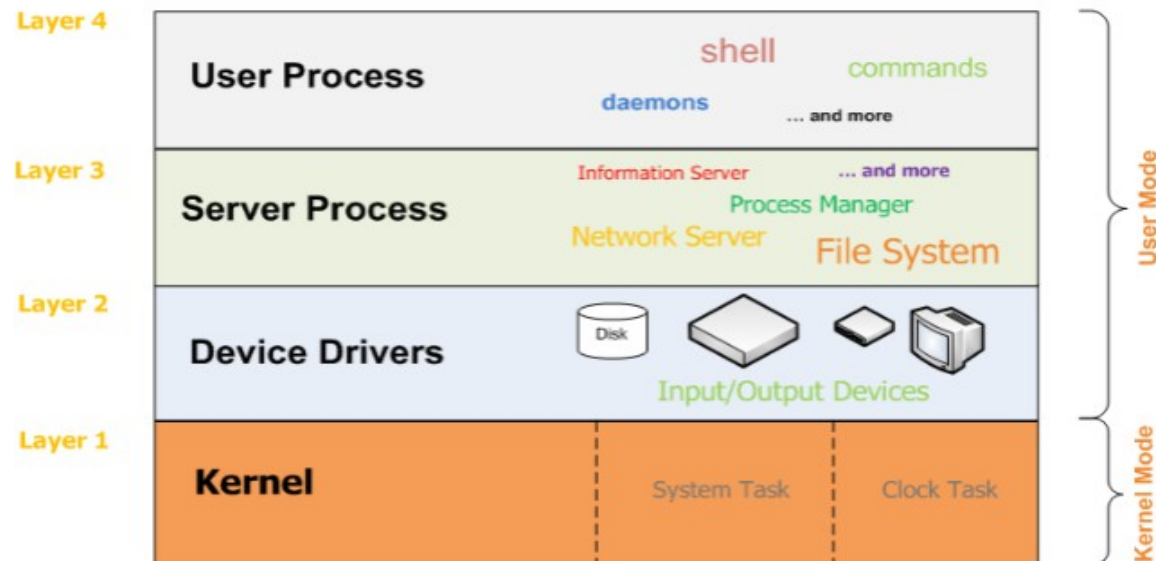
- 4 Privilegienstufen
- Prozesse bleiben innerhalb ihres Ringes
- Ring 0 hat vollen Zugriff
- Kommunikation erfolgt über Gates
- Selten werden alle Ringe involviert

Beispiel: Linux

- Treiber im Kernel und Kernel-Module sind im privilegiertem Modus (Ring 0)
- Wenige Module sind im eingeschränkten Benutzermodus (Ring 3)
- Ring 1 und 2 werden von Linux nicht genutzt

Beispiel(kurz) : Minix

Minix Layered Micro Kernel Architecture

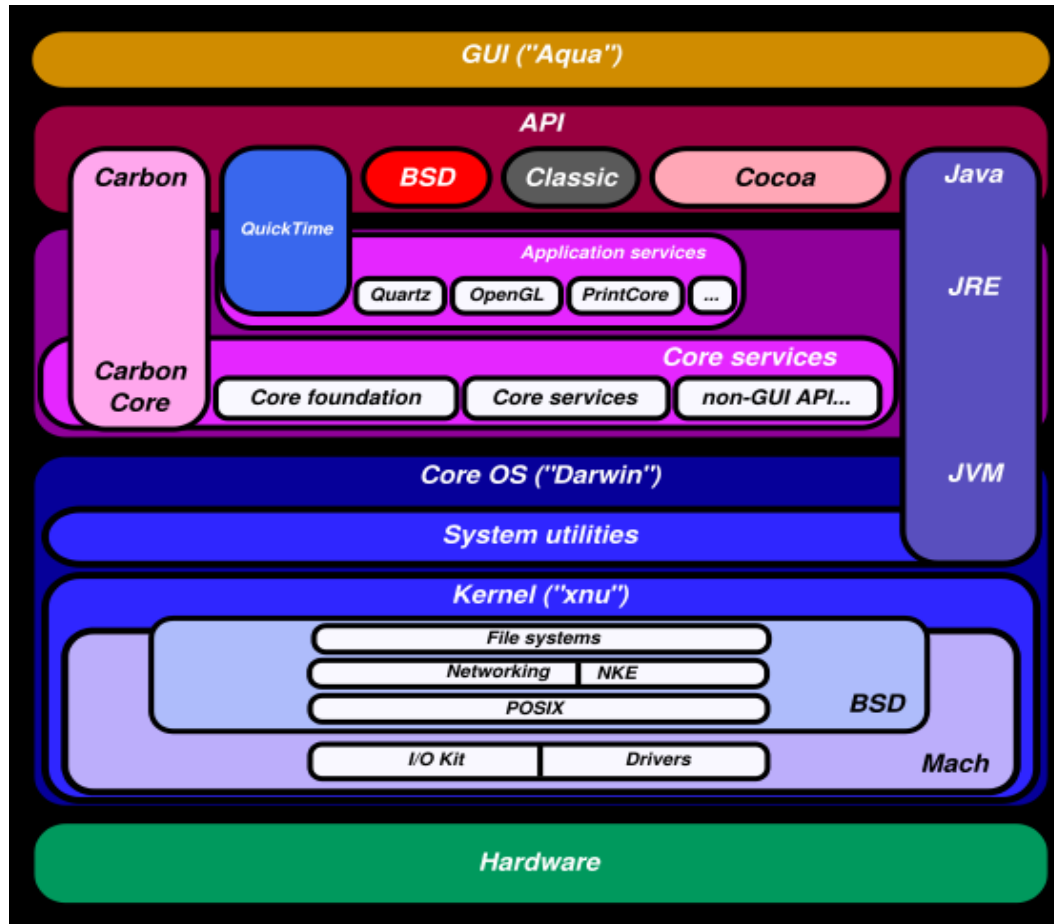


- Beispiel eines Mikrokernelns
- Entwickelt von Andrew S. Tanenbaum
- Steht als Gegenbeispiel, aber auch als Inspiration zu Linux (monolithisch) da
- Es laufen nur wenige Funktionen im Kernelmodus

Beispiel(kurz) : Mac OS X

- Das Betriebssystem wurde von Apple entwickelt
- In abgewandelter Form auch bekannt als iOS
- Es setzt auf einen hybriden Kernel

Beispiel(kurz) : Mac OS X



- Der XNU Kernel ist ein hybrider Kernel
- 1. Teil: der Mach-Kernel (Multitasking, Speicher-verwaltung, Fehlerbehandlung, grundlegende Gerätetreiber)
- 2. Teil: BSD-Kernel (Benutzerverwaltung, Dateirechte, Standarddateisysteme)

Bildquelle: http://de.wikibooks.org/wiki/Mac-OS-Kompendium/_Unter_der_Haube_von_Mac_OS_X:_UNIX/_Architektur_von_Mac_OS_X

Überblick Dateisysteme

- Dateisysteme sind Teil des Betriebssystems
- Kontrolle über die Benennung, Speicherung, und das Organisieren der Daten
- Viele verschiedene Anforderungen = verschiedene Arten von Dateisystemen

Überblick Dateisysteme

Wichtige Beispiele:

- Virtuelles Dateisystem
 - Ist eine Kernel Software Schicht
 - Ist eine Abstraktion des eigentlichen Dateisystems
 - Unterstützt diverse Dateisysteme
 - Bis zu 40, z.B. FAT, NFS, /proc-fs

Überblick Dateisysteme

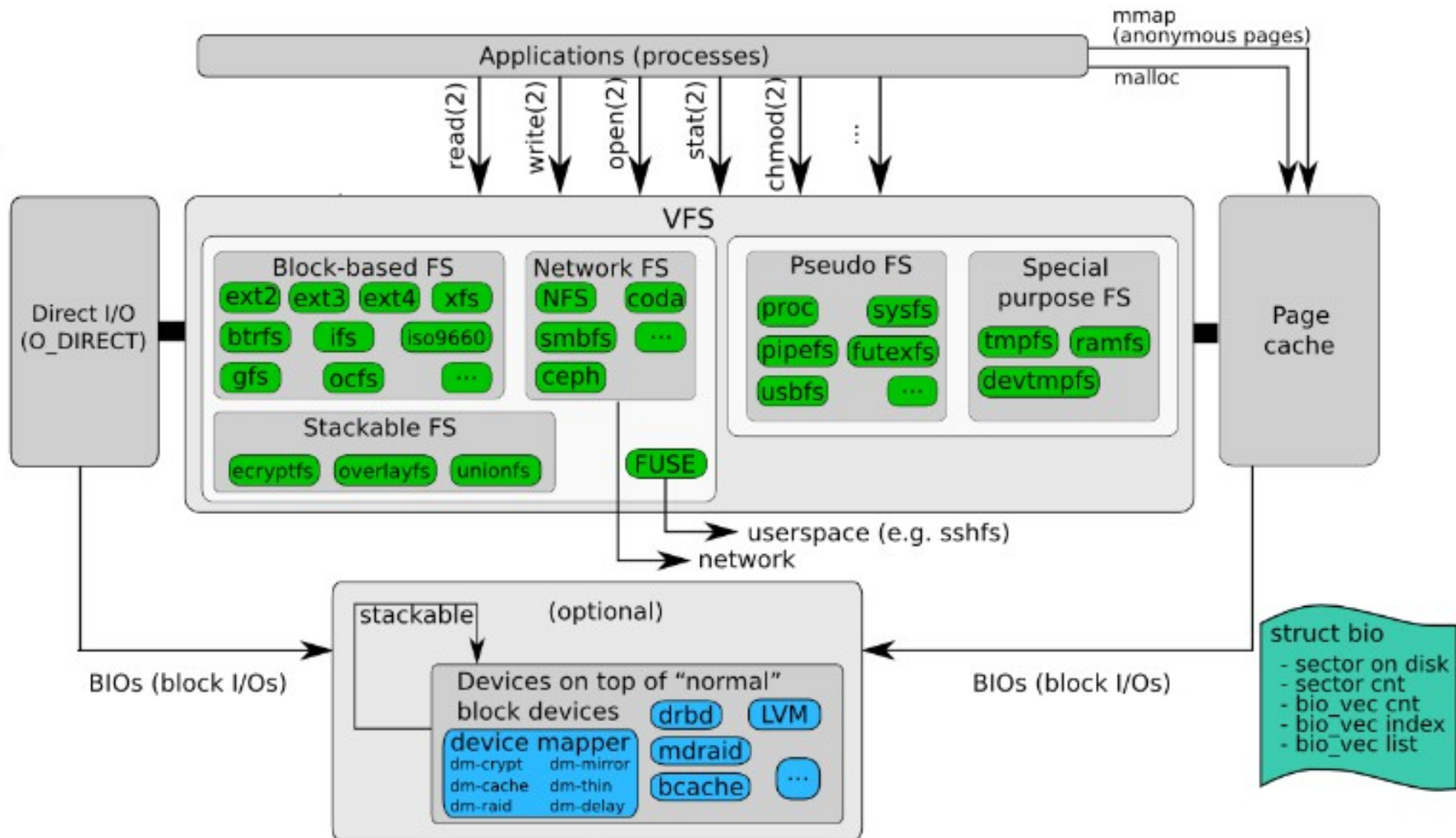


Bild geändert: https://www.thomas-krenn.com/de/wiki/Linux_Storage_Stack_Diagramm

Überblick Dateisysteme

Wichtige Bestandteile eines VFS

- Superblock
 - Informationen zu eingebundenen Dateisystemen
- Inodeobject,
 - Speichert Informationen zur Datei
 - Benutzt eine Nummer zur eindeutigen Identifikation
- Fileobject
 - Speichert Informationen zur Interaktion zwischen einer geöffneten Datei und einem Prozess
- Dentryobject
 - speichert die Verbindung zwischen einem Verzeichniseintrag und der Datei

Überblick Dateisysteme

Wichtige Beispiele:

- FUSE (Filesystem in Userspace)
 - Ermöglicht Dateisysteme in den Usermodus zu verlagern
 - Beschränkungen vom Kernelmodus können umgangen werden
 - „einfache“ Einbindung von den meisten Dateisystemen wird möglich

Überblick Dateisysteme

Andere Beispiele:

- /proc-fs
 - Braucht keinen direkten Speicher
 - Wird vom Kernel dynamisch erzeugt
 - Hat Informationen zu Prozessen
 - Das Arbeitsverzeichnis
 - Der virtuelle Speicher des Prozesses
 - Hat Informationen zur CPU

Überblick Dateisysteme

Andere Beispiele:

- tmpfs
 - Dieses Dateisystem nutzt ausschließlich das RAM
 - Alles Gespeicherte ist temporär
 - Vorteil: Geschwindigkeit
 - Nachteil: Totalverlust bei Systemabsturz

Zusammenfassung

- Es gibt verschiedene Arten von Betriebssystemen bzw. Betriebssystemkernen
- Unterschieden werden können:
 - Monolithischer Kernel
 - Mikrokernel
 - Hybridkernel
- Monolithischer Kernel: Alle Betriebssystemkomponente im Kernelmodus
- Mikrokernel: Wenige Komponente im Kernelmodus
- Hybridkernel: Kompromiss zwischen den beiden Kernelarten

Zusammenfassung

- Linux arbeitet als monolithischer Kernel
- Erfährt aber auch Einflüsse von Mikrokerneln

- Minix ist ein Beispiel für einen Mikrokernel
- Es steht als Kontrast zu Linux da

- Mac OS X ist ein Beispiel für einen Hybridkernel
- Es verbindet die Vorteile und Funktionen von anderen Kernen (Mach-Kernel und BSD Kernel)

Zusammenfassung

- Dateisysteme sind ein Teil des Betriebssystems
- Es gibt viele verschiedene Dateisysteme
- Das Virtuelle Dateisystem abstrahiert bestehende Dateisysteme
- Es unterstützt eine Vielzahl von Dateisystemen
- Wichtige Objekte im VFS sind: Superblock, Inodeobject, Fileobject, Dentryobject

Zusammenfassung

- FUSE verlagert Dateisysteme vom Kernelmodus in den Usermodus
- Einbindung wird unkomplizierter
- /proc-fs und tmpfs sind spezielle virtuelle Dateisysteme
- /proc-fs wird dynamisch erzeugt und zeigt Informationen zu Prozessen
- tmpfs nutzt das RAM und die Dateien sind temporär

Quellen

http://www.operating-system.org/betriebssystem/_german/w-wissen.htm

http://de.wikipedia.org/wiki/Kernel_%28Betriebssystem%29

http://de.wikipedia.org/wiki/Linux_%28Kernel%29

http://baun-vorlesungen.appspot.com/BTM1213/Skript/fohlen_bts_vorlesung_07_WS1213.pdf

http://de.wikibooks.org/wiki/Mac-OS-Kompendium/_Unter_der_Haube_von_Mac_OS_X:_UNIX/_Architektur_von_Mac_OS_X

http://de.wikipedia.org/wiki/OS_X

<http://openbook.rheinwerk-verlag.de/kit/itkomp04001.htm#Rxx355kap04001040002021F04F19F>

http://de.wikipedia.org/wiki/Ring_%28CPU%29

http://de.wikipedia.org/wiki/Monolithischer_Kernel

<http://de.wikipedia.org/wiki/Microkernel>

<http://de.wikipedia.org/wiki/Hybridkernel>

<http://de.wikipedia.org/wiki/Betriebssystem>

http://de.wikibooks.org/wiki/Betriebssystemtheorie/_Einleitung

<https://imma.wordpress.com/2007/04/02/presentation-internal-structure-of-minix/>

http://edoc.sub.uni-hamburg.de/informatik/volltexte/2015/210/pdf/bac_duwe.pdf

https://www.thomas-krenn.com/de/wiki/Linux_Storage_Stack_Diagramm

<http://www.inf.fu-berlin.de/lehre/SS01/OS/LabTalks/filesystem.pdf>