MAS: Betriebssysteme

Entwicklung von Betriebssystemen

T. Pospíšek

Gesamtüberblick

1. Einführung in Computersysteme
2. **Entwicklung von Betriebssystemen**
3. Architekturansätze
4. Interruptverarbeitung in Betriebssystemen
5. Prozesse und Threads
6. CPU-Scheduling
7. Synchronisation und Kommunikation
8. Speicherverwaltung
9. Geräte- und Dateiverwaltung
10. Betriebssystemvirtualisierung

Überblick

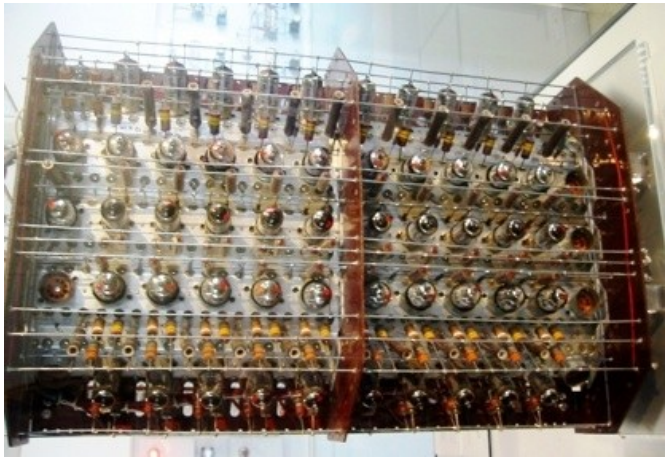
1. **Geschichtliches**
2. Klassische Großrechnerbetriebsarten
3. Fallstudien

Historische Entwicklung von Rechnern und Betriebssystemen

Zürcher Hochschule für Angewandte Wissenschaften
School of Engineering

1. Generation (1945 – 1955)

- Minimale Betriebssysteme
- Röhrencomputer
- Maschinensprache, kein Assembler
- Lochkarten ab 1950



Röhrencomputer der Rechenanlage ORACLE
Deutsches Museum

2. Generation (1955 – 1965)

3. Generation (1965 – 1980)

Lochkartenleser von Control Data
Quelle: Wikipedia

4. Generation (1980 – ...)

Weitere Rechenanlagen:
(Gewicht: Tonnen)

- ZUSE Z22 (BRD)
- D1/D2 (DDR)
- Colossus (GB)
- ENIAC (USA)
- IBM 305 RAMAC

Historische Entwicklung von Rechnern und Betriebssystemen

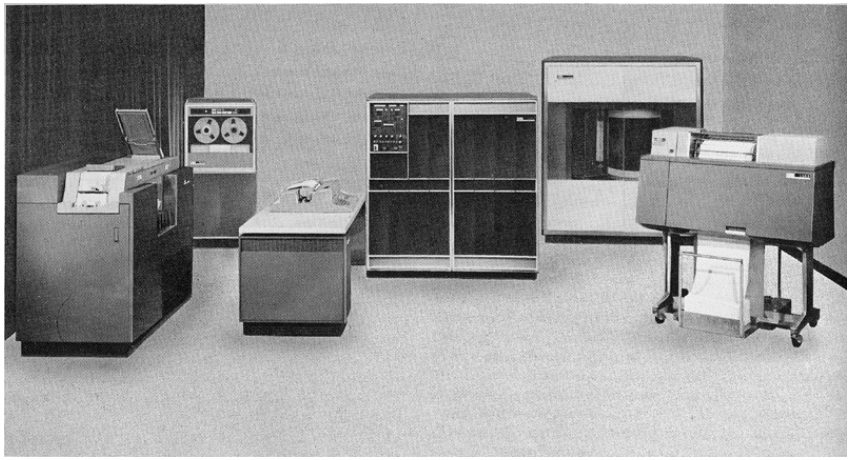
1. Generation (1945 – 1955)

2. Generation (1955 – 1965)

3. Generation (1965 – 1980)

4. Generation (1980 – ...)

- Etwas komplexere Betriebssysteme
- Transistorencomputer
- Assemblersprachen
- Mainframes, Batchverarbeitung: Jobs hintereinander ausgeführt
- IBM 1401, 7094



IBM-1401-Anlage
Quelle: IBM

Historische Entwicklung von Rechnern und Betriebssystemen

1. Generation (1945 – 1955)

2. Generation (1955 – 1965)

3. Generation (1965 – 1980)

4. Generation (1980 – ...)

PDP-11 von digital
Quelle: Wikipedia

- Umfangreiche Betriebssysteme wie OS/360, BS1000, MULTICS, Unix
- Integrated Circuits
- Hochsprachen
- Mainframes, Multiprogramming, Timesharing (Mehrbenutzerbetrieb)
- IBM-Systeme, Siemens-Systeme, DEC PDP-11, ...

S/360-System
Quelle: Wikipedia

Historische Entwicklung von Rechnern und Betriebssystemen

Zürcher Hochschule für Angewandte Wissenschaften
School of Engineering

1. Generation (1945 – 1955)

2. Generation (1955 – 1965)

3. Generation (1965 – 1980)

4. Generation (1980 – ...)

IBM PC, Modell IBM 5150
Quelle: IBM

IBM zSeries
Quelle: IBM

Enterprise Server von Sun
Quelle: Sun Mircosystems

- Komplexe Betriebssysteme
- Large Scale Integration
- Objektorientierte Sprachen
- Deutliche Verbesserung des User Interface
- PCs, Workstations, Server, Mainframes, Verteilte Systeme
- MS-DOS, Unix, Windows, IBM-OS/390, z/OS, Mac OS X, Android ...

Historische Entwicklung der Betriebssysteme

- **1. Generation** (ca. 1945 - 1955)
  - Röhrencomputer (ca. 20.000 Röhren)
  - Programmierung in reiner Maschinensprache (kein Assembler, keine Hochsprache)
  - Lochkarten ab ca. 1950

Röhren als aktive elektronische Bauelelemete
Quelle:www.wikipedia.de

Historische Entwicklung der Betriebssysteme

- **2. Generation** (ca. 1955 - 1965)
  - Transistoren wurden verwendet
  - **Stapelverarbeitung** (Batch-Verarbeitung): IBM 1401, 7094
  - Jobs wurden von Lochkarte auf Magnetband eingelesen und dann hintereinander abgearbeitet
  - Ein Programm nach dem anderen wurde ausgeführt, die Ergebnisse auf Band gespeichert und am Ende ausgedruckt
  - Einfaches Betriebssystem

Historische Entwicklung der Betriebssysteme

- **3. Generation** (ca. 1965 - 1980)
  - Integrated Circuits (ICs), kleinere integrierte Schaltungen
  - IBM System/360 (Serie von Rechnern), IBM System/370, 3080, 3090
  - Einführung von **Multiprogramming** (Mehrprogrammbetrieb, Multitasking):
    - Während I/O-Wartezeit wurde CPU für neuen Job vergeben
  - **Spooling**: Jobs von Platte übernehmen und Ergebnisse auf Platte schreiben
  - Später **Timesharing** (mit Mehrbenutzerbetrieb) als Variante des Multiprogramming:
    - Online-Zugang über Terminal, CPU wird aufgeteilt
    - Am MIT entwickelt: Betriebssystem CTSS, MULTICS
    - Minicomputer DEC PDP-1, PDP-11: Unix

Historische Entwicklung der Betriebssysteme

- **4. Generation** (ca. 1980 - heute)
  - Personal Computer und Workstations
  - Large Scale Integration (LSI-Schaltungen), Millionen von Transistoren auf einem Silizium-Chip (Si, Halbmetall)
  - Betriebssysteme IBM OS/360, MS-DOS, Unix, Unix BSD, Unix System V, IBM OS/2, MS Windows-Derivate und Linux
  - Benutzerfreundlichkeit stieg immer mehr (X-Windows, Motif, OS/2 Presentation Manager)
  - Netzwerkbetriebssysteme und verteilte Betriebssysteme

Überblick

1. Geschichtliches
2. **Klassische Großrechnerbetriebsarten**
3. Fallstudien

Mehrprozessorsysteme

- Mehrprozessorsysteme sind die Basis für echte Parallelverarbeitung
  - Heute sind Mehrkernsysteme schon in Smartphones Standard
- Für Betriebssysteme wird die Verarbeitung komplexer Synchronisation der Zugriffe auf Ressourcen erforderlich
  - Hauptspeicherzugriff
  - Kernel-interne Datenstrukturen
  - Ein-/Ausgabesystem
  - ...

Singletasking und Multitasking

- **Einprogrammbetrieb (singletasking)**
  - Nur ein (Teil-)Programm ist aktiv, das bearbeitete Programm erhält sämtliche Betriebsmittel zugeteilt
- **Mehrprogrammbetrieb (multitasking)**
  - Mehrere Programme sind aktiv, für Dialogverarbeitung
  - Für die Ausführung benötigten Betriebsmittel werden abwechselnd zugeteilt nach Prioritäten oder Zeitscheibenverfahren
    - Zuordnung des Prozessors zu verschiedenen Programmen nach Zeitintervallen → **time sharing**

    - Mehrprogrammbetrieb erfordert **nicht** unbedingt Mehrprozessorsysteme

Stapelverarbeitung versus interaktive Verarbeitung

Zürcher Hochschule für Angewandte Wissenschaften
School of Engineering

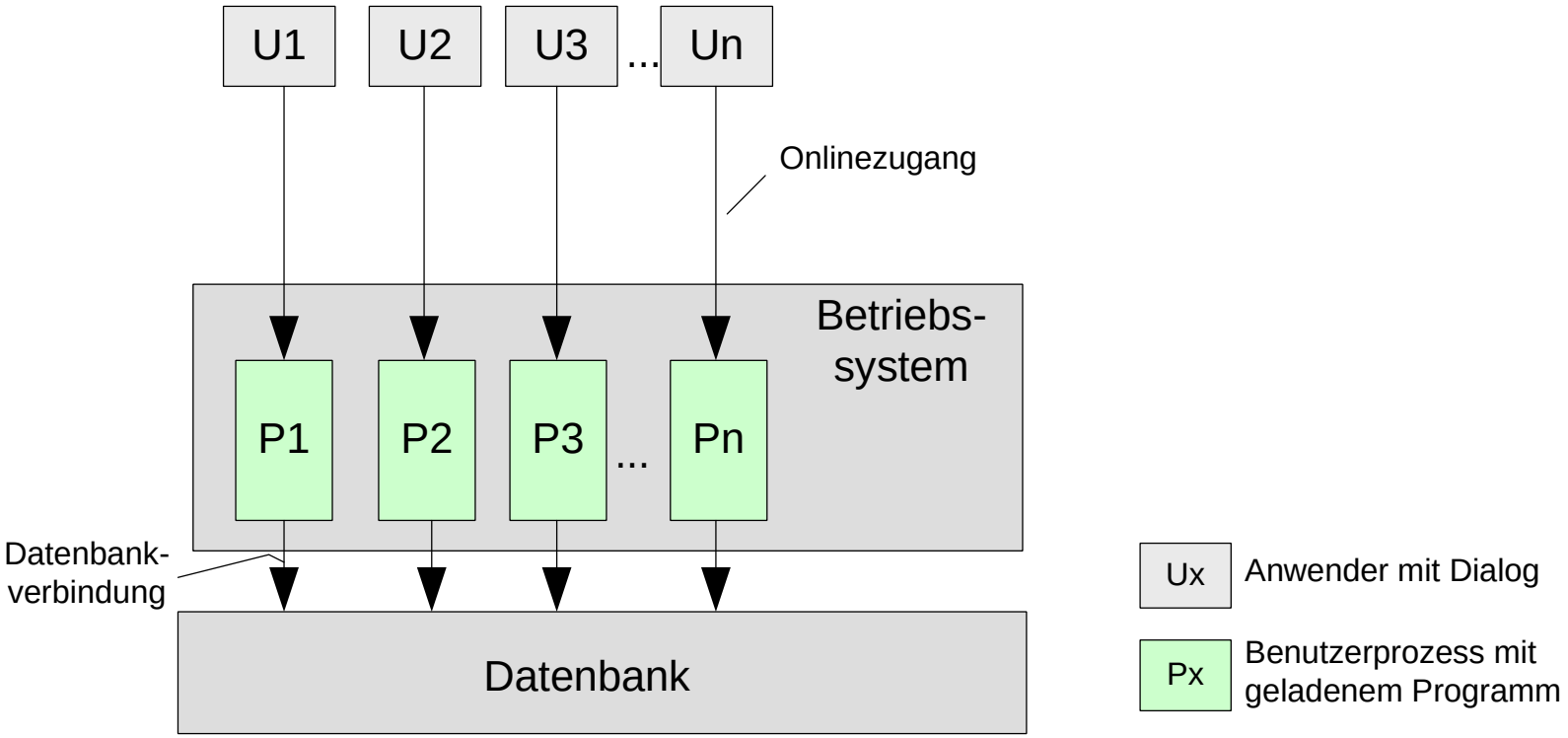
- **Stapelverarbeitung (Batchprocessing)**
  - Zu bearbeitender Auftrag (Job) muss für die Bearbeitung vollständig sein
  - Aufträge werden in einer Warteschlange verwaltet und nach definierter Strategie abgearbeitet
- **Interaktive Verarbeitung**
  - Auftrag muss vor der Bearbeitung nicht vollständig definiert sein
  - Permanente Kommunikation des Nutzers mit dem Betriebssystem über User Interface

Betriebsarten nach der Programmnutzung

- **Teilhaberbetrieb (transaction mode)**
  - Mehrere Anwender arbeiten mit einem **Transaktionsmonitor** (IBM CICS, BS2000 openUTM), d.h. mehrere Anwender arbeiten gleichzeitig an demselben Rechner mit demselben Programm
  - Das System führt die Anforderungen der Anwender in sog. **Transaktionen** aus
  - Transaktionen werden komplett oder gar nicht bearbeitet
  - Beispiel: Zentrale Buchungssysteme
- **Teilnehmerbetrieb (time sharing)**
  - Mehrere Anwender arbeiten mit einem zentralen Rechner, aber mit unterschiedlichen, von einander unabhängigen Programmen und Daten
  - Rechner sieht für jeden Anwender wie *eigener Rechner* aus

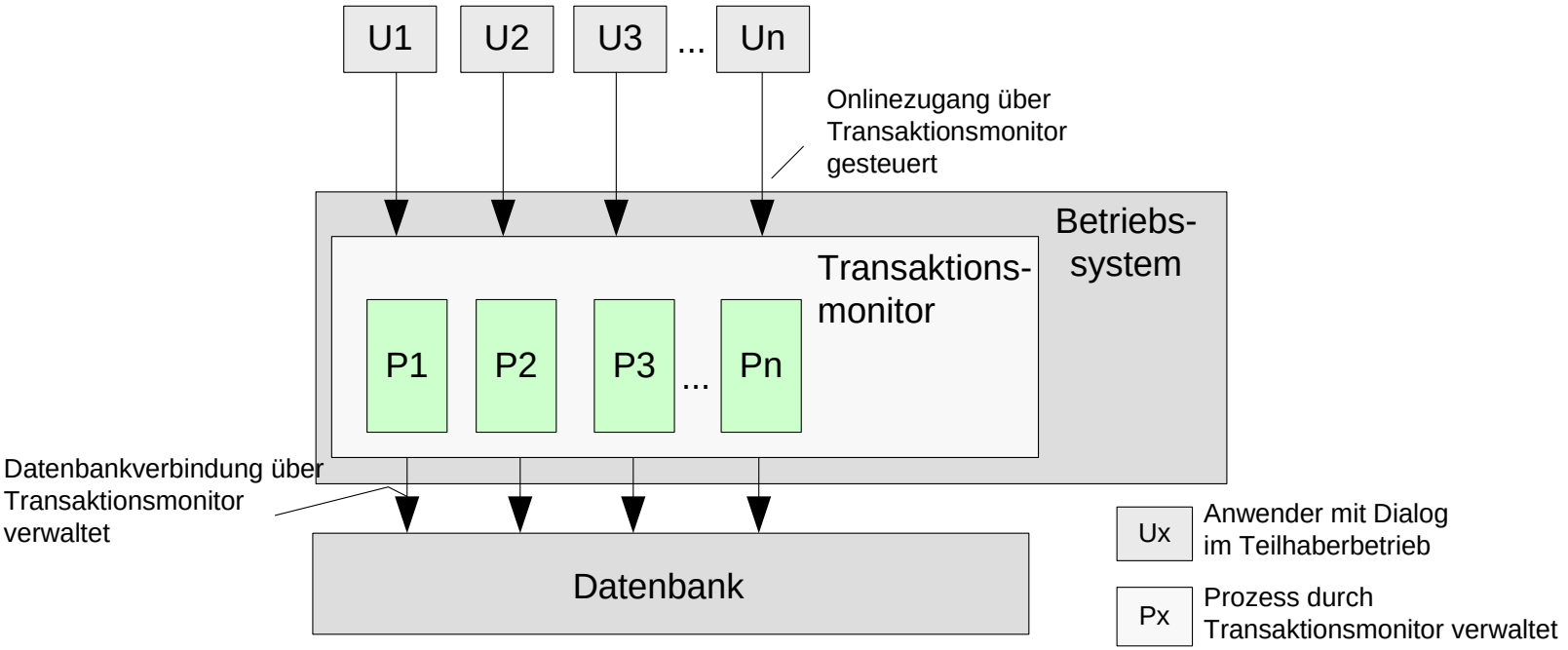
Teilnehmerbetrieb

- Im Teilnehmerbetrieb erhält jeder Anwender einen eigenen Prozess und sonstige Betriebsmittel vom Betriebssystem zugeteilt



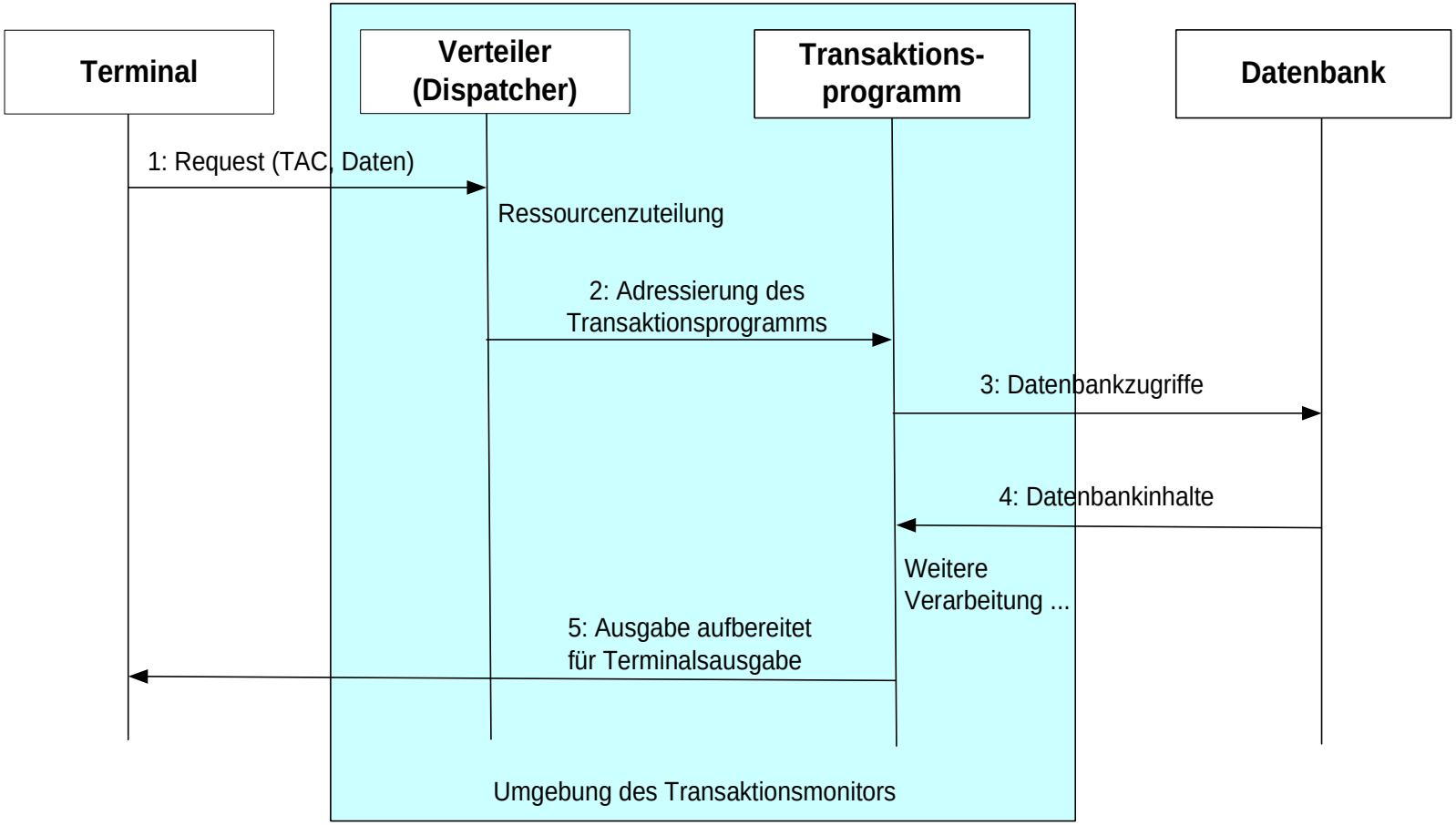
Teilhaberbetrieb

- Teilhaberbetrieb mit Transaktionsmonitor (CICS, UTM,...), auch DB/DC-Systeme genannt
- Prozesse und sonstige Betriebsmittel werden vom Transaktionsmonitor zugeteilt



Teilhaberbetrieb: Ablauf

- Typischer Ablauf beim Aufruf eines Transaktionsprogramms



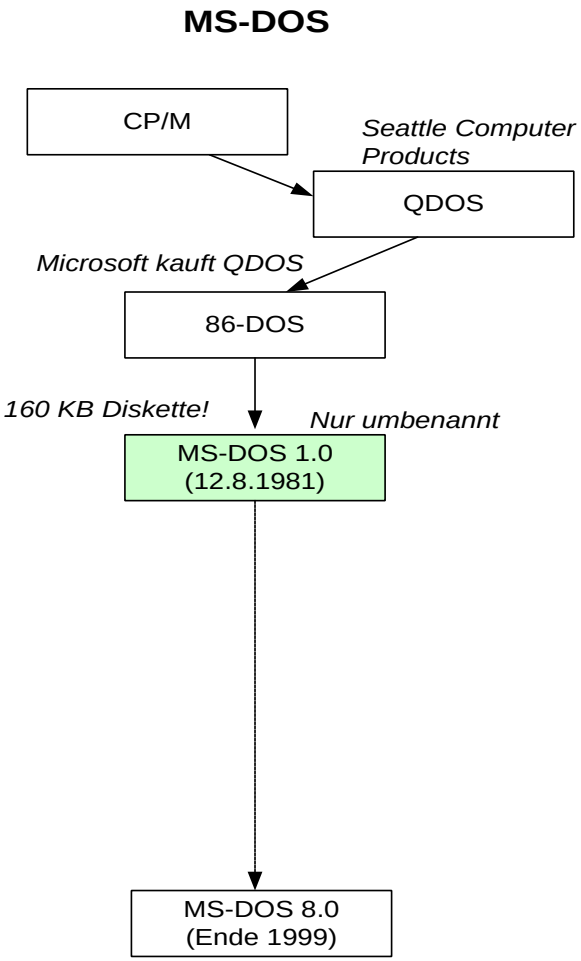
Überblick

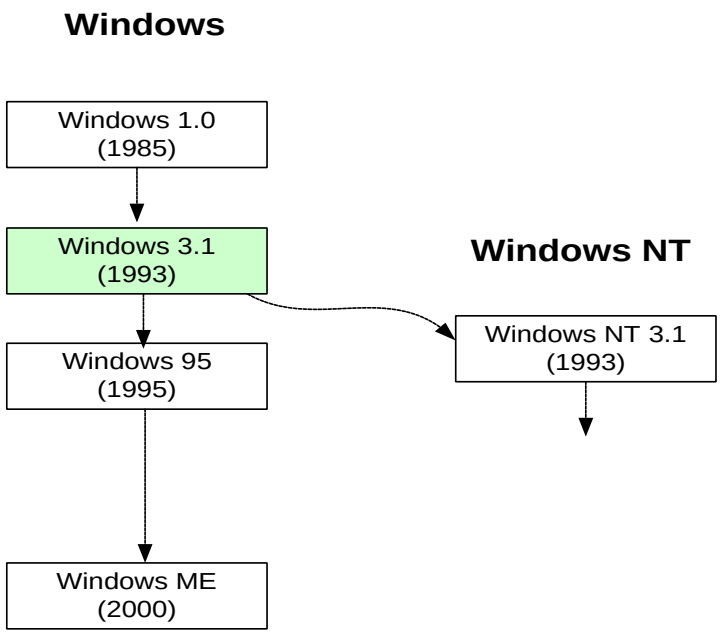
1. Geschichtliches
2. Klassische Großrechnerbetriebsarten
3. **Fallstudien**

Typische Betriebssysteme heute

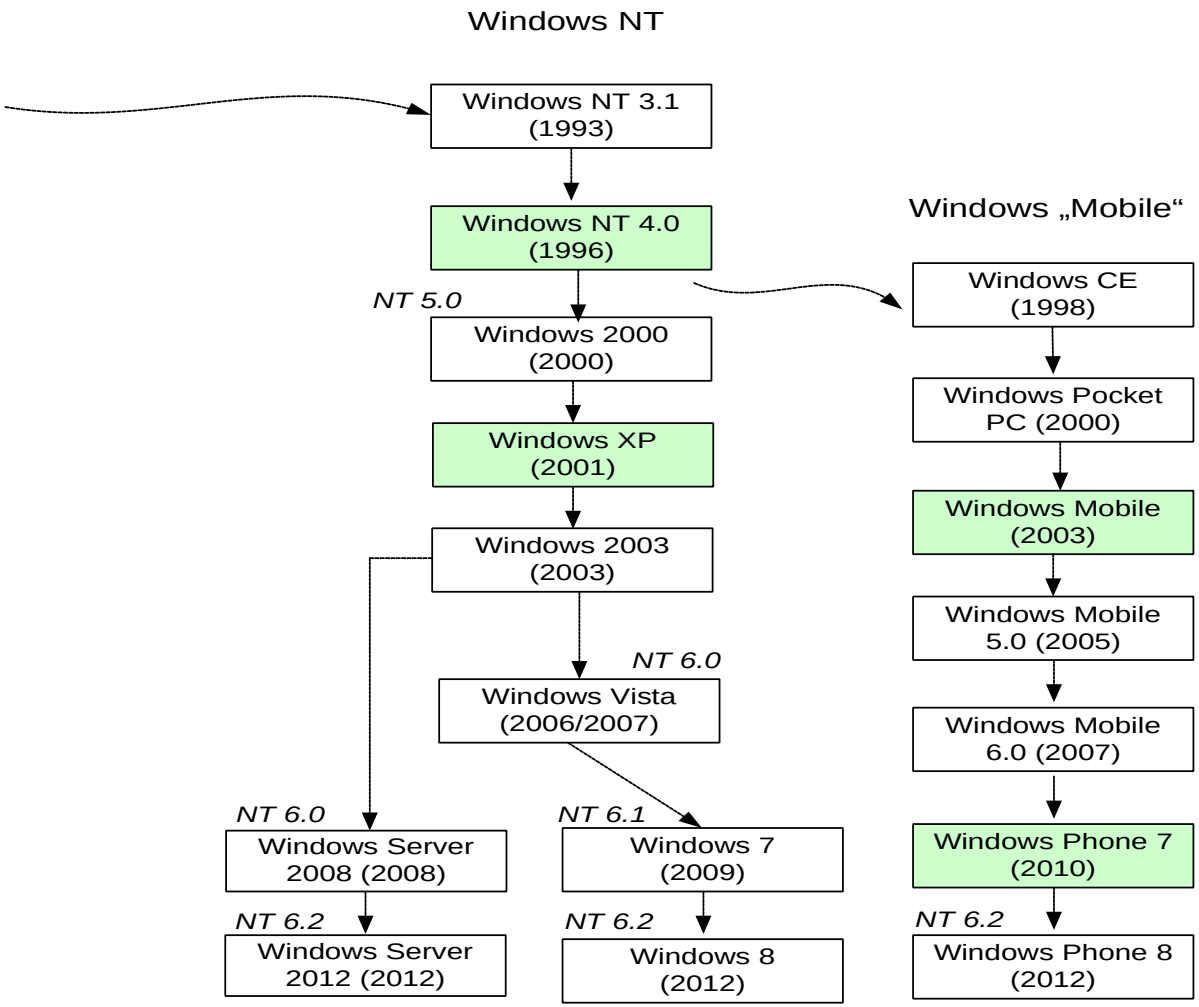
- **MS-DOS** (es soll noch welche geben)
- **Windows**: Windows 2008/2012, Windows XP, Windows 7, 8, 10
- Andere wichtige Systeme:
  - **OS/370** (heute: zSeries S/xxx, z/OS)
  - **OS/400** (heute: iSeries 400)
  - **Unix**: Sun Solaris, HP UX, AIX, Linux, BSDs, Android, Mac OS X, ...
- Wichtiger Grund für eine weite Verbreitung:
  - meist nicht die Qualität sondern die Beliebtheit der Anwenderprogramme
  - Trägheit, Netzwerk-Effekt

Historische Entwicklung: Windows (1)

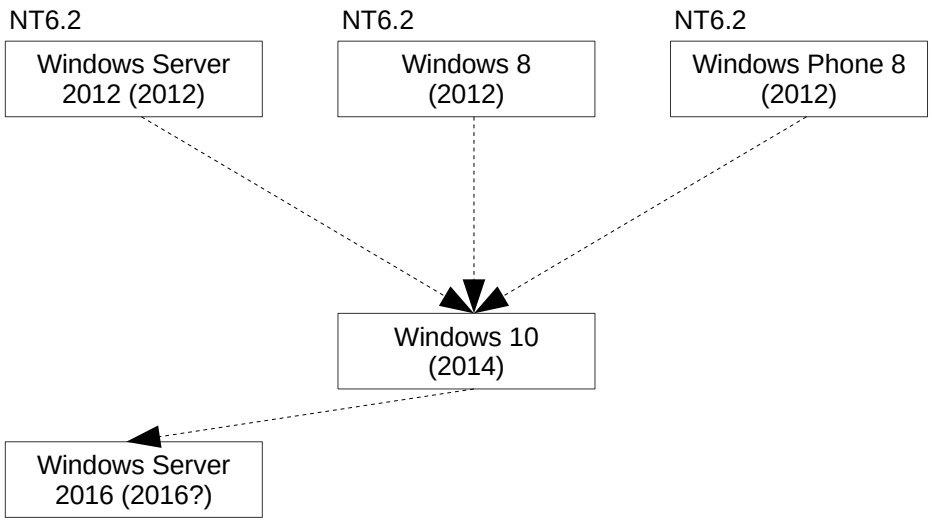




Historische Entwicklung: Windows (2)



Historische Entwicklung: Windows (3)



Kurze Geschichte von Windows (1)

- **MS-DOS V1.0** (Microsoft, Startup) wurde 1981 von IBM mit einem 8088-basierten IBM PC herausgegeben
  - Real-Mode-System und Single-User-System
  - Kommandozeilen-orientiert (von Unix abgeschaut)
  - 8-Bit-Betriebssystem
  - Einfaches Filesystem
- Später kam **MS-DOS V3.0** mit dem **PC/AT** heraus mit
  - 80286-Unterstützung
  - 16 MB realen Adressraum
  - Weiterhin Kommandozeilen-orientiert

Kurze Geschichte von Windows (2)

- **Windows 1.0** (1985) war das erste graphische User-Interface für MS-DOS
- **Windows 3.0** (1990) und die Nachfolger V3.1 und V3.11 waren bereits sehr erfolgreich
  - Weiterhin kein echtes Betriebssystem, sondern mehr eine Benutzeroberfläche
  - MS-DOS war die Basis

Kurze Geschichte von Windows (2)

- **Windows NT 3.1** (New Technology, 1993) wurde von Grund auf als 32-Bit-System konzipiert
  - urpsrüngliches Ziel: OS/2 und POSIX Kompatibilität
  - von VMS Entwicklern mitentwickelt
  - von Microkernel Architektur inspiriert
  - anfangs nicht erfolgreich, daher wurde Windows 95 notwendig

Kurze Geschichte von Windows (3)

- **Windows 95** (1995) brachte dann mehr Features:
  - Virtuellen Speicher und Multiprogramming
  - War aber immer noch mit **MS-DOS (nun V7.0)** verbandelt
  - Weiterhin viele 16-Bit-Codeelemente
  - MS-DOS Filesystem weiter genutzt (8+3 Byte Filenamen)
- **Windows NT 4.0** (1996)
  - User-Interface von Windows 95
  - Recht leistungsfähiges Server-Betriebssystem
  - Neues Filesystem NTFS
  - Keine 100%-MS-DOS-Kompatibilität
  - Erfolgreich!!

Kurze Geschichte von Windows (4)

- **Windows 98** (1998) kam mit einem besseren User-Interface mit Internet-Integration (Monopolvorwurf!)
  - Immer noch mit **MS-DOS (nun V7.1)** verbandelt, weiterhin viele 16-Bit-Codeelemente
  - Kein großer Unterschied zu Windows 95
  - Multiprogramming System, aber **nicht reentrant-fähiger Kernel** → Verwendung von Locks verlangsamte das System
  - Aus Kompatibilitätsgründen mussten MS-DOS-Programme auf den Interrupt-Vektor zugreifen und bekamen 1 MB vom Adressraum, in dem auch Kernel-Daten lagen
  - Systemabstürze durch Fehler in MS-DOS-Programmen waren die Folge

Kurze Geschichte von Windows (5)

- **Windows Me** (Millenium Edition, 2000) brachte nichts wesentlich Neues
- Windows NT 5.0 wurde zu **Windows 2000** umbenannt
  - Vereinheitlichung der Systeme mit Windows 98 User-Interface und volles 32-Bit-System
  - Plug-and-play Devices, USB-, IrDA (Infroarot-Link) und Firewire-Support, Internationalisierung,...
  - Unterstützt bis zu 32 CPUs in symmetrischen Multiprozessorsystemen
- **Windows XP, Windows 2003, Windows Vista, Windows 2008/2012, Windows 7/8, Windows Phone 7/8**
  - Neuere Versionen basieren alle auf NT 6.2
  - 32- und 64-Bit-Systeme

Kurze Geschichte von Unix (1)

- Unix entstand aus **MULTICS** (Multiplexed Information and Computing Service), auch: Unics
- Erste Single-User Version von Unix in den Bell Labs auf einer PDP-7 von **Ken Thompson** und **Dennis Ritchie** entwickelt (1969)
- Zwei inkompatible Hauptversionen entstanden
  - Die Berkeley University entwickelte das **BSD** (Berkeley Software Distribution)
    - Vorbild für Sun OS von Sun Microsystems
    - Heute gibt es viele Nachfolger: FreeBSD, NetBSD, OpenBSD, DragonFly BSD, Mac OSX (sehr erfolgreich)
  - **System V** von AT&T (wechselte mehrfach den Besitzer)
- Weitere Unix-Derivate heute haben ihre Feinheiten: HP UX, Sun Solaris, Sinix, Reliant Unix (Fujitsu Siemens), AIX (IBM),...

Kurze Geschichte von Unix (3)

- IEEE entwickelte einen Standard namens **POSIX**
  - Definiert ein System Call Interface, das von einem kompatiblen Unix unterstützt werden muss
  - Wird von allen Herstellern unterstützt
- Tanenbaum entwickelte 1987 einen kleinen Unix-Clone namens **MINIX** (ca. 12.500 LOC)
  - Heute: MINIX 3 als Forschungsprojekt für zuverlässige Betriebssysteme; Open-Source-Projekt: www.minix3.org

Kurze Geschichte von Unix (3)

- Nach MINIX entstand **Linux** durch **Linus Torvalds** (ehemals finnischer Student) als Open Source Unix

  - Erfreut sich heute immer weiterer Verbreitung über Distributoren, die auch Service anbieten
  - Mischung aus System V, BSD und eigenen Erweiterungen
  - Distributionen heute: Ubuntu, SUSE, Debian, Red Hat, CentOS, Fedora, ...

Kurze Geschichte von Open Source (1)

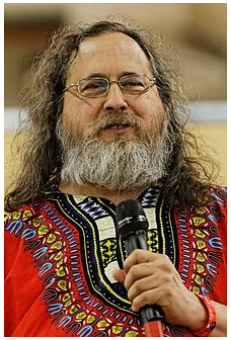
- 1950/1960 wurde Hardware verkauft, Software „gehörte dazu“
  - SW wurde im auch in Quellform geliefert, damit der Benutzer sie anpassen, erweitern und Fehler beheben kann
  - viele Unis unter den ersten Computernutzern, Verbesserungen an SW wurden im akademischen Geiste untereinander geteilt
  - Entwicklung von ARPANET mittels RFCs
- in den späten 1960 war SW bereits „sehr komplex“ und wurde teilweise verkauft
- Software/Sourcecode auf Tapes getauscht
- danach BBS

Kurze Geschichte von Open Source (2)

- AT&T Unix (später System V)
  - Anfangs 70er noch gratis
  - viele Weiterentwicklungen von dritten
  - Anfangs 80er verbreitet, aber Lock-In, AT&T fängt an das System zu verkaufen
- AT&T verklagt andere Berkley
  - „Verletzung von Eigentumsrechten“
  - BSD – Berkeley Software Distribution
  - BSD hat AT&T Lizenz
  - Modifiziert und reimplementiert AT&Ts Unix
  - BSD Unix Familie entsteht
- Entwicklung von ARPANET mittels RFCs

Kurze Geschichte von Open Source (3)

- 1983 GNU Manifesto von Richard Stallmann
  - prägt Begriff „Free Software“ und Philosophie
  - Motive:
    - kann Fehler von SW nicht selbst beheben
    - darf SW nicht benutzen, an welcher er selbst mitgearbeitet hatte



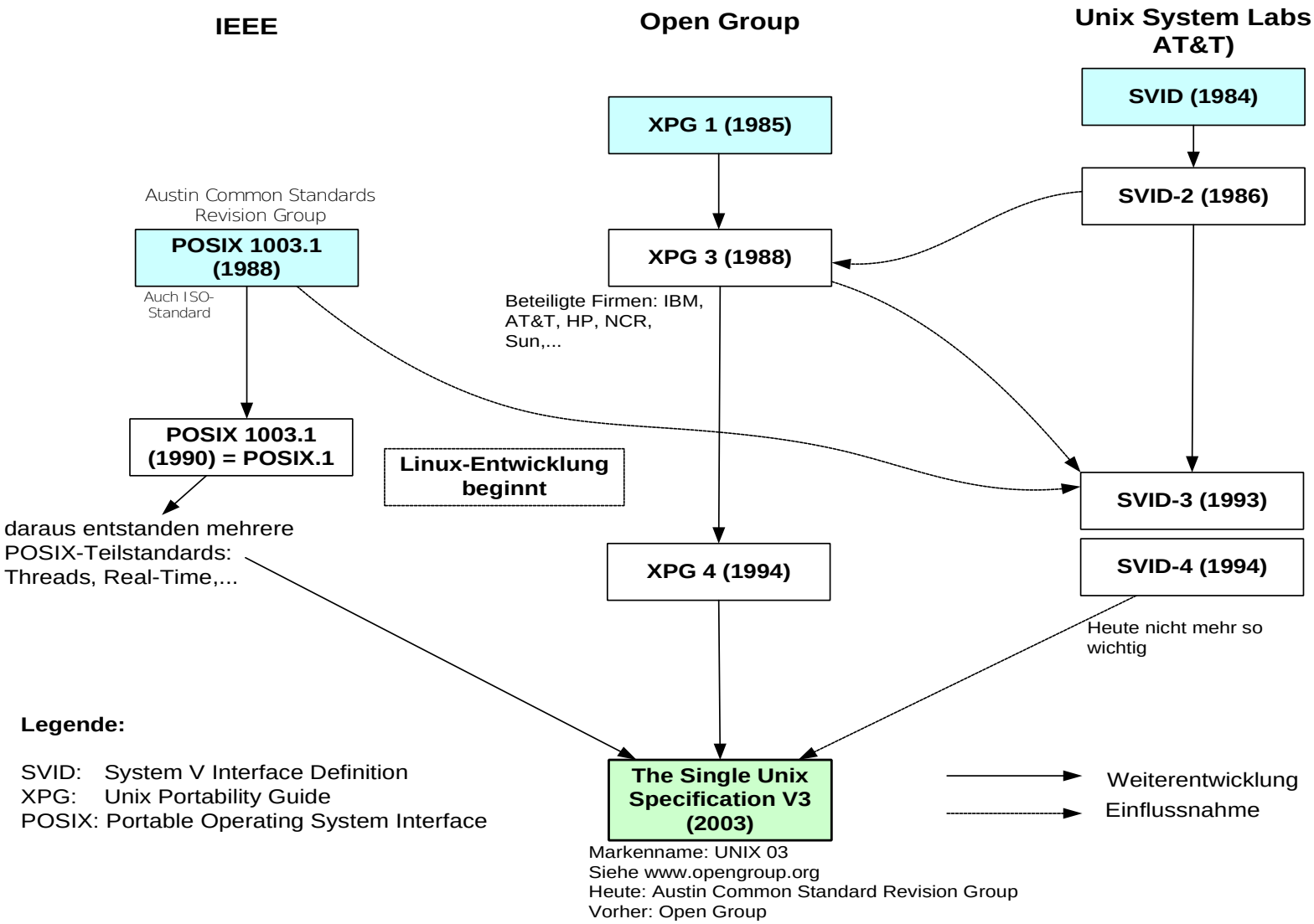
Quelle: Wikipedia

- 1989 GNU General Public License - GPL
  - Nutzer bekommt gleiche Rechte wie Autor
  - darf diese nicht einschränken
- Internet 1990 basiert auf Open Source
  - Linux, Bind, Apache, NTPd, WuFTP, msql, Perl, etc.

Kurze Geschichte von Open Source (4)

- 1991 Linux
- 1999 SourceForge
- 2005 Git
- 2008 GitHub

Unix-Normierung, historische Entwicklung

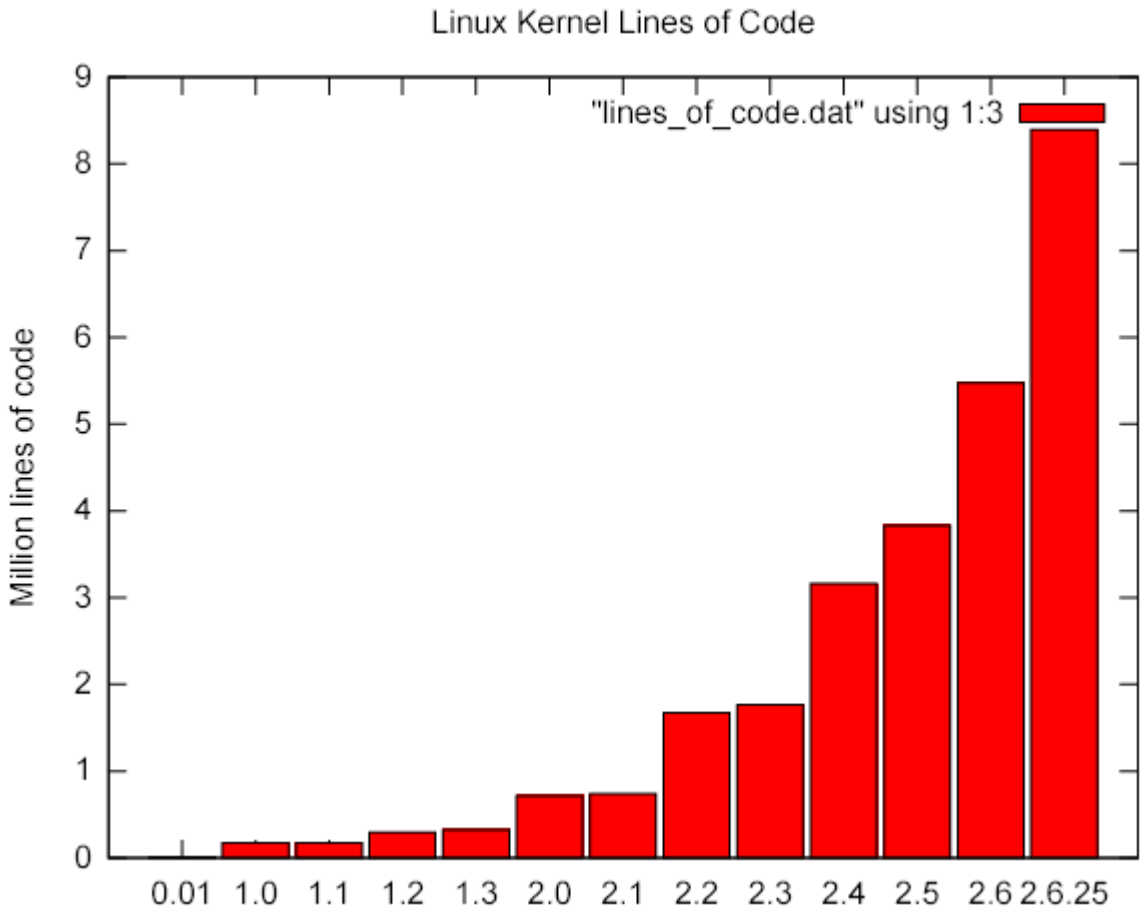


Codeumfang einzelner Betriebssysteme

Jahr	AT&T	BSD	Minix	Linux	Solaris	Win NT
1976	V6, 9K					
1979	V7, 21K					
1980		4.1, 38 K				
1982	Sys III, 58 K	4.2, 98 K				
1984		4.3, 179 K				
1987	SVR3, 92 K		1.0 13 K			
1989	SVR4, 280 K					
1991				0.01, 10 K		
1993		Free 1.0, 235 K				3.1, 6 M
1994		4.4 Lite, 743 K		1.0, 165 K	5.3, 850 K	3.5, 10 M
1996				2.0, 470 K		4.0, 16 M
1997			2.0, 62 K		5.6, 1.4 M	
1999				2.2, 1 M		
2000		Free 4.0, 1.4 M			5.8, 2.0 M	2000, 29 M
2007						Vista, 50 M
2009						Win 7, 70 M
2013				3.10, 15M		

Vgl. auch Tanenbaum, 2002
Lines Of Code (LOC), K = 1'000, M = 1'000'000

Codeumfang Linux-Kernel



Stand: Feb 2016:
Linux-Version 4.4.1:
25 Mio LOC

Stand: Januar 2014,
Linux-Version 3.13:
18 Mio LOC

Stand: März 2013,
Linux-Version 3.8:
16 Mio LOC

Stand: März 2012,
Linux-Version 3.2:
15 Mio LOC

Vergleich: Linux-Version 2.6.26
9 Mio LOC

Quellen: http://de.wikipedia.org/wiki/Linux_(Kernel),
http://kernel.org
http://openhub.net

Überblick

- ✓ Einführung in Computersysteme
- ✓ Entwicklung von Betriebssystemen
3. Architekturansätze
4. Interruptverarbeitung in Betriebssystemen
5. Prozesse und Threads
6. CPU-Scheduling
7. Synchronisation und Kommunikation
8. Speicherverwaltung
9. Geräte- und Dateiverwaltung
10. Betriebssystemvirtualisierung