Real Time Systems

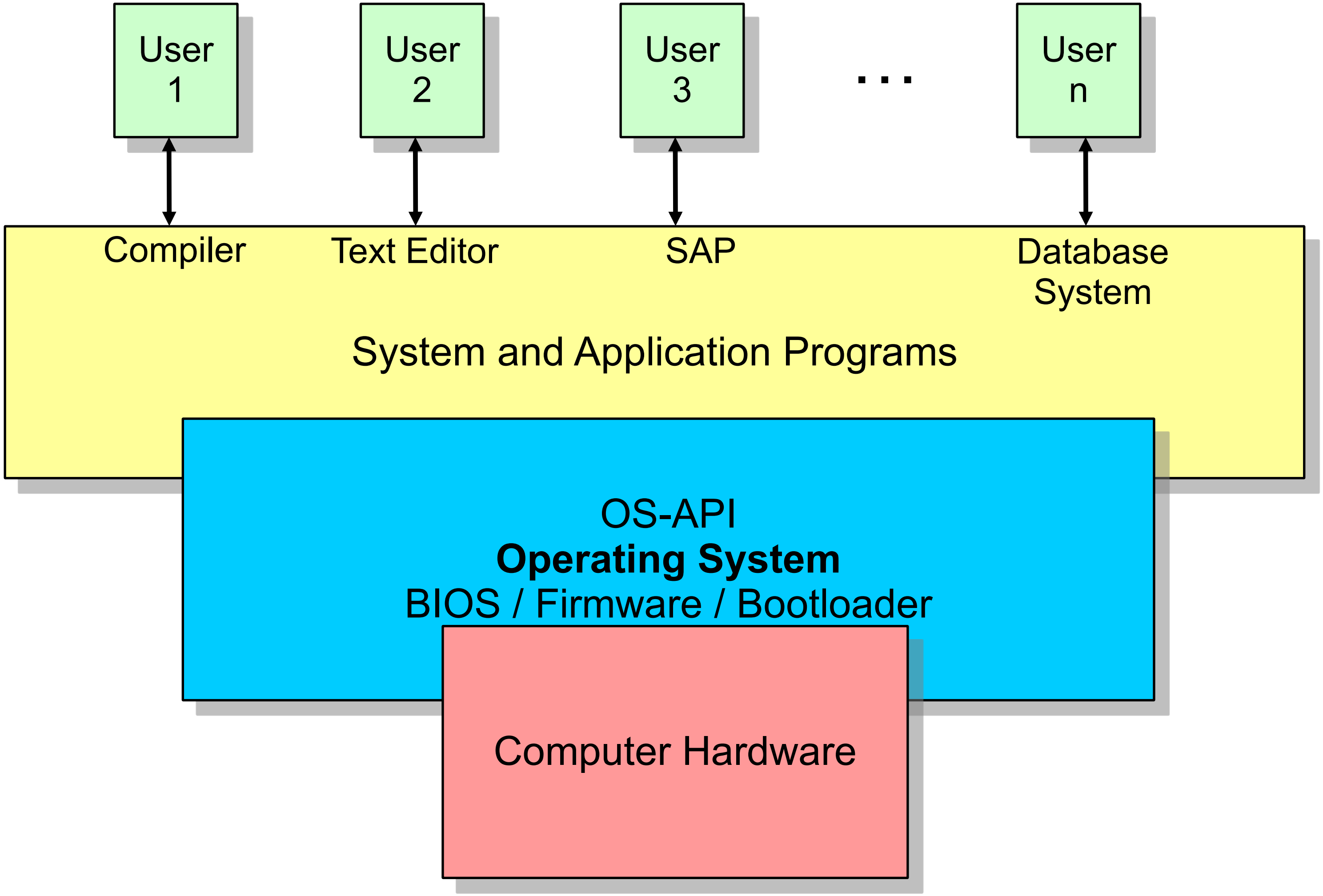
SumSem2024

Prof. Dr. Peter Tawdross
Prof. Dr. Karsten Weronek
Faculty 2
Computer Science and Engineering

Real-Time-Operating Systems (RTOS)

(P)cap: “Computer Architecture: Operating Systems)

The journey between User and Computer-Hardware through different SW-Levels



Quelle: Holten, König, 2004 c/o Silberschatz, Galvin: Operating System Concepts, Fifth Edition, 1999

Main Duties of the Operating System

Interface between hardware and user-programs

Interface between hardware and partner systems

Internal Ressource Management:

- Process-/Processor-Management (mainly CPU)
- Memory-Management (mainly RAM)
- Management (Device- and Filemanagement)

There are much more other duties for an operating system (OS). For those please refer to the lecture "Operating Systems". We focus on real-time relevant topics of OSs.

What about Timekeeping (Zeitverwaltung)?

Timekeeping plays an important role.

Timekeeping is not limited to the OS.

It comprises:

- having a clock:
  - Time Synchronization (e.g. with DCF77 or time-servers)
  - relative time (using tics after systemstart)
  - time measurement (Zeitmessung, Stoppuhr (stopwatch))
- time control for services (Zeitsteuerung)
- Time Monitoring (Zeitüberwachung)

Watchdog (Laufzeit-Überwachung, abs. Wecker)

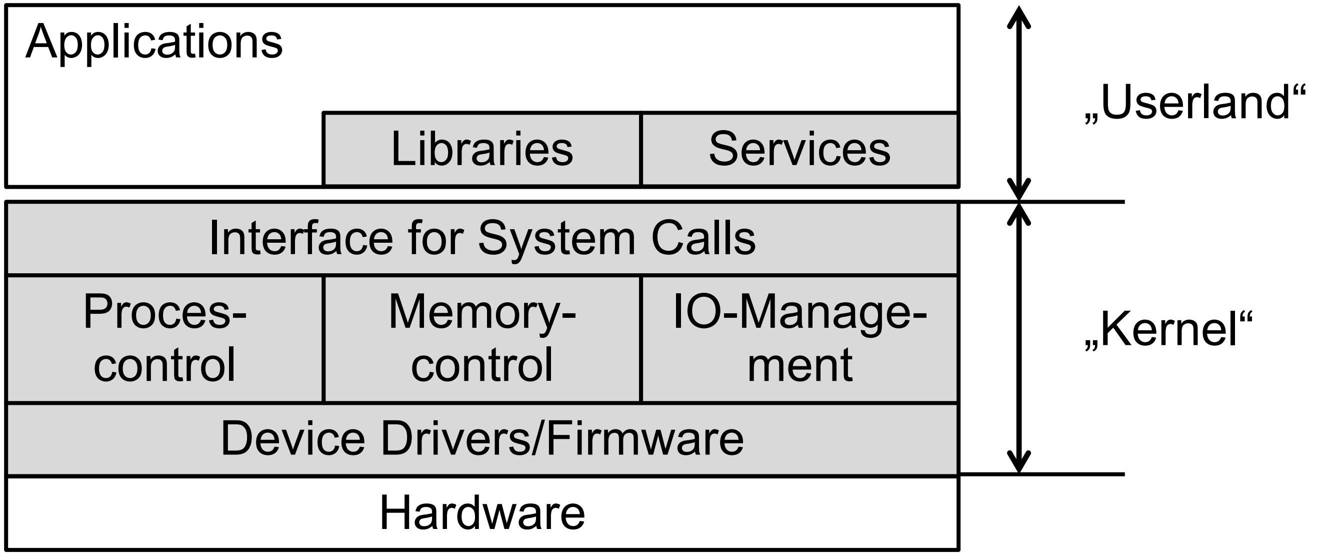
Timer (often hardware, relative time)

(e.g.: egg-/sleep-timer (vgl. Eieruhr, Treppenhauslicht))

System Architecture with Kernel-OS

Requirements for OSs may differ. OSs may differ in architecture.
The understanding what belongs to an OS and what doesn't is different.

For our purposes we us the following architecture model for a OS:

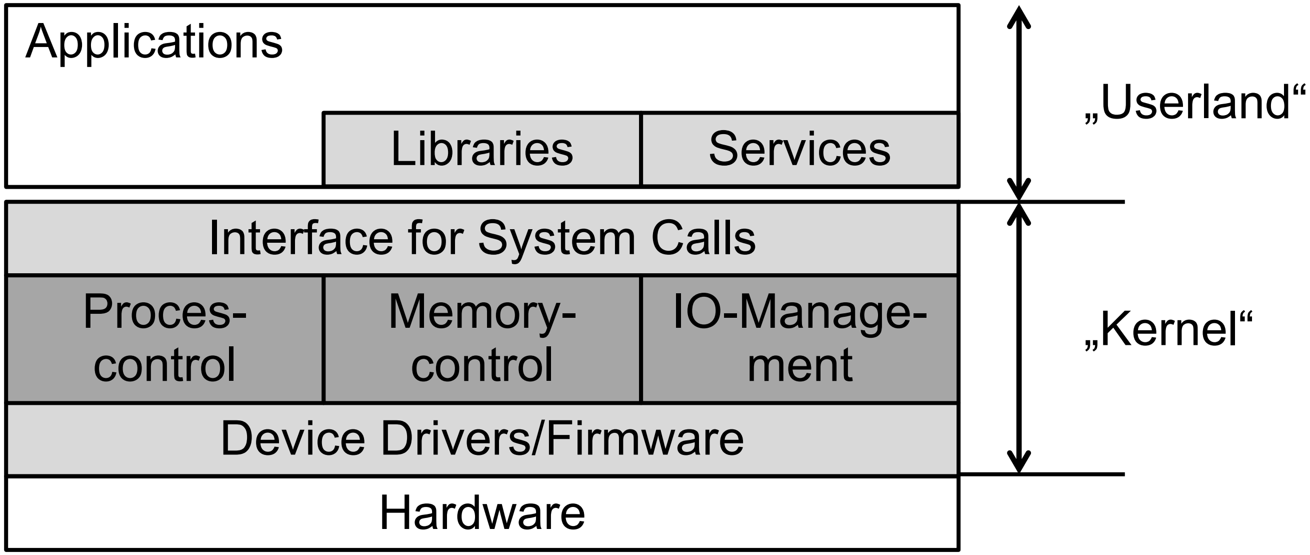


nach J. Quade, M. Mächtel, Moderne Relzeitsysteme kompakt, Eine Einführung mit Embedded LINUX, dpunkt.verlag, Heidelberg 2012

System Architecture with Kernel-OS

Resource management:

- Process-/Processor-Management
- Memory-Management
- IO-Management (Device- and Filemanagement)



Applications

Libraries | Services

„Userland“

Interface for System Calls

Proces-control | Memory-control | IO-Manage-ment

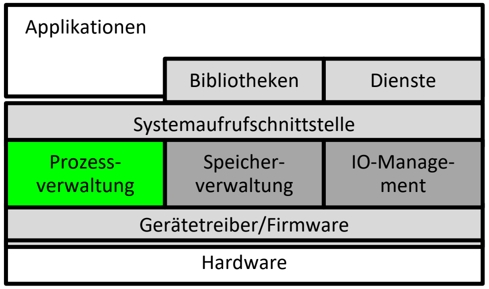
Device Drivers/Firmware

„Kernel“

Hardware

nach J. Quade, M. Mächtel, Moderne Relzeitsysteme kompakt, Eine Einführung mit Embedded LINUX, dpunkt.verlag, Heidelberg 2012

Process Control


Applikationen

Bibliotheken | Dienste

Systemaufrufschnittstelle

Prozess-verwaltung | Speicher-verwaltung | IO-Manage-ment

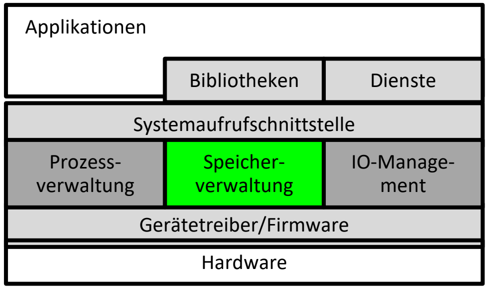
Gerätetreiber/Firmware

Hardware

- (quasi-)parallel information processing of multi computing tasks on a single- or multi-core-processors
- Tasks: Threads and Processes (have their own data segments)
- preemptive multitasking by:
  - static/dynamic scheduling („plan & control“)
  - with static and/or dynamic priorities
- Scheduling Strategies (e.g.):
  - Monocore: FCFS/FIFO; prior. time slices; EDF (earliest deadline first)
  - Multicore: partitioned / global scheduling / or in combination
- State of Tasks (in TCB=Task Control Block: Code-/Data/Stack-Segment)
  - running (also active) (maximal 1 on single-core): actual in processing
  - ready (**enabled**): will be the next one (lauffähig)
  - blocked (also waiting): waits for others (schlafend)
  - suspended: task is not active anymore, usually suspended by calling a task suspend API function and resumed by calling resume function (ruhend)

See: http://www.freertos.org/RTOS-task-states.html

Memory-Management



Applikationen

Bibliotheken

Dienste

Systemaufrufschnittstelle

Prozess-verwaltung

Speicher-verwaltung

IO-Manage-ment

Gerätetreiber/Firmware

Hardware

- Tasks for the Memory-Management-Unit (MMU):
  - Memory Protection

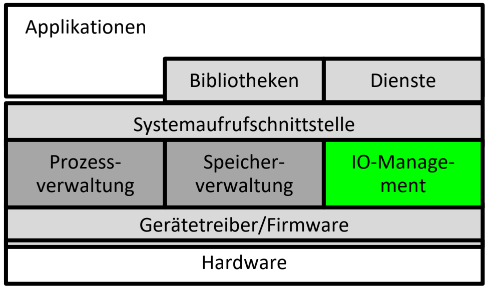
    Applications and processes (not threads) have its own address-space, that is protected against mutual access.
  - Address Translation

    (formerly done by loaders), today the MMU makes sure, that Code-Segments start with memory address 0 : code-share, less memory, faster.
  - Provisioning of extended Memory

    when more memory is required than it is accessible by the bus
  - Provisioning of virtual Memory

    provides more memory than available physical memory by paging und swapping. Often this is not reasonable for critical areas of RT-applications because paging and swapping is not time-deterministic.

IO-Management



- Provision of
  standardized Application Programming Interface (API)
  (create, open, read, write, close)
- Enable a system-conform integration of hardware (driver interface)
- Enable structured storing of information in data und directories (filesystem)

Requirements for a RTOS (2/2)

Requirements with direct time dependence

- Time services
  - absolute and relative Clocks, Timer, Timeouts
- defined reaction times
- RT-compliant scheduling
- Synchronization of processes (Semaphores, Mutexes, etc.)
- RT-compliant process communication (IPC)

Non-functional requirements:

- Availability (7x24)
- Scalability
- minimal memory requirements (Microkernel)

RT-OS (Selection)

- Free-RTOS

  FreeRTOS is a class of RTOS that is designed to be small enough to run on a microcontroller - although its use is not limited to microcontroller applications.

- PREEMPT-RT-Patch for Linux

  adds RT-features that are not migrated yet into the kernel

- LynxOS:

  kommercial RTOS, POSIX-conform for army, aerospace, medical applications etc.

- VxWorks (Wind River Systems)

  prop. OS u.a. for small devices aerospace, defense, health, networks, (Mars-mission)

- QNX (Neutrino)

  kommercial RTOS, POSIX-conform, Open-Source, free not for commercial use

- RTEMS (Open Source)

  for display-less, RT-embedded systems, (US-army)

- eCos:

  Free RTOS, for large number of different processors

- and many more .....

Standards/Organisations

- POSIX
  „Portabel Operating System Interface“
  ist ein von IEEE und der Open Group für UNIX entwickeltes und durch ANSI und ISO standardisiertes Application Programming Interface (API)
- OSEK / VDX
  „**O**ffene Systeme und deren Schnittstellen für die Elektronik im Kraftfahrzeug“
  Gemeinschaftsprojekt der deutschen Automobilindustrie (Hersteller und Zulieferer)
  “Vehicle Distributed eXecutive“, französischer Teil (PSA, Renault)
  Standard für OS, COM, NM, OIL
- AUTOSAR(-OS)
  AUTomotive Open System ARchitecture
  Betriebssystem-Standard für Controller in der Automobilindustrie
- Gesellschaft für Informatik
  Fachgruppe Echtzeitsysteme (real-time), http://www.real-time.de
  siehe auch Peter Holleczek, Birgit Vogel-Heuser (HRSG.), Mobilität und Echtzeit
  ISBN 978-3-540-74836-6, Springer Berlin Heidelberg, 2007

A gallery of disruptive technologies
Estimated potential economic impact of technologies across sized applications in 2025, $ trillion, annual
0
1
2
3
4
5
6
7
8
9
10
11
1. Mobile Internet
2. Automation of knowledge work
3. Internet of Things
4. Cloud
5. Advanced robotics
6. Autonomous and near-autonomous vehicles
7. Next-generation genomics
8. Energy storage
9. 3-D printing
10. Advanced materials
11. Advanced oil and gas exploration and recovery
12. Renewable energy
SOURCE: McKinsey Global Institute