



Embedded Linux- Robocup

Henning Heinold

(heinold@inf.fu-berlin.de)

Holger Freyther

(freyther@inf.fu-berlin.de)

Übersicht

- Einführung GNU/Linux
 - Geschichte
 - Aufbau und Userspace
 - Hardwarebeispiele
- Portierung auf Linux?
 - Hardware-, Softwarestatus
 - Was ist zu tun?
 - Probleme
- Fazit

Geschichte I

- Linus Torvalds Student Finnland 1991
- unzufrieden mit DOS, UNIX an der Uni
- Disput mit Tanenbaum
- schnell fanden sich mehr Leute, auch wegen der freien Lizenz

Geschichte II

- Linux 1.0
- Portierung auf andere Architekturen
- immer mehr Fähigkeiten, z.B. Realtime
- Firmen werden aufmerksam
- aktuell 2.6 Kernel

Aufbau

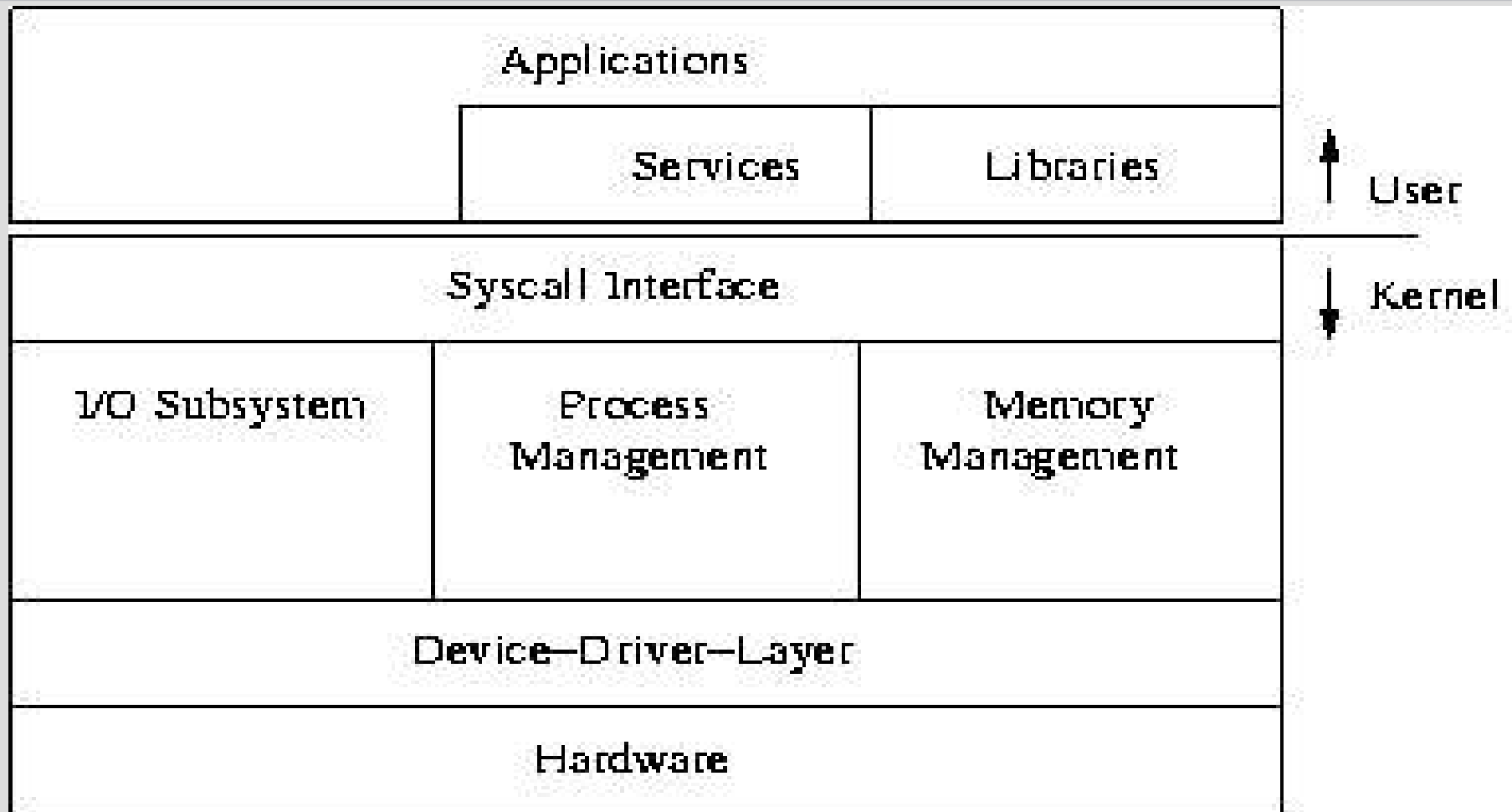


Bild von <http://ezs.kr.fh-niederrhein.de/lectures/treiber/html/c38.html#AEN61>

Userspace

- Compiler, glibc
- Debugger
- viele Bibliotheken
- Qt Embedded

Distributionen

- Zusammenfassung von Linux und den Userspaceprogrammen
- Vorteile: Anwender kommt besser zurecht
- Debian, SuSE, Redhat meist nur I386
- für Embedded Systeme keine Distribution in dem Sinne, dafür aber flexibel

Hardware Beispiele

- Omap2 -Texas Instrument - ARM
 - ARM-Prozessor mit 330 MHz und DSP
- Alchemy - AMD - MIPS
 - Verfügbar, sehr schnell
- Eden – Via - X86
 - preiswert, verschieden Ausführungen

Hardwarestatus

- Platine für die Steuerung von Motoren/Schuss
- Firewire CCD Kamera
- USB-Serial Verbindung zur Platine
- 5Ghz Funknetzwerk

Softwarestatus

- Scryer (Wahrsager)- SharedMemory
- RoboClient in Verbindung zum Server – MFC
 - Vision mit 30 frames pro Sekunde
 - 320x240x16x30 ~ 4MByte/s
- Verhalten mit seinen Ebenen – Qt
 - Umfangreiche Benutzung von Fließkommazahlen

Probleme Software und Hardware

- Hardware
 - Gewicht, Steckverbindungen zur Kamera und Platine
 - Montierung des Laptops und dessen Kühlung
 - Festplatte nicht für Roboter entworfen
- Software
 - XP hat Probleme mit TCP und Funk-Netzwerk
 - Es laufen zu viele Prozesse – Rechenzeit verschenkt
 - Zu wenig Einfluss
 - Eigene Komponenten aktuell halten

Hoffnung Hardware

- spezialisiere Hardware
 - ist kleiner, senkt das Gewicht
 - besser zu montieren
 - besser zu kontrollieren
- Außer der IA-32 Plattform existieren z.B.
 - MIPS
 - ARM

Anforderung an Hardware

- Bildverarbeitung fordert grossen Speicherdurchsatz und Rechenleistung
- Verhalten, insbesondere Pfadplanung braucht ebenfalls Rechenleistung
- Benutzung von float erfordert umdenken
- Ein ARM bzw. MIPS Board kann Anforderung nicht alleine erfüllen

Hardwareplattformen

- X86
 - VIA Eden
 - komplette Eigenentwicklung
- ARM
 - XScale (+ MMX)
 - XScale + IOP331 I/O + 80200
 - OMAP2 + DSP
 - spezieller Grafik-Chip
- MIPS
 - AMD Alchemy

VIA Eden© x86 MiniATX Board

- Low Voltage Prozessor mit 1Ghz
- <http://www.via.com.tw/en/Products/eden.jsp>
- Vorteile
 - Mainboard erfüllt komplett alle Kriterien für Anschlüsse
 - Preis
- Nachteile
 - Stromversorgung noch offen
 - FPU bei halber Taktfrequenz

Intel® XScale®

- Intels ARMv5 Implementierung
- Bis zu 800Mhz Takt
- Vorteile
 - MMX und Flash-Speicher auf dem Core integriert
 - Anschluss von Firewire und WLAN über PCMCIA möglich
 - Intel Performance Primitives
- Nachteile
 - Bildverarbeitung nicht ohne weiteres möglich

Intel® IOP331 I/O Prozessor

- <http://www.intel.com/design/iio/iop331.htm>
- Spezieller XScale basierter IO Prozessor
- Vorteile
 - Anschluss von Firewire über PCI-X Bus möglich
 - Speziell für IO entworfen
- Nachteile
 - Für Bildverarbeitung immer noch eine andere Lösung nötig
 - Verfügbarkeit und Preis

Texas Instruments OMAP2

- ARM Implementierung von TI mit integriertem DSP mit 330 MHz
- <http://www.ti.com/omap2>
- Vorteile
 - Anschluss von Kamera an den DSP möglich
 - Andere Peripherie implementierbar
- Nachteile
 - Verfügbarkeit
- Frage
 - Reicht der DSP aus
 - Reicht das ARM-Core für das Verhalten aus

AMD® Alchemy

- MIPS von AMD mit bis zu 500Mhz
- Vorteile
 - ausgereifte Technik, Verfügbarkeit
- Nachteile
 - kein direkter Firewireanschluss, Anschluss ggf. teuer

Nvidia GOForce 4000

- http://www.nvidia.com/page/goforce_4000.html
- Vorteile
 - YUV422 nach RGB Konvertierung + mehr
 - JPEG Kodierung on the fly
- Nachteile
 - Verfügbarkeit, Preis
- Fragen
 - Kann im Zusammenhang mit einem XScale Vision betrieben werden?

Hardware Zusammenfassung

- VIA Eden sicherste Lösung
- Ein oder zwei ARM + CHIP interessanter

Anforderung Software

- Unsere Komponenten fordern
 - SharedMemory und IPC
 - TCP/IP Stack
 - faire Zuteilung von Rechenzeit
- Linux kann dies auf allen Plattformen befriedigen
- Für Plattformen ohne FPU gibt es Lösungen

Portierung zu Linux I

- Nichts wird weggeworfen
- Gibt die Möglichkeit Probleme zu beseitigen und es modularer zu machen
- Benutzung von Qt beibehalten
- Benutzung von float und double vermeiden, bzw. auf fouble und ffloat umstellen
- RoboClient von MFC befreien

Schrittweise Portierung

- RoboClient
 - RoboClient unter Windows langsam auf Qt umstellen, MFC wrapper für nicht GUI Klassen
 - dann auf Linux laufen lassen -> Board
- Verhalten
 - Trennung von Vision und Verhalten
 - Aktor, Sensoren und Ebenen konsolidieren
 - dann auf Linux compilieren und ggf anpassen

Ist die Zeit reif?

- Hardware können wir langsam auf ein Embedded Board umstellen
- Linux und Entwickler-Werkzeuge stellen eine attraktive plattformunabhängige Basis da
- Wir haben die Chance Probleme mit der Software zu lösen
- Benötigte Werkzeuge legal benutzbar

Fragen und Spenden?
-Danke-