

Vortrag:

# Systemsimulation:

# Beispiel Palm OS Emulator

# Motivation

---

- Verstehen der Palm OS - Architektur
- Starthilfe, um in die Palm OS - Programmierung einzusteigen
- Grundlagen der Palm OS - Programmierung

# Gliederung

---

- 1) Hardware, Architektur
- 2) Vorbereitung zum Programmieren
- 3) GUI - Programmierung
- 4) Weiteres zur Programmierung

- **Motorola 68K-Prozessor der Dragonball Serie**  
(Aktuell ist der MC68VZ328)



- LCD-Controller (bis 640x480 res.)
- Power Management Funktionen
- 32bit interner Adressbus
- Interface für alle Arten von Speicher (SRAM, DRAM, SDRAM, EPROM, FLASH)
- Support für Sound
- Anschlüsse für Peripherie
- Umfangreiche Debuggingoptionen

# Der Palm kurz vorgestellt

---

- 160x160 Display
- Bis 8MB RAM intern
- Graffiti® Technologie
- HotSync® Technologie
- USB Support
- Infrarot Schnittstelle
- Slot für Erweiterungskarten  
(Handspring VISOR, Palm M50x)



# Der Palm kurz vorgestellt

	<b>Palm 1000</b>	<b>Palm 5000</b>	<b>PalmPilot Personal</b>	<b>PalmPilot Professional</b>	<b>Palm III</b>
<b>Backlit</b>			x	x	x
<b>High-Contrast LCD</b>			x	x	x
<b>Memory</b>	128K	512K	512K	1MB	2MB
<b>Built-in Apps</b>	x	x	+ Expenses	+Expenses +Mail	+Expenses +Mail
<b>TCP/IP</b>				x	x
<b>Infared</b>					x
<b>Flash Memory</b>					x
<b>Barcode Scanner</b>					x-symbol only

# Das Betriebssystem

---

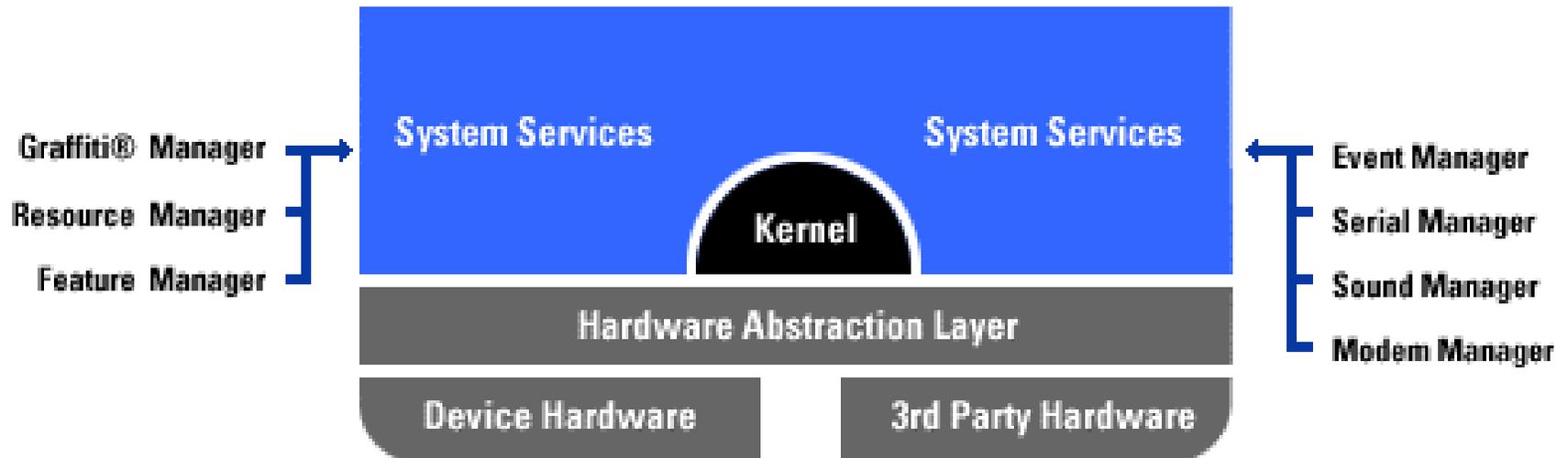
- Hardware Abstraction Layer sagt dem Betriebssystem, welche Hardware es gibt und wie sie angesprochen wird.
- Je nach Hersteller und Modell hat der Palm unterschiedliche Hardwarebausteine



# Das Betriebssystem

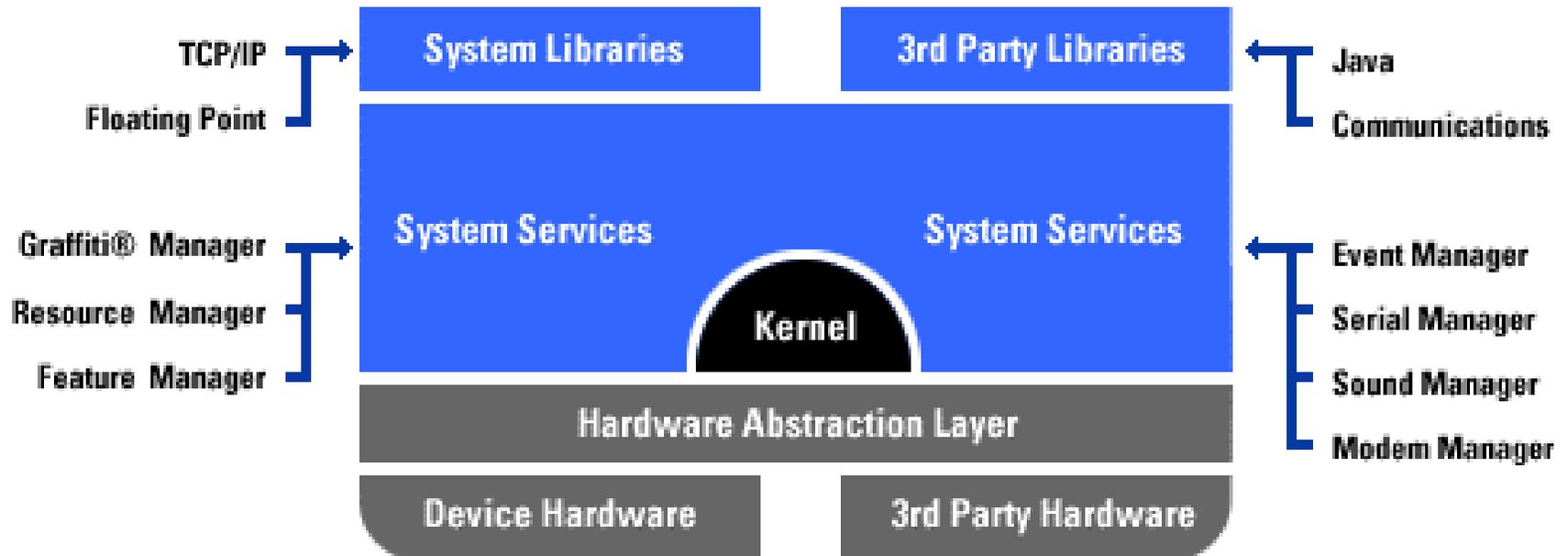
---

- Kernel ist Multitaskingfähig:
  - User Interface
  - Eingabe und Hardwareaktionen
- Systemservices sind palmspezifische OS-Dienste

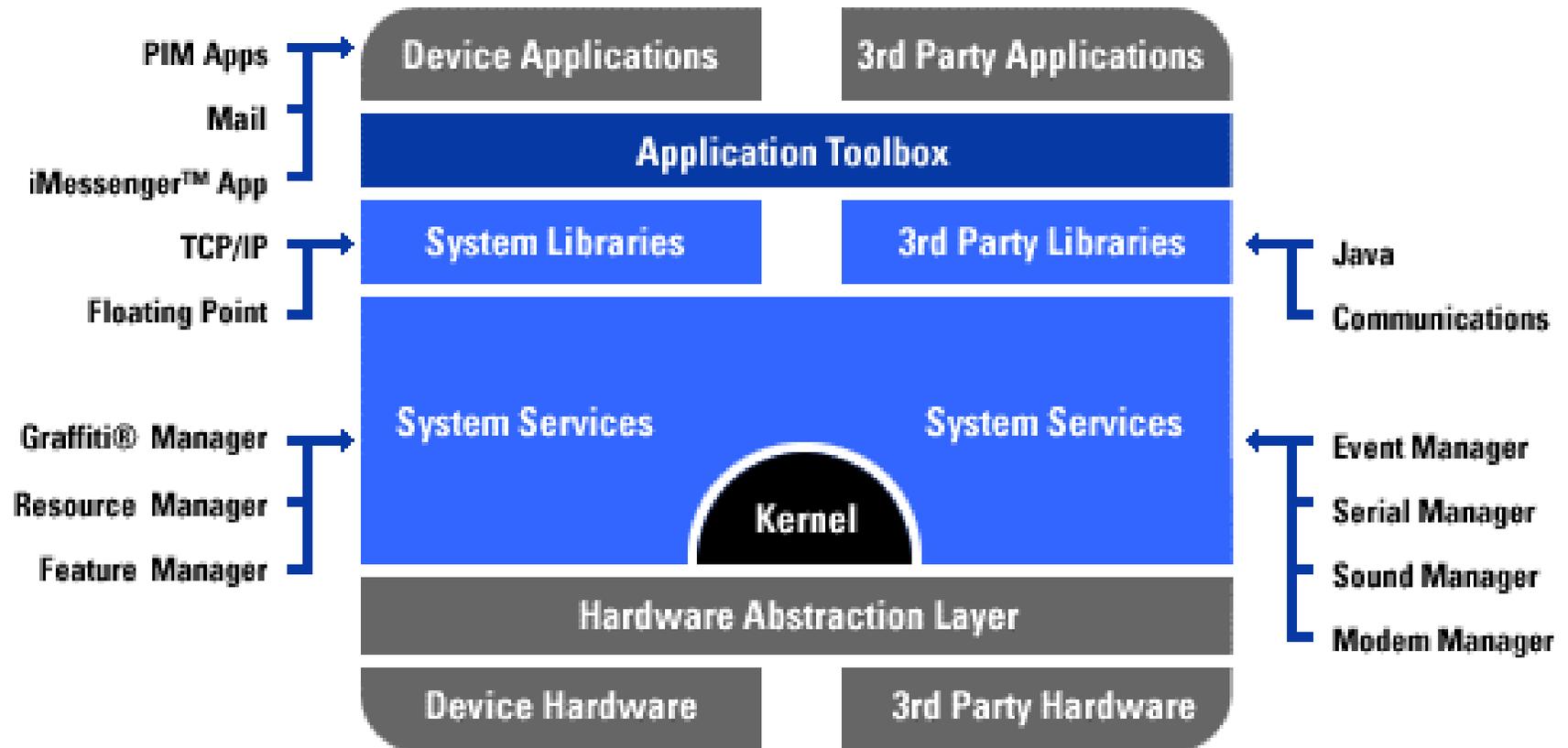


# Das Betriebssystem

- System Libraries erlauben einfachen Zugriff auf Hardware
- Ermöglichen z.B. das Laufen von Javaprogrammen



# Das Betriebssystem



# OS - System Services (Scribble Manager)

---

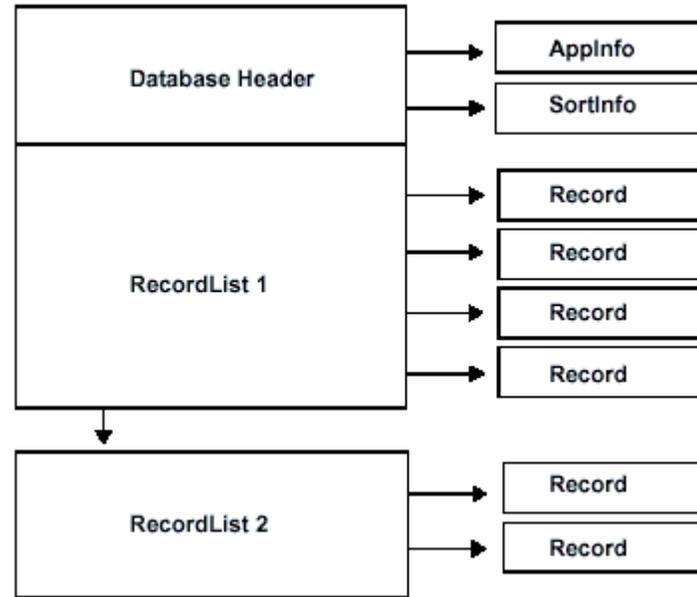
- Aus Platzgründen besitzt der Palm weder Tastatur, noch Maus
- Scribble-Stift ist die einzige Eingabemöglichkeit
- Scribble Manager ermöglicht ‚intuitives‘ Schreiben von Zeichen wie auf Papier

# OS - System Services (Resource Manager)

---

- Es gibt drei Arten von Dateiformaten auf dem Palm
  - PRC (Palm resource)
  - PDB (Palm Database)
  - PQA (Palm Query Application)
- PRCs enthalten GUI-Objekte und Programmcode  
=> Sind also Applikationen
- PDBs enthalten Datensätze (C - structs)
- Die Dateien können auf den PC überspielt werden

# OS - System Services (Resource Manager)



Die Datenstrukturen von PDAs, PRCs und PQAs  
ähneln sich, allerdings sind bei...

- ... PDAs      records = structs
- ... PRCs      records = Ressourcen
- ... PQAs      records = Seiten

# OS - System Services (Event Manager)

---

- Das Palm OS löst Events als Reaktion auf bestimmte Geschehnisse aus
- Es gibt eine Applikationseventschleife
- Es gibt drei Stufen von Events
  - System Events
  - Menu Events
  - Application Events
- SPÄTER MEHR DETAILS HIERZU...

# OS - System Services (weitere Manager)

---

- Weitere System Services sind der...
  - ...SoundManager
  - ...SerialManager (RS-232)
  - ...ModemManager
  - ...MemoryManager
  - ...ExchangeManager (PC-Communication)
  - ...etc.

# Application Toolbox des OS

---

- APIs für GUI - Programmierung
  - Forms
  - Icons
  - Buttons
  - Checkboxes
  - Scrollbars
  - Menus
  - Bitmaps
  - Labels
  - Listboxes
  - ...

# Voraussetzung zur Programmentwicklung

---

- Irgendeine Entwicklungsumgebung für Palm OS (auf PC, Mac, o. ä.)
- SDK für Palm OS
  - compiler
  - resource-creator
  - libs
- Palm OS Emulator für PC, Mac, o. ä.
- ROM eines Palms für den Emulator

# Vorstellen der Entwicklungsprogramme

---

- Emulator
- Entwicklungsumgebung ‚Code Warrior‘
- Palm OS Resource Editor

# Emulatoren

---

- Emulatoren „gaukeln“ einem Computer ein nicht vorhandenes Hardwaresystem vor. Die Hardware wird softwaremäßig simuliert
- Man kann auf Emulatoren fast wie auf dem richtigen System arbeiten
- Einschränkungen liegen natürlich bei hardware-spezifischen Vorteilen, wie der Graphikkarte

# PalmOS Emulator: POSE

---

- Der PalmOS Emulator (POSE) kann alles, was der Palm auch kann (*unter Linux gibt es den xpilot*)
- Installationsmöglichkeiten von Programmen
- Debuggingoptionen
- eigenständiges Testen von Programmen (Gremlins)
- schnelles switchen zwischen Betriebssystemversionen

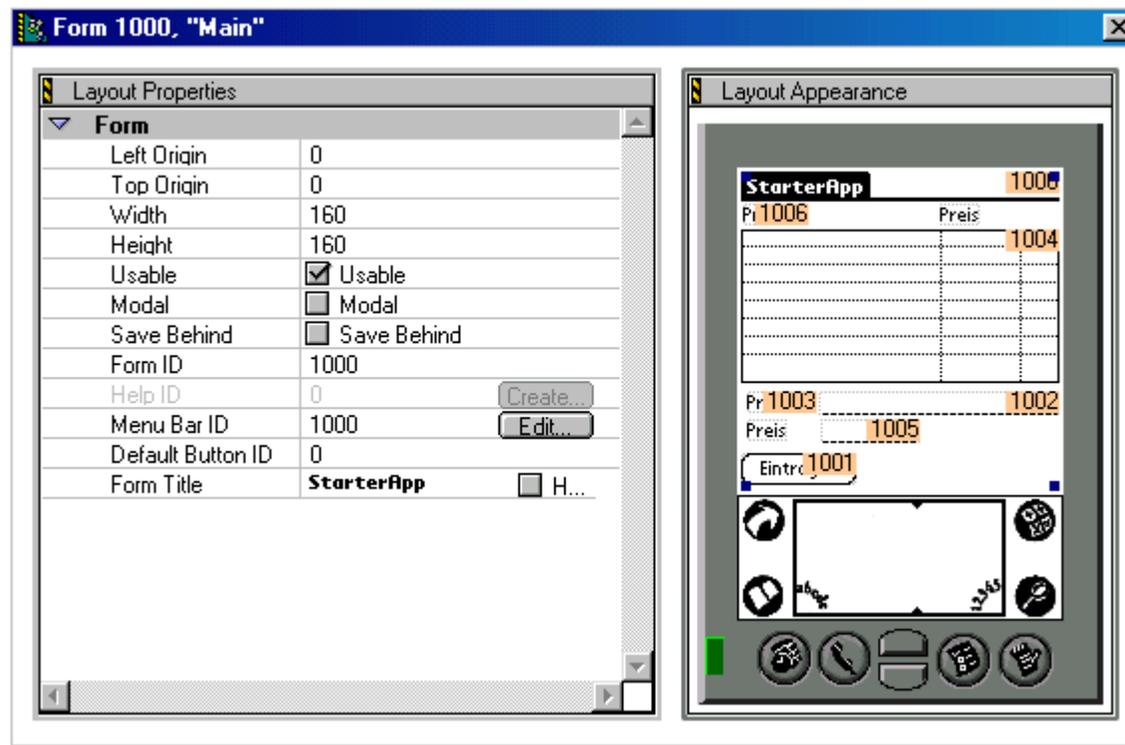
# „Code Warrior“: Ein erster Einblick

---

- Als Entwicklungsumgebung gibt es mehrere Möglichkeiten
- „Code Warrior“ und gcc sind die Marktführer bei C bzw. C++ Programmierung
- Neuerdings auch „AppForge“, ein Add-in für MS VisualStudio

# „Code Constructor“ von Palm

## Designen von Formularen und Anlegen von Ressourcen



# Wie verbindet man Ressourcen mit Code?

---

- Allen Ressourcen liegen Strukturen zugrunde
  - enthalten Informationen über Erscheinungsbild
  - enthalten Ressourcendaten(z.B. Textfeld)
  - enthalten Informationen über Zustände (z.B. Checkbox)
  - mittels Zeiger kann man solche Strukturen referenzieren
- Der Palm legt Ressourcenobjekte selbst an. Der Programmierer arbeitet nur mit Referenzen.
- Ressourcen besitzen KEINE dynamische Speicherverwaltung!!!
- Ressourcen können Events auslösen

# Die Eventstruktur

---

```
typedef struct {
    eventsEnum eType;
    Boolean penDown;
    UInt8 tapCount;
    Int16 screenX;
    Int16 screenY;
    union{
        ...
    } data;
} EventType;
```

# Beispiele für Events

---

**Controls:** ctlEnterEvent, ctlExitEvent, ctlRepeatEvent,  
ctlSelectEvent

**Fields:** fldChangedEvent, fldEnterEvent,  
fldHeightChangedEvent

**Forms:** frmCloseEvent, frmGotoEvent, frmLoadEvent,  
frmOpenEvent, frmSaveEvent, frmUpdateEvent,  
frmTitleEnterEvent, frmTitleSelectEvent

**Graffiti:** penDownEvent, penMoveEvent,  
penUpEvent

# Elementare Programmstruktur

---

- Generell ist prozedurales- und OO-Programmieren möglich (C und C++ eben)
- Elementare Prozeduren
  - PilotMain => OS-Einstiegspunkt
  - MainEventRoutine => Endlosschleife
  - WindowEventRoutine => Callbackfunktion
  - Gewöhnliche unterprogramme

## Tutorial

## Einkaufsliste

# Debugging auf dem Palm

---

- Palm unterstützt integrierte Debuggingoptionen
- Gremlins „helfen“ die Applikationen zu testen
- Den C-Code kann man sich „disassembeln“ lassen, um sich so den Assemblercode anzusehen

# Schlusswort

---

- Wir hoffen, eine kleine Einleitung in das Programmieren des Palms gegeben zu haben
- Fragen???
- Weitere Anmerkungen...

- Hardware
  - [www.Palm.com](http://www.Palm.com) / [www.PalmOS.com](http://www.PalmOS.com) / [www.3com.com](http://www.3com.com)
- Programmieren
  - [www.palmos.com/dev/tech/docs/devguide/TableOfContents.htm](http://www.palmos.com/dev/tech/docs/devguide/TableOfContents.htm)
  - [www.pdaforum.de](http://www.pdaforum.de)
  - [frankscaspage.home.att.net](http://frankscaspage.home.att.net)
- Datenblätter Dragonball
  - [www.motorola.com/SPS/WIRELESS/pda](http://www.motorola.com/SPS/WIRELESS/pda)