



## Windows CE

Process Control and Robotics

Fabian Garagnon

14.01.2009

# Agenda

3

- Geschichte & Timeline
- Echtzeit & Multithreading
- Architektur
- Memory Management & Context Switch
- Entwicklung unter CE
- Interrupts
  - Handling
  - Performance
- Windows CE 6.0
  - Architektur
  - Virtual Memory
- Quellen

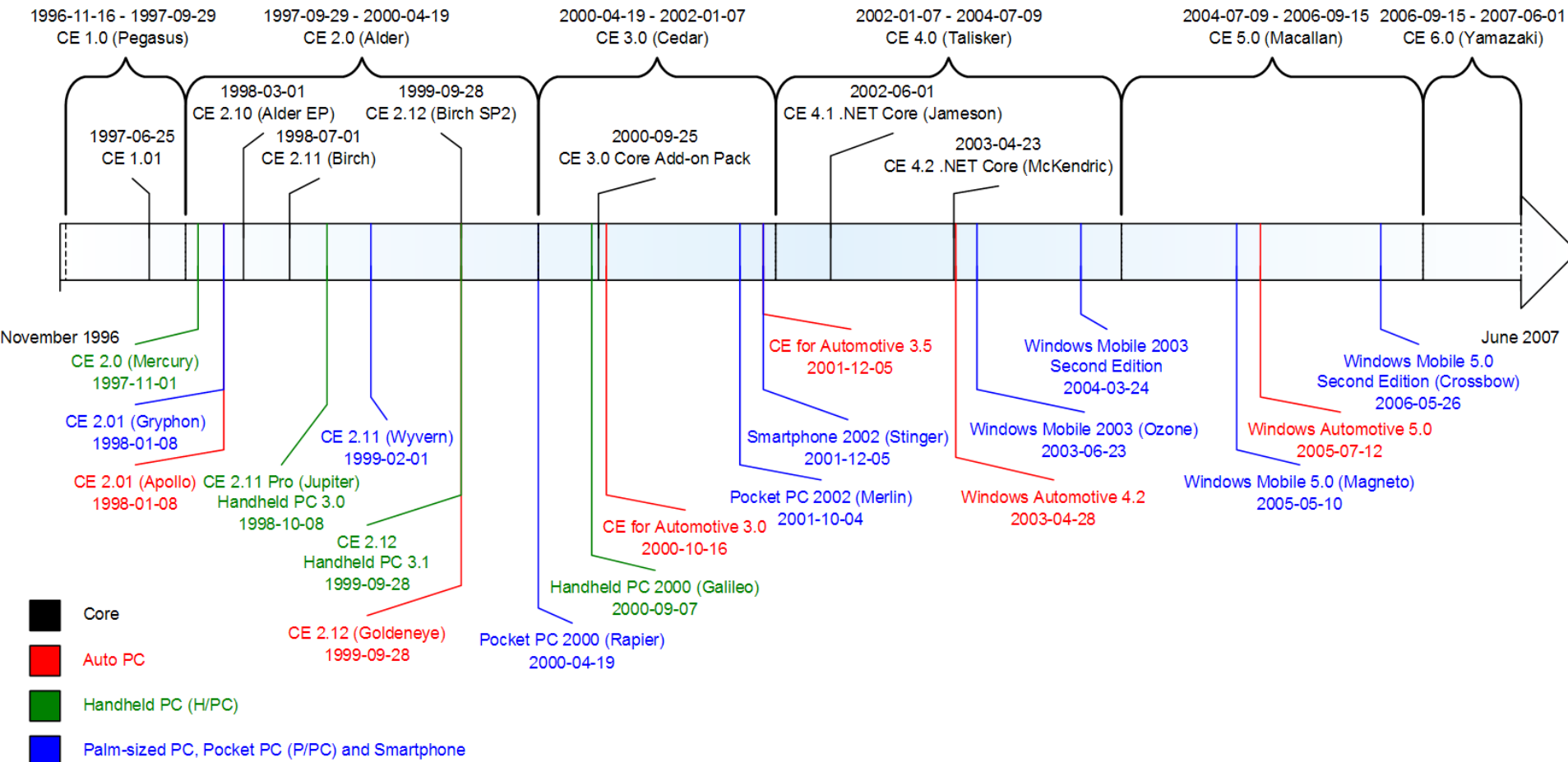
# Geschichte

4

- 1996 CE 1.0
  - kein Drucken und kein Dial-Up
- 1998 CE 2.0
  - Viele CPUs (x86, 82x PPC, SH3, SH4, StrongArm, MIPS)
  - Internationalisierung
  - ActiveSync
  - Modularer Aufbau (durch OEM)
- 2000 Pocket PC (CE 3.0)
  - Stiftbedienung
  - Multimedia
- 2002 CE .NET
  - .NET Konzept hinzugefügt
  - Shared Source

# Windows CE Timeline

Source: "A Brief History of Windows CE" (<http://www.hpcfactor.com/support/windowsce/>), HPC:Factor, retrieved May 21, 2007



# Windows CE und Echtzeit

6

- Kernel unterstützt preemptive Multithreading
- 256 Thread Prioritäten (ab CE 3.0)
- Priority Inheritance
- Mutex, Critical Section, ... (Queue pro Priorität)
- System Timers (Standard Win32 Timer API)
  - „GetTickCount“ Auflösung in ms
  - Performance Counter für höhere Auflösung
- Interrupt Latenzzeit ist vorhersagbar
  - ILTiming.exe
- Zeit für Systemaufrufe vorhersagbar
  - OSBench.exe

# Multithreading

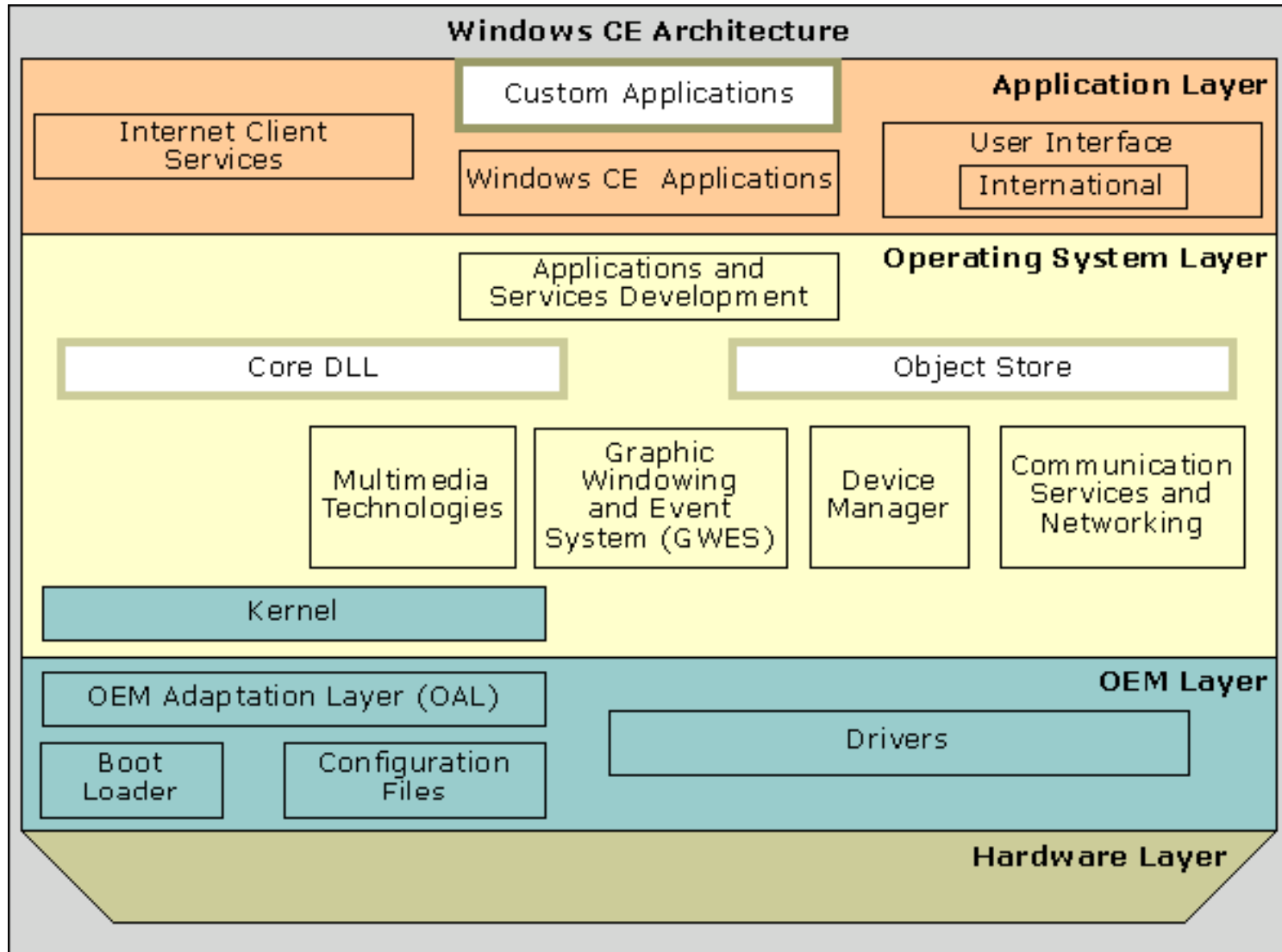
7

- 32 Prozesse
- 32MB pro Prozess
- 256 Prioritäten (ab CE 3.0, davor nur 8)
- Quantum 25ms (ab CE 3.0 einstellbar)
- Threads gleicher Priorität unterliegen Round Robin

Zur Anzeige wird der QuickTime™  
Dekompressor „  
benötigt.

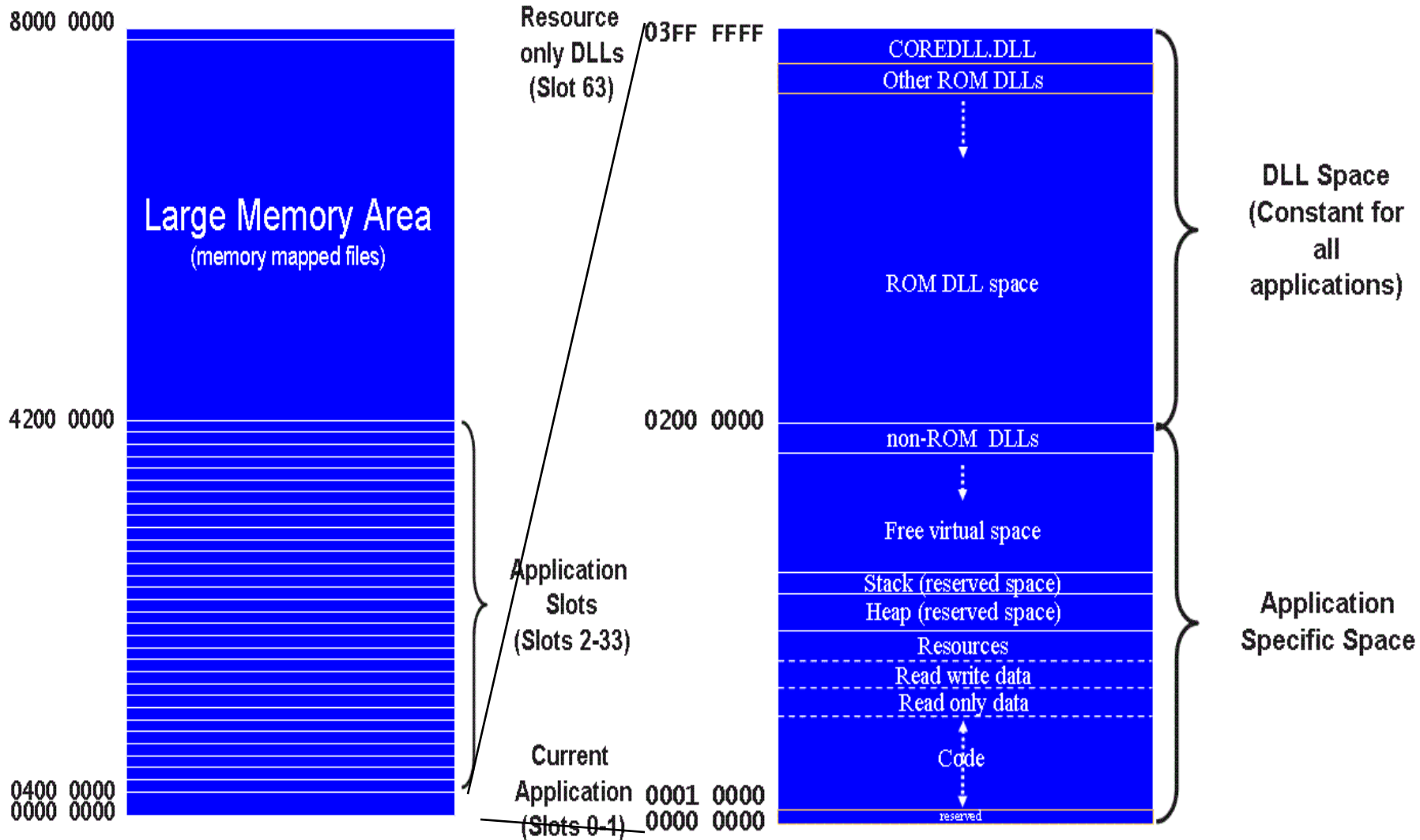
# Windows CE Architektur

8



# Virtual Memory

9





# CE Context Switch

10

Zur Anzeige wird der QuickTime™  
Dekompressor „  
benötigt.

- Kernel manipuliert Pointer beim Kontextwechsel
- Bsp.: GWE kann auf Daten der Shell ohne Kopieren zugreifen

# Speicherzugriffsschutz

11

Zur Anzeige wird der QuickTime™  
Dekompressor „  
benötigt.

- Speicherschutz durch Bitmasken

Zur Anzeige wird der QuickTime™  
Dekompressor „  
benötigt.



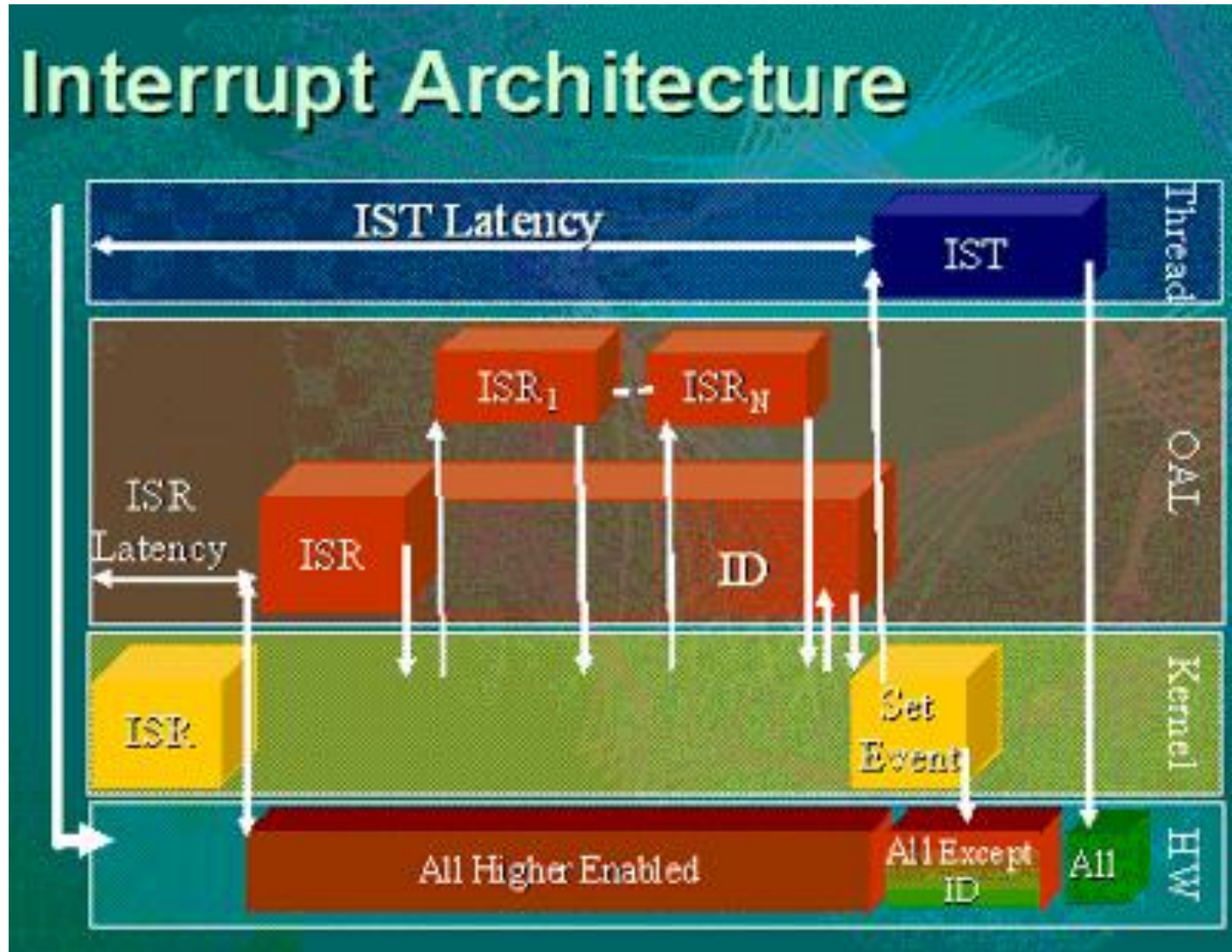
# Memory Layout und Debugging

14

- 32MB in Hex: 0x02000000
- Slot0 0x00000000 - 0x01FFFFFF,  
Slot1 0x02000000 - 0x03FFFFFF, ...
- Slot1 reserviert für Kernel
- 33 Slots - 2 (Kernel und Aktiver Prozess) = 31 Prozess Slots
- Meiste Adressen liegen in Slot0
  - Variable hat Adresse 0x00012345
  - Prozess Adressraum: 0x12000000 - 0x13FFFFFF
  - Gleiche Variable liegt auch in 0x12012345

# Interrupts

15



# Interrupt Performance

16

- Jeweils 1000 Messungen

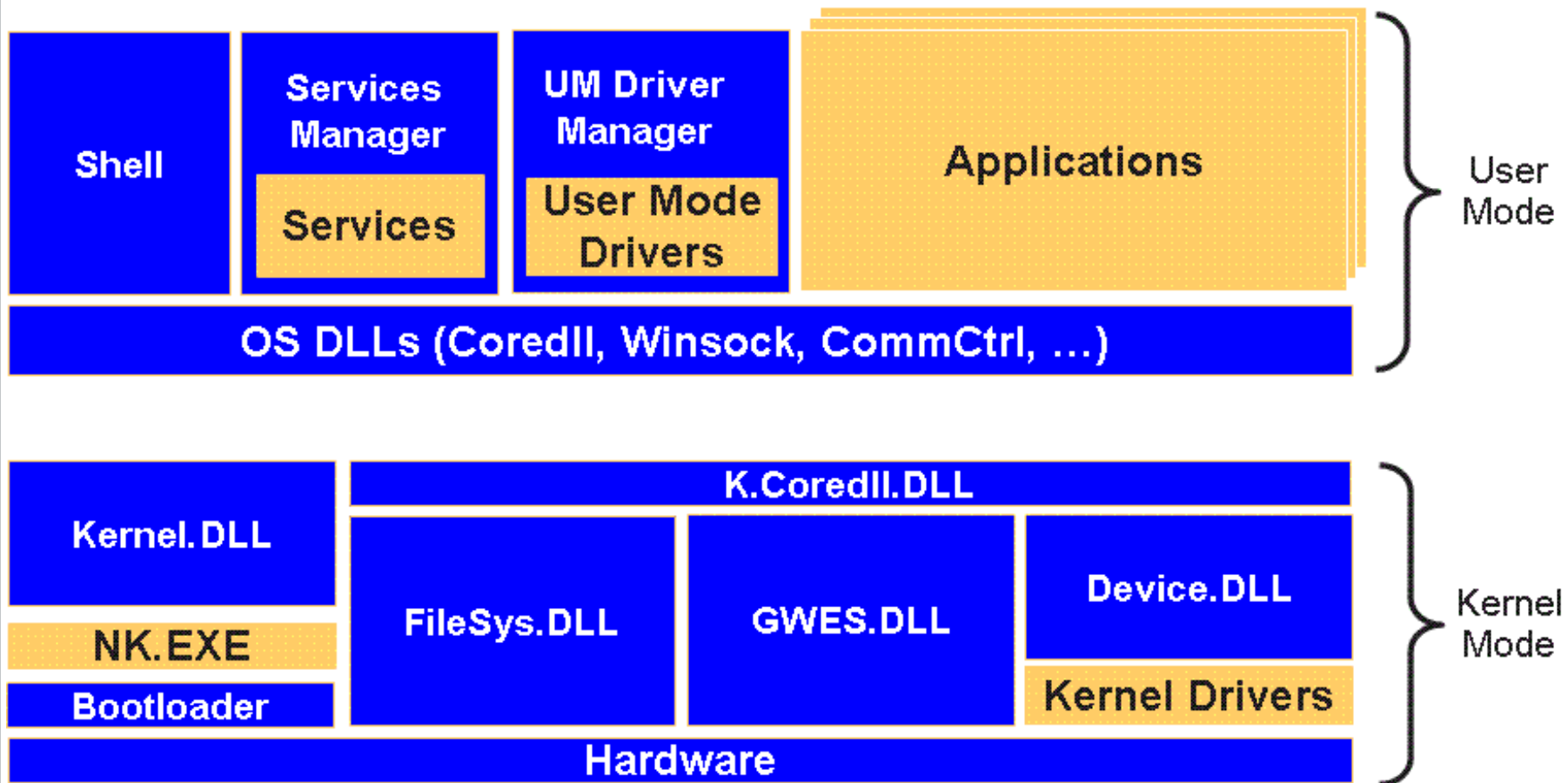
	<b>SH4</b> 198MHz Windows CE 3.0	<b>Pentium</b> 100 MHz Windows CE 3.0	<b>Pentium</b> 100 MHz Windows CE 3.0+ <b>QFE</b>
ISR Min [ $\mu$ s]	0.9	1.9	1.9
ISR Max	9.1	5.7	5.7
ISR Average	2.9	2.8	2.8
IST Min	14.8	12.9	9.6
IST Max	54.3	163.3	55.6
IST Average	38.2	28.7	26.4
Jitter Max	40	150	46

- 2GB Virtual Memory pro Prozess
- 32.000 Prozesse möglich
- Kritische Komponenten sind nun im Kernel
  - Reduziert System-Calls und User Space zu Kernel Space Aufrufe
  - Erleichtert Code Wiederverbenutzung zwischen den Komponenten
- Erhöhte Performance
- Erhöhte Sicherheit
- Abwärts kompatibel
- Treiber im User Mode, sowie im Kernel Mode möglich



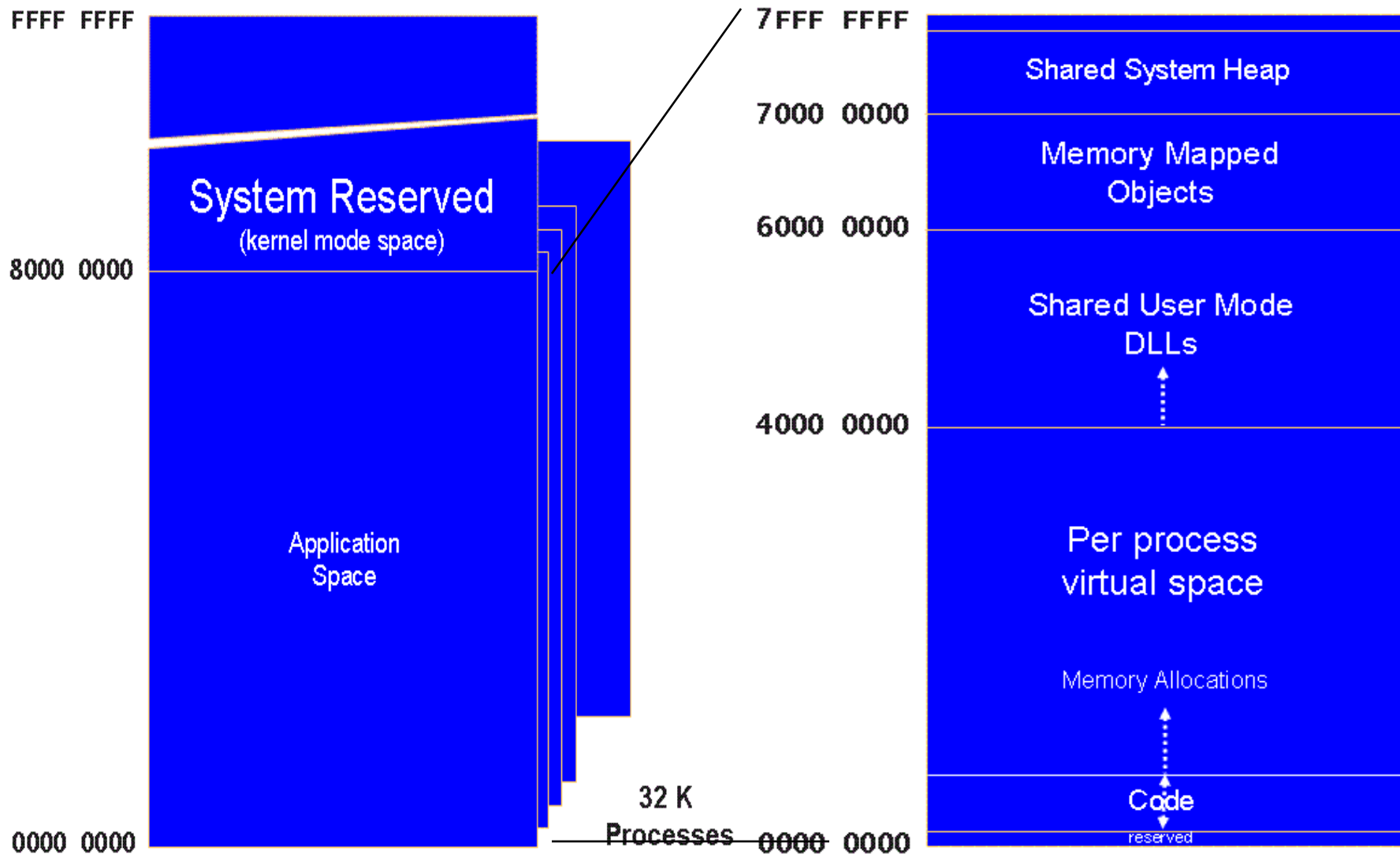
# CE 6.0 Architektur

18



# CE 6.0 Virtual Memory

19



# Quellen

20

- Embedded OS - Windows CE, Prof. Polze, Andreas Rasche
- Windows Embedded CE 6.0,  
<http://msdn.microsoft.com/en-us/library/bb331824.aspx>
- Windows CE 5.0 Architecture,  
<http://msdn.microsoft.com/en-us/library/ms905093.aspx>
- Windows CE .NET Interrupts,  
<http://msdn.microsoft.com/en-us/library/ms836807.aspx>
- Windows CE 3.0 Real-Time Capabilities,  
<http://msdn.microsoft.com/en-us/library/ms834456.aspx>
- Windows CE Memory Layout for Debugging  
<http://blogs.msdn.com/sloh/archive/2005/02/25/380475.aspx>