

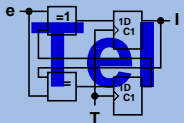
# Einsatz von eingebetteten FPGA-basierten Mikroprozessorkernen in HW-Agentennetzwerken

Vortrag zum Diplom

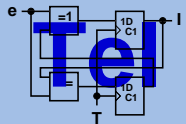
Torsten Schmutzler

s6757418@inf.tu-dresden.de

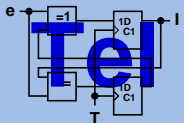
Technische Universität Dresden  
Institut für Technische Informatik



- ❖ Aufgabenstellung
- ❖ Grundlagen
- ❖ Ausgangssituation
- ❖ Konzeption
- ❖ Arbeitsstand

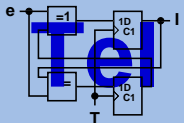


- ❖ Literaturstudium
- ❖ Auswahl einer Betriebssystemumgebung für die Virtex2Pro Architektur
- ❖ Realisierung der Steuerfunktionalität des HW-Agenten. Implementierung des Rekonfigurationsmechanismus.
- ❖ Anbindung an das Agenten-Management-System „amsys“
- ❖ Erstellung von Beispielen und Szenarien
- ❖ Auswertung der Ergebnisse

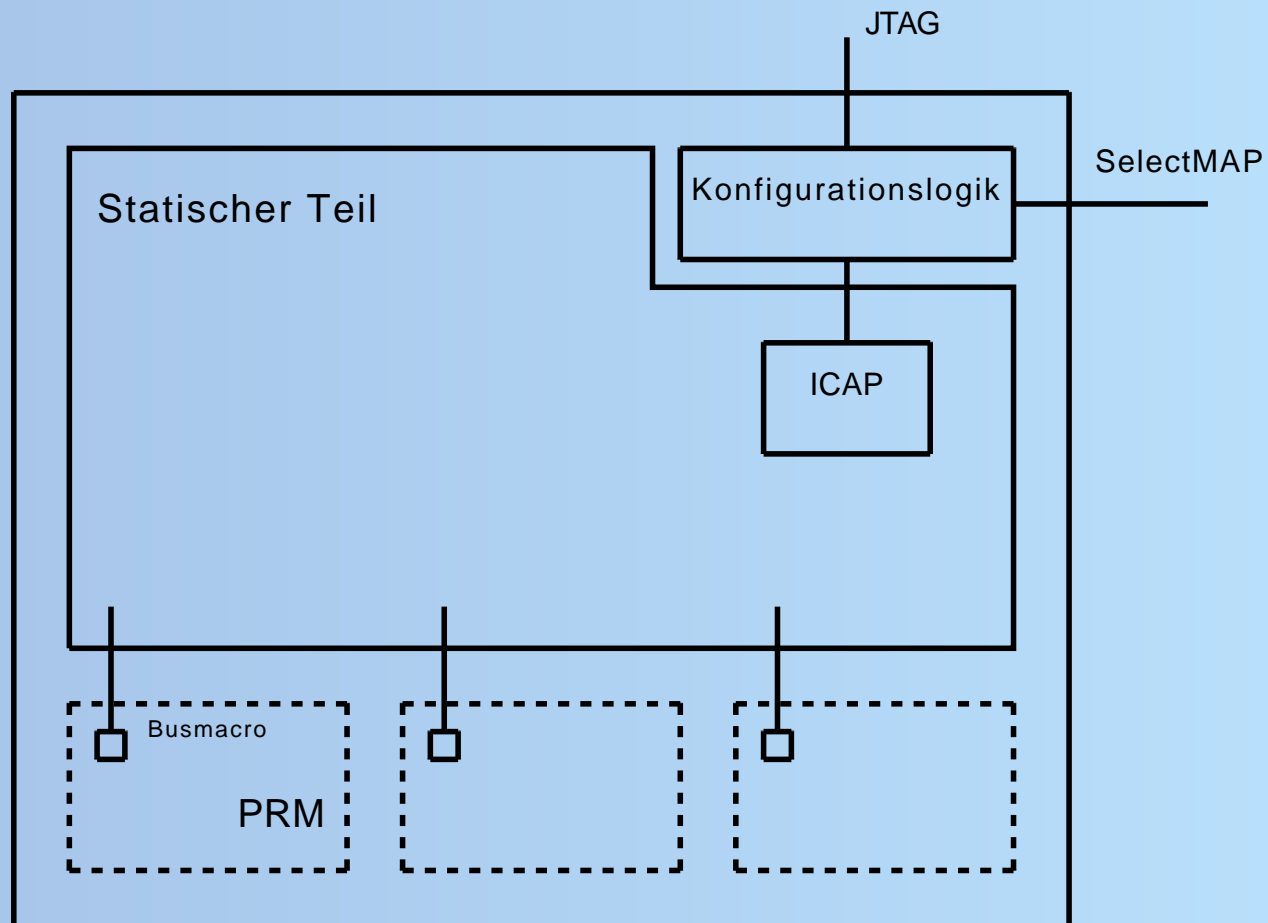


## Partielle Rekonfiguration

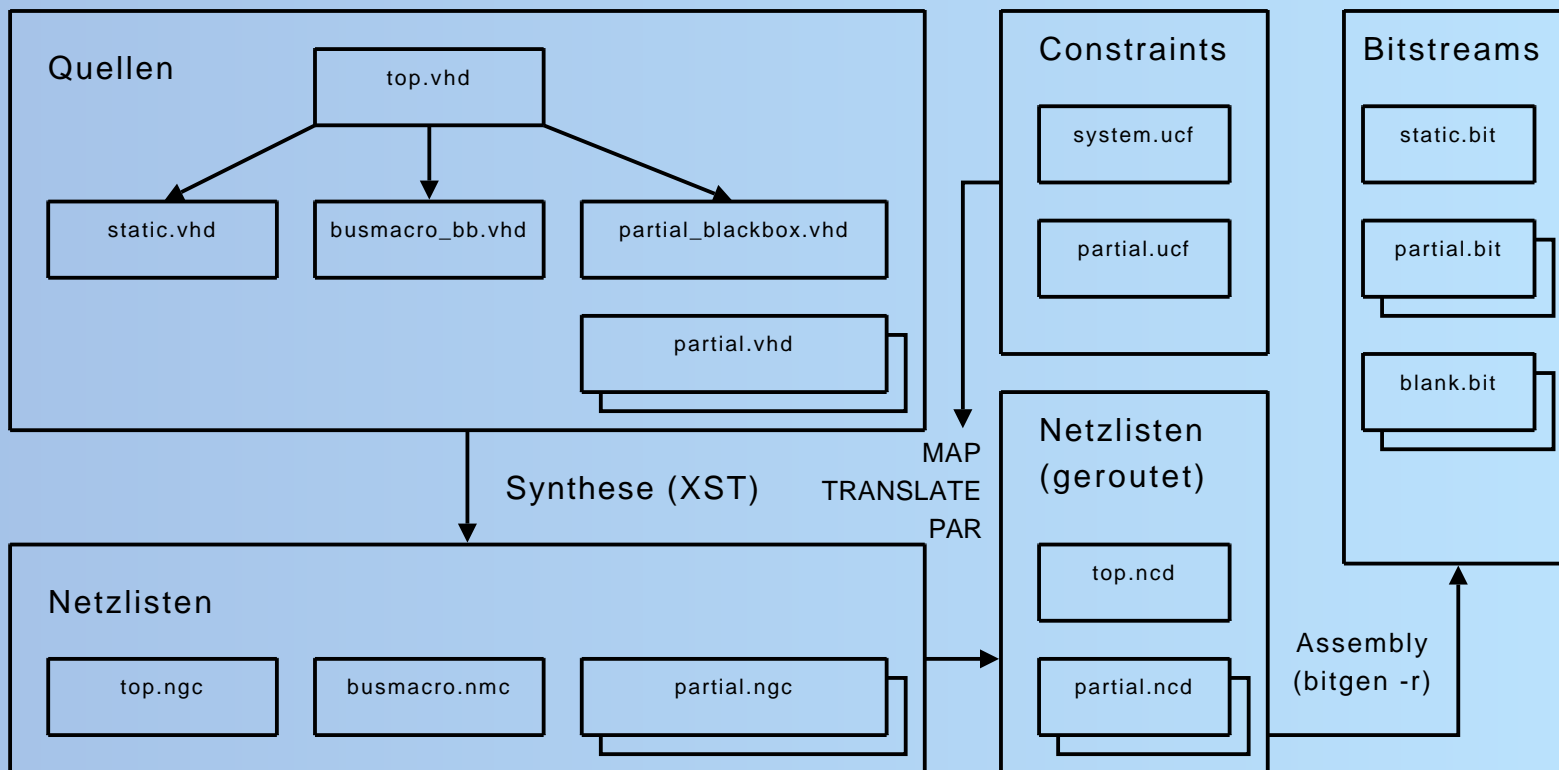
- ❖ Logik und Verbindungen in FPGAs basieren auf SRAM Zellen.
- ❖ Inhalt der Speicherzellen kann zur Laufzeit geändert werden und beeinflusst so das Verhalten der Schaltung.
- ❖ Verschiedene Modelle der Partiellen Reconfiguration
  - Module based
  - Difference based
- ❖ Spezieller Design Flow erforderlich



## Modulbasierte Partielle Rekonfiguration

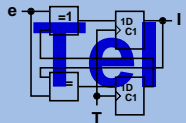


## Partieller Designflow

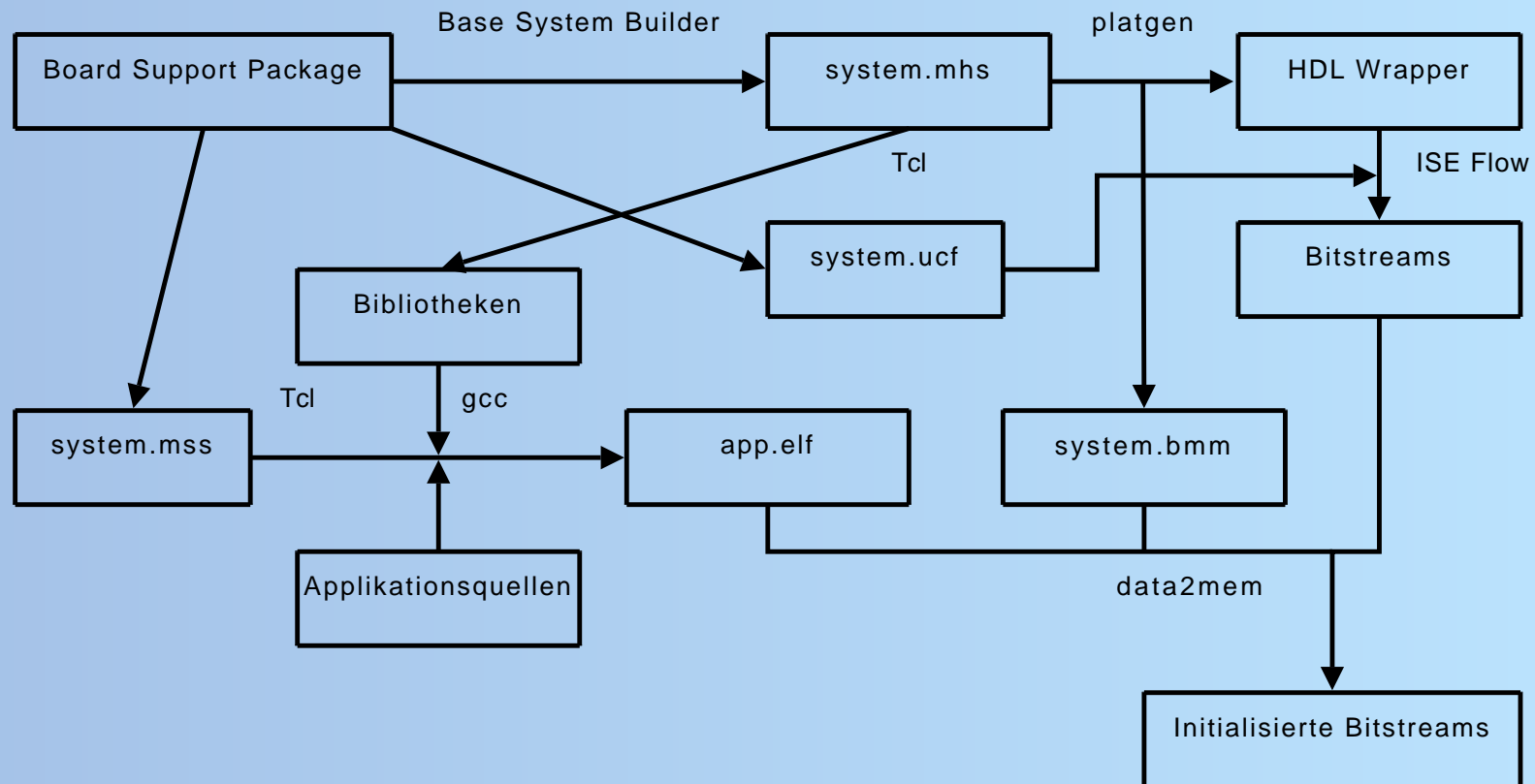


## Embedded Development Kit

- ❖ Sammlung von Xilinx IP Cores zur Implementierung eines Embedded Systems auf FPGAs.
- ❖ Verbindung der Komponenten über ein Bus-System
- ❖ Compiler Toolchain zur Entwicklung von Applikation
- ❖ Schnittstellen zur Einbindung von Betriebssystemen



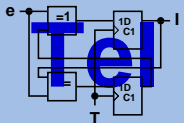
## EDK Design Flow



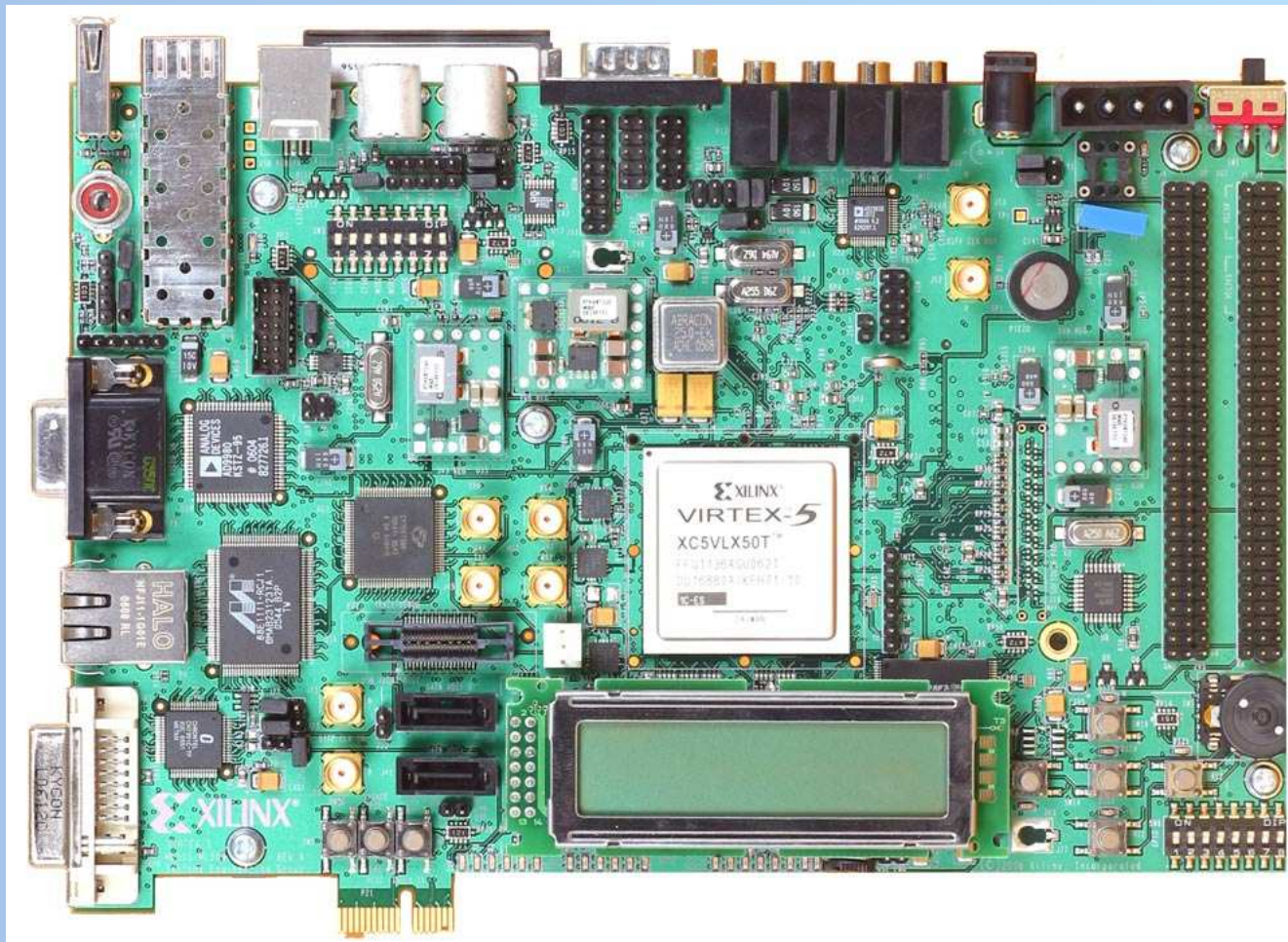


## Betriebssysteme

- ❖ Montavista Linux RTOS
- ❖ Bluecat Linux
- ❖ Petalinux
  - uCLinux-Distribution für Microblaze (PowerPC ohne MMU zur Implementierung auf Xilinx FPGAs)
  - Microblaze gcc - Toolchain
  - Linux 2.4 und 2.6
  - Umfangreiche Sammlung von Applikationen (busybox, uClib)
  - KBuild System, Support Skripte

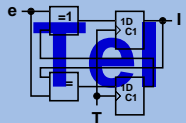


## Hardware - Xilinx ML505 Entwicklungsboard



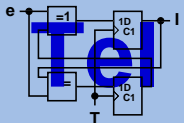
## Bestehende Projekte

- ❖ Java Server „amsys“ zur Steuerung von Atmega32 basierten HW-Agenten
- ❖ GUI und Datenbank Anbindung
- ❖ Hardware Modulmanagement und Verbindungssystem von Marcel Naggaz



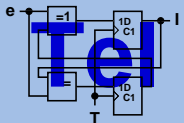
## Entwicklungsumgebung

- ❖ ISE + EDK 10.1 (Systemsynthese)
- ❖ ISE 9.1 + Partial Reconfiguration Early Access + PlanAhead (Partieller DesignFlow)
- ❖ Petalinux v0.30-rc1



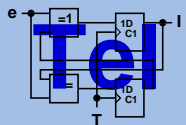
## Lowlevel Features

- ❖ Microblaze basiertes Embedded System auf dem ML505 Virtex5 Board
- ❖ Petalinux Betriebssystem
- ❖ Partielle Rekonfiguration über HWICAP
- ❖ Anbindung an den Serviceprozessor zur Steuerung der partiellen Module (Linux Treiber)

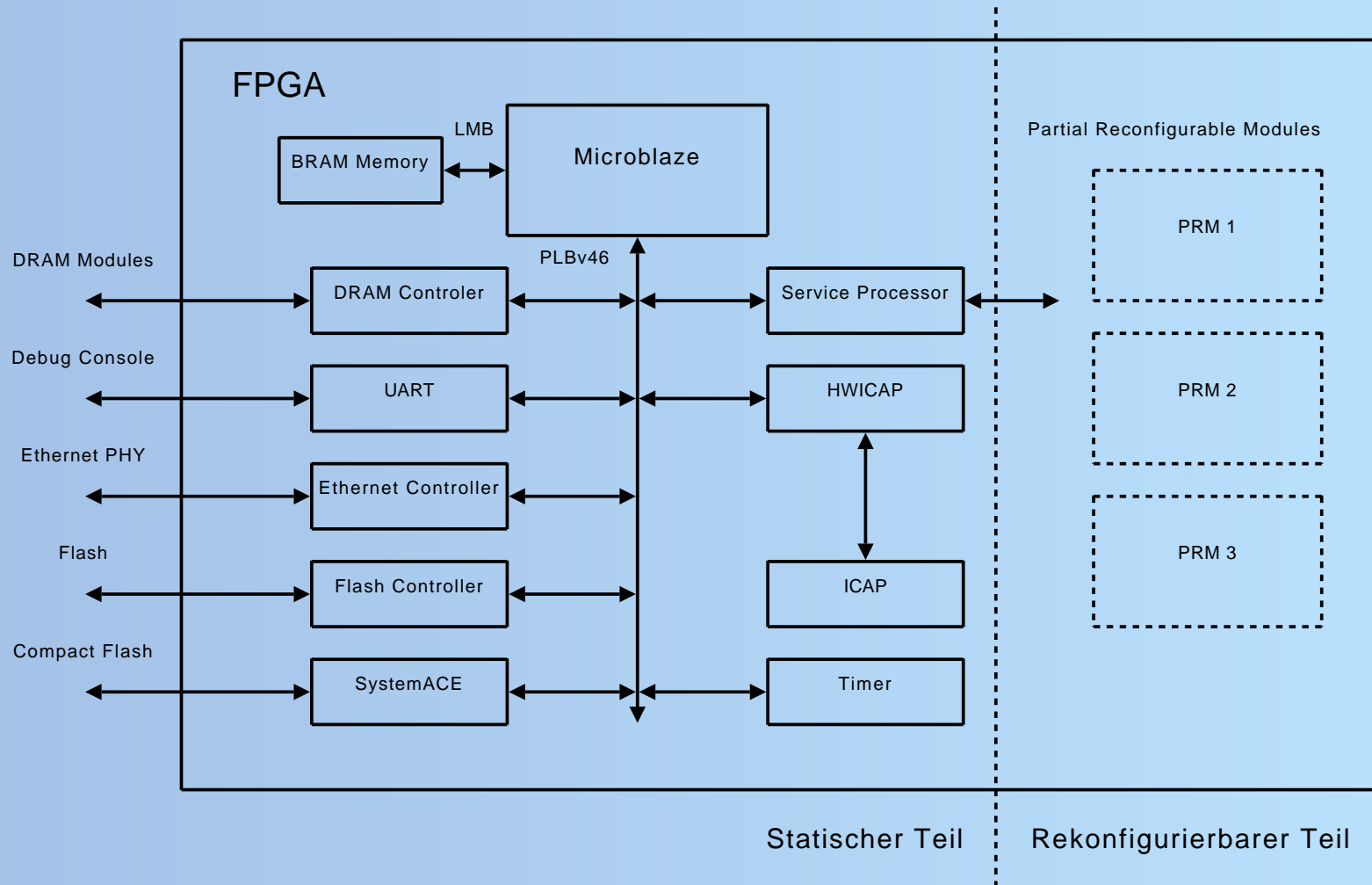


## Bootprozess

- ❖ Firststage Bootloader im BRAM des Microblaze
  - Speicherinitialisierung
  - Laden des Secondstage Bootloaders vom FLASH
- ❖ uboot
  - Kernel und RomFS Updates im FLASH über DHCP/TFTP
  - Laden des Linux Kernels
- ❖ Linux
- ❖ Root Device: FLASH (RomFS) oder SystemACE/CF (ext2)

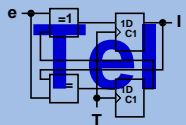


## HW-Agenten Struktur



## Umsetzung der Agenteneigenschaften

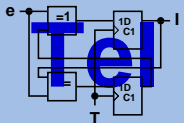
- ❖ autonomes Arbeiten: stärkere Entkopplung vom Supervisor (dem Managementserver)
- ❖ soziales Verhalten: Agentenkommunikation, Taskverteilung
- ❖ Weitere Eigenschaften: implementiert in den Partiellen Modulen





## Implementierung der Taskverwaltung

- ❖ Einsatz einer embedded JVM
- ❖ Scheduler zur Verteilung der Tasks auf die verfügbaren Modulslots
- ❖ Nutzung der Ressourcen anderer Agenten wenn möglich
- ❖ Neustart von Task von ausgefallenen Agenten
- ❖ Hardware Tasks können optional Java Threads enthalten
- ❖ Sicherung der Task und Nutzerdaten mit BerkeleyDB, SQLite oder im Dateisystem
- ❖ Kommunikation mit anderen Agenten und der GUI über Java RMI oder Corba
- ❖ Ereignisse (Task beendet, Fehlersituationen)

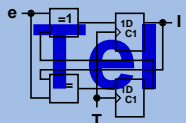


## Kommunikationsprotokoll

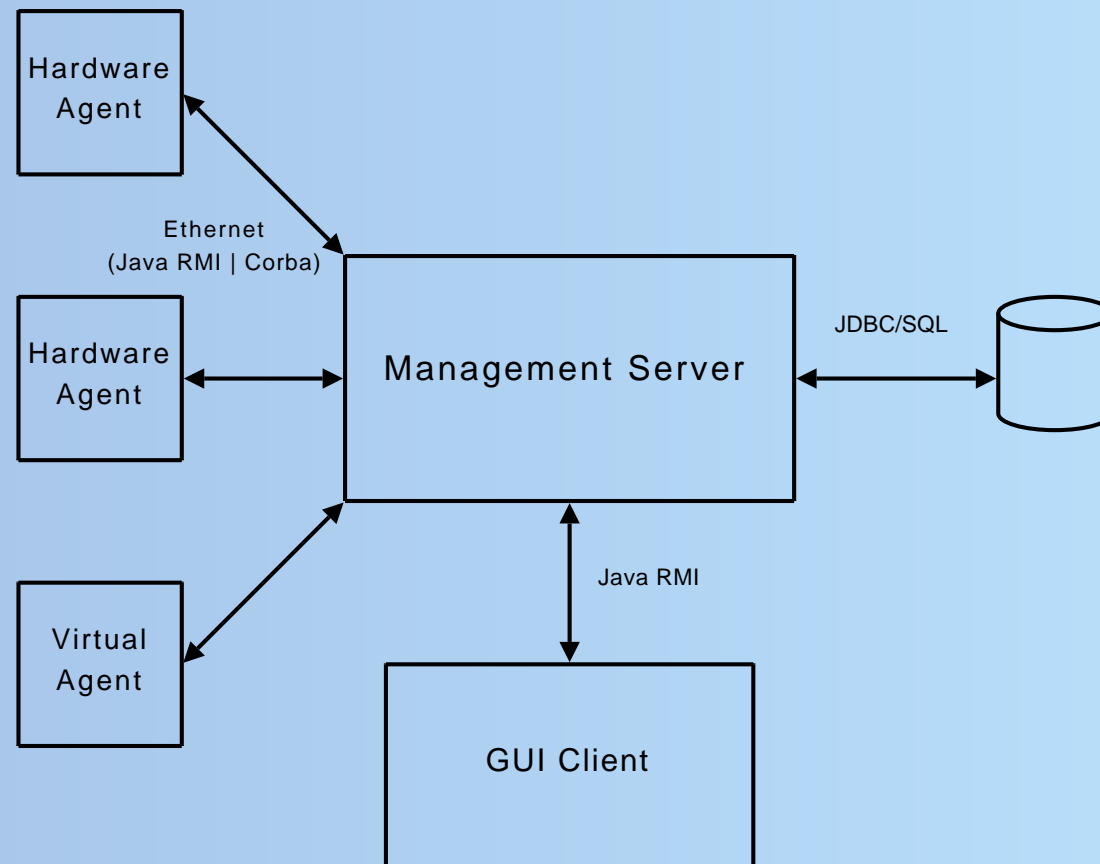
- ❖ Definiert durch die Corba IDL oder Java Remote Interfaces
- ❖ Repräsentierung der Tasks als Objekte

## Capability Management

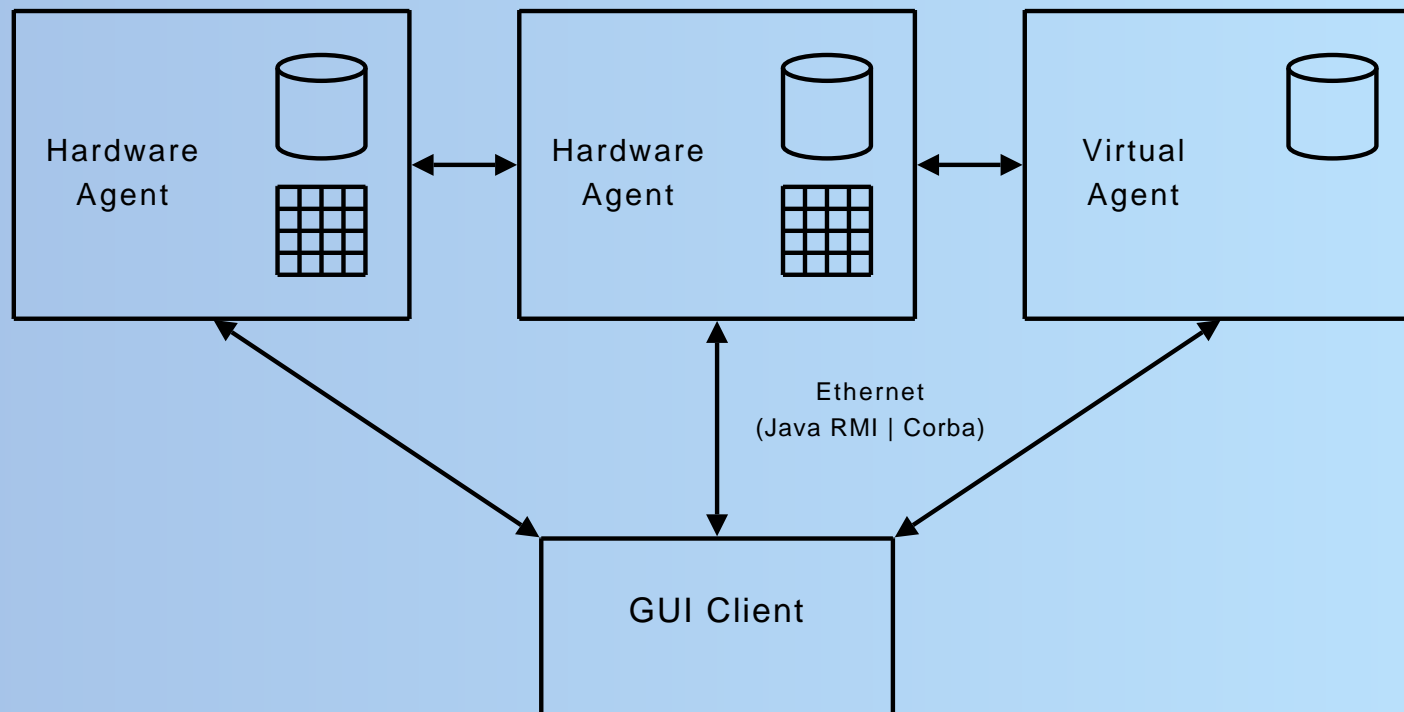
- ❖ Speicherkapazität, Partielles Layout und verfügbare Slots
- ❖ Virtuelle (Software) Agenten haben keine Hardware Ressourcen, können aber Scheduling Aufgaben übernehmen



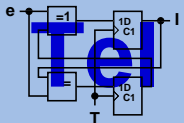
## System Struktur



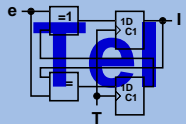
## Verteilte System Struktur



- ❖ Microblaze System
- ❖ Bootprozess über UBoot
- ❖ Lauffähiges Linux
- ❖ Partielle Rekonfiguration über ICAP



# Vielen Dank für Ihre Aufmerksamkeit!



Torsten Schmutzler  
Technische Universität Dresden  
Institut für Technische Informatik  
s6757418@inf.tu-dresden.de