Methodik zur durchgängigen Entwicklung verteilter Systeme mit Echtzeitbedingungen für Rundrufnetze

Dipl.-Inform. Marc Schanne

Präsentation im Rahmen der
Disputation über die Ergebnisse der Dissertation

8. November 2007, Karlsruhe

Kontext und Anforderungen

- **Verteilte sicherheitskritische Systeme**
  - besserer Skalierbarkeit und höherer Zuverlässigkeit
  - dezentrales Anwendungsdesign

Kontext und Anforderungen

- **Verteilte sicherheitskritische Systeme**
  - besserer Skalierbarkeit und höherer Zuverlässigkeit
  - dezentrales Anwendungsdesign
- **Softwareentwicklung für eingebette Systeme**
  - asynchrone Kommunikation für beschränkte Ressourcen
  - Standardmethoden zur Komponentenbeschreibung
  - Nichtfunktionale Anforderungen mit objektorientiertem Komponentendesign

Kontext und Anforderungen

- **Asynchrone Kommunikationsinfrastruktur**
  - Generierung von Standardprogrammcode und äquivalenten Systemmodellen für statische Analyse
  - zuverlässige Nachrichtenkommunikation mit Zeitgrenzen für die Verarbeitung in der Anwendungsschicht

Kontext und Anforderungen

- **Asynchrone Kommunikationsinfrastruktur**
  - Generierung von Standardprogrammcode und äquivalenten Systemmodellen für statische Analyse
  - zuverlässige Nachrichtenkommunikation mit Zeitgrenzen für die Verarbeitung in der Anwendungsschicht
- **Durchgängige Softwareentwicklung**
  - Unterstützung für Anwendungsentwurf, Implementierung und Analyse
  - zuverlässige Anwendungsentwicklung für verteilte sicherheits kritische eingebettete Systeme

Ergebnisse der Arbeit

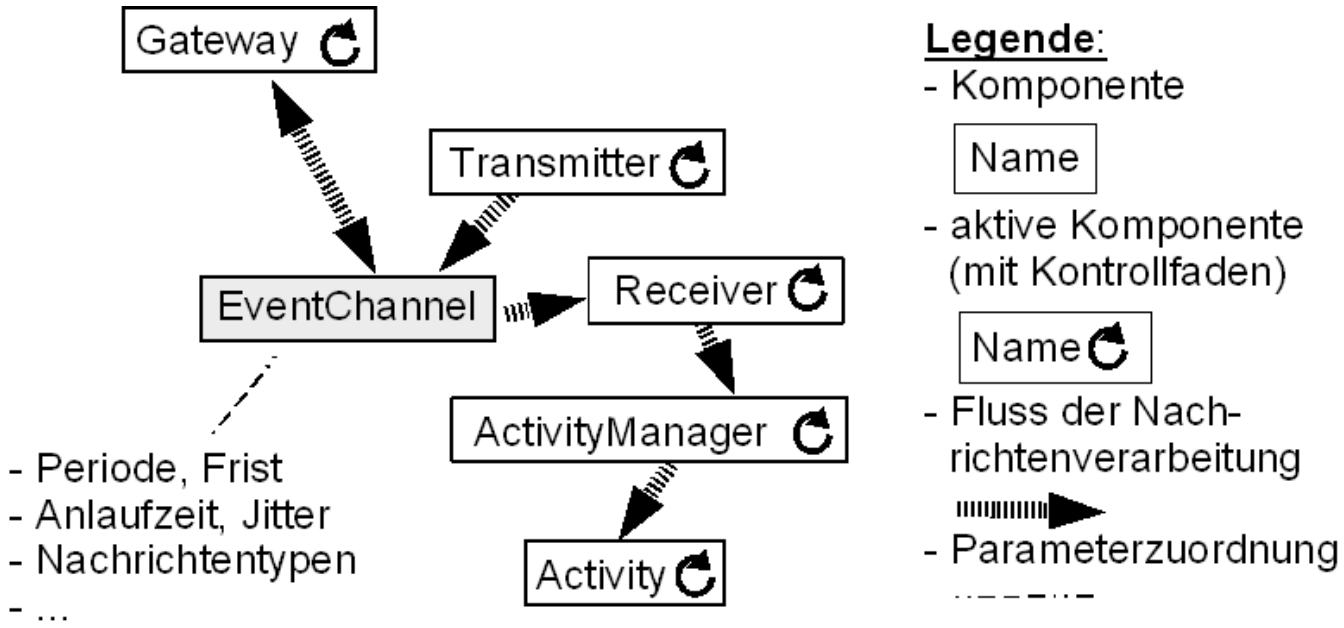
- **Methodik** zur **durchgängigen Entwicklung verteilter Systeme** mit **Echtzeitbedingungen** für **Rundrufnetze**

- **Methodik** für einzelne Phasen der Softwareentwicklung
  - **Entwurf**
    - Nachrichtenkanäle für netzunabhängigen Komponentenentwurf
  - **Implementierung**
    - Systembeschreibung erlaubt automatische Code-Generierung
  - **Analyse** und Tests
    - Statische Analyse mit generierten Modellen für Ablaufpläne

Thesen und Ergebnisse: Entwurf

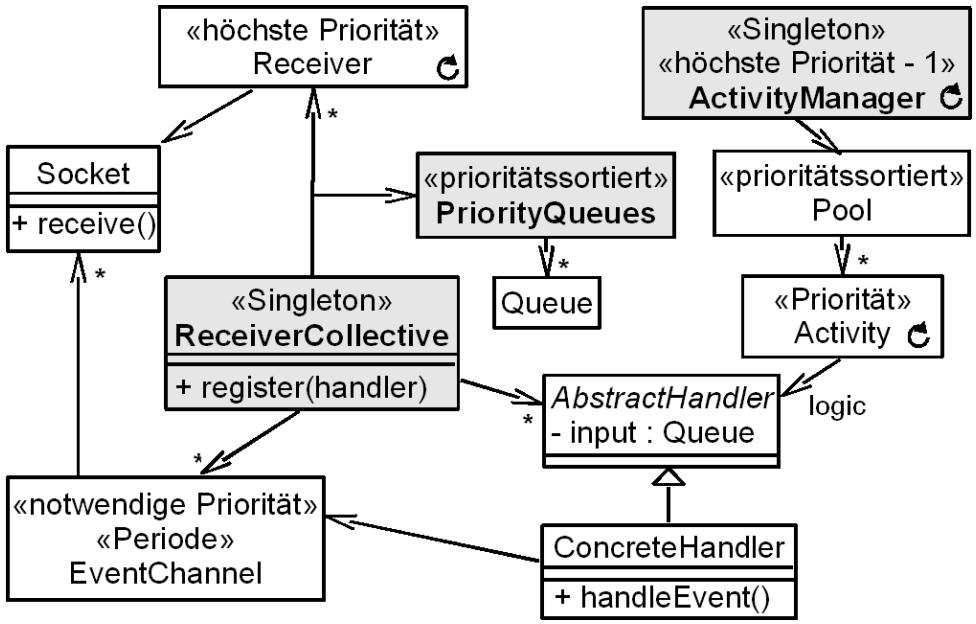
(1) Konzept von Nachrichtenkanälen vereinfacht die Entwicklung verteilter Systeme auf Basis von asynchroner Nachrichtenkommunikation in Rundrufnetz- und Feldbus-Infrastrukturen.

- Netzunabhängige, zentrale Abstraktion des Nachrichtenkanals beschreibt Kommunikationcharakteristika (z.B. Periode, Ungenauigkeit) und Echtzeitbedingungen (z.B. Verarbeitungsfrist und -zeit)



Thesen und Ergebnisse: Entwurf

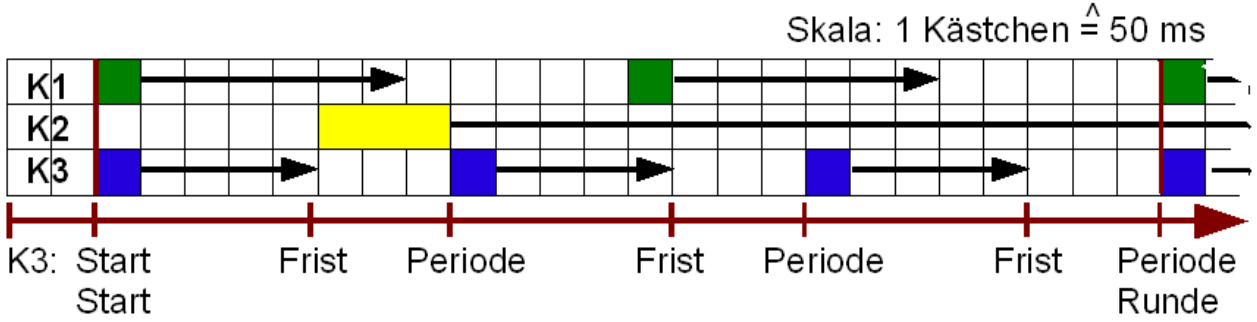
- Nebenläufiger Nachrichtenempfang verlangt Planung:
  - Plattformunabh. Entwurfsmuster: Trennung zwischen umgehender Entgegenahme der Nachrichten mit einem Kontrollfaden höchster Priorität und der anschließenden Verarbeitung durch Kontrollfäden mit Prioritäten umgekehrt proportional zur geforderten Frist.



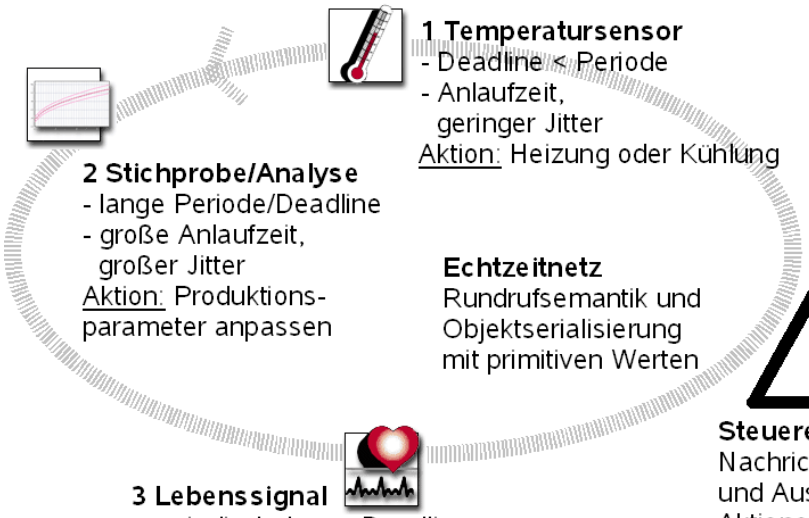
  - **Empfängerkollektiv** Empfang an Netzzugangsschnittstelle bei Verfügbarkeit und Einreihung der Nachricht in Warteschlangen
  - **Aktivitätsmanager** Verwaltung der Behandlerlogik mit Zuteilung von Nachrichten an verfügbare Aktivitätskontrollfäden
  - **Warteschlangen** sortiert nach Prioritäten

Thesen und Ergebnisse: Entwurf

- Beispiel für nebenläufigen Empfang von drei Kanälen (K1-K3)



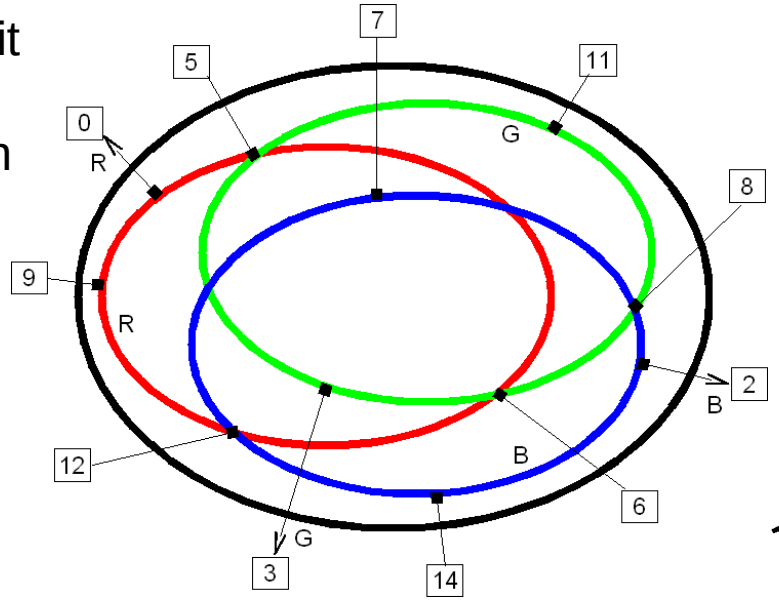
	K1 Temp.	K2 Probe	K3 Leben
Periode	600	1200	400
Start	100	350	100
Jitter	50	150	50
Frist	300	1200	200
WCET	50	250	50



Steuereinheit mit Nachrichtenempfang und Ausführung von Aktionen inEchtzeit

Thesen und Ergebnisse: Entwurf

- Echtzeitgarantien im Kommunikationsprotokoll der Transportschicht erlauben Fristen bei der Nachrichtenverarbeitung in der Anwendungsschicht der Kommunikationsknoten einer verteilten Anwendung.
  - Netze mit Rundruflogik oder Feldbusse mit Echtzeitgarantien bei der Paketvermittlung werden vorausgesetzt.
  - Komponentenorientierter Entwurf der Applikation mit loser Kopplung: Echtzeitbedingungen werden an die Verarbeitung von Nachrichten geknüpft, die über Publiziere/Abonniere-Kommunikation für ein Thema (d.h. einen Nachrichtenkanal) verteilt werden.
  - Flexibler Anwendungsentwurf mit Knoten und Nachrichtenkanälen für Hin- und Rückkommunikation über einem Rundrufnetz.

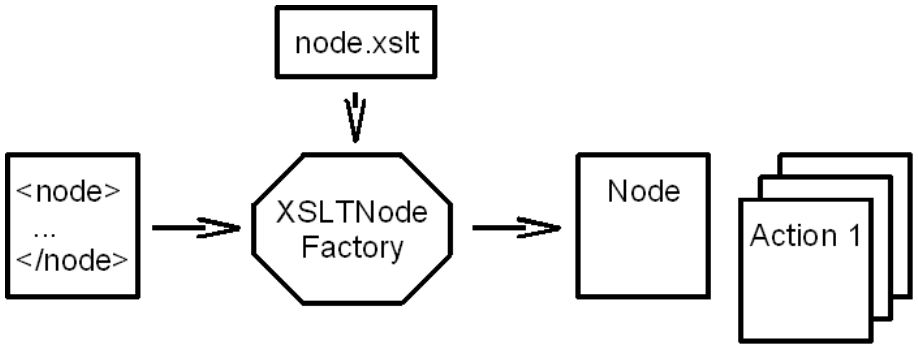


(Beispiel einer Gleiche-zu-Gleichen-Kommunikationsanwendung mit elf Knoten und drei Nachrichtenkanälen.)

Thesen und Ergebnisse: Impl.

(2) Eine deskriptive Entwurfsmethode unterstützt die Entwicklung verteilter Systeme und die Wiederverwendung von Kompontenten (auch der Bibliothek) ist leichter möglich.

- Generierung von Programmrahmen vereinfacht die Anwendungsentwicklung. Anwendungsspezifische Verarbeitungslogik kann dank komponentenorientierten Anwendungsdesigns leicht wiederverwendet werden.

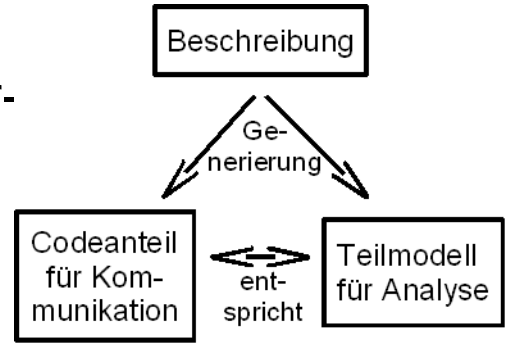


- Entwurfsmuster für Anwendungsdesign erlaubt plattform- und netzunabhängigen Entwurf.
  - Die Generierung von Standardprogrammcode für die Kommunikation verschiebt Entwicklungswissen in die Rahmenarchitektur bzw. die Laufzeitumgebung.

Thesen und Ergebnisse: Analyse

(3) Statisch vorhersagbare harte Echtzeitbedingungen ermöglichen auch die Erzeugung und Analyse von Modellen für Ablaufpläne der Kontrollfäden in jedem Knoten.

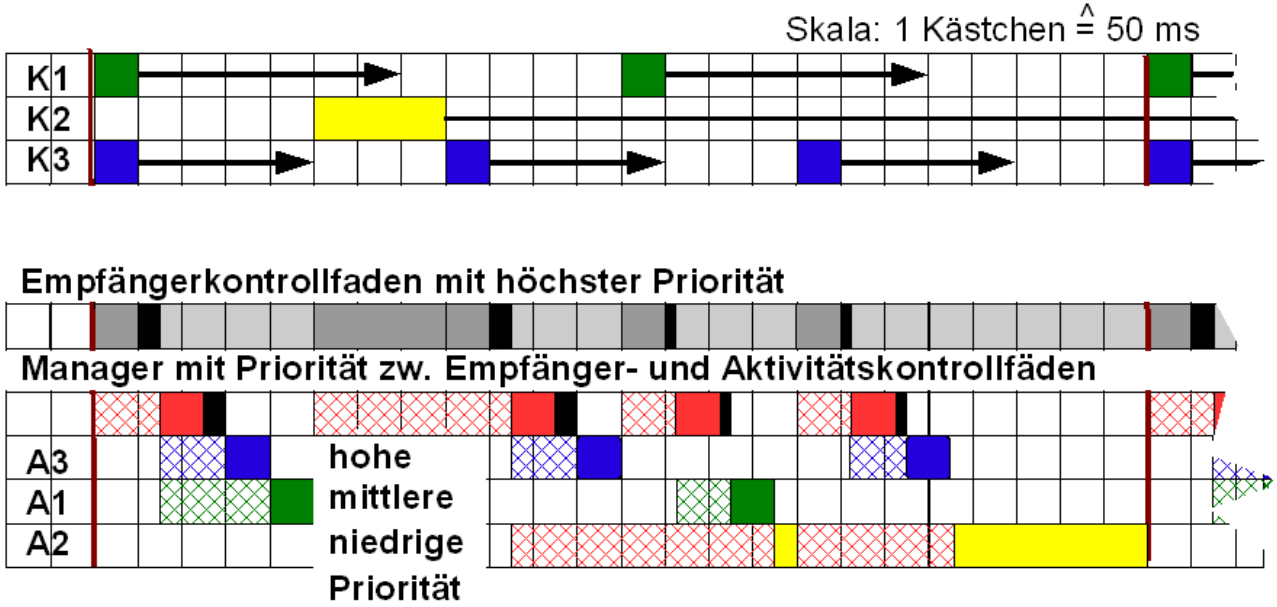
- Die Generierung von Modellen äquivalent zum Programmcode erzeugt im Prototyp eine Steuerdatei für ein Werkzeug zur Analyse der Ablaufkoordinierung von nebenläufigen Kontrollfäden mit einem linearen Modell (MAST entwickelt an der Universität von Kantabrien, Spanien).



  - Voraussetzung für die statische Analyse ist eine Laufzeitumgebung mit einem Verfahren der Ablaufkoordinierung mit festen Prioritäten.
- Verwendung von Ergebnissen einer Zeitabschätzung für die Ausführung von Bibliothek und Anwendungskomponenten im ungünstigsten Fall (WCETA) bei der Analyse der Ablaufpläne.
  - Die Einhaltung von Fristen hängt vom Programmcode (Bibliothek der Rahmenarchitektur, anwendungsspezifischen Behandlerlogik) und der verwendeten Laufzeitumgebung und Hardware ab.

Thesen und Ergebnisse: Analyse

- Modell im ungünstigsten Fall für die Verarbeitung mit den notwendigen Kontrollfäden.
  - Überlappungsfreie Aktivität der Empfängerkontrollfäden (mit höchster Priorität).
  - Zweithöchste Priorität des Kontrollfadens für den Aktivitätsmanager zur Verwaltung von Verarbeitungslogik und Aktivitätskontrollfäden.
  - Aktivitätskontrollfäden (A2, A1, A3) mit Prioritäten umgekehrt proportional zur Verarbeitungsfrist der Nachrichtenkanäle (K2, K1, K3).



Verwandte Arbeiten

- **Unified Modeling Language (UML)**
  - Standardisierte Diagramme für Modellierung mit Unterstützung in unterschiedlichen Phasen des Softwareentwicklungsprozesses.
  - Model-Driven Architecture (MDA) für modellgetriebene automatisierte Entwicklung.
- **UML Profile for Schedulability, Performance and Time (UML-RT)**
  - UML-Erweiterung für die Beschreibung von Systemen mit Echtzeitbedingungen.
- **Domain Specific Language/Modelling (DSL/DSM)**
  - Entwicklung von Anwendungen für klar begrenzte Fachbereiche (z.B. verteilte Echtzeitsysteme).

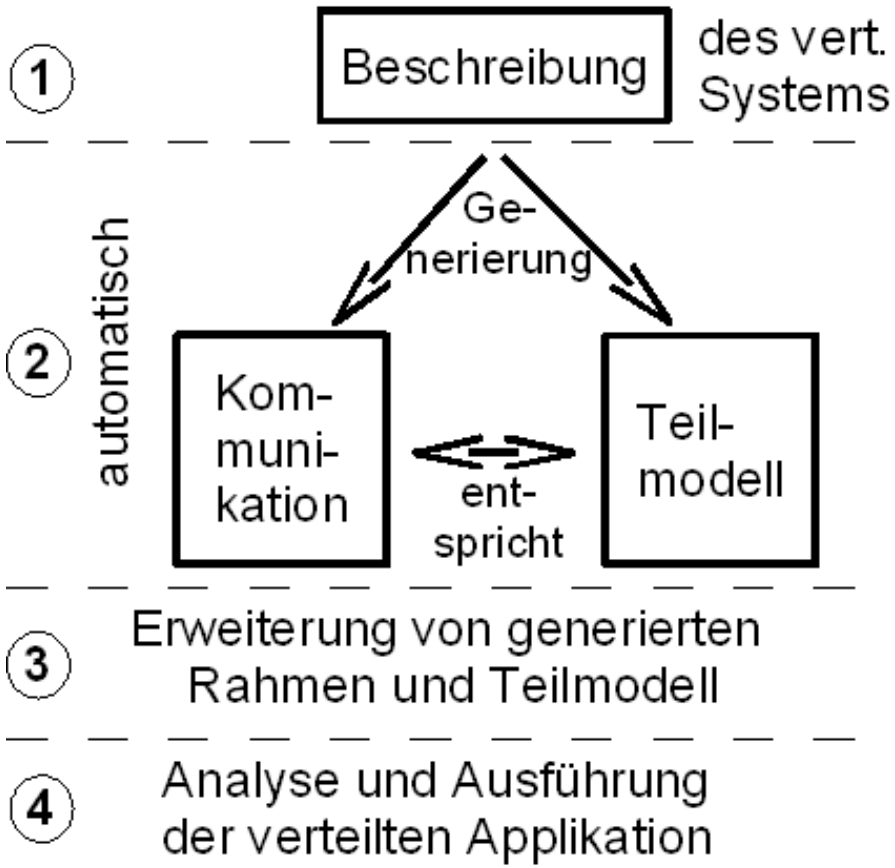
Methodik im Überblick

(1) Beschreibung
in XML-Dateien für die Festlegung von Nachrichtenkanälen mit Perioden und Fristen.

(2) Generierung
von Standardprogrammcode für die Kommunikation und äquivalenten Analysemodellen.

(3) Erweiterung
generierter Rahmen mit spezifischer Logik und Modelldaten.

(4) Analyse
der Ablaufkoordinierung aller Kontrollfäden auf Einhaltung aller Fristen in jedem Anwendungsknoten vor Ausführung.



Evaluierung und Grenzen

- **Softwareentwicklungsmethodik**
  - Evaluierung der Methodik ist schwierig.
  - Keine Parallelversuche zum Vergleich mit anderen Methoden und Kommunikationsrahmenarchitekturen.
  - Erfahrungen aus EU-Forschungsprojekten.

- Evaluierung der Rahmenarchitektur und vorgeschlagener Methodik unter verschiedenen Gesichtspunkten
  - **(1) Eignung der Rahmenarchitektur** für den Entwurf verteilter Anwendungen (Basis einer Gleiche-zu-Gleichen-Bibliothek).
  - **(2) Generierung von Programmcode und -rahmen** aus XML-Beschreibungen für Kommunikation und Echtzeitbedingungen.
  - **(3) Möglichkeit der Modellprüfung** für Ablaufpläne notwendiger (d.h. generierter) Kontrollfäden.

Evaluierung und Grenzen

- **Eignung asynchroner Nachrichtenkommunikation**
  - Einsatz in zwei EU-Forschungsprojekten für die Entwicklung verteilter eingebetteter Systeme mit einer Hochsprache und Laufzeitumgebung am Beispiel von Echtzeit-Java (RTSJ)
    - High Integrity Distributed OO Realtime Systems (HIDOORS)
    - High Integrity Java (HIJA)
- **Nutzen von Codegenerierung mittels XSLT**
  - Vorstufe für Codegenerierung aus UML im Projekt HIDOORS
- **Statische Analyse unter harten Echtzeitbedingungen**
  - Einsatz einer einheitlichen Kommunikationsrahmenarchitektur mit harten oder weichen Echtzeitbedingungen und Erweiterung einer umfassenden Werkzeugkette für die statische Analyse bei harten Echtzeitbedingungen.

Evaluierung und Grenzen

- **Grenzen / Voraussetzungen (für harte Echtzeitbedingungen)**
  - Netz mit Echtzeitgarantien in der (Rundruf-)Transportschicht
  - Laufzeitumgebung mit einem Verfahren für Ablaufkoordinierung mit festen Prioritäten und Vermeidung von Verklemmungen z.B. mit Prioritätshöchstmaß. Ergebnis: Definition HRT-Profil für RTSJ
  - Verwendung von Zeitabschätzungen für Programmteile unter ungünstigsten Voraussetzungen (z.B. FZI Gromit, aiT von AbsInt)
  - Ende-zu-Ende-Analyse evtl. über Modellprüfung (UPPAAL) oder Erweiterung der Ablaufplananalyse (MAST) auf Kommunikation
  - Änderung bestehender Entwicklerkulturen: Statische Analyse, Systembeschreibung, asynchrone Kommunikation in Echtzeit

Ausblick und Auswirkungen

- **Entwicklungen bei Echtzeit-Java**
  - Standardisierung asynchroner Nachrichtenkommunikation mit Echtzeitbedingungen bei der Verarbeitung.
  - Verwendung asynchroner Kommunikation in kommenden Profilen für Echtzeit-Java.
  - Standard-Annotationen für Werkzeugunterstützung und Programmcodegenerierung.

- **Softwareentwicklung von sicherheitskritischen Systemen**
  - Einsatz deskriptiver Entwicklungsmethoden für plattform- und netzunabhängige Entwicklung sicherheitskritischer Systeme.
  - Statische Analyse der Ablaufkoordinierung aller Kontrollfäden auf Einhaltung von harten Echtzeitbedingungen für die Zertifizierung von sicherheitskritischen Systemen.

Vielen Dank! - Noch offene Fragen?

- **HIJA Projektwebseite:**
  http://www.hija.info

- **HIDOORS Projektwebseite:**
  http://www.hidoors.org

- **Nachrichtenkanalnetz event channel network:**
  http://www.eventchannelnetwork.org
  http://sourceforge.net/projects/ecn/