Year 2 Review
Brussels, February 12, 2010

Transversal Activity

Achievements and Perspectives

Design for Adaptivity

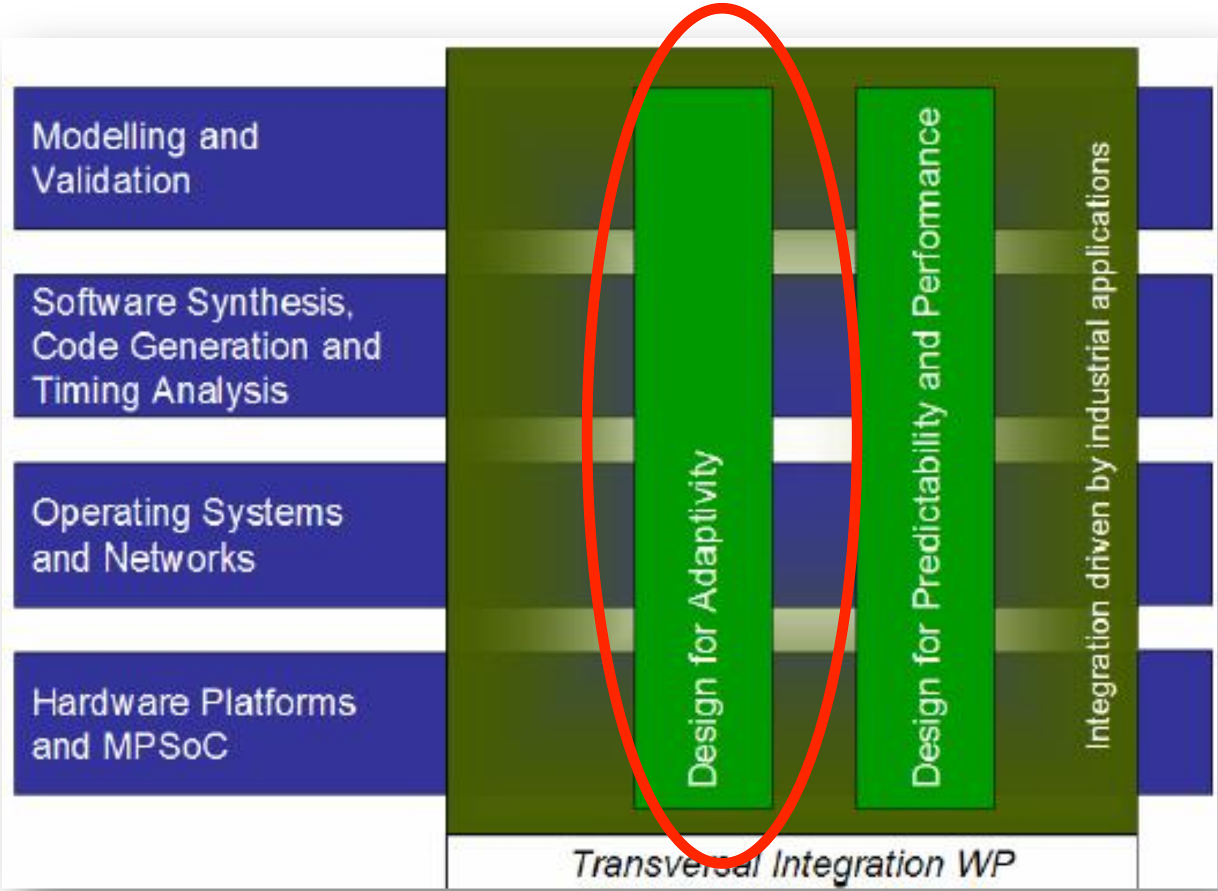
Activity Leader : Karl-Erik Årzén

Lund University

Outline

- Structure
- Why adaptivity?
- Objectives and Vision
- Assessment of Year 2
- Project Classification
- Scientific Highlights
  - DySCAS - Dynamically Self-Configuring Automotive Systems
  - ACTORS – Adaptivity and Control of Resources in Embedded Systems
- Plans for Year 3

Structure



- Around half the size of a thematic cluster

Involved Partners

Core Partners:

- Karl-Erik Årzén (ULUND)
- Gerhard Fohler (TUKL)
- Giorgio Buttazzo (SSSA)
- Luis Almeida (UPorto)
- Luca Benini (UBologna)
- Stylianos Mamagkakis (IMEC)
- Eduardo Tovar (Porto)
- Björn Lisper (MdH)
- Alan Burns (York)
- Lothar Thiele (ETH-Z)
- Hamid Brahim (CEA)
- Axel Jantsch & Martin Törngren (KTH)
- Jan Madsen (TU Denmark)
- Rolf Ernst (TUBraunschweig)
- Joseph Sifakis (VERIMAG)

Affiliated Partners:

- Alejandro Alonso (UPM)
- Lucia Lo Bello (UCatania)
- Pau Martí (UPC)
- Johan Eker (Ericsson)
- Liesbeth Steffens (NXP)

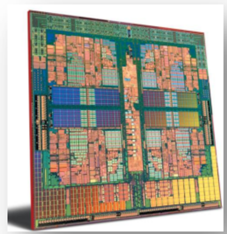
Definitions

*"An embedded system is **adaptive** if it is able to adjust its internal strategies to meet its objectives"*

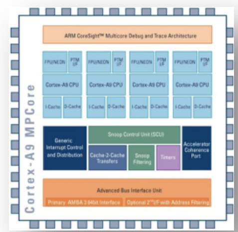
*"An embedded system is **robust** if it meet its objectives under changing conditions without modifying its internal strategies"*

Why Adaptivity?

- Increasing complexity of embedded systems
  - Higher requirements on autonomous behaviour
- Increasing uncertainty in use cases and resource requirements
  - Designs based on worst-case prior information unfeasible
- Hardware development makes adaptivity a possibility
  - Reconfigurable hardware
  - Power saving technologies



- Hardware development increases the need for adaptivity
  - Multi- & many-core platforms
  - Variability of 10-20 nm chips



- Hardware development makes adaptivity more complicated
  - High performance on, e.g., multi-cores, for communication-heavy applications requires careful optimization and complicates on-line modifications

Motivation for Adaptivity

- Cope with uncertain resource requirements (CPUs, network)
  - Unknown resource requirements
  - Varying resource requirements
  - Changes in total workload (multiple applications)
- Cope with uncertainties in resource availability
  - Changes in the amount of resources (# cores, # nodes, clock frequency, …)
    - To save power, minimize heat, ….
  - Changes in the quality of resources (network variability, ….)

Goals for Adaptivity

- Maximize the service delivered with a fixed level of resources
- Minimize the resources used while maintaining an acceptable service level
- Increase dependability
  - Reliability, safety, availability, maintainability, ….

Levels of Adaptability

- Adapability can be applied at several levels:
  - Application level
  - Distributed system level
  - Middleware level
  - OS level
  - SoC level
  - HW level

Problems of Adaptivity

Adaptivity can introduce new problems:

- The adaptation mechanism itself consumes resources
- Harder to provide formal guarantees about the system
- Adds to the complexity
- May complicate the design process (modeling, V&V, …)
- Requires tuning
- Bad tuning might lead to stability problems
- Sensors and actuators are necessary
- Models are necessary
  - Of the system that we adapt
  - Of the adaptation mechanism itself

High-Level Objectives

- Integrate the efforts and combine the competences related to adaptivity in embedded systems within the thematic clusters of ArtistDesign.
- Create suitable interfaces, meeting points, and research contacts between the partners and the communities.
- Define the ontology for adaptivity in embedded systems,
  - Define relationship between adaptivity, reconfigurability, flexibility, sustainability, and robustness
  - Define relationship between adaptivity and predictability.

Long Term Vision

To generate a substantial advance in theory, methods and tools of relevance to adaptivity in embedded systems and to disseminate this into industry and to the scientific community at large.

State of the Integration in Europe

- Adaptivity is a very general concept
- Most research on embedded systems relates to adaptivity in some way
- However few forums that are specifically aimed at adaptivity in embedded systems
- Adaptivity is of highest concern in consumer electronics and telecommunications (multimedia & soft realtime)
- However, also in the more hard and safety-critical sectors one finds needs and efforts related to adaptivity
  - E.g. the DySCAS project

Building Excellence

- Joint and individual research projects
  - Funded by other sources → Networking and contacts
- Annual general meeting for the activity
  - SSSA (Pisa), 2-3 April 09
  - 21 participants representing 15 partners
  - http://www2.control.lth.se/ArtistAdapt/index.php/Main_Page/Meetings/Pisa_Apr_2009/
- Smaller meetings and workshops organized by the partners
- The common wiki has been extended
  - http://www2.control.lth.se/ArtistAdapt/
  - Much more required, however

Overall Assessment and Vision at Y0+2

- Well attended annual meeting
- Numerous research activities
- Contributed to education about adaptive and feedback-based approaches.
  - Summer schools or special courses
- Several industrial contacts
  - E.g. NXP, Ericsson, Volvo, Evidence, Enea
- Two successful workshops organized:
  - FeBID ’09, San Francisco, April 16 2009 (CPSWEEK)
  - APRES ’09, Grenoble, October 11 2009 (ESWEEK)

Quantitative Assessment of Y2

- At least 14 joint publications (Goal: 10)
- At least 9 research collaborations involving more than one partner, including several European projects (Goal: 15 but uncertainties in counting)
- At least 7 meetings or workshops organized or co-organized by the partners. (Goal: 10)
- One summer schools organized by the partners (sometimes jointly with Artist2) (Goal: 3)
- The content of the wiki substantially expanded

→ At, or somewhat, below our ambitions

Meetings, Workshops & Courses

General Meetings:

- Activities Meeting, Pisa, 2-3 April 2009

Workshops:

- Fourth Int. Workshop on Feedback Control Implementation and Design in Computing Systems and Networks (FeBID 2009)
  - San Francisco - April 16, 2009
  - Cofunded by Artist (co-chaired by Årzén, several of the partners in the PC)
  - http://controlofsystems.org/febid2009/
  - 1 plenary talk + 12 submitted papers

Meetings, Workshops & Courses

Workshops:

- Second Workshop on Adaptive and Reconfigurable Embedded Systems (APRES 09)
  - Grenoble - Oct 11, 2009
  - Cofunded by Artist (co-chaired by Almeida, Årzén)
  - 1 plenary (Mamagkakis) + 12 submitted papers
    - Papers will be published by SIGBED
  - http://www.artist-embedded.org/artist/Overview.1765.html
- DySCAS Open Workshop
  - Brussels - February 18, 2009
  - Dissemination of the results of the DySCAS (Dynamically Self-configuring Automotive Systems) FP6 project
  - http://www.artist-embedded.org/artist/DySCAS-2009.1555.html

Meetings, Workshops & Courses

Courses:

- ARTIST Graduate Course on Embedded Control Systems
  - Pisa - June 8-12, 2009
  - SSSA, ULUND, UPorto, UAveiro
  - http://www.artist-embedded.org/artist/Overview.1673.html

Smaller Meetings:

- BIP-DOL Meeting, Grenoble Oct 13, 2009
  - VERIMAG, ETHZ
- Real-Time Parameters
  - ETHZ, SSSA
- Numerous smaller meetings involving the partners in projects
  - ACTORS, FRESCOR, DySCAS, ...

Tools and Platforms

- SHARK and ForSyDe reported in Y1
- SWEET (SWEdish Execution Time tool)
  - Parametric WCET analysis
  - Mälardalen and Usaarland
- Hardware setup
  - Demonstrate self-protection and adaptability of embedded Real-Time Systems
  - TUBraunschweig, UErlangen

Tools and Platforms

- TrueTime Simulator
  - Networked embedded control simulation in Simulink
  - ULUND + several Artist partners as users
  - Four new releases during 2009
    - GPL license
    - Support for Constant Bandwidth Servers (hard and soft)
    - FlexRay and PROFINET IO network models
    - Multi-core support with partitioned scheduling
    - Network models ported to Modelica/Dymola within ITEA 2 project EUROSYSLIB

Project Classification

Modeling:

- Modeling and analysis of adaptive systems (KTH, OFFIS)
  - ANDRES
- Model-Based Implementation of Real-Time Applications in BIP (VERIMAG)

Middleware:

- QoS-Aware Adaptive Cooperative Systems (IPP)
- Adaptation in service-oriented architectures (UPM)

Project Classification

Analysis:

- Dynamic Changes of Real-Time Parameters (ETHZ, SSSA)
  - Combine: Real-Time Calculus + Real-Time Interfaces + server-based hierarchcal scheduling
  - Goal: To achieve more dynamic scheduling strategies allowing online parameter changes while meeting real-time constraints
- Parametric WCET analysis (MDH)

Control:

- Analysis of Event-Driven Control Systems (UPC, SSSA, ULUND)
- Feedback control of web servers (ULUND)

Project Classification

WSN:

- Adaptive Topology Management in WSNs (UCatania)
- Wireless Technologies for Automation (UCatania)
  - flexWARE project
- Adaptive energy management in WSN (UBologna, ETHZ)
- Graceful Degradation in Real-Time Wireless Protocols (UPorto, UAveiro)

Media Processing:

- Adaptive MPEG-2 decoding (TUKL, ULUND)

Project Classification

Resource Management:

- Adaptive Resource Management in FRESCOR and ACTORS (UYork, TUKL, ULUND, Evidence, Ericsson, UCantabria)
- Temporal Isolation in Real-Time Systems (IPP)

Distributed systems:

- Adaptivity in Distributed Systems (UPorto, UAveiro, MDH)

Project Classification

Self-configuration - self-optimization:

- Self-configuring Embedded Systems (KTH, Volvo, Offis)
  - DySCAS
- Reference implementations of self-configuring embedded systems (KTH)
- In-system self-optimization for real-time systems (TU Braunschweig)
  - Self-protection through autonomous assignment of execution priorities

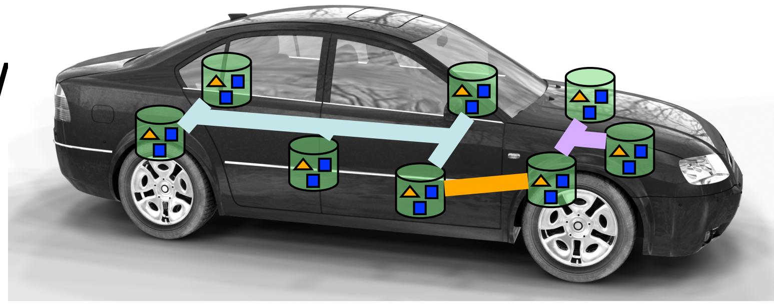
Project Classification

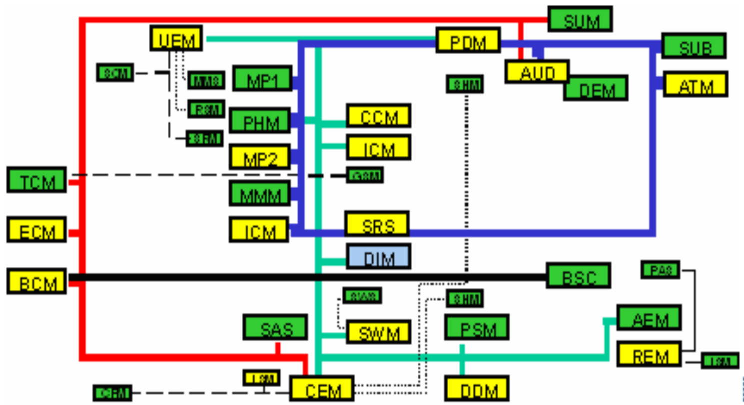
Hardware:

- eDNA: Reconfigurable self-organising and self-healing hardware (DTU)
- Dynamic Resource Management in Embedded Systems (IMEC, NTNU, NTUA)

Dynamically Self-Configuring Automotive Systems

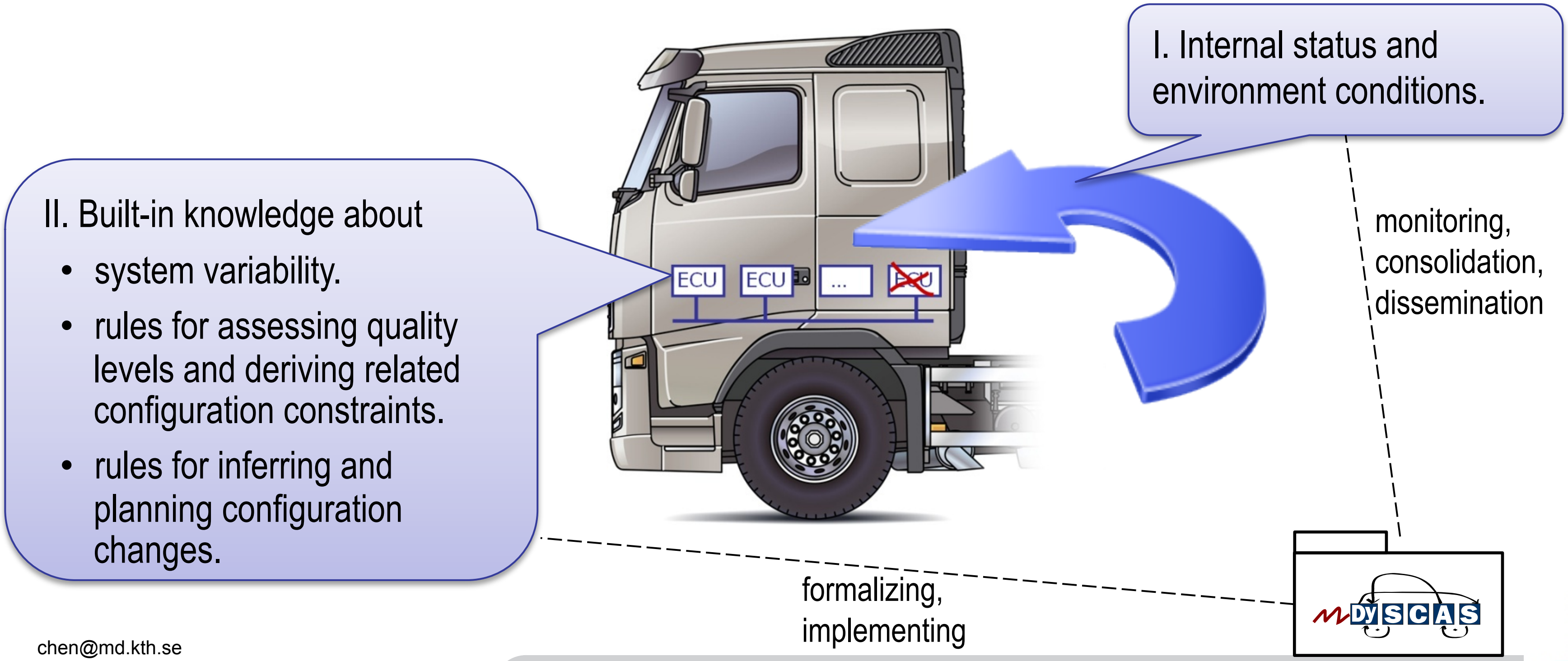
- FP6 project ended in Feb 09
  - Volvo Technology, Daimler, Enea , Bosch, UGreenwich, UPaderborn, KTH, Movimento, Systemite, UOldenburg
- Systematic approach to a Dynamic Architecture through MV
  - Efficient and reliable field maintenance and upgrades
  - Support for information and functionality sharing
  - On-line configuration adaptation
    - Performance, dependability, …
    - Well-delimited
      - Specification includes allowed, preferred and prohibited alternatives





Research Objective: a middleware approach to *self-configuration*

- An embedded system is *self-configuring* if it is able to adjust by itself its structures, behaviors, and data to meet its functionality and quality requirements.



chen@md.kth.se

System architectures: From Static to Dynamic

DySCAS:
- Run-time reconfigurable applications & resource deployment.
- DySCAS MW itself will have a fixed topology.

Static Architecture | Constrained Dynamic Architecture | Dynamic Architecture → Degree of configuration flexibility

Static Architecture:
- A fixed choice of configuration solutions (e.g., topology) in a setup.
- Only offline configured.

Dynamic Architecture:
- Multiple alternative configuration choices in a setup.
- Run-time reconfiguration in reacting to events or requests.

V&V complexity →

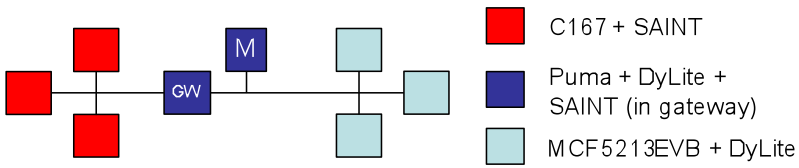
DyLite

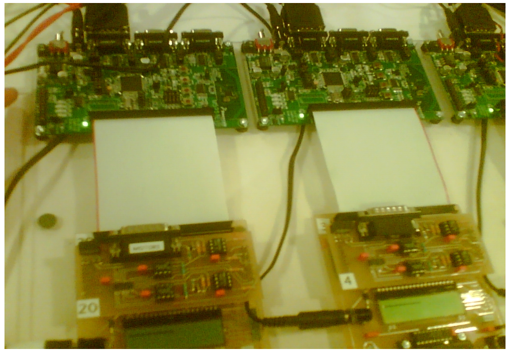
- Proof-of-concept implementation of the DySCAS concepts on QoS and reconfigurability
- Real-world implementation of reconfiguration algorithm [Feng, CDC 2008]
- Design exploration – how small can an implementation along of the DySCAS ideas become?
  - current footprint in slaves: 15 kB
- *Hard* real-time reconfigurability
  - Formal model for QoS
  - Including ability of real-time guarantees through resource reservation
  - Supervision and control of applications’ resource usage

Demonstrator

Ultrasonic sensors
Tachometer
Motor
ECU
ECU
ECU
ECU
Steering
Angle sensor
CAN Bus
Laser
Steering
Scales
ECU
ECU
ECU

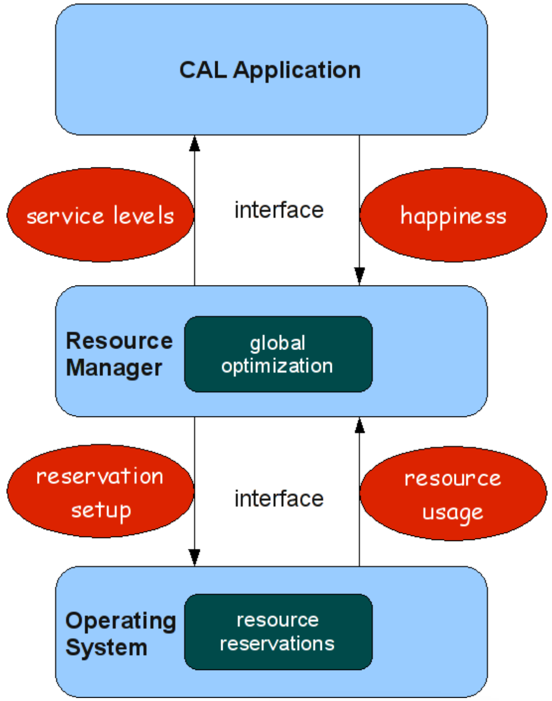
SAINT truck

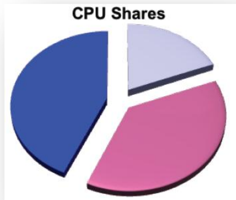




Feedback-Based Resource Management

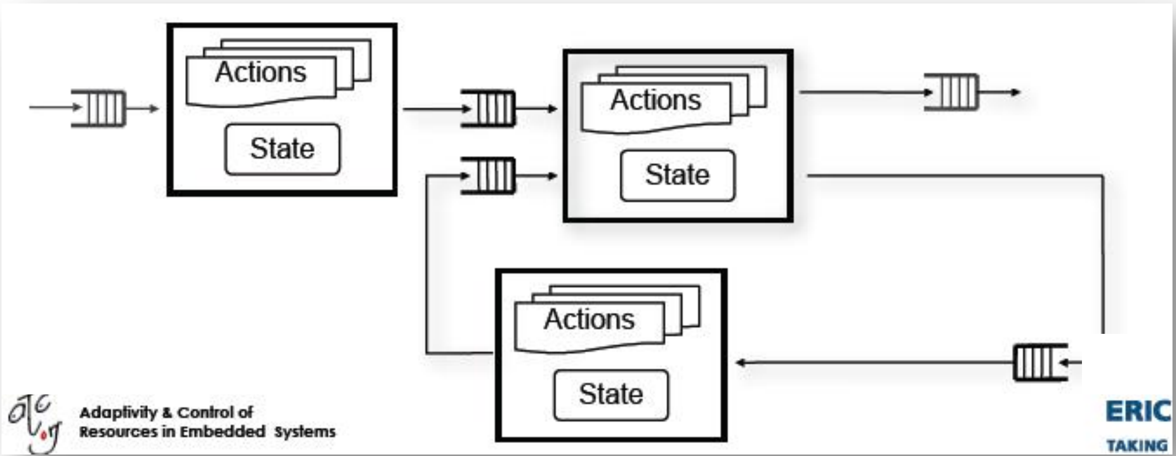
- ACTORS – Adaptivity and Control of Resources in Embedded Systems
  - Ericsson (coord), SSSA, TUKL, Lund, EPFL, Akatech, Evidence
- Levels:
  - Applications: CAL Dataflow Language
  - Resource Manager
  - Operating System: SCHED_EDF Linux scheduler (hard CBS) and Linux CFS schedule
- Demonstrators
  - Media streaming on cellular phones, control, high-performance video
- Platforms: ARM 11 & x86 multicore with Linux > 2.6.26





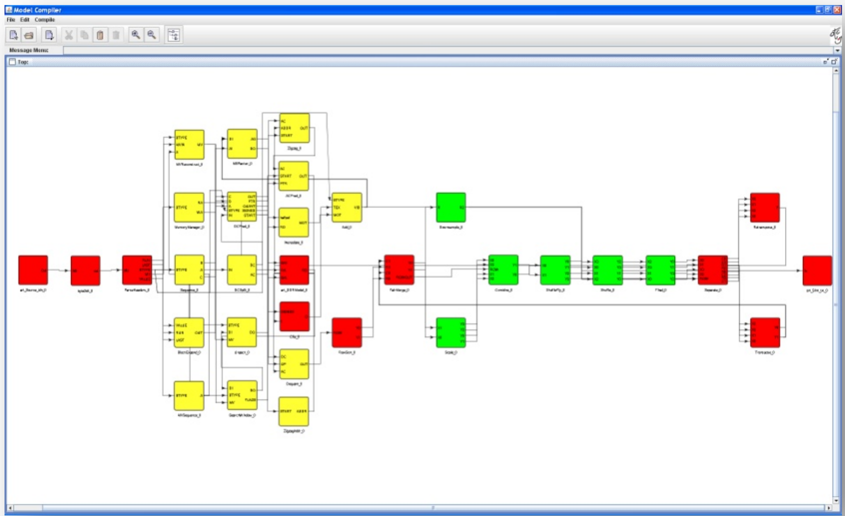
ACTORS: Dataflow Modeling

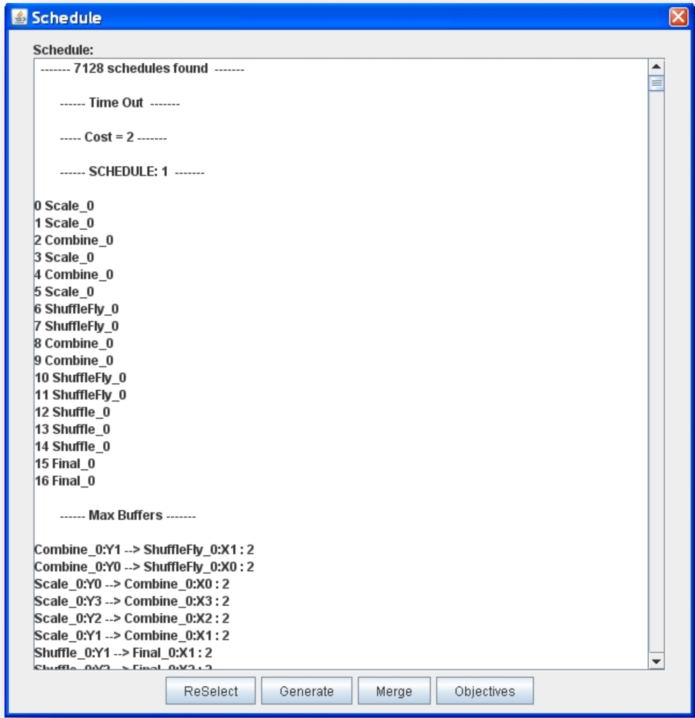
- Data flow programming with actors
  - Associate resources with streams
  - Clean cut between execution specifics and algorithm design
  - Strict semantics with explicit parallelism provides foundation for analysis and model transformation
- CAL Actor Language (UC Berkeley, Xilinx) http://opendf.org
  - Part of MPEG/RVC



ACTORS: Model Transformations

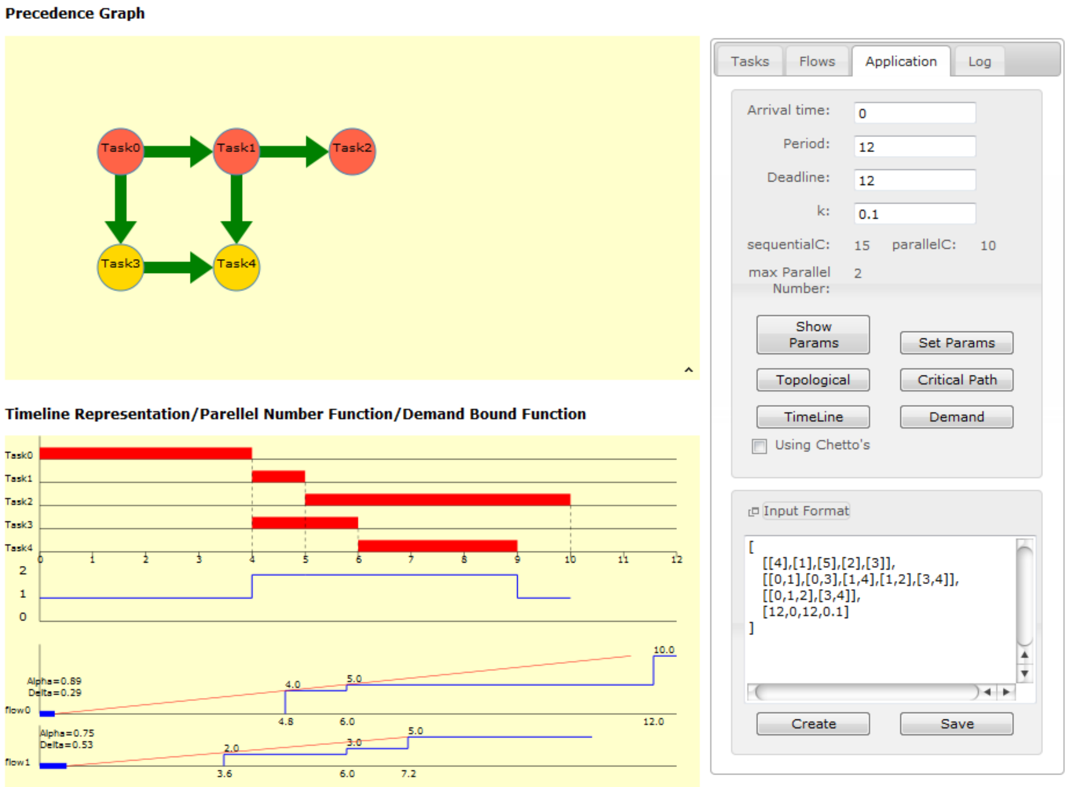
- Merging of actors within statically schedulable regions
  - Tool for classification of actors (statically schedulable, dynamic, time-dependent), selection of statically schedulable subnetworks, calculation of sequential schedules, and merging of actors





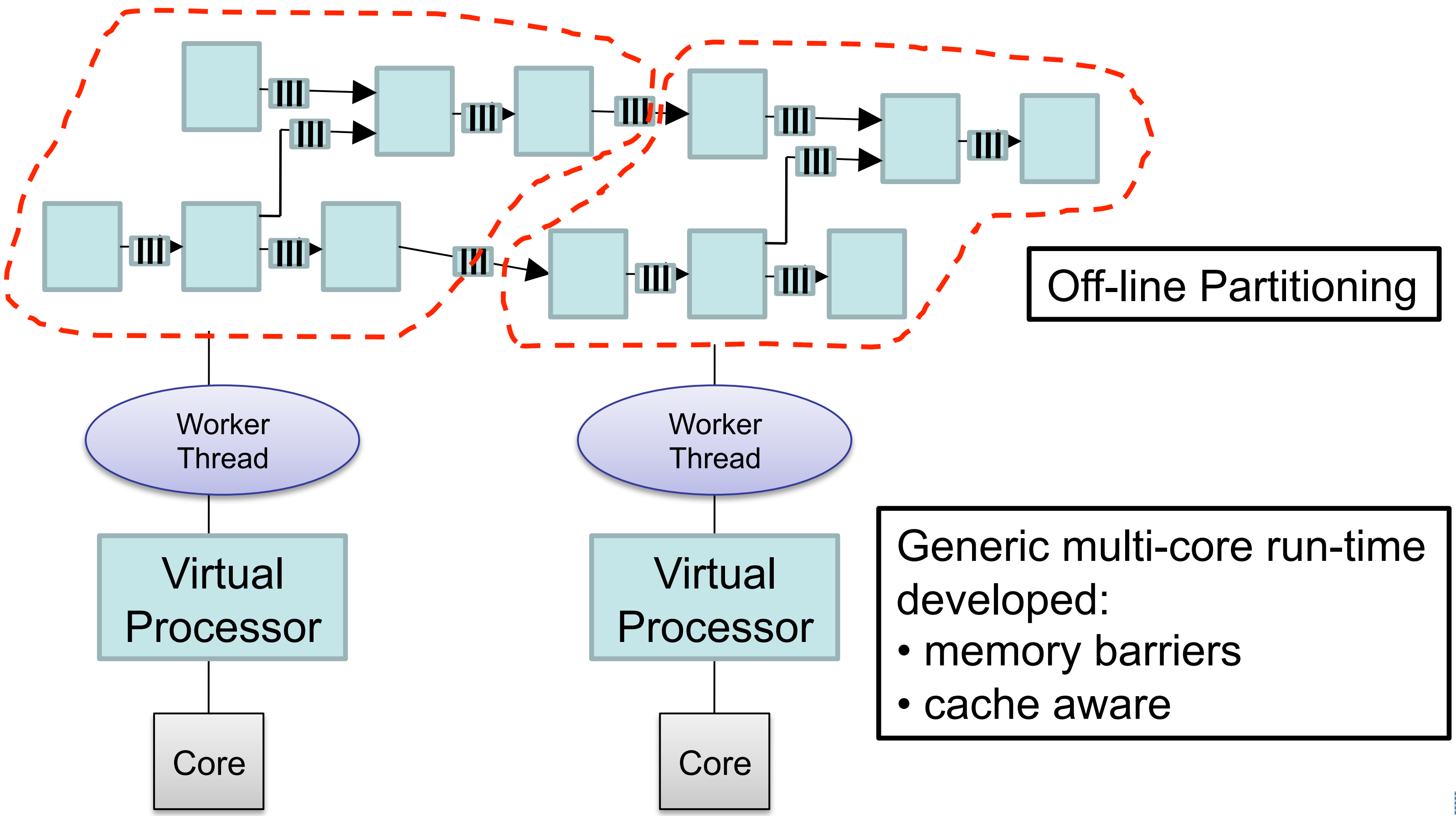
ACTORS: Scheduling

- Off-line schedulability analysis tool for CAL applications that can be translated into static precedence graphs (DAGs)



ACTORS: Scheduling

- Best-effort scheduling with dynamic processor allocation for dynamic CAL applications on multi-core platforms



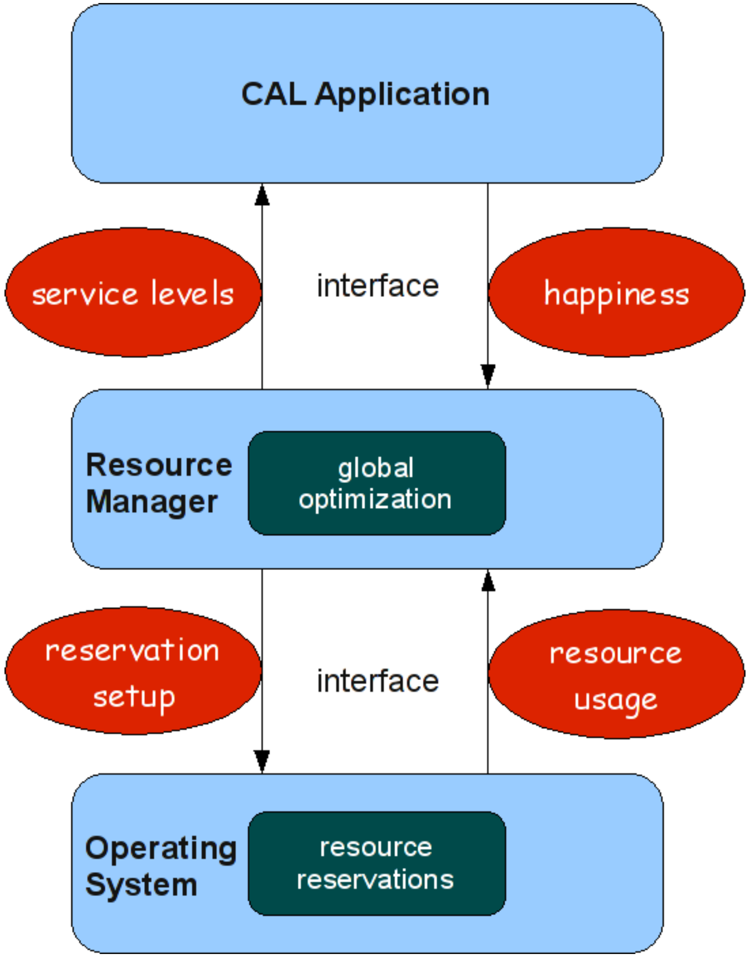
Resource Manager

- Inputs:
  - Service Level Tables

Service Level	Quality Level	Resource 1 Requirement	Resource 2 Requirement	Timing Granularity
1	100	100	1	20 ms
2	70	90	1	10 ms
3	40	30	1	5 ms

  - Application importance
  - Happiness

Appl.	Importance
Appl 1	10
Appl 2	10
Appl 3	100



Resource Manager Tasks

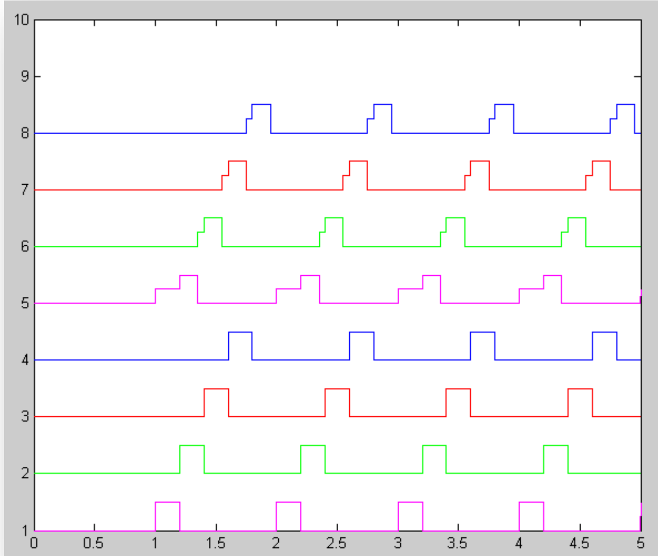
- Service Level Assignment
  - Selects the service levels of the applications taking into account the application importance and the total amount of resources available (total CPU bandwidth)
- Mapping
  - Map the virtual processors to physical processors (cores)
- Bandwidth Distribution
  - Distribute the total bandwidth of each application onto the virtual processors of the application according to some criteria
- Bandwidth Adaptation
  - Adjust the allocated bandwidth for each virtual processor based on measurement of the used bandwidth and application happiness

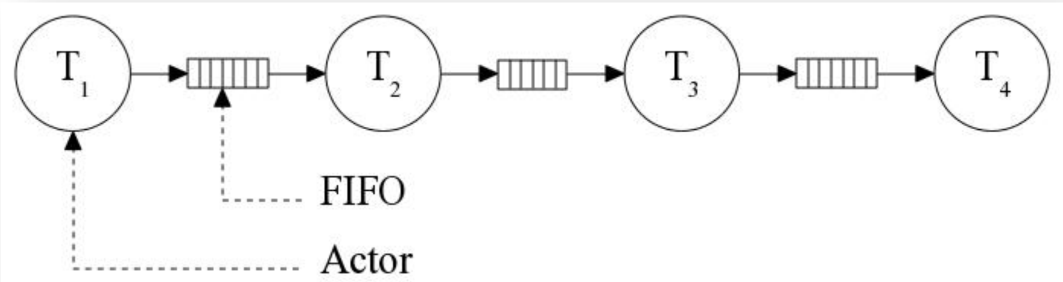
Feedback Control Possibilities

- Feedback only
  - Challenge - how to respect application importance and limited resource availability
- Feedforward + feedback
  - Feedforward = service level assignment and bandwidth distribution solved using ILP optimization
  - Feedback = bandwidth adaptation
  - Challenge - how to handle uncertainties and variations
- Feedback within applications
- Feedback-based partitioning

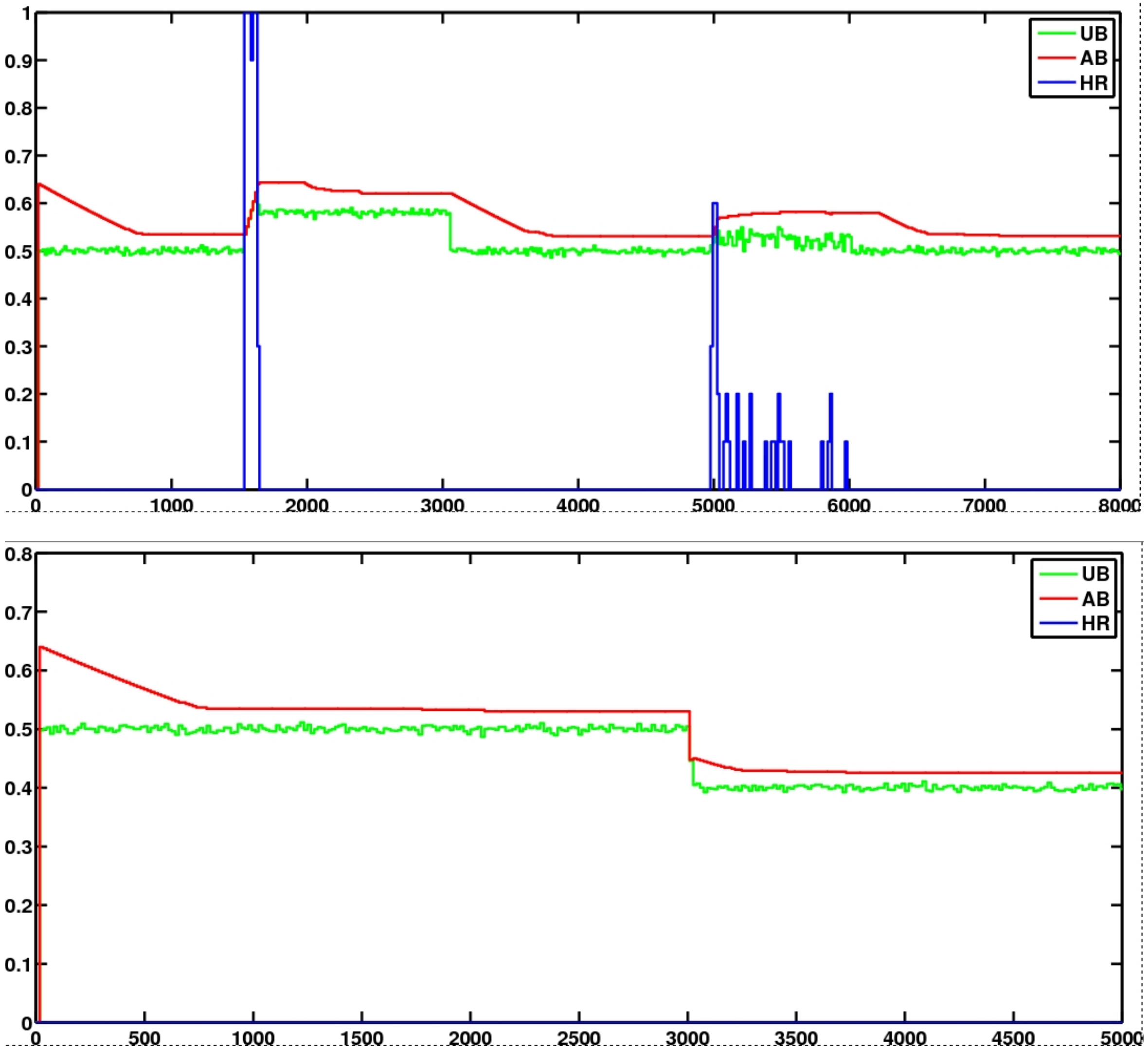
TrueTime Simulations

- In parallel with the C++ implementation of the RM a TrueTime implementation is performed
- TrueTime model:
  - CAL application models
  - Resource manager logic
  - Hard CBS servers and EDF
  - Multiple cores





Bandwidth Adaptation



Plans for Y3

- Continued integration of the work related to adaptivity in ArtistDesign
- At least 10 joint publications
- More than 10 research collaborations
- More than 10 meetings or workshops organized by the partners.
  - WARM 2010, 12 April 2010, Stockholm (CPSWEEK)
    - Workshop on Adaptive Resource Management
    - Integrated with general meeting
  - FeBID 13 April, Paris (EUROSYS)
- Two educational events (incl. the Artist Graduate School on Embedded Control)
- The content of the wiki will be further expanded.
- Guest edit a special issue of J. of Real-Time Systems on Adaptive Embedded System during 2010/2011