Technische Universität Braunschweig

INSTITUTE OF COMPUTER AND NETWORK ENGINEERING

Embedded performance analysis for organic computing (EPOC)

Tobias Michaels, Steffen Stein, Moritz Neukirchner, Rolf Ernst

September 15, 2011

Outline

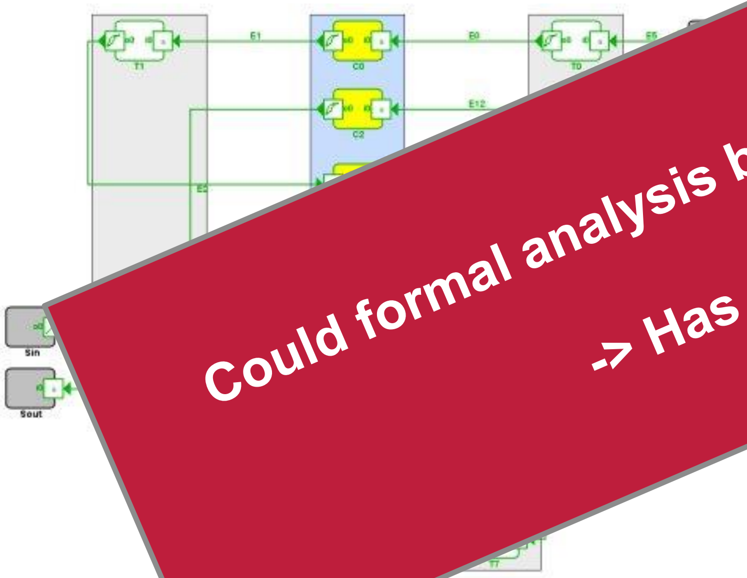
- The past: EPOC – Evolving Critical Systems since 2005
- Current work: In-System Model Exploration
  - Self-Configuration
  - Sensitivity Analysis
- Outlook: Future Building Automation Systems

Outline

- The past: EPOC – Evolving Critical Systems since 2005
- Current work: In-System Model Exploration
  - Self-Configuration
  - Sensitivity Analysis
- Outlook: Future Building Automation Systems

Previous Work -- The Foundations

Available: Formal Performance Analysis Tool (S

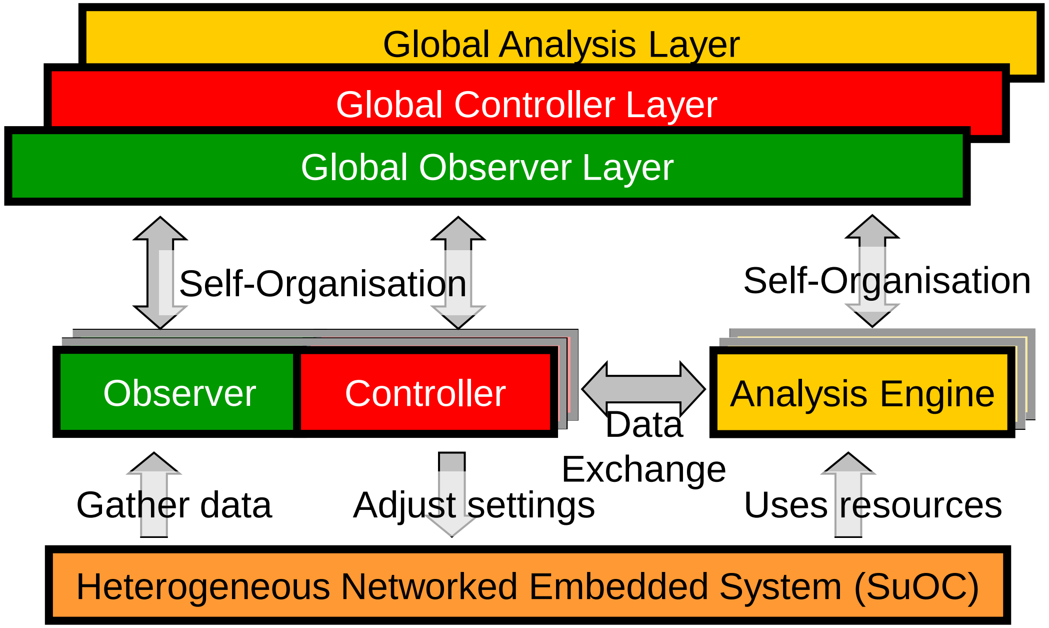


- …ework
- …y SymtaVision 2004
- …ccess Stories in Automotive Industry
- Used for Network and ECU Design at several OEMs and Suppliers

Could formal analysis be used to guide organic systems?

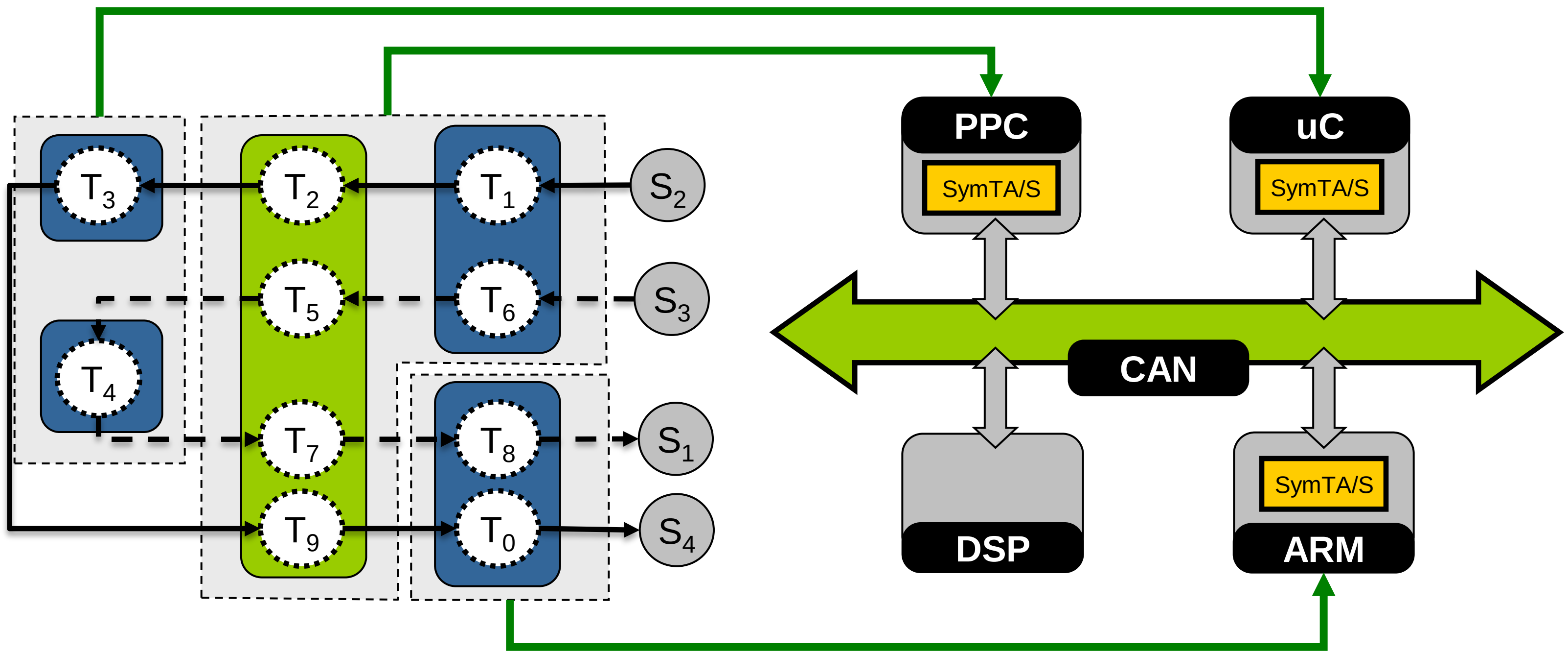
-> Has to be done at runtime!

Phase I: Generic Organic Architecture to do the analysis



- Use Analysis for System Supervision
- Distribute Observation/ Control and Analysis over System to form Control Layer Plane
- Focus on Distributed Analysis Algorithm

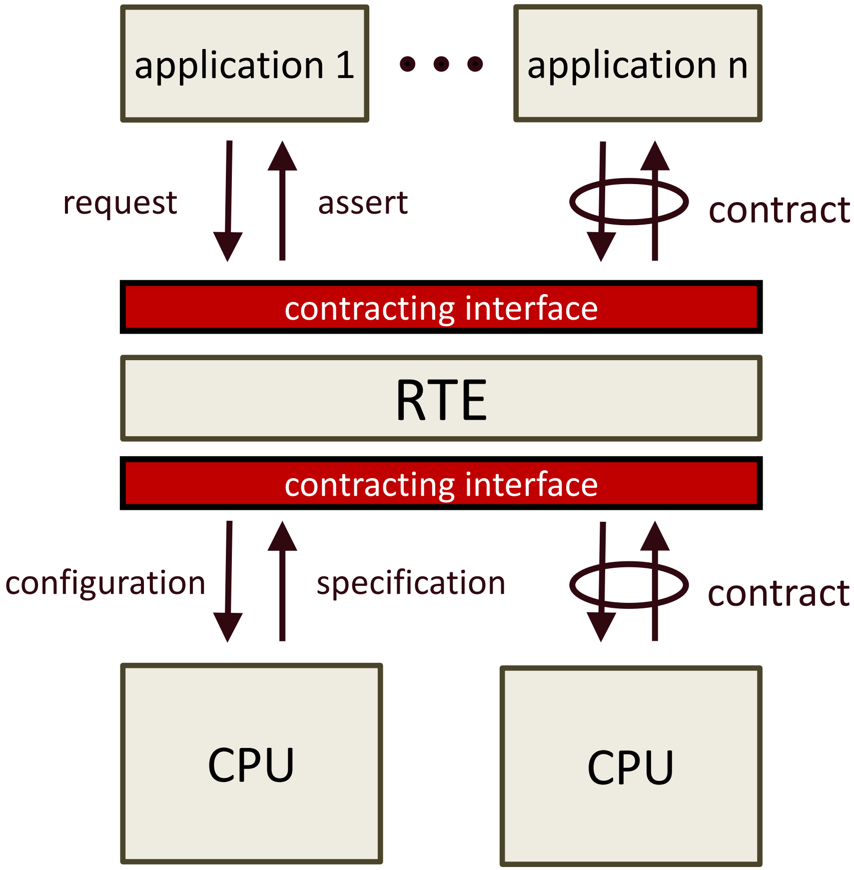
Phase I: Distributed Modeling/ Analysis



Global Model

Real System

Phase II: Use-Case: Contracting at Update Time

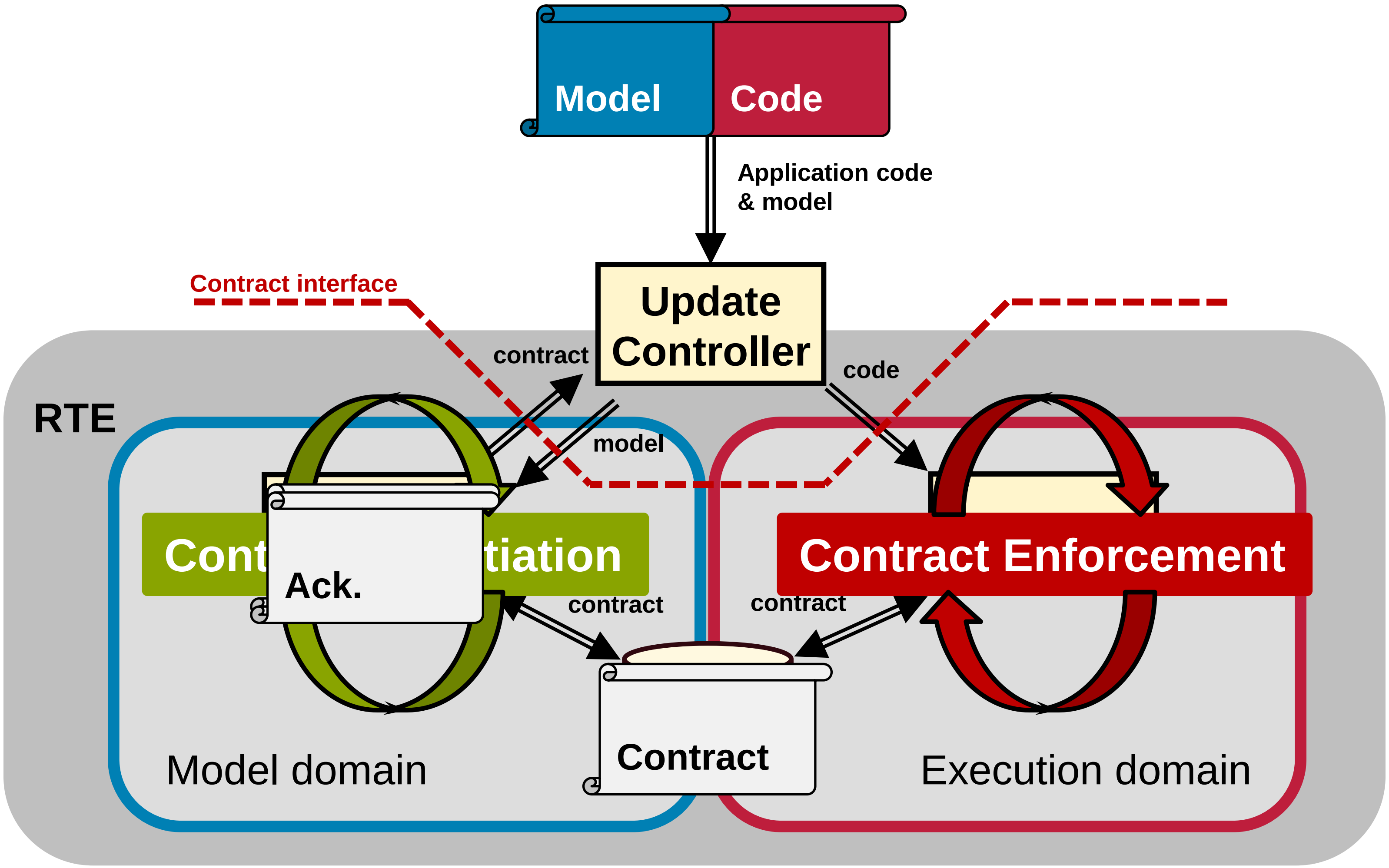


- Software has to **declare** its **behavior**, requirements and constraints
- The **system verifies itself** based on software description and platform capabilities
- The **system supervises adherence** of the software to its descriptions

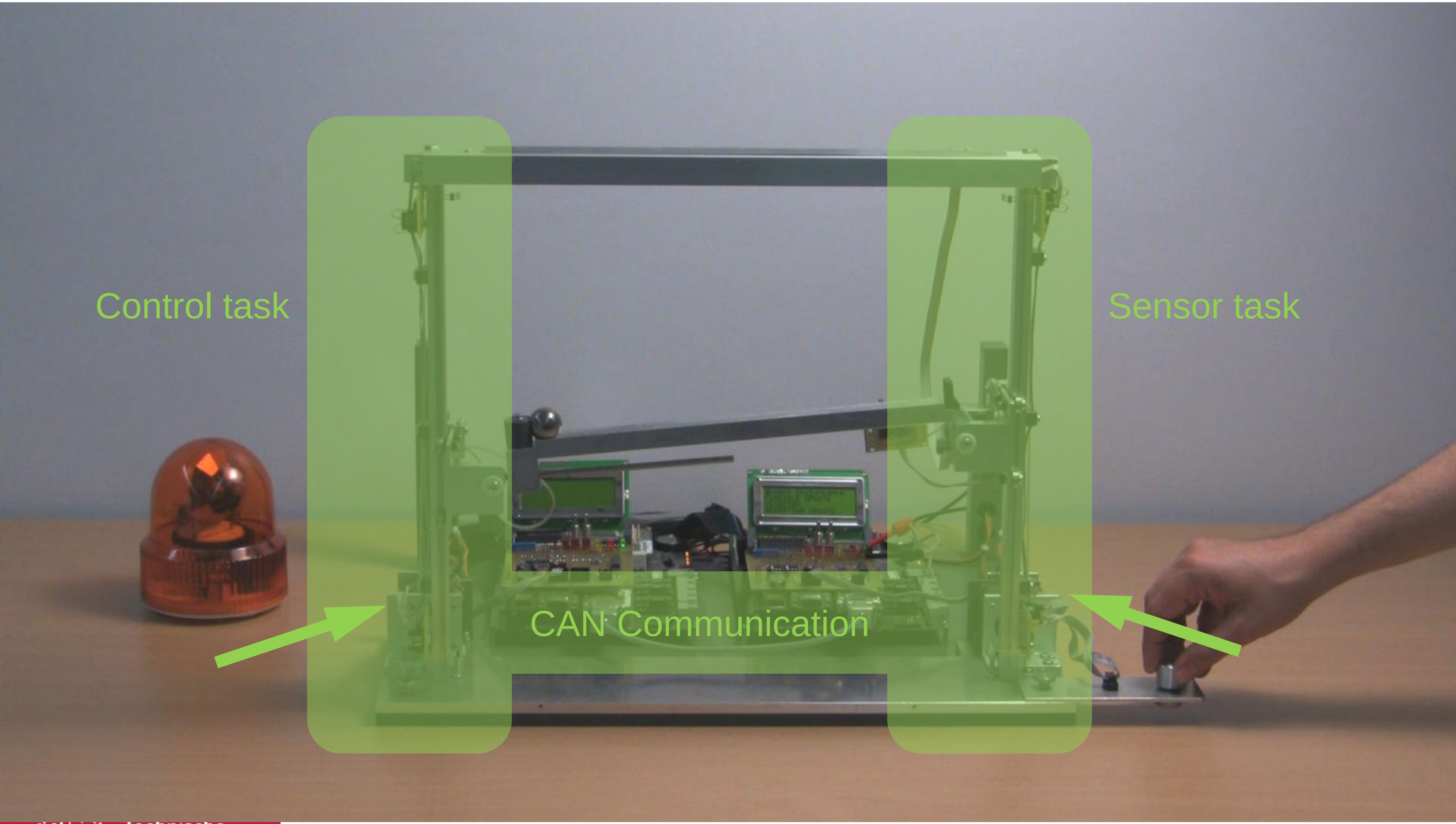
⇒The system protects itself from infeasible changes which enables evolution of critical systems

⇒Building safer systems at update time!

Phase II: Software Architecture



Phase II: Demonstrator



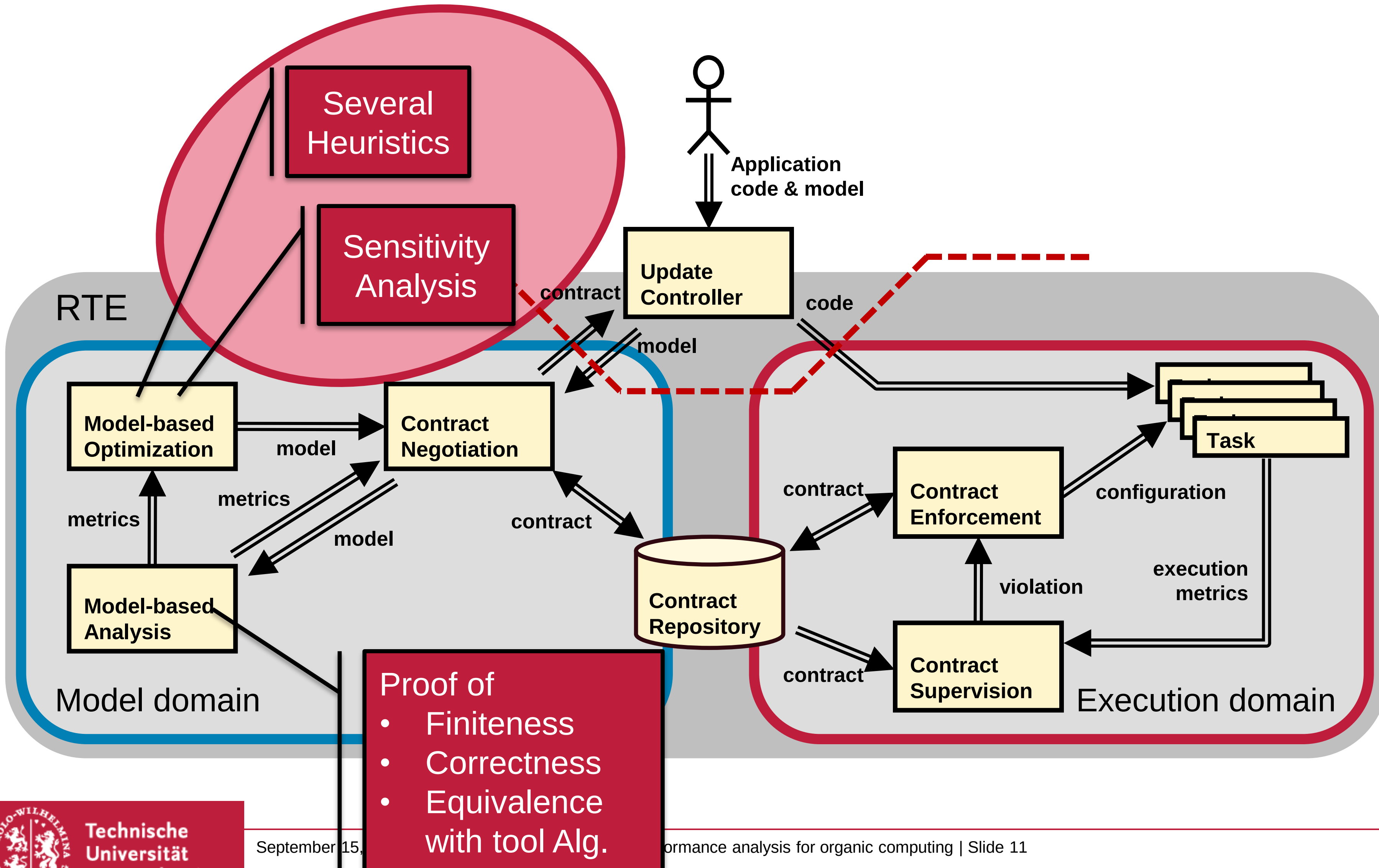
Outline

- The past: EPOC – Evolving Critical Systems since 2005

- Current work: In-System Model Exploration
  - Theoretical Work
  - Self-Configuration
  - Sensitivity Analysis

- Outlook: Future Building Automation Systems

Phase III: Theoretical Background



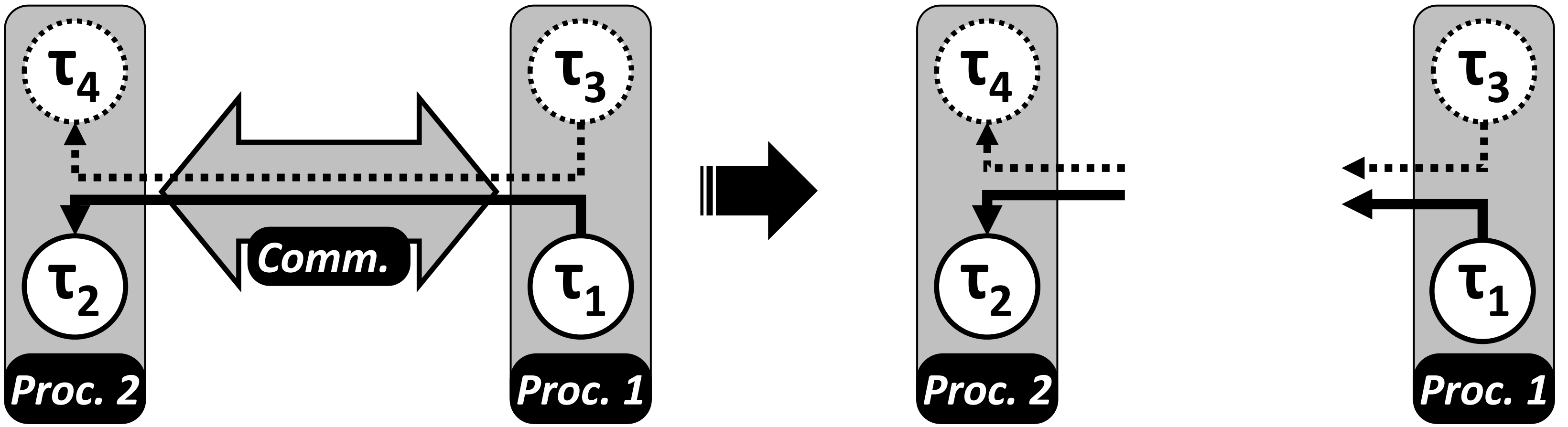
Heuristic Distributed Self-Configuration

Objective:

Find a priority assignment on all processors and busses, such that all end-to-end path latency constraints are satisfied.

Constraint:

As the model is held distributedly in the system, no resource has a complete view of the system model.



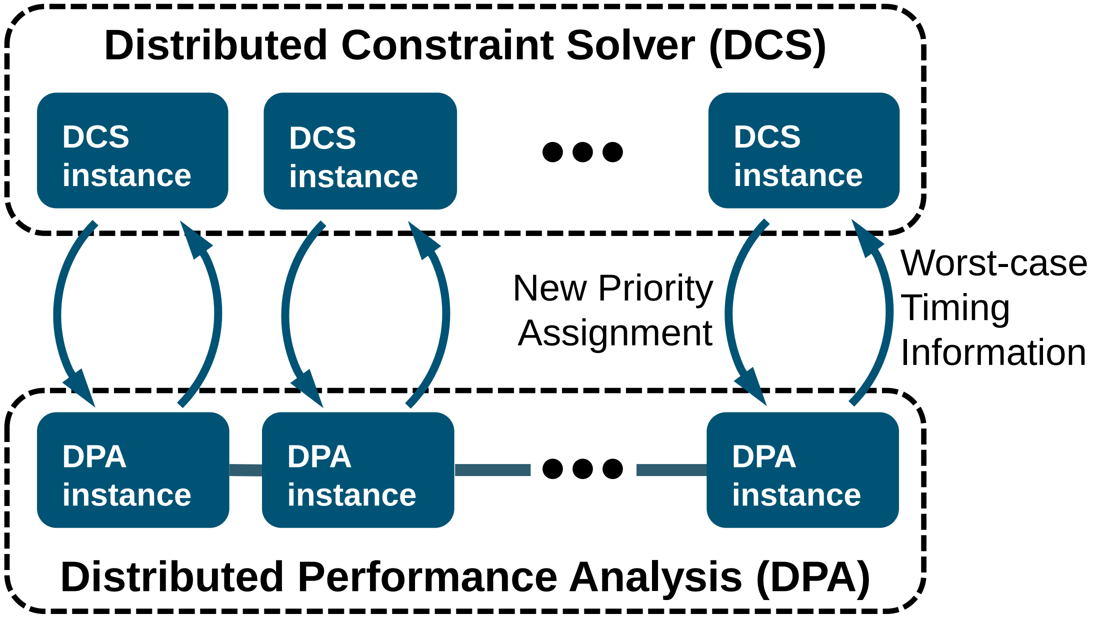
Priorities?

Priorities?

Heuristic Distributed Self-Configuration

Approach:

Extend each instance of the Distributed Performance Analysis (DPA) Algorithm with an instance of a heuristic Distributed Constraint Solver (DCS)



Heuristic Distributed Self-Configuration

- Metric based on path constraint violation and task worst-case response time
  $\Rightarrow$ No explicit communication required between DCS instances, local gradient search

Path latency

Task WCRT

Path latency constraint

$$\delta = \max\left(0, \frac{\omega}{\lambda} * (\lambda - \chi)\right)$$

Path latency share

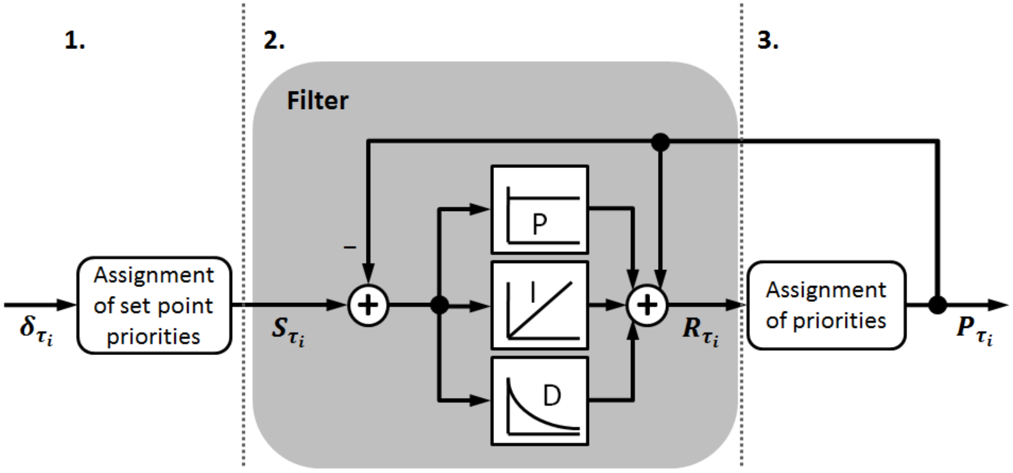
Constraint Violation

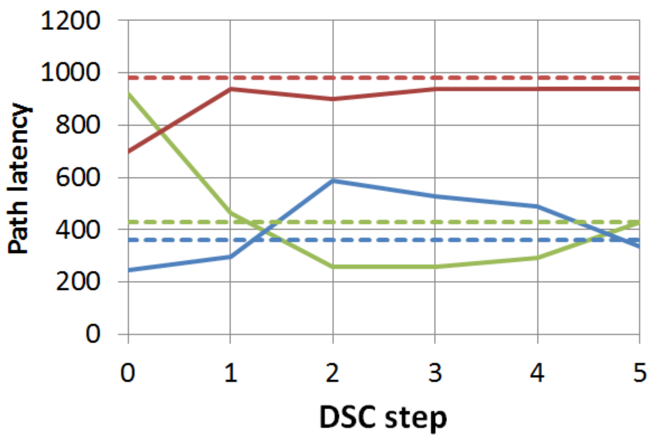
- Oscillations in exploration process can occur in distributed concurrent execution
  $\Rightarrow$ Oscillations have to be damped

Damping Oscillations

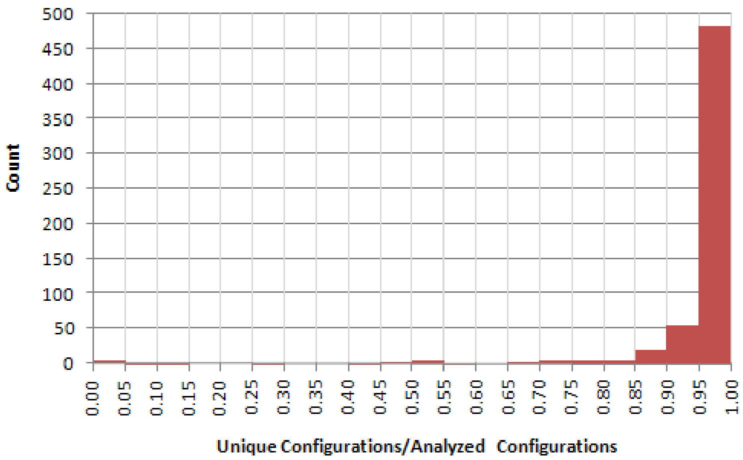
- Naive Solution: Damp oscillations by skipping actions from time to time
  - „Lazy-Algorithm“ (Last Colloquium)

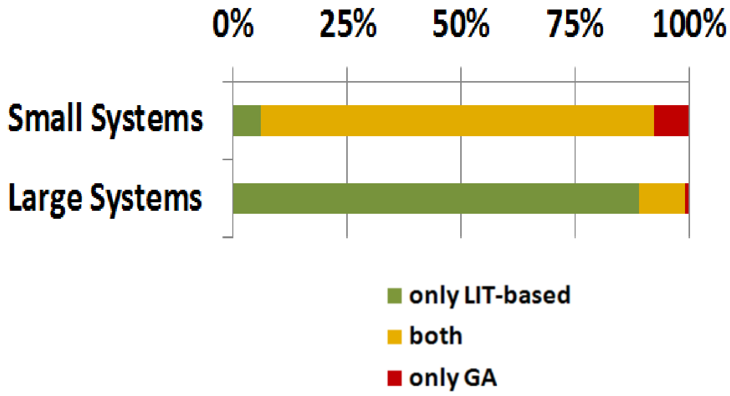
- Sophisticated Solution: Damp oscillations by PID filtering
  - Gain values determined by experiments





Results



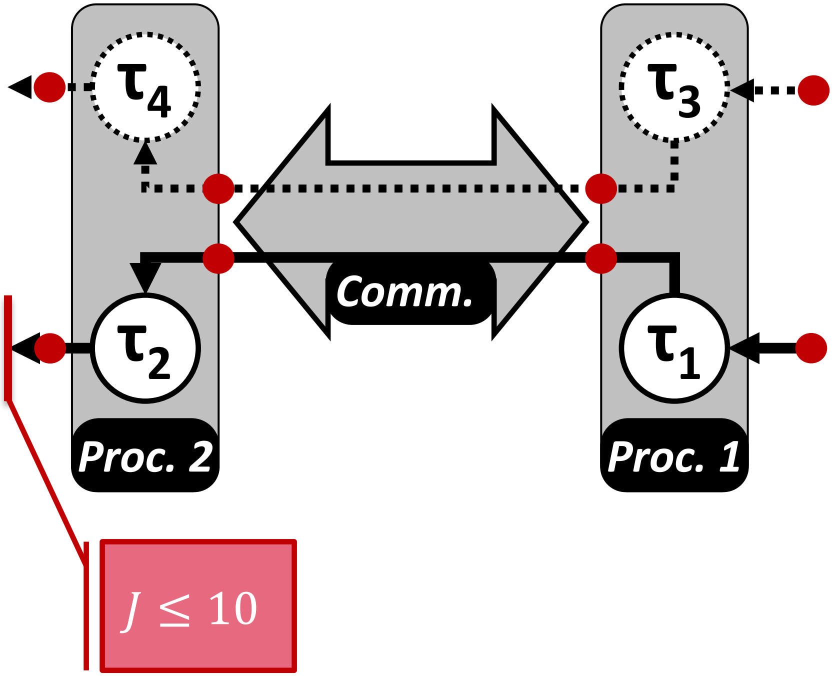


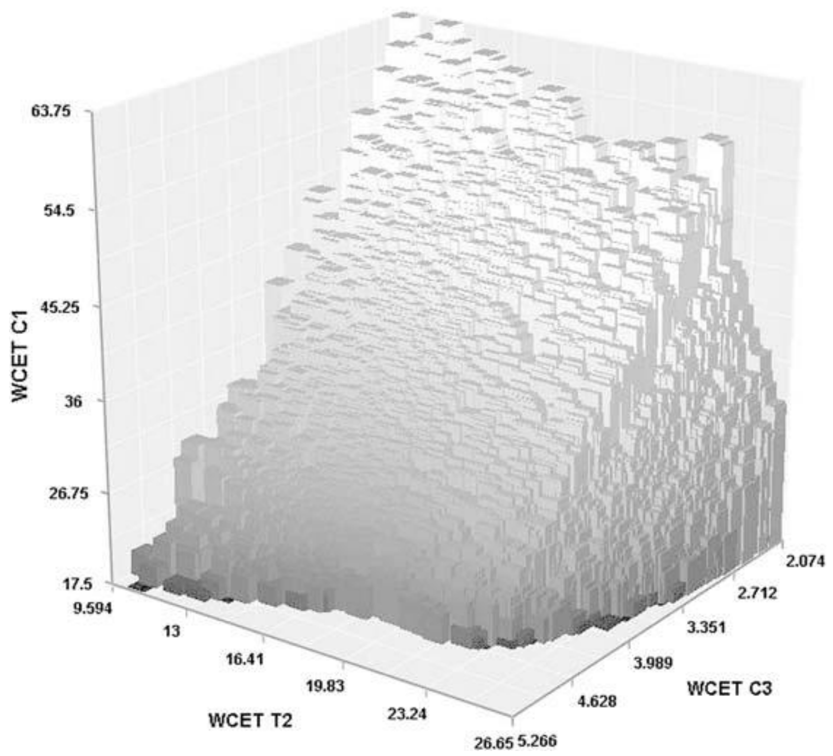
No oscillations

Bounded Computing Time
for large Systems

Deriving Monitoring Bounds – The Problem

- How much input jitter can each task tolerate so that no constraint is violated?
- Multi-Dimensional Pareto-Problem
- Each point can be used for Monitoring



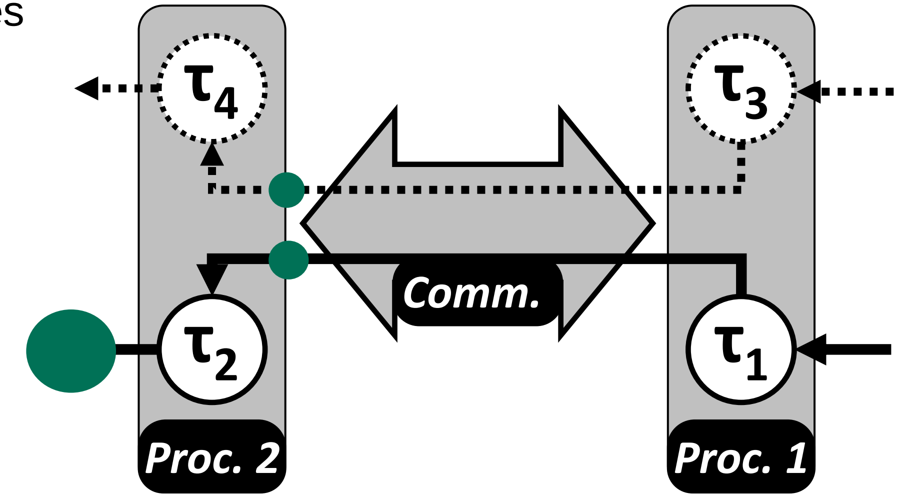


Constructing a Pareto-Point

- Constructive Algorithm
- Greedily Assign available Slack „From Back to Front“

- Theory provided for
  - Correctness
  - Finiteness of Algorithm

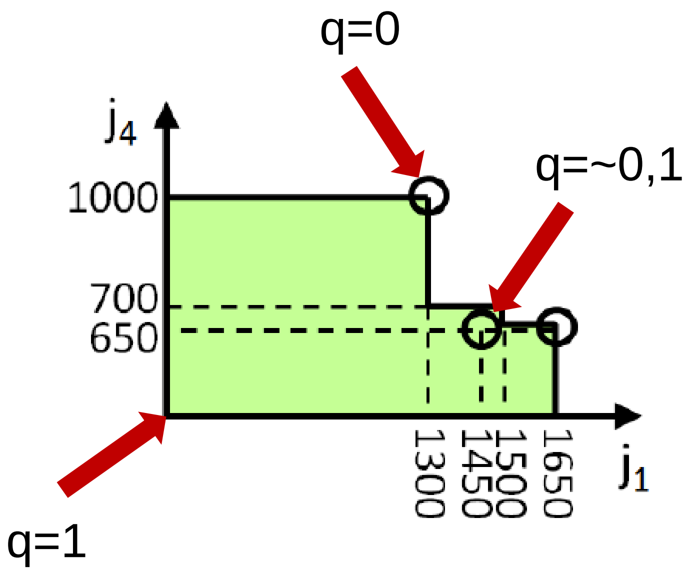
- Extension: Iterative adaptation of Greedyness
- Better slack-utilization for cyclic dependencies

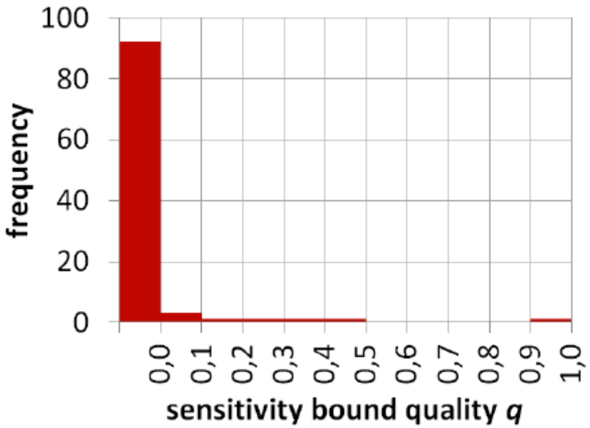


Results

Manhattan-Distance

$$q = \min_{\boldsymbol{G}_p \in \mathcal{G}_p} \left(\frac{|\boldsymbol{G}_p - \boldsymbol{G}_s|_m}{|\boldsymbol{G}_p|_m} \right)$$





EPOC – Result Exploitation

- Evaluated use cases:
  - Automotive
    - many concerns (certification, computational resources, design process)
    - Central Integrator - Manual „Admission Control“ by dozens of engineers and countless hours of testing still viable
  - Smart Buildings:
    - Less complicated, but far more distributed design process
    - No central integrator
    - Very long lifetimes (>20yrs)
    - Permanent adaptation needed
    - Performance Analysis already applied (EkReit / BmBF)
  - Smart Grid
    - Similar Problems, but higher safety requirements

Outline

- The past: EPOC – Evolving Critical Systems since 2005

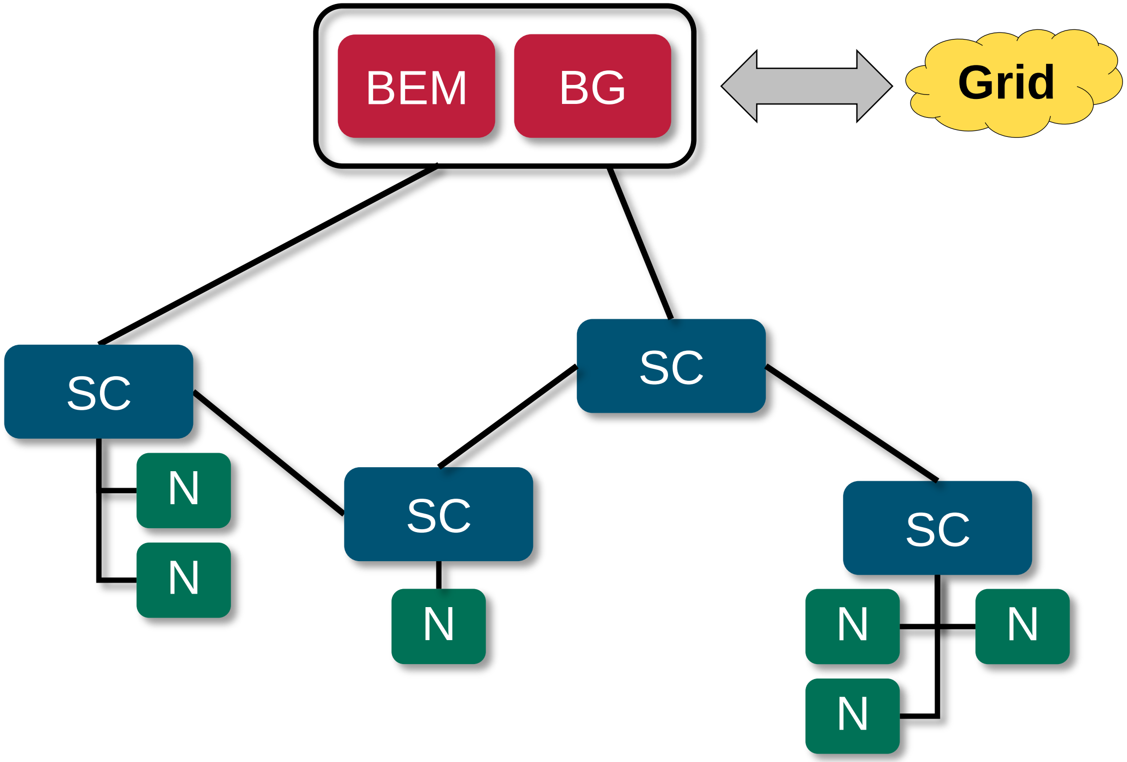
- Current work: In-System Model Exploration
  - Self-Configuration
  - Sensitivity Analysis

- Outlook: Future Building Automation Systems

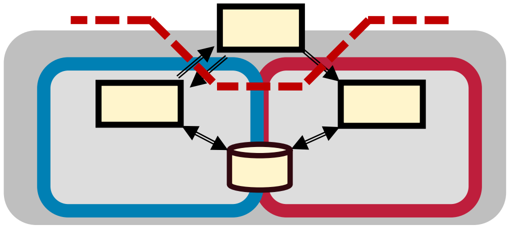
Exploitation: Smart Buildings

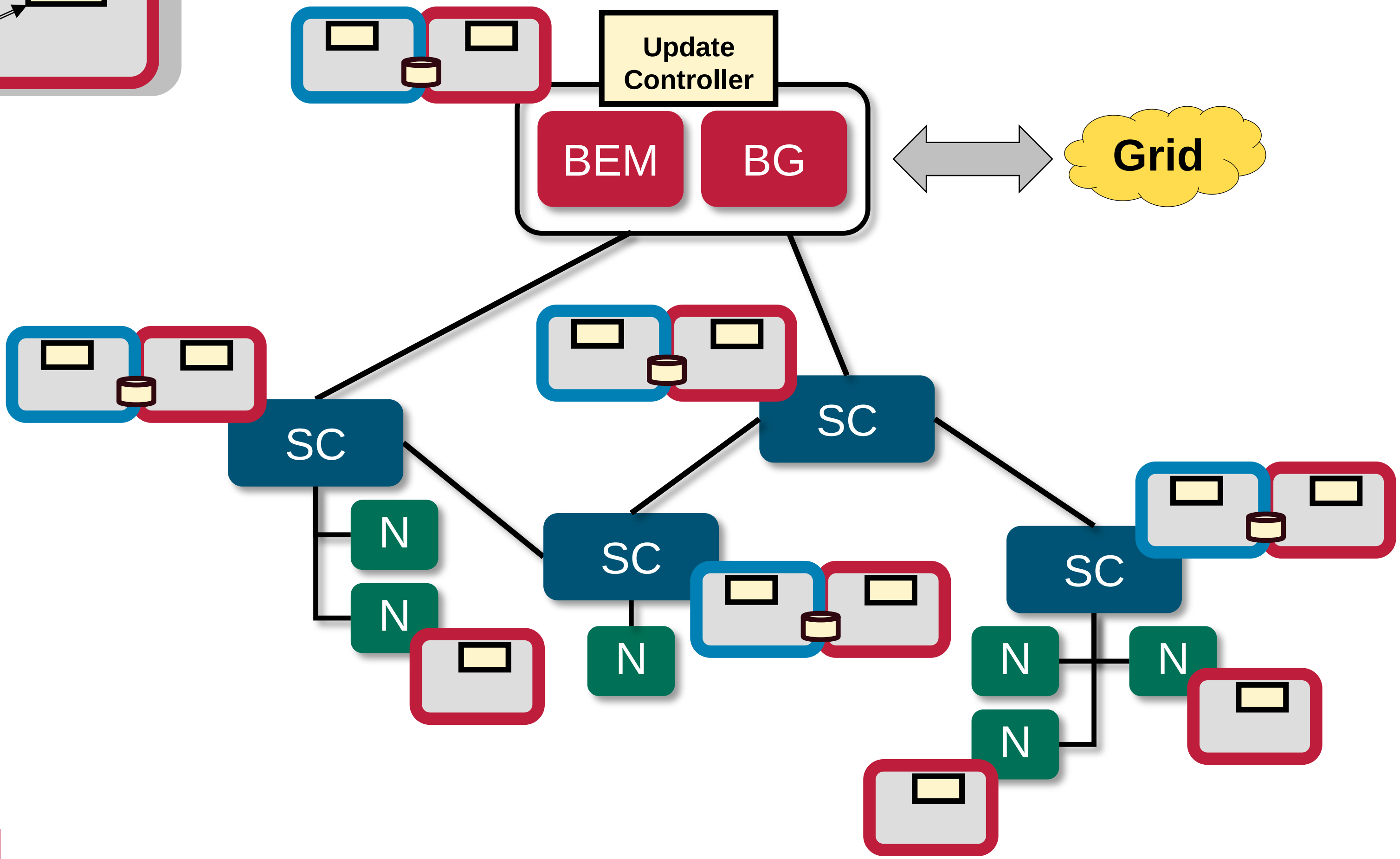
Building Energy Manager

- Coordinating Load / Generation of Smart Buildings
- Frequent Reconfiguration of internal and external networked embedded system
- Several companies want to run software on the BEM
- No central integrator
- Security becomes an issue

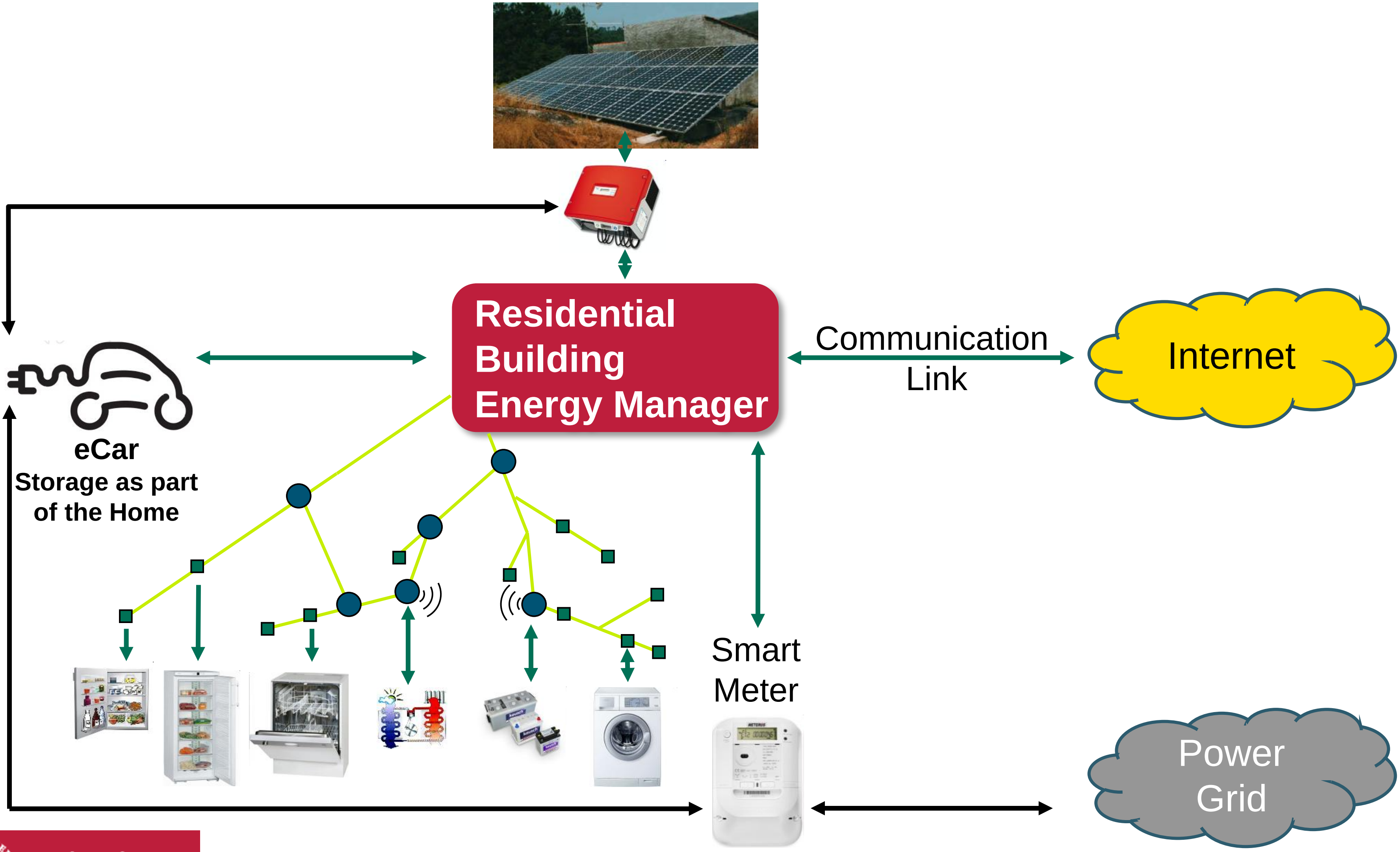


EPOC in Smart Buildings





Application to current project – Internet of Energy



Dissemination

- Mircea Negrean, Moritz Neukirchner, Steffen Stein, Simon Schliecker, und Rolf Ernst, **"Bounding Mode Change Transition Latencies for Multi-Mode Real-Time Distributed Applications"** in *Accepted for Publication in Proc. of Emerging Technologies and Factory Automation (ETFA)*, September 2011
- Moritz Neukirchner, Steffen Stein, und Rolf Ernst, **"SMFF: System Models for Free"** in *2nd International Workshop on Analysis Tools and Methodologies for Embedded and Real-time Systems (WATERS)*, (Porto, Portugal), July 2011
- Steffen Stein, Moritz Neukirchner, und Rolf Ernst, **"Admission Control and Self-Configuration in the EPOC Framework"** in *Proc. of International Conference on Embedded Computer Systems: Architectures, Modeling, and Simulation (SAMOS XI)*, July 2011
- Moritz Neukirchner, Steffen Stein, Harald Schrom, Johannes Schlatow, und Rolf Ernst, **"Contract-based Dynamic Task Management for Mixed-Criticality Systems"** in *6th IEEE International Symposium on Industrial Embedded Systems (SIES)*, June 2011
- Moritz Neukirchner, Steffen Stein, und Rolf Ernst, **"A Lazy Algorithm for Distributed Priority Assignment in Real-Time Systems"** in *Proc. of 2nd IEEE Workshop on Self-Organizing Real-Time Systems*, No. 126-132, May 2011

Dissemination

- Harald Schrom, Tobias Michaels, Steffen Stein, und Rolf Ernst, **"SmallCAN - A Reliable, Low-Power and Low-Cost Distributed Embedded System for Energy Efficient Building Automation"** in *Energy2011*, May 2011
- Moritz Neukirchner, Steffen Stein, und Rolf Ernst, **"The EPOC Architecture - Enabling Evolution under Hard Constraints"** in *Organic Computing - A Paradigm Shift for Complex Systems* (Christian Müller-Schloer and Hartmut Schmeck and Theo Ungerer, Ed.), chapter 4, Birkhäuser Science, 2011
- Steffen Stein, Matthias Ivers, Jonas Diemer, und Rolf Ernst, **"A polynomial time algorithm for computing response time bounds in static priority scheduling with convex event models"** in *Euromicro Conference on Real-Time Systems (ECRTS'10)*, July 2010
- Steffen Stein, Moritz Neukirchner, Harald Schrom, und Rolf Ernst, **"Consistency Challenges in Self-Organizing Distributed Hard Real-Time Systems"** in *Workshop on Self-Organizing Real-Time Systems (SORT)*, May 2010
- Moritz Neukirchner, Steffen Stein, Harald Schrom, und Rolf Ernst, **"A Software Update Service with Self-Protection Capabilities"** in *Proc. of Design, Automation, and Test in Europe (DATE)*, (Dresden, Germany), March 2010

Dissemination

- Steffen Stein and Rolf Ernst, **"Distributed Performance Control in Organic Embedded Systems"** in *IEEE 5th International Conference on Autonomic and Trusted Computing (ATC-08) Autonomic and Trusted Computing (LNCS)*, vol. 5060/2008 of series Lecture Notes in Computer Science, pp. 331-342, Springer Berlin / Heidelberg, June 2008
- Simon Schliecker, Steffen Stein, und Rolf Ernst, **"Performance Analysis of Complex Systems by Integration of Dataflow Graphs and Compositional Performance Analysis"** in *Proc. of Design, Automation and Test in Europe (DATE)*, April 2007
- Steffen Stein, Arne Hamann, und Rolf Ernst, **"Real-time Property Verification in Organic Computing Systems"** in *Proc. of the 2nd International Symposium on Leveraging Applications of Formal Methods, Verification and Validation (ISoLA-06)*, November 2006
- Steffen Stein, Arne Hamann, und Rolf Ernst, **"Real-time Management in Emergent Systems"** in *36. Jahrestagung der Gesellschaft für Informatik*, (Dresden, Germany), October 2006

Conclusion

- EPOC framework was developed
  - Distributed Performance Analysis incl. proofs
  - Software Architecture (Model / Execution Domain)
  - Distributed Constraint Solver with several heuristics
  - Methods to derive monitoring bounds
  - Demonstrator built

- Future use in building automation systems / smart grid
  - Internet of Energy (Artemis project, 42 partners, 40 million €)
    - Architecture Re-Use
    - Additional Analysis Challenges

Thank you for your attention!

Detailed EPOC Architecture