

Engineering of Adaptive Communication Systems

Prof. Dr.-Ing. Reinhard Gotzhein

Computer Science Department
University of Kaiserslautern

**Signature de l'accord de coopération Franco-Allemande en informatique
INRIA Lorraine**

7 Février 2007

Topics

- **Motivation**
- Model-driven Design
- Model-driven Implementation
- Model-driven Performance Simulation
- Conclusions

Motivation

- Wireless Ad-hoc Networks -

- **Characteristics**

- set of small, low-power nodes: scarce resources (energy, bandwidth, CPU power, memory)
- wireless communication: shared medium, frame collisions, hidden stations, varying channel quality, asymmetrical links
- dynamic formation: no infrastructure, decentralized control, nodes within range
- mobile nodes: dynamic topology, network partitioning and merging

- **Implications**

- adaptive route establishment: robustness against mobility
- adaptive QoS mechanisms: QoS monitoring and QoS scaling
- adaptive bandwidth assignment: hierarchical coding of payload data
- adaptive networked applications: robustness against fluctuating communication services
- extensive feedback cycles: cross-layer integration of functionalities

Motivation

- Networked System „Airship“ -



Objective

- development of remotely controllable devices for deployment in disaster areas

Features

- remote control of the airship by joystick
- transmission and projection of video streams

Focus

- adaptive QoS mechanisms
- cross-layer integration

Status

- fully operational prototype
- several demonstration flights

Motivation

- Networked System „Assisted Bicycle Trainer“ -



Objective

- improvement of training effects of a group of cyclists

Features

- collection and communication of cyclist status data during training
- trainer dynamically monitors and adjusts training parameters

Focus

- adaptive QoS mechanisms
- robustness against mobility

Status

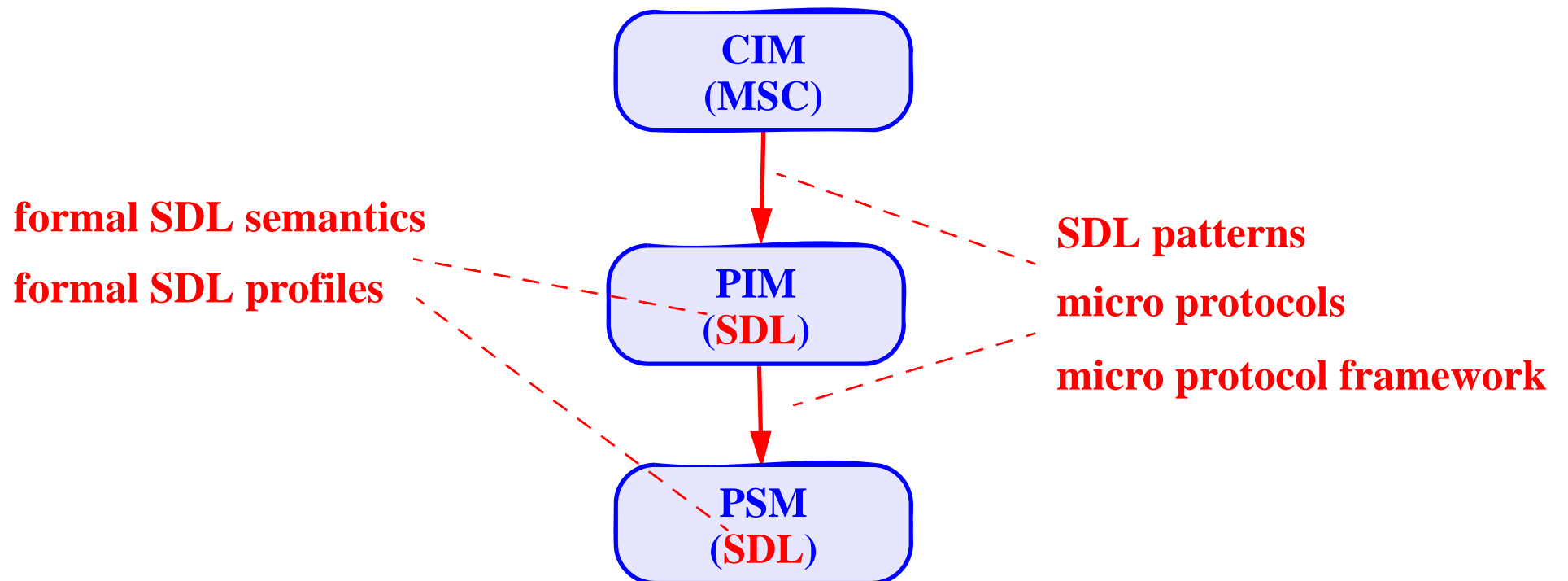
- fully equipped bicycles, based on Berkeley notes
- training sessions about to start

Topics

- Motivation
- **Model-driven Design**
- Model-driven Implementation
- Model-driven Performance Simulation
- Conclusions

Model-driven Design

- Process Model -



Model-driven Design

- Specification and Description Language (SDL) -

- SDL = FSA++
 - SDL system = set of asynchronously interacting extended FSAs
 - explicit hierarchical system architecture
 - explicit connectivity of system components
 - data types and context variables
 - transition triggers (input, priority input, spontaneous transitions, etc.)
 - control structures
 - timers
 - exceptions
 - hierarchical states
 - object-orientation
 - reuse
 - ...

Model-driven Design

- SDL Patterns -

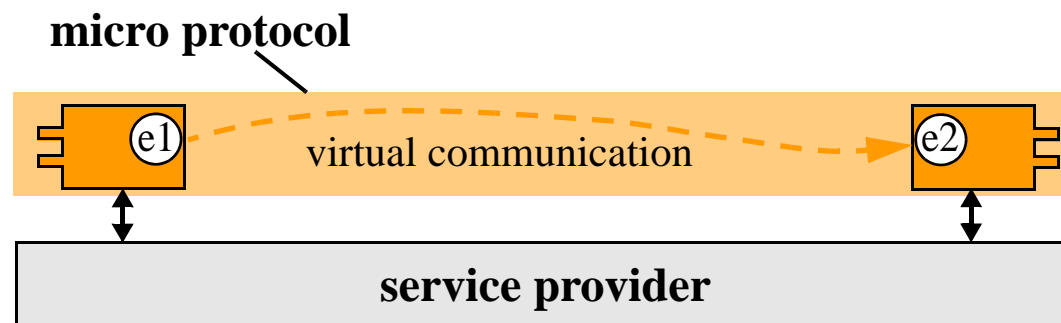
An **SDL pattern** is a reusable software artefact, representing a generic solution for a recurring design problem, with SDL as design language. An SDL pattern is selected from a pattern pool, adapted to a specific context, and embedded into that context.

- **SDL pattern approach**
 - process model, description template, PA-SDL, pattern pool
- **Applications**
 - reengineering of Internet protocols (e.g., ST2+, IPv6, RTP)
 - development of specialized communication systems (e.g., CAN-CSS, airship, UMTS-RNC)
- **Tool support**
 - SPT - The SDL Pattern Tool

Model-driven Design

- Micro Protocols -

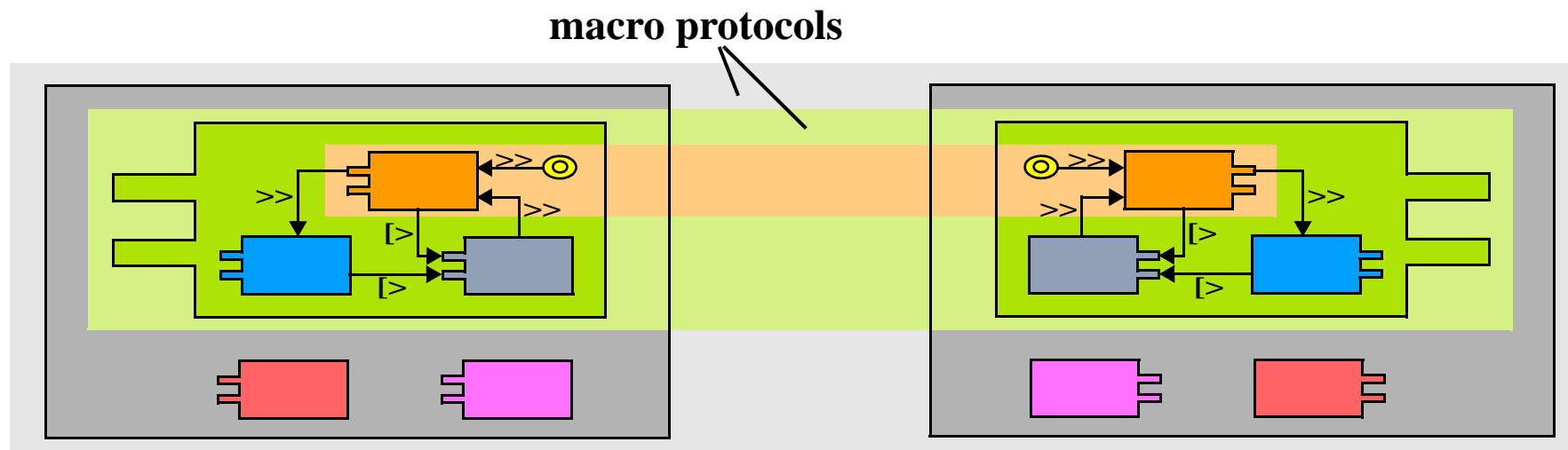
- A **micro protocol** is a communication protocol with a single (distributed) functionality and the required protocol collaboration.
 - **protocol functionality**: e.g., flow control, loss control, resource reservation
 - **protocol collaboration**: synchronization/causality relationships of protocol instances



Model-driven Design

- Micro Protocol Framework -

- A **generic micro protocol framework** is a set of general principles and rules for the composition of micro protocols.
 - **generic composition:** concurrent, sequential („pipelining“), disrupting, hierarchical
 - **specific composition:** additionally based on internal aspects of a micro protocol

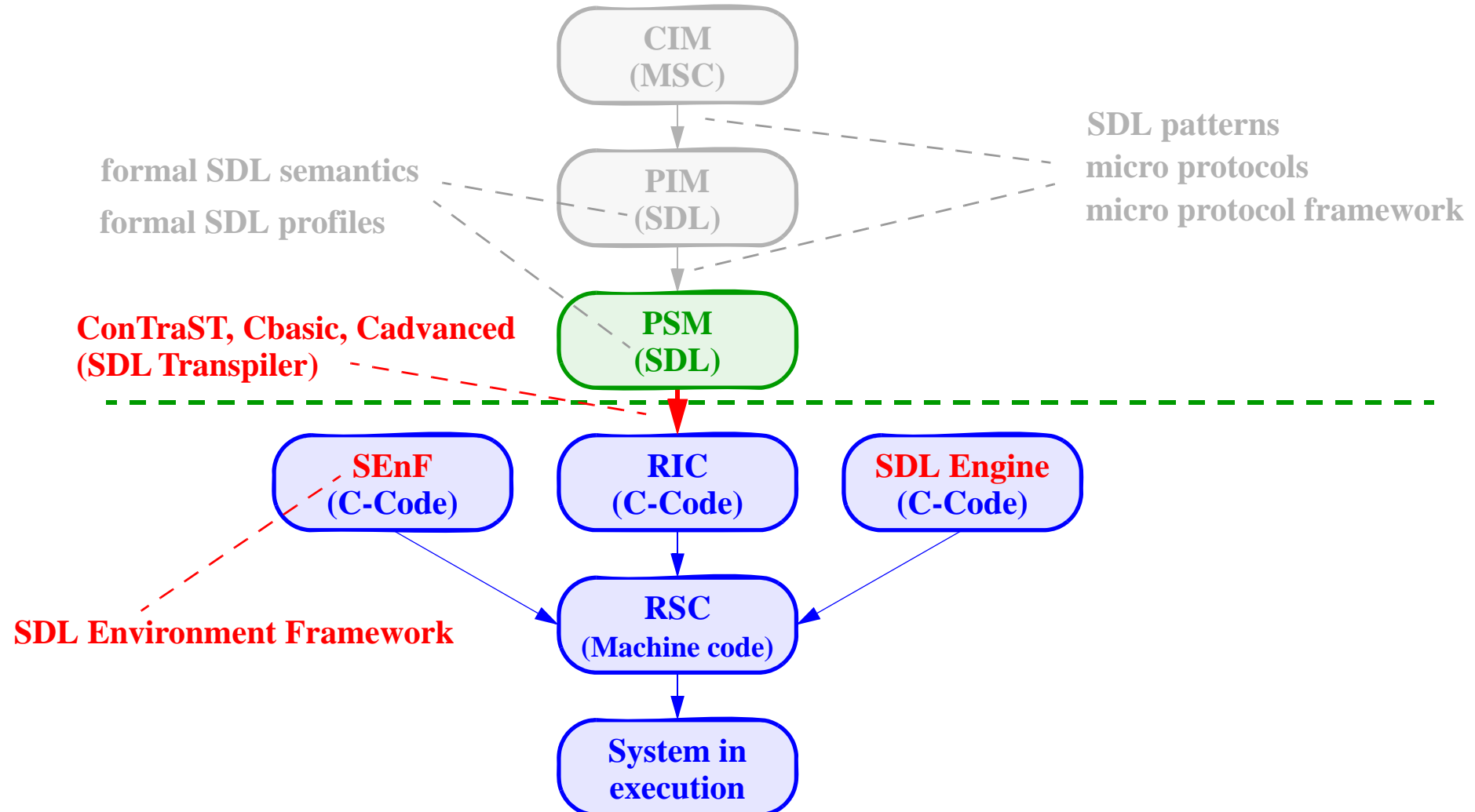


Topics

- Motivation
- Model-driven Design
- **Model-driven Implementation**
- Model-driven Performance Simulation
- Conclusions

Model-driven Implementation

- Process Model -



Model-driven Implementation

- Tool Support -

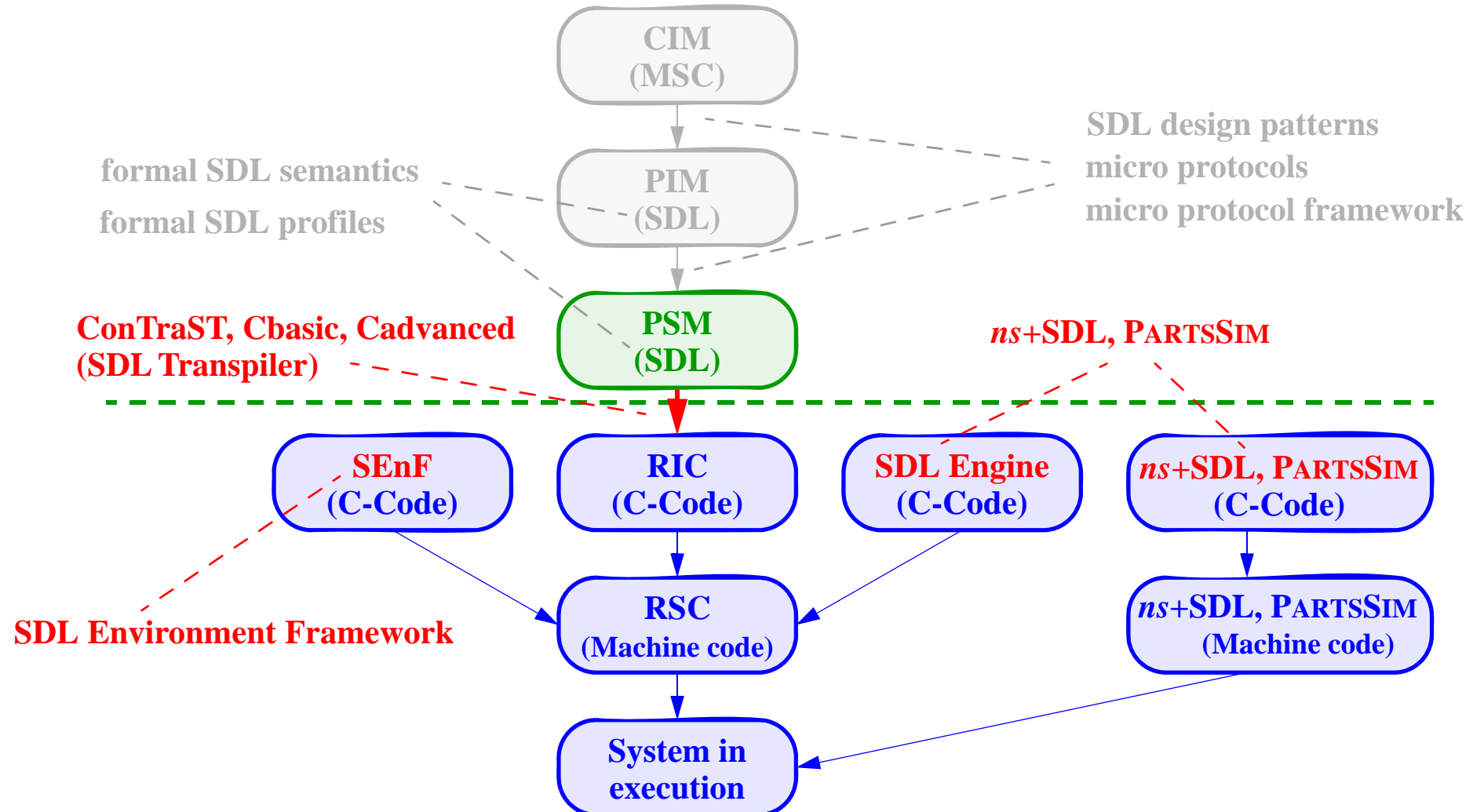
- **SDL Transpiler**
 - generate runtime-independent code (RIC) from the SDL model
 - e.g., ConTraST, Cadvanced, Cbasic
- **SDL Engine**
 - SDL runtime environment
 - SDL transition scheduler
- **SDL Environment Framework (SEnF)**
 - interfacing SDL systems with hardware devices (e.g., IEEE 802.11a/b/g (WLAN), IEEE 802.15.1 (Bluetooth), IEEE 802.15.4 (ZigBee), RS-232 (UART), LEDs, Netcam)
 - interfacing SDL systems with operating systems (e.g., Linux, Windows NT/2000/XP)

Topics

- Motivation
- Model-driven Design
- Model-driven Implementation
- **Model-driven Performance Simulation**
- Conclusions

Model-driven Performance Simulation

- Process Model -



Model-driven Performance Simulation

- Tool Support: *ns+SDL* and *PARTSSIM* -

- **Early: *ns+SDL* and *PARTSSIM* are simulators for platform-specific models (PSM)**
 - platform-specific models in SDL are transpiled to C
 - simulator-specific SDL Environment and SDL Engine are linked to the generated code
 - executable loaded by *ns+SDL* and *PARTSSIM*
- **Faithfully: SDL model as basis both for production system and simulations**
 - the same compilers are used for the generation of production and simulation code
 - accurate resource and environment models (e.g., propagation models, platform models)
- **Reusable: simulation scripts**
 - environment model and movement patterns separated from the core simulation script

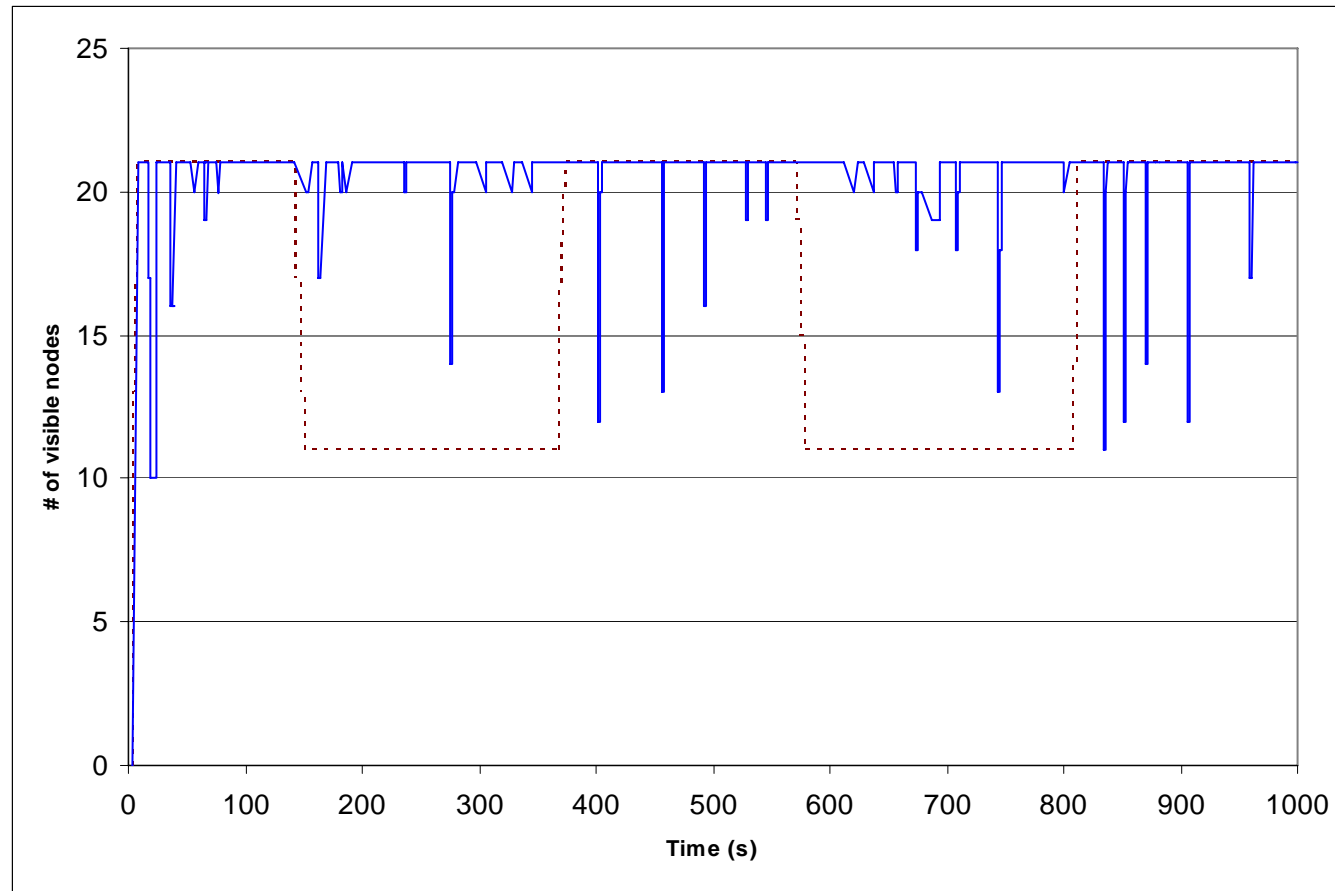
Model-driven Performance Simulation

- Performance Simulation Studies with *ns*+SDL and PARTSSIM -

- **Routing protocols for ad hoc networks**
 - AODV (Ad-Hoc On-Demand Distance Vector)
 - DSDV (Destination-Sequenced Distance Vector)
 - NXP/MPR (Neighbourhood Exchange Protocol / Multipoint Relaying)
- **Adaptive QoS protocols for ad hoc networks**
 - Bandwidth adaptation of application data streams with enhanced best effort
- **Adaptive networked applications based on ad hoc networks**
 - Assisted Bicycle Trainer
 - Airship control and video transmission

Model-driven Performance Simulation

- Assisted Bicycle Trainer: Adaptive Route Establishment -

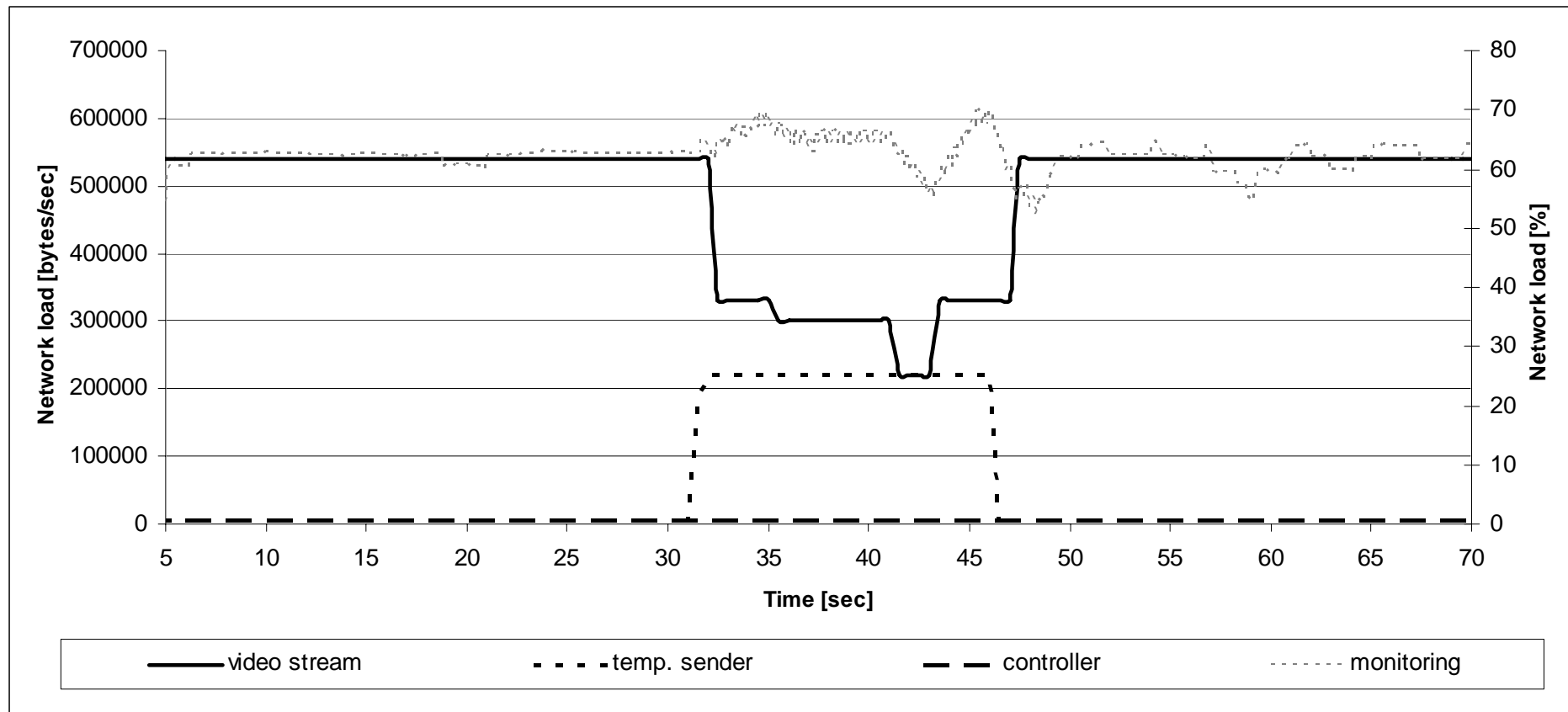


..... local broadcast

—— global broadcast (NXP/MPR)

Model-driven Performance Simulation

- Airship: Bandwidth Adaptation with Enhanced Best-Effort -



Topics

- Motivation
- Model-driven Design
- Model-driven Implementation
- Model-driven Performance Simulation
- **Conclusions**

Conclusions

- **Long term adaptivity**
 - development of specialized communication systems
 - reuse techniques, product line engineering
- **Medium term adaptivity**
 - learning algorithms
 - dynamic change between algorithms
- **Short term adaptivity**
 - parameter based
 - dynamic tuning

