Technische Universität Wien

An Encapsulated Communication System for Integrated Architectures

Architectural Support for Temporal Composability

Roman Obermaisser

Real-Time Systems Group

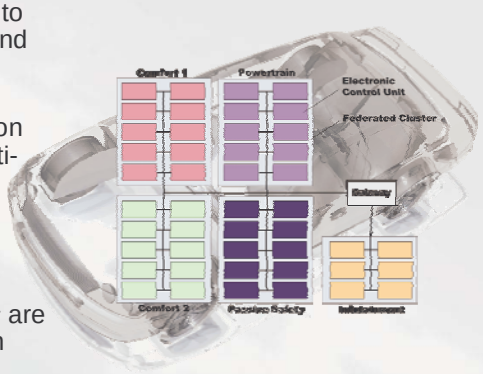
Technische Universität Wien

Overview

- Introduction
- Federated and Integrated Architectures
- DECOS Architecture
- Encapsulated Virtual Networks
- Automotive Example
- Experimental Evaluation

Introduction

- *Federated architectures* have lead to high numbers of deployed nodes and communication networks
  - dedicated distributed computer systems for individual application subsystems (e.g., comfort, multi-media, powertrain, passive safety domain in a car)
  - "1 Function – 1 ECU" design philosophy
- As a result *integrated architectures* are gaining more and more momentum (e.g., IMA, AUTOSAR, DECOS)



Introduction (2)

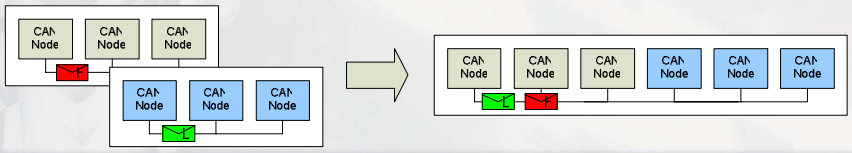
- *System complexity* in distributed embedded real-time systems causes increasing cost of design, verification, integration and maintenance
- *Time-triggered networks* widely accepted as communication infrastructure for safety-critical applications (e.g., aerospace, currently introduced in automotive domain)
- Foundation for integrated system architectures that improve resource utilization, coordination of application subsystem, and complexity management

Federated and Integrated Architectures

- Federated architectures provide each application subsystem with its own dedicated computer system
  - Natural separation of application subsystems
  - Complexity control
  - Fault isolation between computer systems
  - Service optimization
- Integrated architectures support multiple application subsystems within a single distributed computer system
  - Reduced hardware cost
  - Dependability
  - Flexibility

Challenge in Moving Towards the Integrated Architectural Paradigm

- Inherent application complexity
- Accidental complexity through integration-induced interference between application subsystems
  - example: integration of two CAN-based application subsystems
  - invalidation of prior services

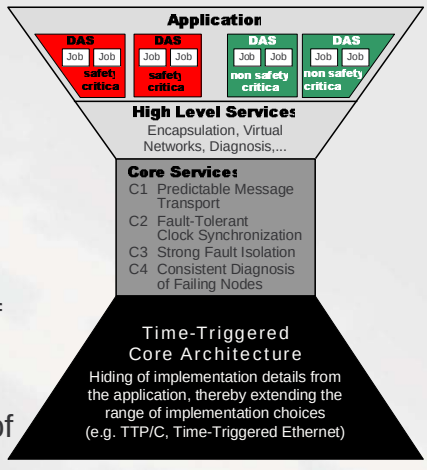


Temporal Composability

- Divide-and-conquer strategies reduce the mental effort for understanding large systems using subsystems that can be developed and analyzed in isolation
- Requirement of a framework for smooth integration and reuse of independently developed components is needed in order to increase the level of abstraction in the design process
- Notion of composability refers to the stability of component properties across integration
- Temporal composability
  - instantiation of the general notion of composability
  - temporal correctness is not refuted by the system integration

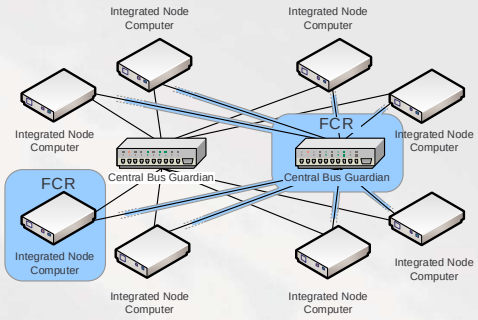
DECOS Architecture

- Distributed Application Sub-systems (DASs)
  - nearly independent distributed subsystem
  - exploit specific platform services
- A DAS consists of a number of jobs interacting cooperatively
- Virtual network as the communication infrastructure of a DAS



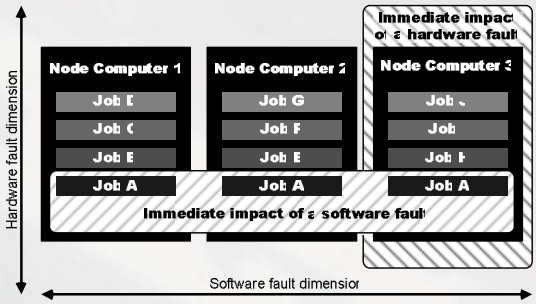
Fault Hypothesis – Hardware Faults

- **Fault containment region:**
  - complete node computer due to shared physical resources (e.g., processor, memory, power supply, oscillator)
  - communication channel
- **Failure mode assumption:**
  - arbitrary node failure
  - no spontaneous generation of correct frames by a communication channel



Fault Hypothesis – Software Faults

- **Fault containment region:**
  - jobs
  - system software considered to be free of design faults
- **Failure mode assumption:**
  - communication system: arbitrary value and timing message failures
  - execution environment: arbitrary timing and value failures



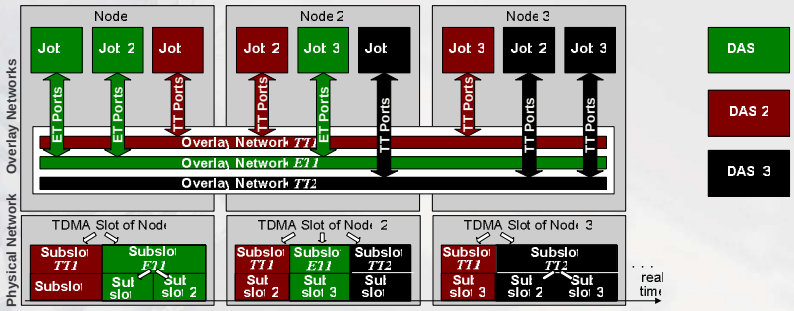
Encapsulation in the DECOS Architecture

- Computational resources and communication resources
  - *partitions* within a node computer with hardware support (e.g., multi-core processors) and software support (e.g., operating system)
  - *virtual networks* with guaranteed temporal properties (bandwidths, latencies)
- Temporal partitioning and spatial partitioning

Virtual Networks

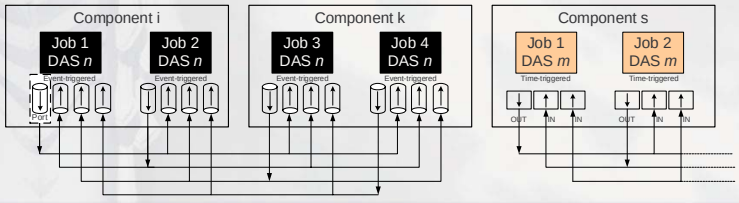
- Overlay network on top of a time-triggered physical network
- Communication according to requirements of a particular DAS (e.g., bandwidth, control paradigm)
- Time-triggered virtual networks for safety-critical DASs
  - periodic broadcast of state messages
  - bounded latency and jitter
- Event-triggered virtual networks for non safety-critical DASs
  - sporadic exchange of event messages
  - emulation of existing event-triggered protocols (e.g. CAN)
  - flexibility

Realization of Virtual Networks



Partitioning of Comm. Resources

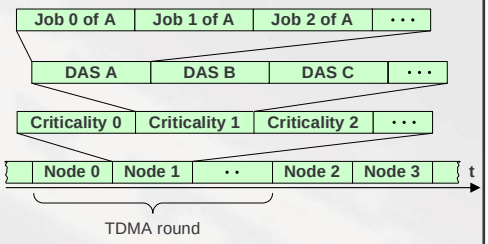
- *Sender-centric view*: non interference of the message transmissions between sender jobs, while abstracting over interference between message transmissions from the same sender job.
- Separate input ports
  - independent queuing delays
  - no spatial interference



Partitioning of Comm. Resources (2)

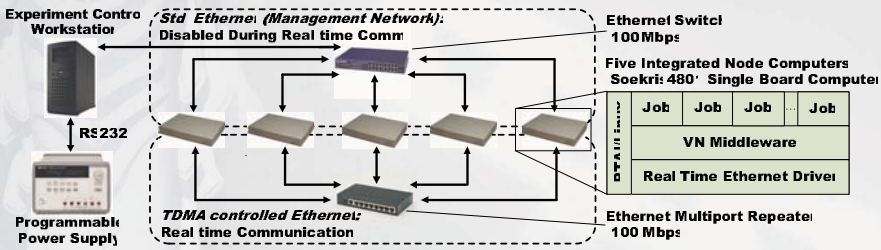
Protection of statically reserved slots in the underlying TDMA scheme

- Protection between nodes by time-triggered communication protocol (e.g., local or central guardian in TTP or FlexRay)
- Protection within a node, e.g., using virtual network middleware
  - encapsulation of criticality domains by protecting criticality-domain slots
  - encapsulation of DAS by protecting DAS slots
  - encapsulation of jobs by protecting job slots



Implementation and Experimental Evaluation

- Prototype implementation of DECOS architecture
- Time-triggered communication protocol: Ethernet with TDMA scheme
- Evaluation of partitioning at communication system using 20,000 testruns

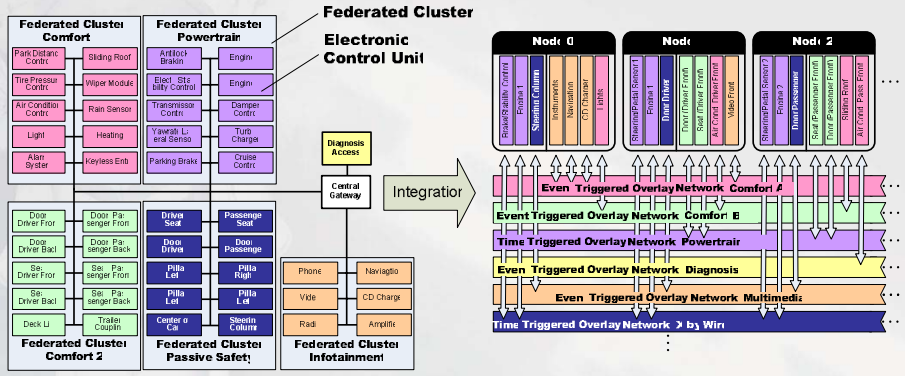


Automotive Example

- SAE classification of in-vehicle networks based on performance
- Instances of all four network classes in present-day luxury cars (e.g., BMW7 series, Volkswagen Phaeton)
- BMW7 series
  - multiple class A networks (LIN fieldbuses)
  - two class B networks (peripheral CAN and body CAN)
  - one class C network (powertrain CAN with 500 kbps)
  - two class D networks for multimedia (MOST) and safety functions (Byteflight)

Network Class	Exemplary Protocols	Bandwidth	Exemplary Application Domains
Class A	LIN	< 10 kbps	sensor/actuator access
Class B	CAN	10kbps-125kbps	comfort domain
Class C	CAN	125kbps-1Mbps	powertrain domain
Class D	FlexRay, Byteflight	> 1 Mbps	multimedia, X-by-wire

Mapping to an Integrated Architecture

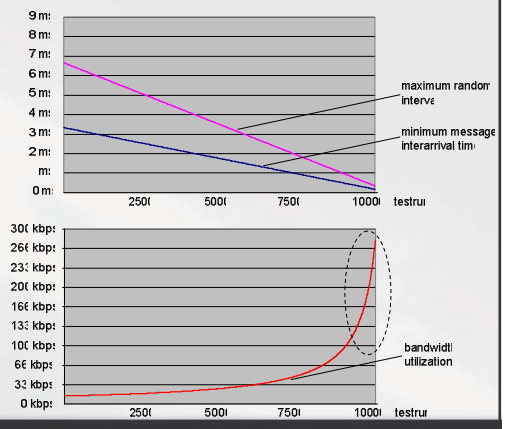


Temporal Requirements

- **Performance:**
  - minimum bandwidth
    - 2 class B (e.g., for comfort domain)
    - 1 class C (e.g., for powertrain domain)
    - 2 class D (e.g., for multimedia and X-by-wire)
  - maximum latencies
    - 10 ms to 100 ms in the comfort domain
    - in the order of ms in the powertrain domain
    - reaction time of 5 ms for safety functions realized with class D networks
- **Encapsulation:**
  - temporal partitioning to guarantee temporal properties (i.e. bandwidths, latencies, variability of latencies)
  - temporal partitioning between DASs
  - temporal partitioning within DASs

Experiments

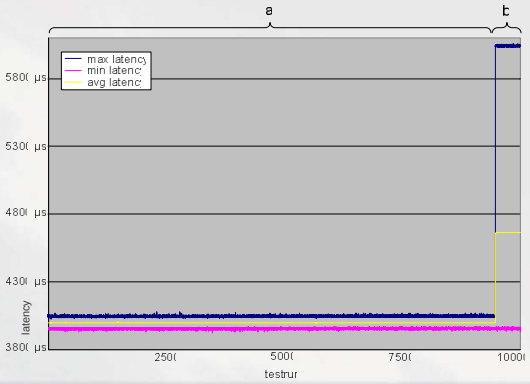
- Sporadic and periodic message transmissions controlled by
  - minimum interarrival time
  - random interval with uniform distribution for sporadic msgs.
- Probe job
  - comfort subsystem (virtual network with 125 kbps)
  - increasing bandwidth utilization
- Reference jobs
  - invariant minimum interarrival time and random interval
  - 50% bandwidth utilization



Experimental Results: Latencies for Messages from Probe Job

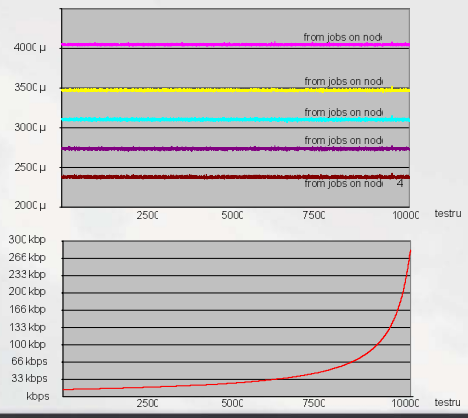
(a) Transmission behavior of probe job complying with bandwidth limit
- latencies approx. 4ms
- no omission failures

(b) Transmission behavior of probe job exceeding the bandwidth limit
- message omissions
- increased latencies



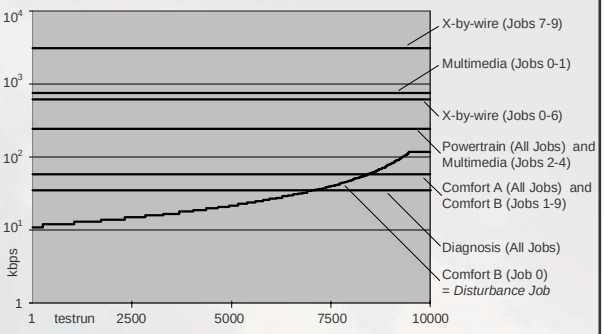
Latencies for Messages of Reference Jobs

- Transmission latencies independent from behavior of probe job
- Variability for sporadic message transmissions due to random message interarrival times
- Latency determined by phase relationship between sender and receiver node
- Performance requirements (<5ms) satisfied



Experimental Results: Bandwidth

- Bandwidth of reference jobs independent from behavior of probe job
- Variability for sporadic message transmissions due to random message interarrival times
- Performance requirements w.r.t. SAE classification satisfied



Conclusion

- Increasing importance of integrated architectures facilitating correctness-by-construction
  - growing complexity of distributed embedded real-time systems
  - avoid accidental complexity in integrated architectures
- Encapsulation of communication resources key technology for temporal composability
- Prototype implementation of DECOS architecture based on a time-triggered communication protocol demonstrates
  - competitive performance with
  - rigid temporal and spatial encapsulation