Technische Universität Wien

TU WIEN

Supporting Heterogeneous Applications in the DECOS Integrated Architecture

Roman Obermaisser

Real-Time Systems Group

Overview

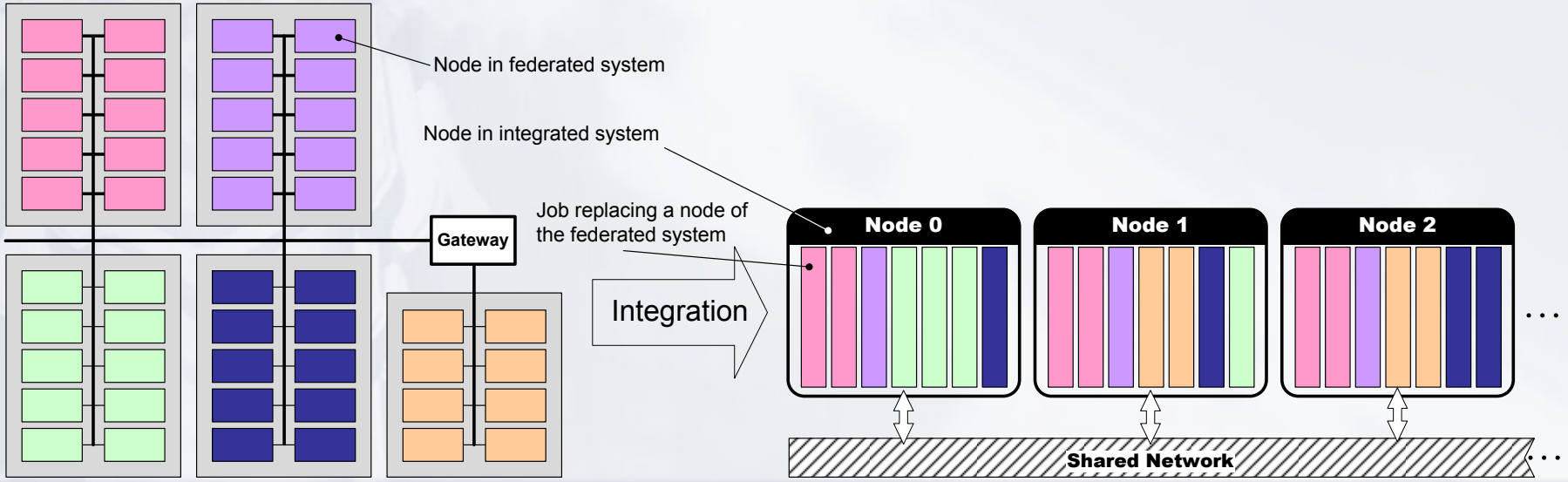
- Federated and Integrated Architectures
- DECOS Architecture
- Model-Based Development Process
- Architectural Services
  - Virtual Networks
  - Diagnostic Services
- Implementation and Results

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

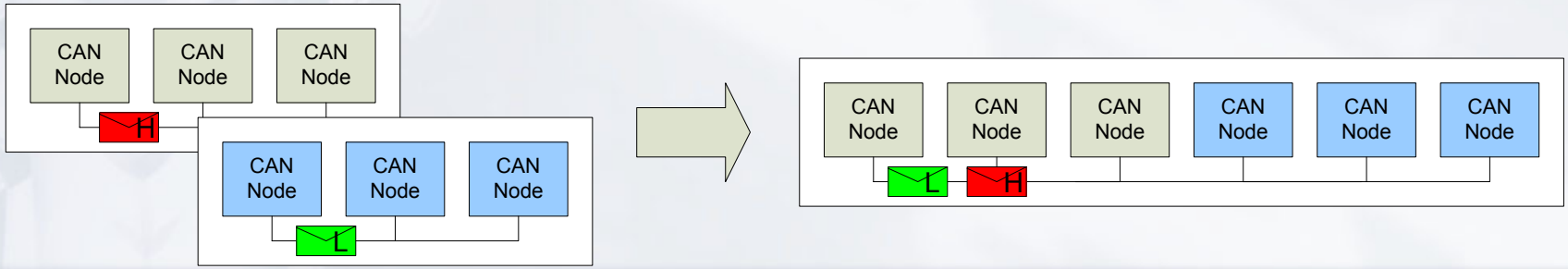
Shift to Integrated Architectures

- *Federated architectures* lead to high numbers of deployed nodes and networks: “1 Function – 1 Node” design philosophy
- As a result *integrated architectures* are gaining more and more importance (e.g., IMA, AUTOSAR, DECOS)



Challenges in Moving Towards the Integrated Architectural Paradigm

- System complexity in distributed embedded real-time systems causes increasing cost of design, verification, integration and maintenance
- Inherent application complexity
- Accidental complexity through integration-induced interference between application subsystems
  - example: integration of two CAN-based 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

Challenges in Moving Towards the Integrated Architectural Paradigm (2)

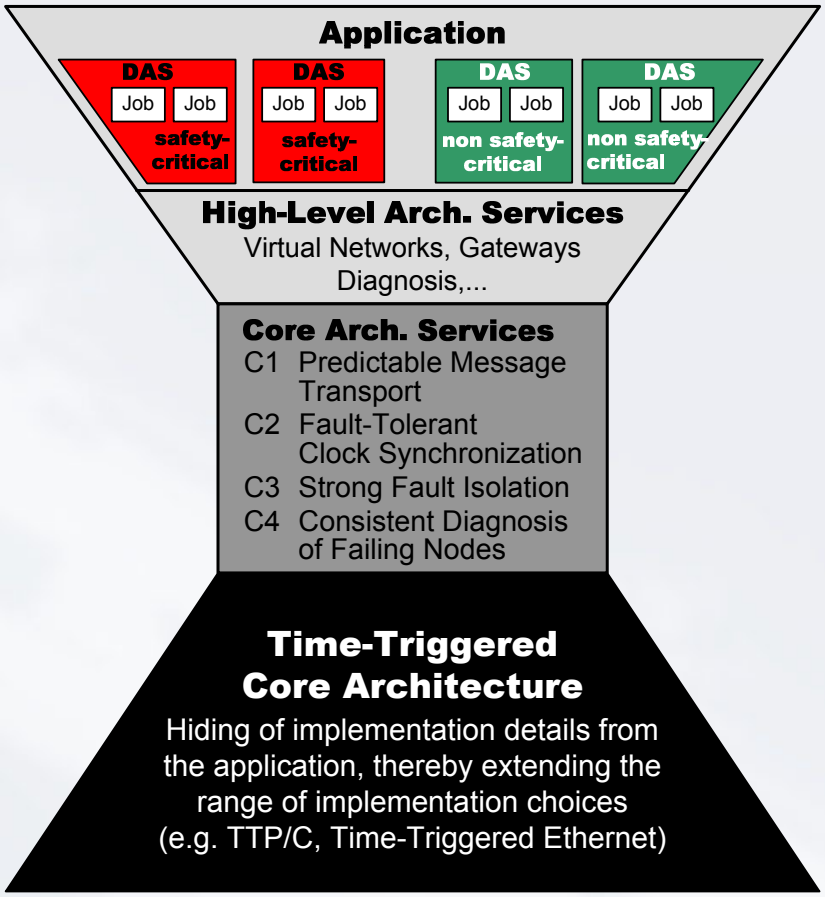
- Heterogenous application subsystems
  - safety-crtical and non safety-critical application subsystems
  - different programming models (e.g., synchronous data flow, client/server)
  - designed for different platforms (e.g., legacy systems)
- Divergent requirements concerning the services of the underlying platform
  - functionality of communication system (e.g., communication topology, control flow, connection-oriented/connection-less)
  - temporal properties (e.g., bandwidth, latencies)
  - Application Programming Interfaces (e.g., CAN-based API)
  - guaranteed services vs. best-effort services

Time-Triggered Integrated Architectures

- Time-triggered networks are widely accepted as communication infrastructure for safety-critical applications (e.g., aerospace, currently introduced in automotive domain)
  - temporal predictability
  - fault tolerance
- Foundation for integrated system architectures that support heterogeneous application subsystems
  - improved resource utilization
  - coordination of application subsystem
  - complexity management

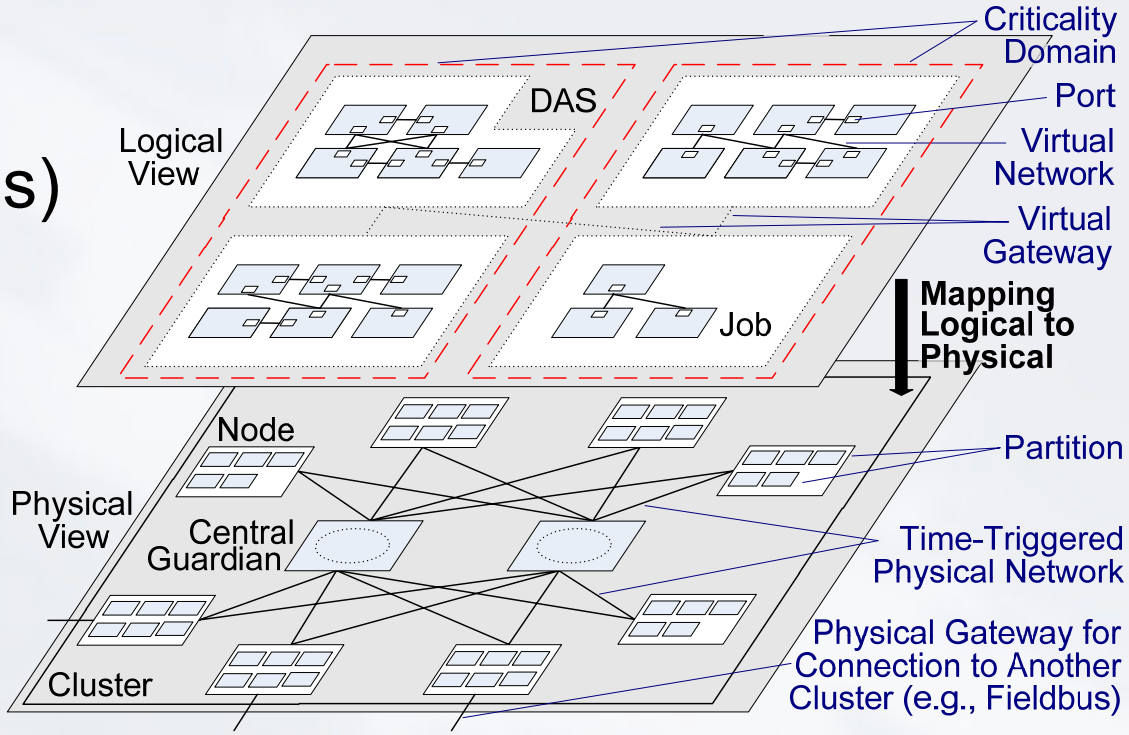
DECOS Architecture

- Application consisting of Distributed Application Subsystems (DASs)
- Core architectural services abstract from the implementation of the underlying network
- High-level architectural services
  - specific to certain types of application subsystems
  - provide support for heterogenous application subsystems
  - different types of high-level arch. services to handle contradicting requirements (e.g., flexibility vs. predictability)



Structuring of a DECOS System

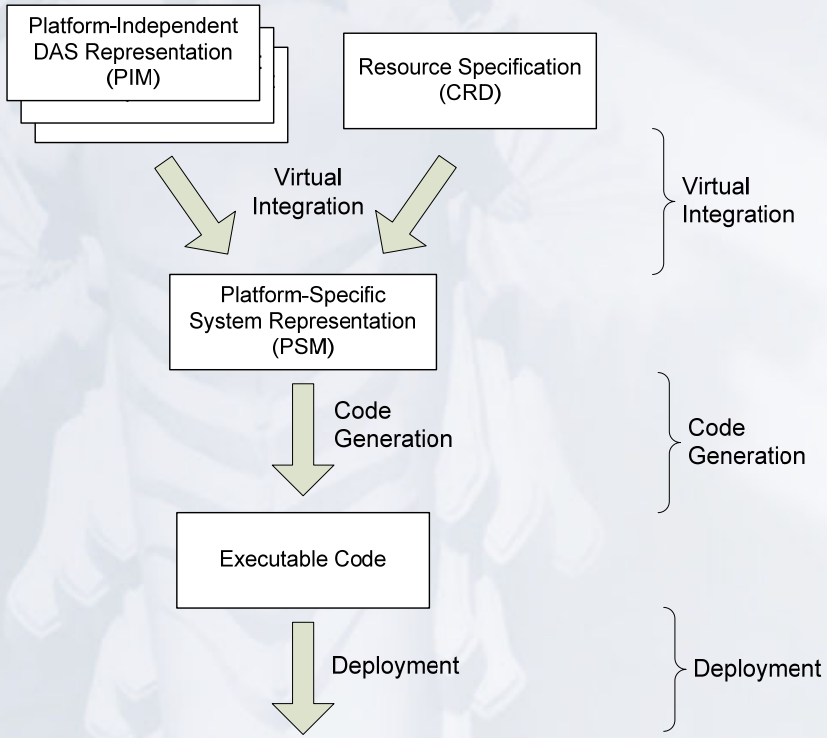
- Logical View
  - Distributed Appl. Subsystems (DASs)
  - jobs
  - specification of linking interfaces
- Physical View
  - node computers
  - partitions



Encapsulation in the DECOS Architecture

- Partitioning for computational 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)
- Partitioning in value and time domain
  - spatial partitioning to ensure data integrity (e.g., memory protection)
  - temporal partitioning to ensure guaranteed temporal properties of resources (e.g., CPU time)

Design Methodology



PIM: Formal specification of the structure and function of a system that abstracts away technical details.

CRD: Specification of the available resources on the hardware platform implementing the DECOS architecture

PSM: Extension of the PIM covering the details how the integrated system (architectural services as well as application jobs) use the available resources.

Platform Independent Model (PIM)

- Functional structuring of the system into Distributed Application Subsystems (DASs) and jobs.
- Identification of DASs is guided by functional coherence and common criticality of subsystems.
- Major focus of the PIM lies in the specification of the Linking Interfaces (LIFs) between jobs of the same DAS as well as between DASs
  - *port requirements*: definition of information semantics (state or event) and data direction
  - *communication topology*: connection between output and input ports
  - *temporal requirements*: bandwidth and latency requirements

Platform Specific Model (PSM)

- **Allocation of jobs to node computers**
  - *dependability requirements*: allocation of jobs to partitions of independent fault containment regions
  - *resource constraints of nodes*: sufficient computational resources
- **Mapping of virtual networks to the core network:**
  - *resource constrains of physical time-triggered network*: number and performance of virtual networks
  - *scheduling:* creation of a time-triggered message schedule
  - *higher protocols:* selection of the protocol of a virtual network (e.g., CAN, TCP/IP)
- **Parameterization of high-level services:** The DECOS high-level services (e.g., virtual networks, gateways, and diagnosis) have to be instantiated and configured

Hardware Specification Model (HSM)

- Besides performance and dependability aspects the transformation of the PIM to the PSM is mainly guided by the availability of resources.
- **Computational Resources:** Sufficient processing power and memory capacity are mandatory prerequisites for the allocation of a job to a particular partition.
- **Communication Resources:** A job-to-partition allocation is only considered as valid, if the communication demands (e.g. latency, bandwidth) can be fulfilled, e.g. a communication schedule can be found.
- **Special Purpose HW:** The availability of particular sensor/actuator devices or special purpose hardware (e.g., DSPs) highly influences the virtual integration.

Exemplary High-Level Architectural Services

- **Virtual Networks**
- Diagnostic Services

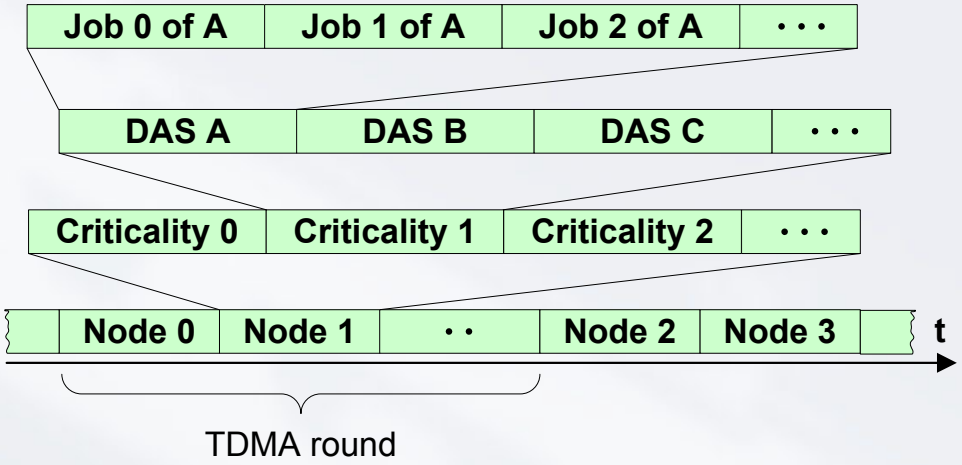
High-Level Architectural Service 1: Virtual Networks

- 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

Partitioning of Comm. Resources

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



Higher Protocols to Support Heterogenous Application Subsystems

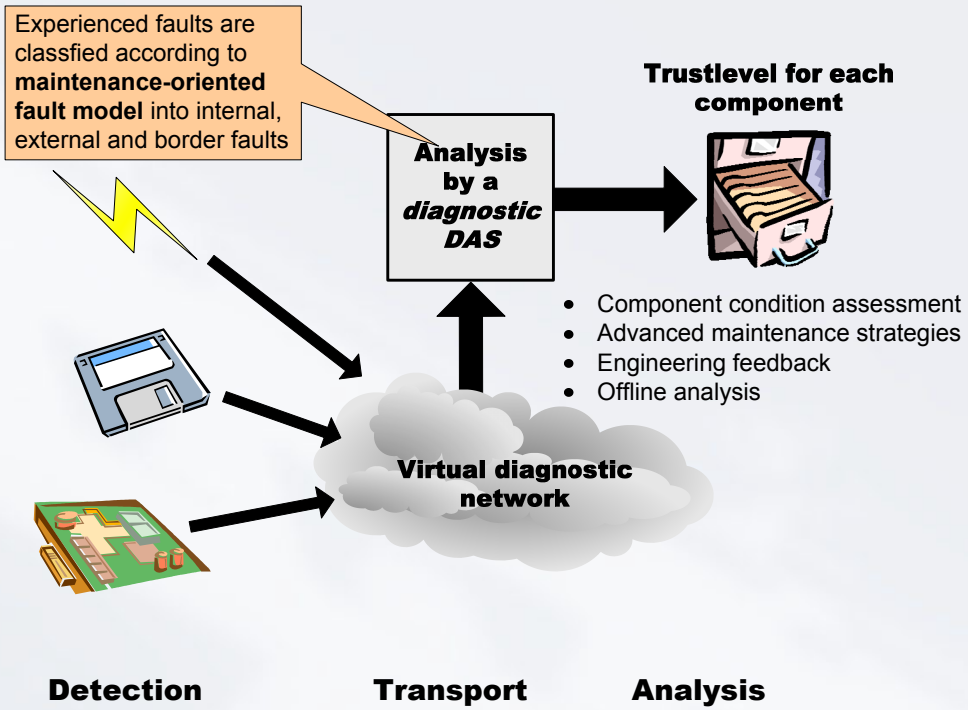
- Time-triggered virtual network (e.g., safety-critical application subsystems)
- Virtual CAN network (e.g., CAN-based legacy applications)
- Virtual TCP/IP network (e.g., diagnosis, infotainment services)
- Virtual network with CORBA transport protocol

Exemplary High-Level Architectural Services

- Virtual Networks
- **Diagnostic Services**

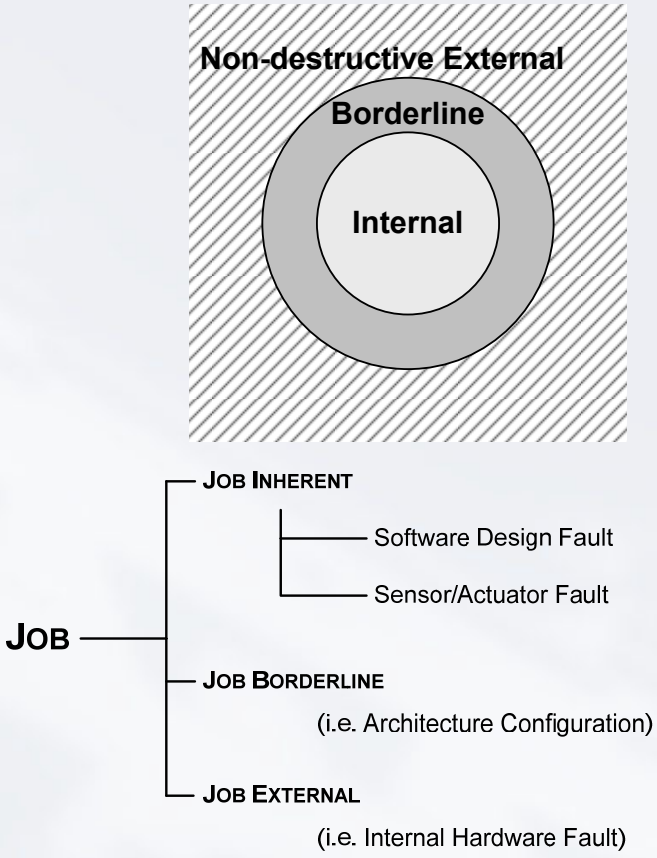
High-Level Architectural Service 2: Diagnosis

- Operation on the interface state of node computers and jobs to protect intellectual property
- Avoidance of probe effects (e.g., transport, analysis)
- Detection of correlated errors
- Analysis of the information according to a fault model for maintenance



Maintenance-Oriented Fault Model

- We stop “fault-error-failure” chain at Field Replaceable Unit (FRU) level
- Node computer as unit of replacement for **hardware faults**:
  - Internal (e.g., crack in PCB, faulty processor)
  - Borderline (e.g., connector failures)
  - Non-destructive external (e.g., EMI)
- Jobs as unit of update for **software faults**:
  - Inherent
  - Borderline
  - External



Out-of-Norm Assertions (ONAs)

- Capture consequences of faults on the distributed state
- Characteristic manifestation of a fault in the time, value and space domain (*fault pattern)*
- Fault patterns on the interface state (*symptoms*) are characteristic for particular classes of faults
- Interface state is defined between the intervals of activity on the sparse time base

Dimension	Fault Patterns	
	Massive Transient	Connector Fault
Time	approximately at the same time (within a small delta)	arbitrary
Space	multiple node computers with spatial proximity	one node computer
Value	multiple bit flips	message omissions, syntactic invalid frames on a channel

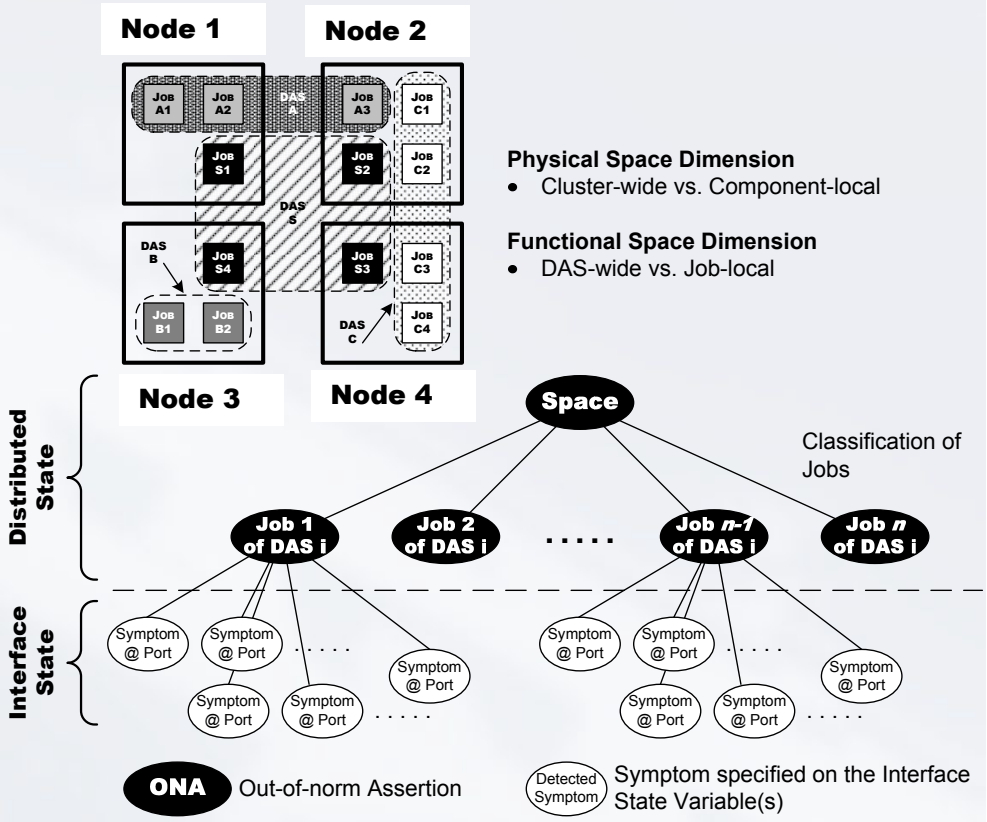
- **Architecture-level:** Detection and analysis of architecture level failures and anomalies (e.g., diverging replicas, channel failures, lost messages);
- **Application-level:** Detection and analysis of anomalous and faulty job behavior

Detection of Out-of-Norm Behavior in Heterogenous Application Subsystems

- Time-Triggered Applications
  - a priori knowledge about points in time of the periodic message transmissions
  - exploited for error detection and the establishment of membership information
- Event-Triggered Applications
  - inherent impreciseness of temporal specifications
  - gathering and correlating information about improbable behavior denoted as *out-of-norm behavior*
  - analysis process concentrates information about out-of-norm behavior occurrences to conclude whether an error has occurred

Analysis

- Global analysis process on action lattice of the sparse time base of time-triggered core system
- Integrated architectures provide the inclusion of structural information into the analysis process
  - functional vs. physical
  - job vs. node computer
- Analysis algorithms exploit architectural services like other jobs (protection mechanisms)

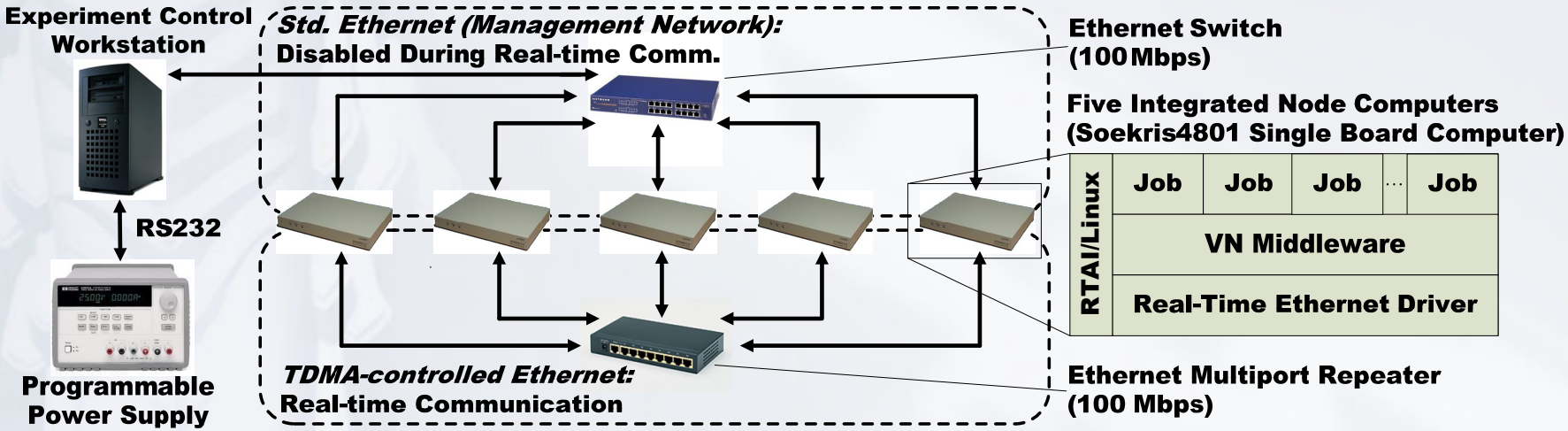


Overview

- Federated and Integrated Architectures
- DECOS Architecture
- Fault Assumptions
- Model-Based Development Process
- Architectural Services
  - Virtual Networks
  - Diagnostic Services
- **Implementation and Results**

Implementation of DECOS Architecture

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



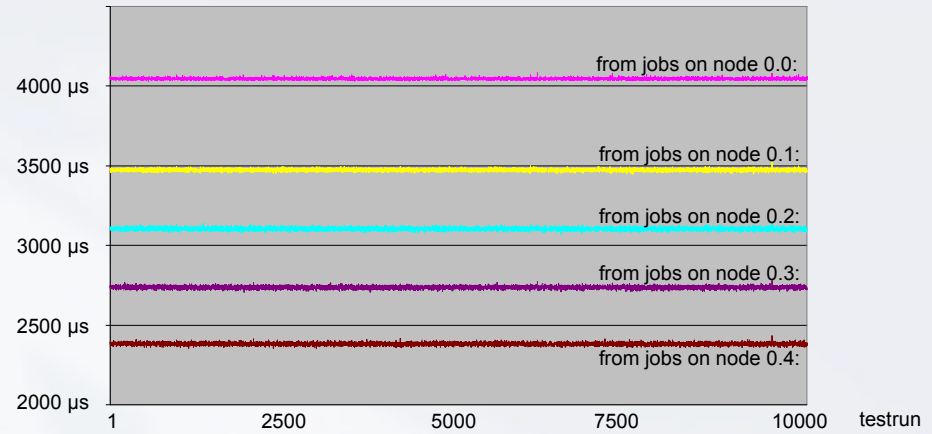
Example Configuration

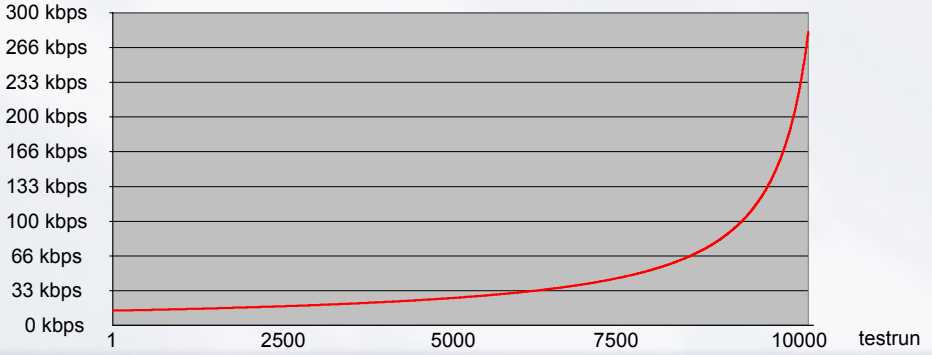
- Two time-triggered virtual networks (class D, approx. 3Mbps)
  - periodic message transmissions
  - safety-Critical applications or multimedia services
- Two virtual CAN networks (class B, 125kbps)
  - sporadic message transmissions
  - non safety-critical application services with low comm. bandwidth
- Two event-triggered virtual network (class C, 500kbps)
  - sporadic message transmissions
  - non safety-critical application services with higher comm. bandwidth

Network Class	Exemplary Protocols	Bandwidth
Class A	LIN	< 10 kbps
Class B	CAN	10kbps-125kbps
Class C	CAN	125kbps-1Mbps
Class D	FlexRay, Byteflight	> 1 Mbps

Latencies of Messages

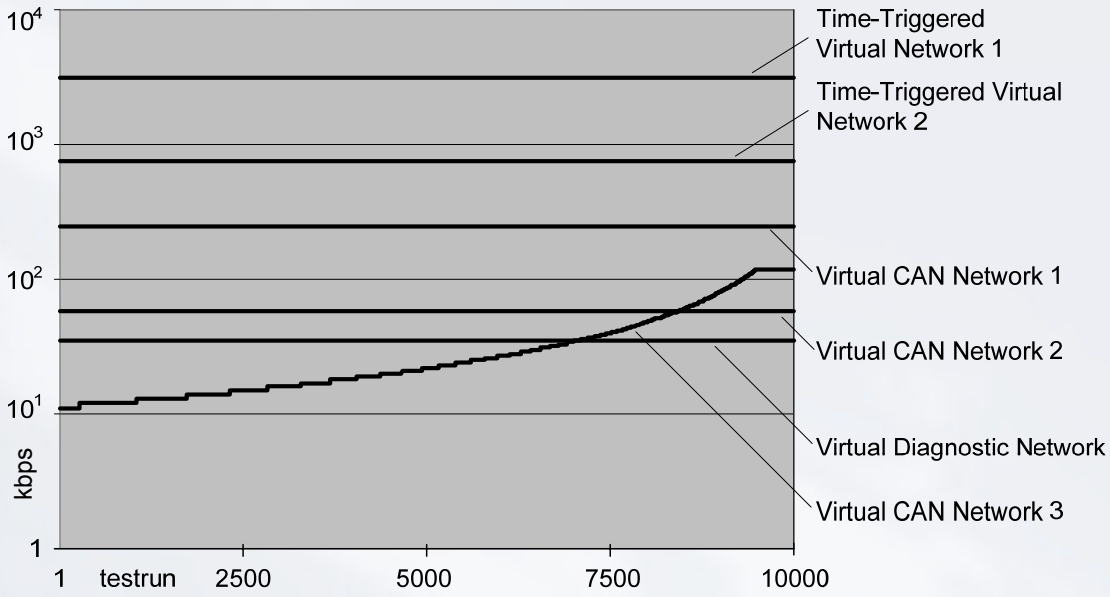
- Sporadic and periodic msg. transmissions controlled by minimum interarrival time random interval with uniform distribution for sporadic msgs.
- Probe job in virtual network with 125 kbps: increasing bandwidth utilization
- Reference jobs: invariant minimum interarrival time and random interval 50%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





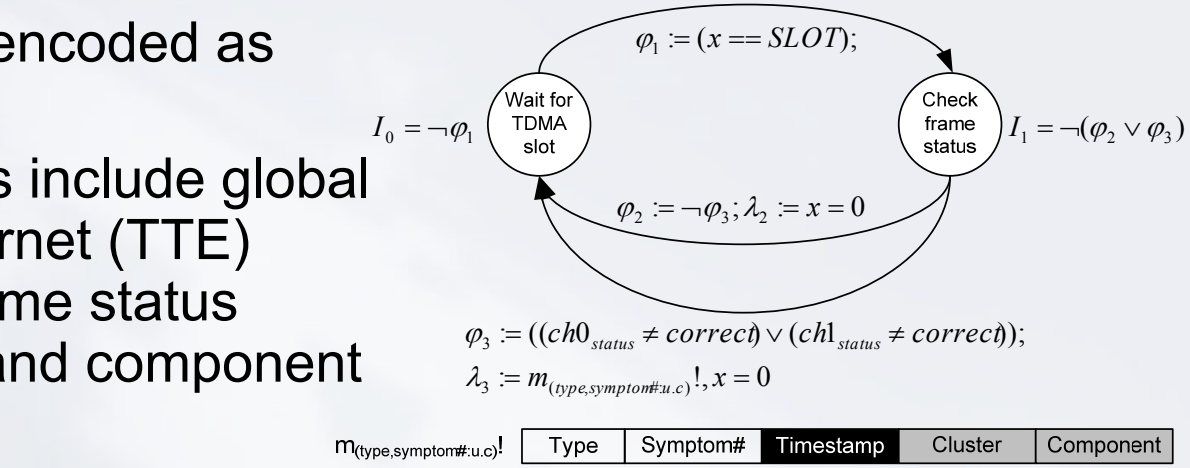
Experimental Results: Bandwidth

- 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



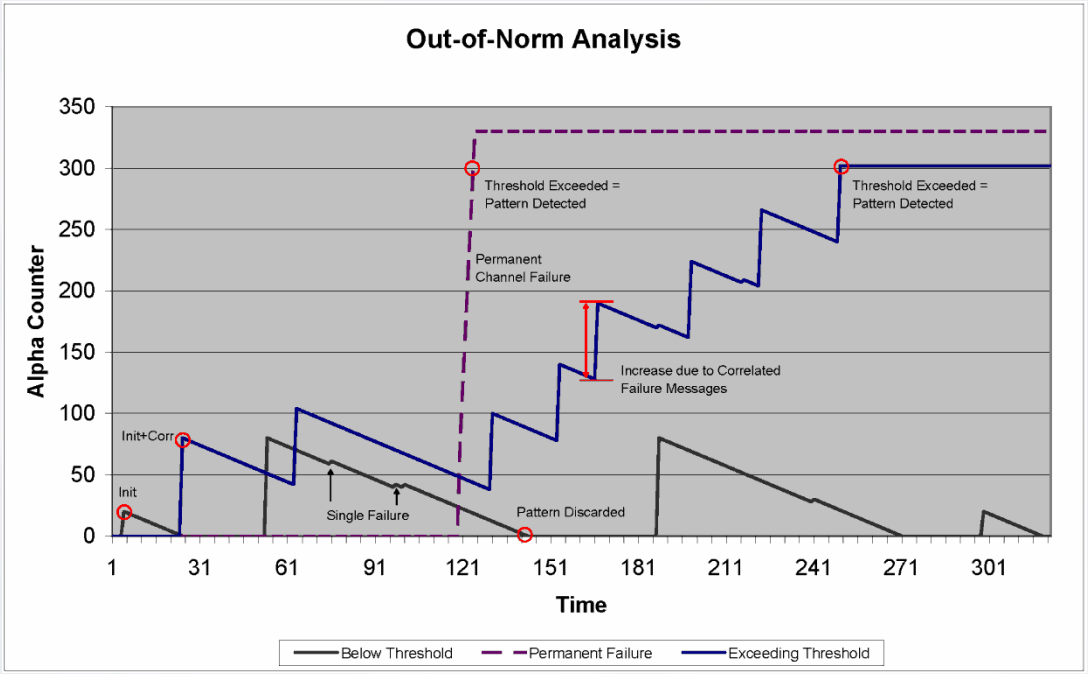
Diagnostic Services: Detection and Transport

- Symptom collectors encoded as timed automata
- Diagnostic messages include global Time-Triggered Ethernet (TTE) timestamp (time), frame status information (value), and component information (space)
- Using a virtual diagnostic network a part of the available bandwidth is reserved for diagnosis



Diagnostic Services: Analysis

- Encoded as timed state machine
- Executed on action lattice of sparse time base
- Implements threshold-based analysis techniques
- Inclusion of the time, value, and space domain into analysis
- Fault classification:
  - permanent internal
  - transient internal
  - transient external



Conclusion

- Increasing importance of integrated architectures facilitating correctness-by-construction
- DECOS architecture provides generic high-level architectural services
- Encapsulation of communication and computational resources key technology for temporal composability
- Virtual networks on top of a time-triggered physical network as the comm. infrastructure for integrated architectures
  - predefined temporal properties (e.g., bandwidth, latency)
  - rigid temporal and spatial encapsulation