

DOCUMENT

| | | |
|------------------------------|---|------------|
| Document / Deliverable Name: | Analysis and steering of cross-domain reusability regarding the identified architectural services across INDEXYS application domains | |
| Document / Deliverable Nr. | D 5.1 | |
| Version: | <input type="checkbox"/> draft <input checked="" type="checkbox"/> final | 1.0 |
| Document Type: | <input type="checkbox"/> confidential <input checked="" type="checkbox"/> public | |
| Responsible: | C. Fidi, TTTech | |
| Date of creation: | 03 May 2010 | |
| Last modification | 22 October 2010 | |

List of INDEXYS Beneficiaries

| No | Name | Short | Country |
|----|--|---------|-------------|
| 01 | TTTech Computertechnik AG | TTT | Austria |
| 02 | AUDI AG | AUDI | Germany |
| 03 | Delft University of Technology | DUT | Netherlands |
| 04 | EADS Deutschland GmbH | EADS-IW | Germany |
| 05 | NXP Semiconductors Netherlands B.V. | NXP-NL | Netherlands |
| 06 | OptXware Research and Development Ltd. | OPT | Hungary |
| 07 | Thales Austria GmbH | TRSS-AT | Austria |
| 08 | Technical University of Darmstadt | TUDA | Germany |
| 09 | Technical University of Kaiserslautern | UNIKL | Germany |
| 10 | Vienna University of Technology | TUVI | Austria |

Author(s)

| Name | Company |
|--------------------|---------|
| Christian Fidi | TTT |
| Christoph Scherrer | TRSS-AT |
| Gerhard Fohler | UNKL |
| Rodrigo Coelho | UNKL |
| György Csertán | OPT |
| Stefan Burger | EADS IW |
| Stefan Schneelee | EADS IW |

Project Coordination

TTTech Computertechnik AG

Schoenbrunner Strasse 7
1040 Vienna, Austria

Technical Matters:

D.I. Andreas ECKEL, MBA
Mail-to: andreas.eckel@tttech.com
Tel: +43 1 585 34 34 – 16
Fax: +43 1 585 34 34 – 90

Financial Matters:

D.I. Andreas BAUMGARTNER
Mail-to: andreas.baumgartner@tttech.com
Tel: +43 1 585 34 34 – 942
Fax: +43 1 585 34 34 – 90

Copyright 2010: The INDEXYS Consortium
www.indexys.eu

Revision chart and history log

| Version | Date | Reason |
|---------|------------|--|
| 0.1 | 2010-03-21 | Initial Version |
| 0.2 | 2010-05-03 | Contribution TTT (Eckel), editorial |
| 0.3 | 2010-09-10 | Contribution TRSS-AT, UNKL, TUVI, OPT, EADS IW |
| 1.0 | 2010-10-22 | Released version |

Table of Contents

| | | |
|----------|---|-----------|
| 1 | INTRODUCTION..... | 7 |
| 2 | ANALYSIS AND EVALUATION OF DOMAIN RELEVANT INDUSTRIAL STANDARDS | 8 |
| 2.1 | Automotive ISO CD 26262..... | 8 |
| 2.2 | Railway Standards..... | 11 |
| 2.3 | Aerospace DO 178C..... | 15 |
| 3 | CROSS ACCEPTANCE ANALYSIS | 18 |
| 3.1 | Similarities of the Standards | 18 |
| 3.1.1 | Certification | 18 |
| 3.1.2 | Safety..... | 18 |
| 3.1.3 | Development Assurance/Integrity Levels..... | 19 |
| 3.1.4 | System Verification | 20 |
| 4 | LINKING OF CROSS-DOMAIN ARCHITECTURAL SERVICES TO THE TARGET OF THE ARTEMIS-JU WORK PROGRAM..... | 21 |
| 4.1 | Automotive Domain..... | 21 |
| 4.1.1 | General | 21 |
| 4.1.2 | CAN Router..... | 21 |
| 4.1.3 | FlexRay Multi-Router | 23 |
| 4.2 | Aerospace Domain | 24 |
| 4.2.1 | General | 24 |
| 4.2.2 | TTP in the Remote Data Concentrator (RDC)..... | 24 |
| 4.2.3 | Network Access Controller (NAC)..... | 25 |
| 4.3 | Railway Domain | 27 |
| 4.3.1 | General..... | 27 |
| 4.3.2 | TTEthernet..... | 28 |
| 5 | MODEL-DRIVEN DEVELOPMENT | 30 |
| 6 | REFERENCES | 31 |
| 7 | ANNEX A: ABBREVIATIONS | 32 |

List of Figures

| | |
|---|----|
| Figure 1: Overview of ISO26262 | 10 |
| Figure 2: Scope of the main CENELEC railway standards | 11 |
| Figure 3: Structure of the Safety Case | 14 |
| Figure 4: Safety acceptance and approval process | 15 |
| Figure 5: Relationship between system- and software development process | 16 |
| Figure 6: TTP HW-COM Layer | 25 |
| Figure 7: Block diagram of Network Access Controller (NAC) | 26 |

List of Tables

| | |
|--|----|
| Table 1: Design Assurance Levels | 17 |
|--|----|

1 Introduction

A key goal of INDEXYS is to enable cross-domain reusability of architectural service implementation which is instantiated for platforms in the automotive, aerospace and railway domains.

Cross-domain reusability is the ability to reuse architectural service instantiations which have been implemented for one of the targeted domains (i.e., automotive, aerospace, railway) in other domains.

The objective of work package 5 is to analyse, steer, and evaluate cross-domain reusability during the actual project work, thereby directly contributing to the cross-domain aspects of the ARTEMIS-JU work programme which mentions that “*strong cross-domain studies and exchanges should be undertaken so as to achieve conceptual and technological sharing between domain specific solutions*

(ARTEMIS-JU work programme, section 3.2.5)”. A particular focus will be the reusability within other targeted domains of INDEXYS. However, reusability can also be applied to domains that are not directly in the scope of INDEXYS such as consumer electronics or industrial control systems.

The objective of this document is not only allowing the implementation of reusable services (this might be technically or economically infeasible for certain domain specific services).

However, in all cases where it is feasible to implement services in such a way, that they can be exploited within multiple domains, this shall be done in favour of domain specific implementations.

Thereby industry standards shall be analysed and evaluated (also emerging draft standards to assure sustainable project solutions) including the analysing of the safety case concept from Railway (EN 50129), the new “Safety Elements” approach from the emerging IEC 61508, related work from ISO CD 26262 and AUTOSAR (Automotive), and DO178C (Aerospace).

2 Analysis and Evaluation of Domain Relevant Industrial Standards

This section summarizes and evaluates the different concepts of the Automotive, the Railway and the Aerospace domain, with an analysis of how well cross acceptance is supported from one industry to the other.

2.1 Automotive ISO CD 26262

ISO 26262 is intended to be applied to safety-related systems that include one or more E/E systems that are installed in series production passenger cars with a max gross weight up to 3.5 t. ISO 26262 does not address unique E/E systems in special purpose vehicles such as vehicles designed for drivers with disabilities. Systems developed prior to the publication date of ISO 26262 are exempted from the scope.

ISO 26262 addresses possible hazards caused by malfunctioning behaviour of E/E safety-related systems including interaction of these systems. It does not address hazards as electric shock, fire, smoke, heat, radiation, toxicity, flammability, reactivity, corrosion, release of energy, and similar hazards unless directly caused by malfunctioning behaviour of E/E safety related systems.

ISO 26262 does not address the nominal performance of E/E systems, even if dedicated functional performance standards exist for these systems (for example active and passive safety systems, brake systems, adaptive cruise control (ACC)).

ISO 26262 is the adaptation of IEC 61508 to comply with needs specific to the application sector of E/E systems within road vehicles.

This adaptation applies to all activities during the safety lifecycle of safety-related systems comprised of electrical, electronic, and software elements that provide safety-related functions.

Safety is one of the key issues of future automobile development. New functionality, not only in the area of driver assistance, but also in vehicle dynamics control and active and passive safety systems increasingly touch the domain of safety engineering. Future development and integration of these functionalities will even strengthen the need for safe system development processes and the possibility to provide evidence that all reasonable safety objectives are satisfied.

With the trend of increasing complexity, software content and mechatronics implementations, there are increasing risks from systematic failures and random hardware failures. ISO 26262 includes guidance to avoid these risks by providing feasible requirements and processes.

System safety is achieved through a number of safety measures, which are implemented in a variety of technologies (i.e.: mechanical, hydraulic, pneumatic, electrical, electronic, programmable electronics etc.). Although ISO 26262 is concerned with E/E systems, it provides a framework within which safety-related systems based on other technologies can be considered.

ISO 26262:

- provides an automotive safety lifecycle (management, development, production, operation, service, decommissioning) and supports tailoring the necessary activities during these lifecycle phases;
- provides an automotive specific risk-based approach for determining risk classes (Automotive Safety Integrity Levels, ASILs);
- uses ASILs for specifying the item's necessary safety requirements for achieving an acceptable residual risk;
- provides requirements for validation and confirmation measures to ensure a sufficient and acceptable level of safety being achieved.

Functional safety is influenced by the development process (including such activities as requirements specification, design, implementation, integration, verification, validation and configuration), the production and service processes and by the management processes.

Safety issues are intertwined with common function-oriented and quality-oriented development activities and work products. ISO 26262 addresses the safety-related aspects of the development activities and work products.

Figure 1 shows the overall structure of ISO 26262. ISO 26262 is based upon a V-Model as a reference process model for the different phases of product development. The shaded "V"s represents the relations between ISO 26262-3, ISO 26262-4, ISO 26262-5, ISO 26262-6 and ISO 26262-7.

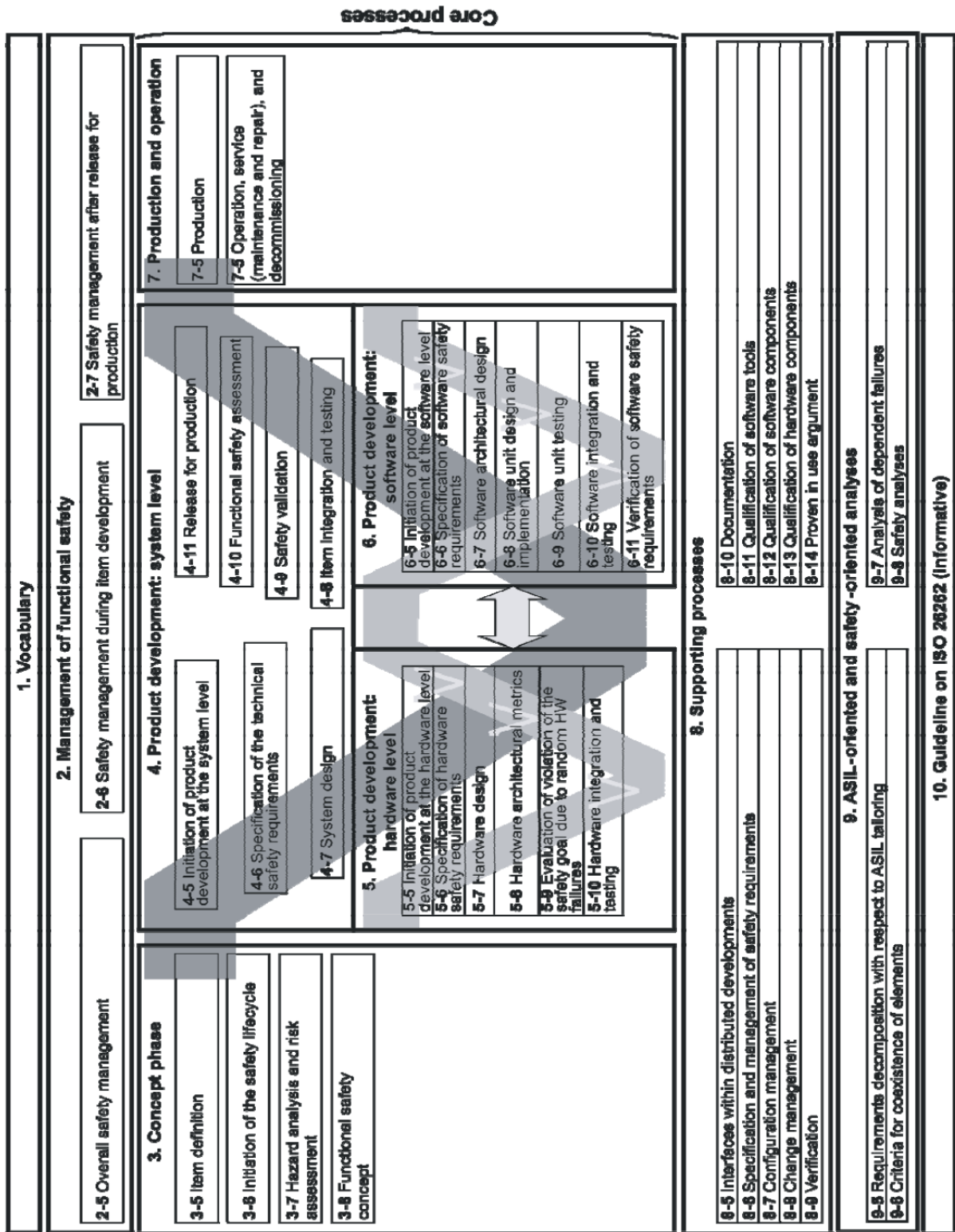


Figure 1: Overview of ISO26262

2.2 Railway Standards

The applicable standards for the railway domain with focus on Reliability, Availability, Maintainability and Safety (RAMS) and moreover the implementation of safety related equipment (electronic systems and software) is specified in the following standards:

- EN 50126 Railway Applications – Specification and Demonstration of RAMS;
- EN 50128 Railway Applications – Software for railway control and protection systems;
- EN 50129 Railway Applications – Safety related electronic systems for signaling;
- EN 50159 Railway Applications –Communication, signaling and processing systems.

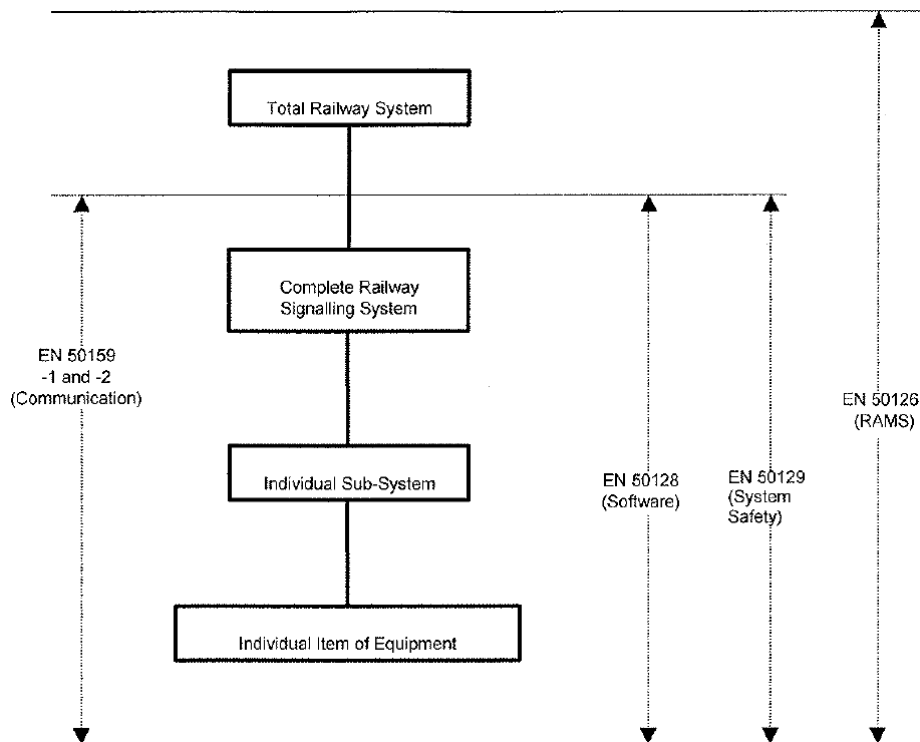


Figure 2: Scope of the main CENELEC railway standards

These standards and their influence of the development process in the railway domain will be described and in this section.

The EN 50126 Standard

Railway systems aim at achieving all rail traffic at defined levels within a given time frame following specific processes for safe operation. Railway RAMS describe the confidence levels particular systems can achieve and guarantee. Railway RAMS has a clear influence on the Quality of Service (QoS) that can be delivered to the customer. In more detail, QoS is influenced by other characteristics concerning functionality and performance, for example frequency of service, regularity of service and fare structure.

The EN 50126 standard:

- defines RAMS in terms of reliability, availability, maintainability and safety and their interaction;
- defines a process, based on the system lifecycle and tasks included for managing RAMS;
- enables conflicts between RAMS elements to be controlled and managed effectively;
- defines a systematic process for specifying requirements for RAMS and demonstrating that these requirements are achieved;
- does not define RAMS targets, quantities, requirements or solutions for specific railway applications, but defines the methods to elaborate these quantities;
- does not specify requirements for ensuring system security;
- does not define an approval process by safety regulatory authority. However, the standard serves as a guideline on which evidence has to be provided for the approval process.

The standard is applicable:

- to the specification and demonstration of RAMS for all railway applications and at all levels of such an application, as appropriate, from complete railway routes to major systems within a railway route, and to individual and combined sub-systems and components within these major system, including those containing software; in particular:
 - to new systems;
 - to new systems integrated into existing systems in operation prior to the creation of this standard, although it is not generally applicable to other aspects of the existing system;
 - to modifications of existing systems in operations prior to the creation of this standard, although it is not generally applicable to other aspects of the existing system;
 - at all relevant phases of the lifecycle of an application;
 - for use by Railway Authorities and the railway support industry.

The EN 50128 Standard

The EN 50128 standard specifies procedures and technical requirements for the development software for programmable electronic systems for use in railway control and protection applications. It is aimed at use in any area where there are safety implications. These may range from the very critical safety aspects, such as safety signalling to the non-critical ones, such as management information systems. These systems may be implemented using dedicated microprocessors, programmable logic controllers, multiprocessor distributed systems, larger scale central processor systems or other architectures.

It is applicable exclusively to software and the interaction between software and the system of which it is part. Software safety integrity levels above zero are for use in systems in which the consequences of failure could include severe injury and even loss of life. Economic or environmental considerations, however, may also justify the use of higher software safety integrity levels.

This can be software used in development and implementation of railway control and protection systems including:

- application programming;
- operating systems;
- support tools;
- firmware.

EN 50128 moreover describes the use of “Commercial Of The Shelf” (COTS) available software and tools and addresses the requirements for systems configured by application data.

The safety case, as defined in EN 50129, contains the documented safety evidence for the system/sub-system/equipment, and shall be structured as illustrated in Figure 3.

The EN 50129 Standard

Safety-related electronic systems for signalling include hardware and software aspects. To install complete safety-related systems, both parts within the whole life-cycle of the system have to be taken into account. The requirements for safety-related hardware and for the overall system are defined in the EN 50129 standard. Other requirements are defined in associated CENELEC standards (EN 50128 for software and EN 50159 for communication).

The EN50129 standard is applicable to safety-related electronic systems (including sub-systems and equipment) for railway signalling applications. The scope of this standard, and its relationship with other CENELEC standards, are shown in Figure 2.

The standard applies to all safety-related railway signalling systems, sub-system and equipment.

However, the hazard analysis and risk assessment processes defined in EN 50126 and this standard are necessary for all railway signalling systems, sub-systems and equipment, in order to identify any safety requirements. If analysis reveals that no safety requirements exist (i.e. that the situation is non-safety related), and provided the conclusion is not revised as a consequence of later changes, this safety standard ceases to be applicable.

The EN50129 standard applies to the specification, design, construction, installation, acceptance, operation, maintenance and modification/extension phases of complete signalling systems, and also to individual sub-systems and equipment within the complete system. Furthermore this standard applies to generic sub-systems and equipment (both application-independent and those intended for a particular class of application), and also to systems, sub-systems and equipment for specific applications.

Safety Case

The EN 50129 defines the conditions that shall be satisfied in order that a safety-related electronic railway system, sub-system or equipment can be accepted as adequately safe for its intended application.

The conditions for safety acceptance are presented in the following categories:

- Evidence of quality management;
- Evidence of safety management;
- Evidence of functional and technical safety.

All of these conditions shall be satisfied, at equipment, sub-system and system levels, before the safety related system can be accepted as adequately safe.

The safety case contains the documented safety evidence for the system/sub-system/equipment, and shall be structured as illustrated in Figure 3.

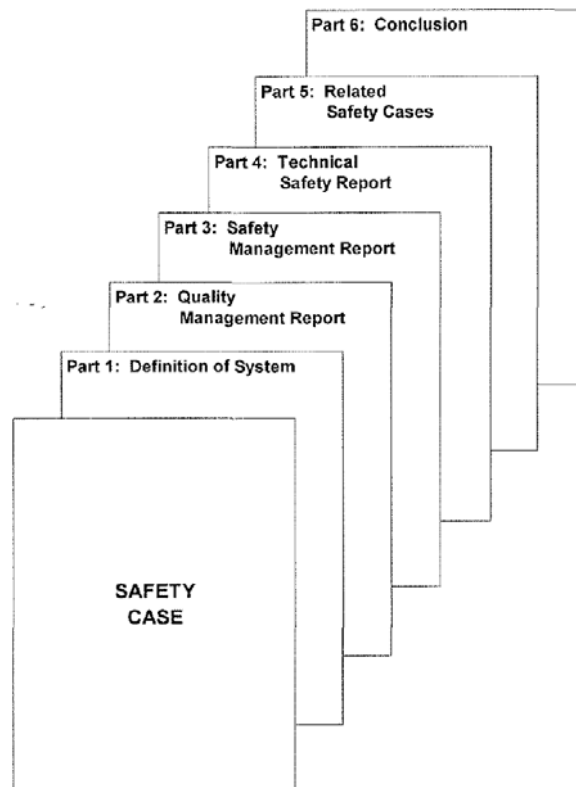


Figure 3: Structure of the Safety Case

Part 5 of this structure refers to the related safety cases which shall contain references to the Safety Cases of any the sub-systems or equipment on which the main Safety Case depends.

It shall also demonstrate that all the safety-related application conditions specified in each of the related sub-system/equipment Safety Cases are:

- either fulfilled in the main Safety Case;
- or carried forward into the safety-related application conditions of the main Safety Case.

In case one safety authority grants a safety approval for a generic product (i.e. independent of application) and a generic application (i.e. class of application) other safety authorities can accept such approval and automatically can grant the approval as well.

The safety approval process, for all three categories of Safety Case, is illustrated in Figure 4.

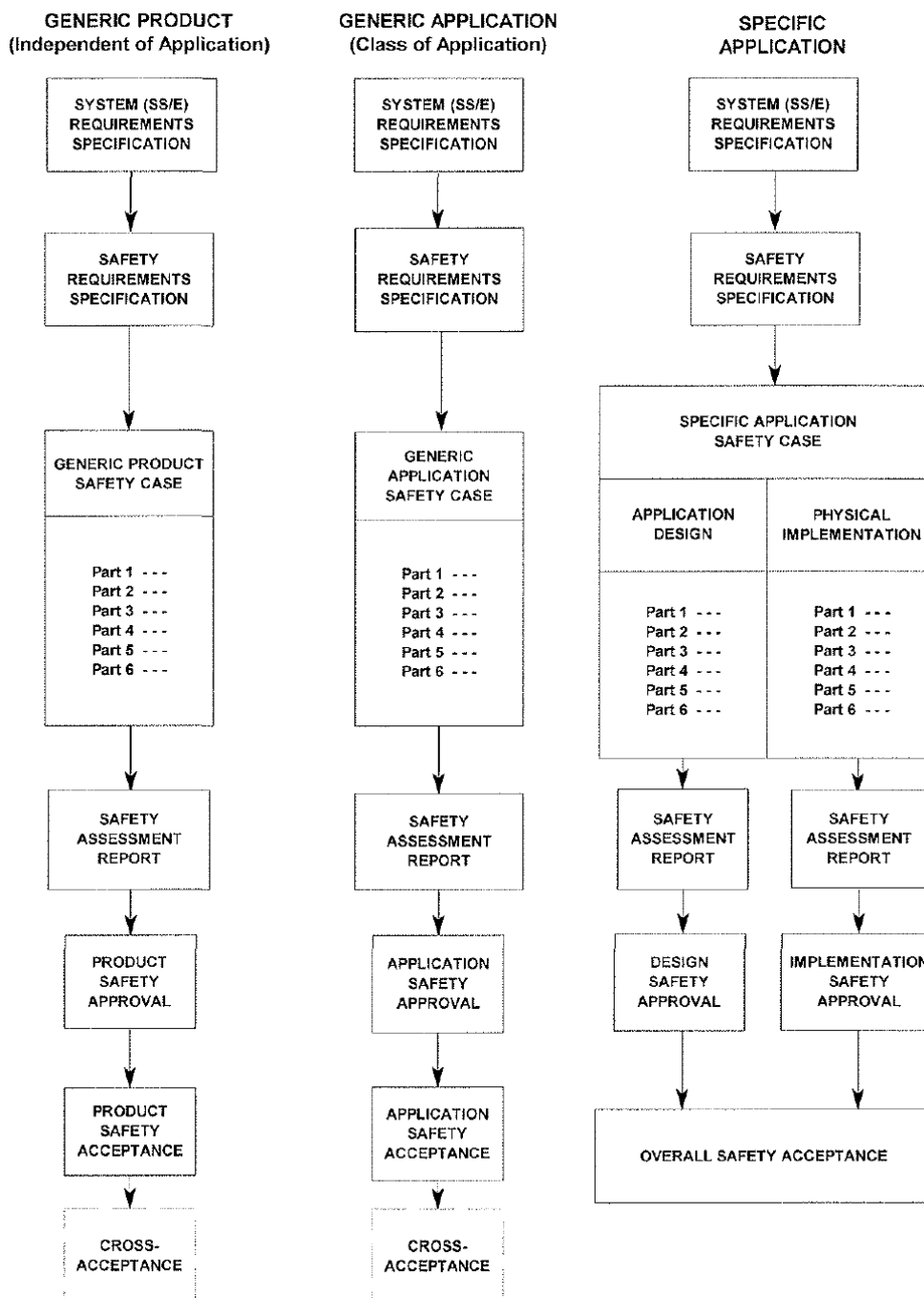


Figure 4: Safety acceptance and approval process

2.3 Aerospace DO 178C

The DO-178C document is currently under construction and will be finalized in late 2010. Due to this reason the DO178B standard will be taken into account.

DO-178B provides guidance for the production of software for airborne systems and equipment asking for a specific level of confidence for correct functioning of such software in compliance with airworthiness requirements. DO-178B represents the industry consensus opinion on the best way to ensure

safe software. It should also be noted that although DO-178B does not discuss specific development methodologies or management activities, there is clear evidence that by following rigorous processes, cost and schedule benefits may be realized. The verification activities specified in DO-178B are particularly effective in identifying software problems early in the development process. The guidelines address the concerns of the aviation industry. However the aviation law does not mandate such guidelines.

DO-178B:

- Offers a strict certification requirement for software where anomalous behavior could cause a catastrophic failure condition;
- Is used internationally to specify the safety and airworthiness of software for avionics systems;
- Is used in the development, supply, acquisition, evaluation/certification, and operation of software products to be integrated in airborne systems and equipment;
- Describes techniques and methods appropriate to ensure the integrity and reliability of such software;
- Focuses on the entire software lifecycle environment;
- Has been used to secure FAA approval of digital computer software;
- Is published by RTCA, a private, non-profit organization;
- Because of its intended international audience, uses generic terms and minimizes references to specific national regulations and procedures.

Applying the guidelines of DO-178B to the industrial software development processes perfectly integrates into larger frameworks of development guidelines including hardware, system and safety development. The system level standard is SAE ARP4754, Certification Considerations for Highly-Integrated or Complex Aircraft Systems. The relationship between system, software, and hardware processes is illustrated in Figure 5.

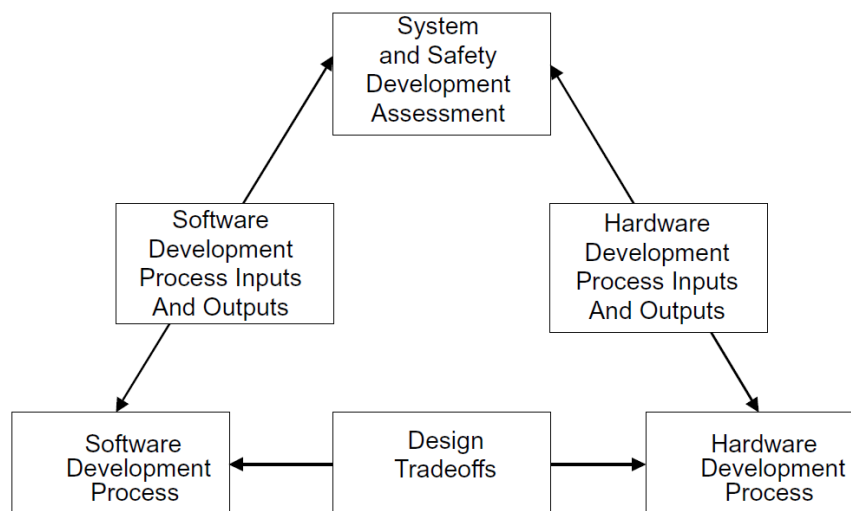


Figure 5: Relationship between system- and software development process

DO-178B specifies the information flow between system processes and software processes. The focus of the information flow from the system process to the software process is to keep track of requirements allocated to software, particularly those requirements that contribute to the system safety. The focus of information flow from the software process to the system process is to ensure that

changes in the software requirements, including the introduction of derived requirements (those not directly traceable to a parent requirement), do not adversely affect system safety.

The idea of system safety, although outside the scope of DO-178B, is crucial to understand how to apply DO-178B. The regulatory materials governing the certification of airborne systems and equipment define five levels of failure conditions. The Design Assurance Level (DAL) is determined from the safety assessment process and hazard analysis by examining the effects of a failure condition in the system. The most severe of these is catastrophic, meaning failures that result in the loss of ability to continue safe flight and landing. The least severe is “no effect”, where the failure results in no loss of operational capabilities and no increase in crew workload. The intervening levels define various levels of loss of functionality resulting in corresponding levels of workload and potential for loss of life. These five levels map directly to the five levels of software defined in DO-178B (Table 1).

| Software Level | Impact of failure condition on the system |
|----------------|--|
| A | Catastrophic |
| B | Hazardous, severe |
| C | Major |
| D | Minor |
| E | No effect on aircraft operational capability or pilot workload |

Table 1: Design Assurance Levels

It is important to note that software is never certified as a standalone entity. A parallel exists for the hardware development process and flow of information between hardware processes and system process.

Design trade-offs between software processes and hardware processes are also taken into consideration at the system level. Software levels may be lowered by using protective software or hardware mechanisms elsewhere in the system. Such architectural methods include partitioning, use of hardware or software monitors, and architectures with built-in redundancy.

3 Cross Acceptance Analysis

Software failures arise as a result of systematic (design) faults that have been introduced during software development. In recognition of this, the approach taken by many of the existing software safety standards is to define requirements and constraints for the software development and assurance processes.

This approach is found in standards such as ISO 26262, IEC 61508, EN 50128 and DO178B. By stipulating or constraining the processes to be used in the development, verification and validation of software, their intent is to reduce the number of faults *introduced* by the process (e.g. through increased rigour in specification), and to increase the number of faults *revealed* by the process (e.g. through increased rigour in verification) in order that such faults can subsequently be removed.

In addition, some standards (e.g. IEC 61508, ISO 26262) also recommend defensive measures (e.g. architectural strategies) to mitigate faults which may remain post-development and assurance, although such guidance is surprisingly uncommon. Software standards dictate the degree of rigour required in software development and assurance, according to the criticality of the software within the system application. The degree of rigour is typically expressed in terms of Safety Integrity Levels (SILs), or Development Assurance Levels (DALs) in the case of DO178B. In ISO 26262, IEC 61508 the focus is on protection systems and the SIL required is determined according to the acceptable failure rate of the protection system in question. While IEC 61508 makes a distinction between the system under control and the protective system, EN 50128/129 refers to the system as a whole.

For example, in IEC 61508 a requirement for SIL 3 is defined as corresponding to an equivalent failure rate range of 1×10^{-7} to 1×10^{-8} failures per hour of continuous operation. In DO178B the DAL is determined according to the worst-case severity of the system hazard to which failure of the software can contribute, together with some consideration of the extent of possible mitigation external to the software. In the civil aerospace domain, acceptable failure rates, from random causes, are also determined by hazard severity. Therefore, implicitly, there is a correspondence between DALs and acceptable failure rate targets. For example, the requirement of DAL A corresponds to a failure rate requirement of 1×10^{-9} per flying hour.

3.1 Similarities of the Standards

3.1.1 Certification

Certification is generally defined as the provision of assurance of safety or dependability that stems from two types of evidence:

- evidence that the developer has followed a certain system development and safety assessment process which is defined within the standard;
- technical evidence that the engineered system itself is of the required integrity.

3.1.2 Safety

Safety is in all cases defined in relation to the concept of risk. The common notion of safety as freedom from unacceptable risk is the basis of significant commonalities between all standards. The most decisive influence is on the processes which are recommended to establish the system safety requirements.

System Safety Requirements

Standards generally agree on a common frame work for the derivation of safety requirements which combines hazard assessment and risk analysis techniques. The aim of the analysis is to determine:

- critical system functions, i.e. functions the loss or malfunction of which is hazardous;
- safety requirements for these functions, i.e. the maximum tolerable failure probabilities;
- demands, if any, for additional safety functions in order to achieve acceptable levels of risk for the system.

As already mentioned, EN 50128/129 does not strongly differentiate between the system under control and the protective system many functional requirements directly translate into safety requirements. These safety requirements are related to system operation and thus formulated as functional requirements. In this sense, safety requirements in the railway domain define necessary technical measures for implementation.

Development and Safety Process

At a detailed level, the suggested development and safety processes vary in terms of content between the different standards. However, important similarities can be identified:

- At a structural level, there is a similar specification of processes. A process is defined as a linear progression of phases. The standards specify in detail the information flow between successive phases. Each phase is defined in terms of its input data, activities, a plan to execute these activities and deliverable data;
- At a semantic level, we have identified a common underlying logic and a common core between the various models of development and safety processes. We have proceeded to elicit and model this common logic into a new process for system development and assessment which can be acceptable in the framework of each standard in consideration. The model is discussed in section 4 of this paper.

3.1.3 Development Assurance/Integrity Levels

Integrity levels are used as a mechanism to allocate functional safety requirements to systems. They associate failure probabilities of critical functions with requirements for the implementation of these functions in systems. The lower that the required failure probability of a function is, the higher is the integrity level of its implementation. Standards define five integrity levels. Level zero (or E) signifies no integrity requirements, while level 4 (or A) is the highest.

During the system architectural decomposition, integrity levels provide a mechanism for the allocation of the overall system integrity to subsystems or components of the system architecture.

The general rule defines that subsystems or architecture components inherit the system integrity level. However, reduced sub-system/component integrity levels can be achieved by using partitioning techniques or fault tolerant architectures.

Partitioning is the separation of critical from non-critical functions. If partitioning is applied, each partition can be developed to the integrity level that corresponds to the most severe failure mode for this partition. Proof of independence and elimination of common cause failure is required when partitioning or fault tolerant architectures are employed to achieve the required integrity level.

If a function is implemented in a programmable electronic system, which incorporates hardware and software then standards define that the Software Integrity Level = Hardware Integrity Level = Function Integrity Level Standards agree that only with respect to hardware integrity will it be possible to quantify and apply failure rate prediction in accessing whether integrity levels have been met. As far as systematic safety integrity is concerned, qualitative techniques have to be applied and judgements have to be made to meet the target integrity levels.

3.1.4 System Verification

The main commonalities in this area can be summarised as follows:

- System verification activities are driven by the functional safety and integrity requirements;
- The type of activities is defined by the commonly accepted dichotomy between systematic and random failure;
- Probabilistic treatment is required for the assessment of the impact of random hardware failure;
- Assurance against systematic failure can be obtained by the application of certain qualitative techniques and measures in the development lifecycle.

4 Linking of Cross-Domain Architectural Services to the Target of the ARTEMIS-JU Work Program

A generic methodology will then be derived with generic recommendations. The generic recommendations shall be traceable to the standards to allow easy tailoring to the respective industry needs and simplified demonstration of standards-compliance if so needed.

During the implementation phase the methodology will be applied to the automotive implementation of the FlexRay Multi-Router to test its usability in real life. This is proposed as FlexRay is seen today as an enabler of future automotive safety-related services like x-by-wire. This includes specific tailoring of the generic recommendations as well as introduction of the concept to the development team in the initial phase of implementation.

4.1 Automotive Domain

4.1.1 General

During the implementation phase the methodology will in particular be applied to the automotive implementation of the FlexRay multi-router to test its usability in real life. This is proposed as FlexRay is seen today as an enabler of future automotive safety-related services like x-by-wire. This includes specific tailoring of the generic recommendations as well as introduction of the concept to the development team in the initial phase of implementation.

4.1.2 CAN Router

Controller Area Network (CAN) was originally developed for automotive applications and is now used in a wide area of application fields including also avionics, factory automation and medical devices. These domains share the limitations of CAN with respect to fault isolation, bandwidth, wire length, namespaces and diagnosis (cf. INDEXYS Deliverable D2.1). The CAN router allows to overcome these limitations, thereby offering significant advantages in all application domains using the CAN protocol.

Fault Isolation

An example of a hazard to reliability is the missing fault isolation for babbling idiot failures. A faulty CAN node can disrupt the communication abilities of all other nodes by continuously transmitting high-priority messages. Hence, a single CAN bus does not support the construction of embedded systems where the correct operation of the communication services is required to ensure safety.

Fault isolation is required in safety-relevant applications (e.g. to prevent common mode failures of replicas), improves robustness and establishes clear integration responsibilities. The configuration of the CAN router includes a priori knowledge about the permitted behavior of CAN nodes in the time domain (i.e., minimum message interarrival times) and value domain (i.e., message identifiers). This a priori

knowledge is used to block faulty messages. In order to achieve fault isolation in the temporal domain, the CAN router ensures that message transmissions comply with specified minimum message interarrival times. In a properly configured system, the CAN router limits the effect of messages sent by one CAN segment onto the temporal properties of messages sent by other CAN segments (e.g. latencies, variability of latencies).

Furthermore, the CAN router ensures that a CAN node from one CAN segment cannot masquerade as a CAN node from another CAN segment. Therefore, the configuration of the CAN router includes for each CAN segment a pool of permitted message identifiers. A message with an identifier that is not contained in this pool is blocked by the CAN router.

Improved Diagnosis

Fault isolation is also an important factor for effective diagnosis. Without fault isolation, the tracing of experienced errors back to the origin is complicated because a faulty node can cause secondary failures in other nodes. This diagnostic deficiency of CAN is one of the reasons why today's automotive breakdown logs do not assist the technician adequately in the identification of faulty Electronic Control Units (ECUs).

The CAN router records violations of the specification in the value and time domain. In the time domain, violations of minimum message interarrival times are recorded. In the value domain, information about naming incoherence is collected. Using a management port, the CAN router supports the retrieval of this diagnostic information. The retrieval of diagnostic information does not disrupt the routing of CAN messages and can also occur during normal operation. An external device can store the retrieved diagnostic information for an off-line analysis or use the diagnostic information as a basis for on-line diagnosis (e.g. fault detection followed by the erroneous node's isolation and system reconfiguration).

Exceed Limits of CAN

From the point of view of scalability, bandwidth and wire length limitations, which are caused by CAN's arbitration mechanism, result in tight bounds for the scale of CAN-based systems. The CAN router multicasts messages only to the nodes that require a particular message. Most messages need not be broadcast to all nodes, but only to a subset of nodes. Thus, the overall bandwidth (i.e., counting the different messages in the CAN segments) can exceed 1 Mbps, although the bandwidth in each individual segment is limited to 1 Mbps.

Through the presence of multiple wires at the overall system-level instead of a single one, the CAN router also permits a longer overall network length than a bus. Only the wire length of each CAN segment is limited to 40m at 1 Mbps.

Furthermore, the CAN router extends the CAN namespace. For the conversion of message identifiers, the configuration of the CAN router contains tables that define the mapping of identifiers between different CAN segments.

Compatibility to Existing Networks

Many existing CAN-based nodes are available. The development of these nodes represents major investments, which should not be compromised through the migration to a new CAN communication system based on the CAN router. Therefore, the CAN router must be fully compatible to a CAN bus. For a CAN segment the CAN router appears as a conventional CAN node. Thus, investments in existing CAN nodes are preserved.

4.1.3 FlexRay Multi-Router

FlexRay was originally developed as a replacement for automotive CAN networks owing to the increased bandwidth demands of highly integrated vehicles that CAN would in the future no longer be able to support.

Market demand

Since its introduction as the de-facto standard in automotive for high bandwidth real-time in vehicle networking FlexRay has found itself in use in an ever increasing number of vehicles. At the very cutting edge of automotive development some car manufactures have four or more vehicles in series production that use FlexRay.

It has become clear however that the most recent generation of high-end vehicles that use FlexRay the available bandwidth has become once again a concern for network designers. This has been driven by the continued demand by applications to exchange more and more data between ECU's, it has also come from the large amounts of diagnostic data that manufactures wish to access in order to find and fix faults.

The most beneficial feature of the FlexRay Multi-Router is its capability to increase the bandwidth of FlexRay networks by operating as an "intelligent" star allowing the parallel transmission of data in the same time slot in on the same channel.

By implementing such a device additional features such as "bit reshaping" and "bus guardian" functionality also become accessible which bring benefits to network signal quality in electrically noisy environments but also the system safety in safety critical applications.

For these reasons technology such as the FlexRay Multi-Router is necessary to protect investments and leverage the decreasing costs of implementing FlexRay hardware in the automobile.

Compatibility

Many existing FlexRay based nodes are available. The development of these nodes represents major investments, which should not be compromised through the migration to a new FlexRay communication system based on the FlexRay Multi-Router. Therefore, the FlexRay Multi-Router must be fully compatible to a FlexRay network. For a FlexRay ECU the FlexRay Multi-Router appears as a conventional FlexRay node and can work as a drop in replacement for existing active star devices. Thus, investments in existing FlexRay nodes are preserved.

Another important factor about the FlexRay Multi-Router is about not only its compatibility at the protocol level but also at the chip level. Where by the FlexRay Multi-Router has been designed to be pin compatible with existing quad transceivers so as to reduce development times since new hardware does not need to be designed.

4.2 Aerospace Domain

4.2.1 General

The GENESYS architectural services have been compared with the aerospace domain needs for the two developments in the INDEXSYS project, the Remote Data Concentrator and the Network Access Controller.

These two developments should extend the current Integrated Modular Avionics (IMA) approach and will address several GENESYS principles such as:

- complexity management i.e., reduction of cognitive complexity is achieved through small and easily understood interfaces between core processing modules and associated transducers;
- component based design i.e., separation of processing devices and input/output devices and communication over linking interfaces;
- hard and soft components i.e., FPGA based implementation of Remote Data Concentrator;
- message passing i.e., message based Remote Data Concentrator interface;
- composability i.e., deterministic communication over time-triggered network;
- the concept of a common time i.e., global time which is shared across the network.

4.2.2 TTP in the Remote Data Concentrator (RDC)

In the past TTTech developed a Table Driven Communication Layer (TD-COM Layer) for Time-Triggered Protocol (TTP) in software to make use of the reduced certification effort during reuse. The TD-COM Layer implements a high-performance communication layer between TTP networks and host applications. The TD-COM Layer can support up to two TTP networks, each being connected by a separate TTP controller. To meet these requirements the TD-COM Layer is divided into two sub-layers:

- On the TTP network side, there is the Frame-Copy Layer (FCL), which copies data between the CNI of the TTP controller and the frame buffer;
- On the application side, there is the Message-Handling Layer (MHL), which copies data between the frame buffer and the message-handling buffer.

The TD-COM is designed in such a way that the FCL runs synchronously with TTP networks, whereas the MHL can run asynchronously with TTP networks.

The application can make use of the TD-COM Layer by using defined API calls. Configuration data tables generated by the configuration tool provide the TD-COM Layer with all information needed for data handling. A pointer makes reference to those configuration data tables.

For simple distributed communication nodes as Remote Data Concentrators are, a solution with the TD-COM layer requires a lot of CPU power for executing the packing and unpacking of messages. This hints to a TD-COM implementation based on a FPGA or moreover as ASIC. Figure 6 shows a line replaceable unit with implemented HW-COM.

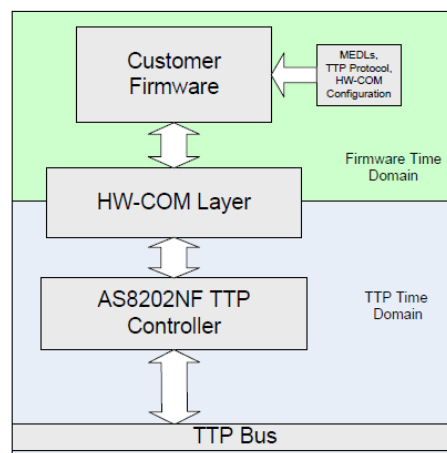


Figure 6: TTP HW-COM Layer

However TTP is a powerful bus protocol that fits perfectly into the GENESYS environment:

- Very fast and simple integration;
- Composability/Modularity;
- Complexity Reduction;
- Scalability;
- Reusability;
- Strong Partitioning;
- Reduction of certification risk/cost;
- Multiple SW DALs;
- Verifiability & Testability;
- Higher bandwidth compared to CAN, MIL-STD-1553, ARINC 429;
- Higher integration possible;
- Weight savings;
- Less connectors;
- Less cabling;
- Higher Reliability;
- Tightly coupled distributed controls;
- Minimal latency, small and bounded jitter.

4.2.3 Network Access Controller (NAC)

The existing processes to design aviation software are well established and proven. Nevertheless, these methods are a trade off between state-of-the-art techniques, like model-driven development (i.e. SCADE, which is widely used to develop avionic software) and well know and proved techniques like structural programming. The recent advances in the world of information technologies, especially in the field of entertainment, have also changed the needs and wishes of airline passengers. In the 1980s, passenger's wishes were a hot meal and a quiet flight. At the beginning of the 21st century passengers demand functionalities like video on demand, video games and Wi-Fi / GSM access during the flight. Also the requirements of airlines have changed. They demand fast maintenance times and rapid update cycle. Furthermore, airplanes themselves have changed. More and more electronics

and sensors have found a way into several airborne systems. The number of sensors and actuators has increased rapidly. Assuming the validity of Moore's law, during this period the computing industry experiences at least several dozen of generations of new processors. Therefore it is very likely that the same applies for the number of electronics modules in an airplane.

Another example of technology progress is the cabin illumination techniques. In older planes this illumination units were simple fluorescent tubes. In contrast to this fluorescent tubes use modern airplanes, like the Airbus A350, use state-of-the-art LED techniques, which are not only able to enlighten the cabin. By combining different colours of light and brightness levels, airlines are able to minimize the effect of jet lag and more. A reasonable request of airlines is to install new features into older airplanes. The product life circle of an airplane is longer than 30 years. For example the first A320 has assembled in the year 1987 and is still in production.

All this new development leads to a demanding request for more extendable and flexible software architectures and networks. The GENESYS approach might support this demand.

The goal in this project is to develop a network component which is very scalable and adaptable for future needs, especially in terms of increase of network performance and bandwidth.

The network consists of one or more Central Units (CU), Network Access Controllers (NAC) and Passenger Service Units (PSU). A high data rate backbone connects up to 12 NACs with the CU. Each NAC provides at least four subnets for connecting to the network up to eight PSUs per subnet.

With this configuration it is possible to connect 32 passenger oriented devices to one NAC. In total up to 384 passengers oriented devices are possible with the use of 12 NACs. The NAC itself (see Figure 7) connects the high data rate backbone with the sub-networks. The NAC has a modular based structure with generalized interfaces to the backbone and to the sub-networks. The NAC Core Module provides gateway functionality between the backbone and the sub-networks (see Figure 7).

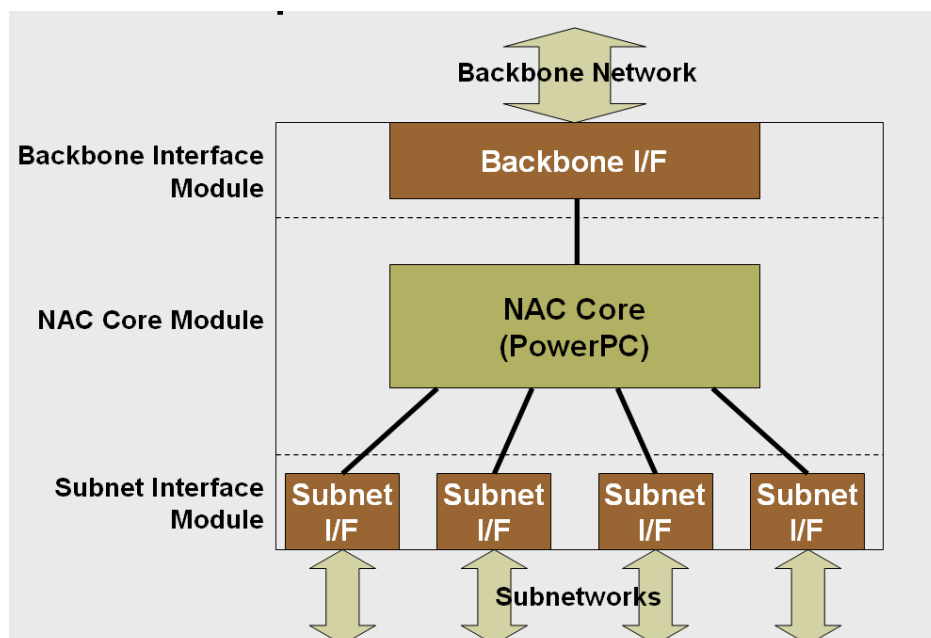


Figure 7: Block diagram of Network Access Controller (NAC)

The overall benefits of the NAC architecture based on COTS are:

- Scalability;
- Reusability;
- Modular;
- Fault tolerant on NAC and backbone;
- Bandwidth up to several Gbit/s;
- Robust against Electro Magnetic Compatibility (EMC) and lightning strikes.

The NAC uses open standards, such as 802.3 (Ethernet), at the physical layer. The usage of such open standards has several advantages for the mentioned environment. At first, new devices can be attached in a short time into the airplane environment, because the devices and also the environment use the same language. Another advantage of open standards is that the industry can use it without additional cost and in case of IEEE standards, they are able to work together on further versions. Openness also allows changing details in the standard. This makes it possible to adapt the standard to special use cases.

As mentioned above, the NAC uses the IEEE 802.3 standard. The backbone interfaces (IF) implements the optical Ethernet standard 802.3z (1000Base-X). The subnet interface implements the 802.3u (100Base-TX).

The standard Ethernet equipment has to be modified for usage in the cabin. To fulfill the security requirements, the standard connectors (RJ-45) are replaced by special connectors. These connectors have fewer problems with respect to "bad connection" failures.

Ethernet already implements methods, like CRC and padding, to secure the data flow. For higher level data flow, the cabin shall be able to implement further standards like Secure Socket Layer (SSL) and IPSec.

The current data rate can be easily increased by changing the Ethernet chipset in the NAC. The cabling either the optical as well the electrical cable, is able to provide higher data rates up to 10Gbit. This is an investment into the future.

Naturally, it is a known fact that Ethernet is not able to provided real-time support. A meaningful utilization of such a network will lead to determinism in the sense that all timing requirements of the functions are met with the required probability.

4.3 Railway Domain

4.3.1 General

In the railway domain the cross-domain usability of GENESYS architectural services will be studied with the help of the planned demonstrator for the railway domain. This demonstrator integrates a safety related communication protocol (called OCS- One Channel Safe) provided by Thales on top of software based TTEthernet developed by TTEch.

In the current implementation of OCS all safety related procedures are implemented on-top of a non-trusted IP communication stack. The aim of the demonstrator is to examine to which extent GENESYS services implemented in TTEthernet can be utilized to build a safe communication stack for the railway domain.

The communication related CENELEC 50159 standard defines safety procedures – on top of a non-trusted communication protocol. It will have to be assessed which GENESYS services have to be “trusted” services such that upper protocol layers responsible for implementing these safety procedures can rely on these services.

The guidelines to create technical evidence for the safety of a system/subsystem are stated in the EN 50126/8/9 standards. The key for the applicability of GENESYS services in the railway domain is the ability to provide a technical safety report in accordance with this standard. Therefore an EN 50129 compliant technical safety concept for communication based on GENESYS services will be elaborated, especially taking into account the design options stated in Annex B and E of the EN 50129 standard. This analysis will be based on railway typical reference system architecture in terms of number of nodes and redundancy schemes on component level.

This safety concept for the railway domain shall serve as a basis for the comparison with the approach achieving technical safety in standards relevant to the automotive and aerospace domain. Once a comprehensive view on how to prove technical safety in the various industry domains is established, the preconditions for cross-domain applicability of GENESYS service implementations with respect to standards compliance can be identified.

4.3.2 TTEthernet

GENESYS specifies mechanisms for deterministic communication in distributed systems with mixed timeliness requirements. In such a scenario, real-time applications should be able to make use of guarantees provided by the communication services to perform deterministic transmission of messages. Considering this mixed-criticality scenario, GENESYS classifies the message exchange into three subservices of communication services:

- **Periodic exchange of messages:** guarantees the periodic transmission of messages, at pre defined moments in time, from one source to one or more destinations. Messages transmitted in this category follow the state paradigm: the content of a message is not consumed when transmitted and a copy of the state is sent periodically regardless of changes on its contents;
- **Sporadic exchange of messages:** guarantees the transmission of messages according to the sporadic concept based on the minimum inter-arrival time of events, i.e., messages are not transmitted in intervals of time shorter than the minimum inter-arrival time. Transmission of this category of messages follows the event paradigm: each event enqueues a message that will be transmitted as soon as the communication is ready to accept the transmission. Messages are consumed at transmission and no transmission is required if no event occurs;
- **Primitive real-time streaming:** is utilized for transmission of variable size elements with temporal requirements. As in sporadic exchange of messages, this type of messages makes use of output queues to smooth “bursty” traffic.

TT-Ethernet [9] is a network protocol providing deterministic, real-time communication and guarantees compliance with the GENESYS communication requirements. More detailed information on how TT-Ethernet (TTE) relates to the basic communication services is presented in Deliverable D1.3.

Within the scope of INDEXYS, the proposed TTE scheduler generates a table containing the points in time when the Time-Triggered messages will be transmitted. This schedule must further ensure that the guaranteed Rate Constrained messages meet their timeliness requirements. Furthermore, output

buffer sizes must be considered. Finally, Best-Effort traffic must be accepted on-line to be transmitted during the period of time when neither TT nor RC messages occupy the transmission path.

Despite the guarantees provided by the scheduler, the timeliness communication requirements will be compared against the actual transmission times. This practical comparison should be applied for different application domains that make use of the TTE scheduler and thus provide an evaluation “on-prototype” of the cross-domain implementation of the proposed scheduler.

5 Model-Driven Development

Model Driven Development (MDD), a multi-paradigm approach, is focusing on models as primary artefacts. Architectural views form the basis of architecture modeling, i.e., the reference architecture template is organized by the views, which conform to the viewpoints covering concerns of the interested stakeholders. The Model Driven Architecture (MDA) is an Object Management Group (OMG) initiative designed to provide a standardization framework for MDD. MDA comprises a family of OMG standards that provide foundational languages and formats for meta-modeling, model transformation, general purpose modeling (UML), a number of extensions addressing specific concerns (e.g. real time and quality of service).

A model-driven development strategy separates cleanly the logic of an application from implementation decisions that may change over a system's lifetime. Therefore, INDEXYS applications can be developed according to the model-driven design paradigm, by distinguishing between a Platform Independent Model (PIM) and a Platform Specific Model (PSM). The PIM captures all behavioral aspects (i.e., value and timing) and non functional (e.g. dependability, energy) aspects of a particular application without concern for the concrete execution platform. The PSM is expressed in terms of the specification model of the target platform. The PSM is expressed with respect to specific programming models (e.g. supported by run-time libraries on the target platform).

A key lesson learnt from the experiences gained in the DECOS project is the recognition that the industry needs domain specific modeling languages (DSL) because the application of pure UML or a similar general-purpose modeling language is infeasible in the design of large HW/SW systems. Hence, domain specific extensions of UML (e.g. based on UML MARTE) can be instantiated for particular platform contexts.

- AUTOSAR for the automotive demonstrator;
- AADL for the aerospace demonstrator;
- SysML for the railway domain.

Since project resources are limited implementation will be done for one selected domain. In WP5 AUTOSAR will be used to evaluate cross-domain usability of tools and methods.

In order to provide cross-domain reusability in model-driven development, the different domain specific models should be transformed to a single meta model. This single meta model serves as a joint basis of the entire tool-chain. General purpose model transformation/code generation tools are built on top of the single meta model and can be reused for most of the domain specific languages. This facilitates cross-domain reuse of the evaluation methods and tools.

6 References

- [1] ISO 26262 Road Vehicle - Functional Safety, 8.December 2009
- [2] IEC 61508, INTERNATIONAL STANDARD, First edition 1998-12
- [3] EN 50126, Railway applications - The specification and demonstration of Reliability, Availability, Maintainability and Safety (RAMS), September 1999
- [4] EN 50128, Railway applications -Communications, signalling and processing systems_Software for railway control and protection systems, March 2001
- [5] EN 50129, Railway applications -Communication, signalling and processing systems_Safety related electronic systems for signalling, February 2003
- [6] The Potential for a Generic Approach to Certification of Safety-Critical Systems in the Transportation Sector, Y. Papadopoulos, J. A. McDermid, Department of Computer Science, University of York, Heslington, York YO1 5DD, UK, e-mail {yiannis, jam}@cs.york.ac.uk, 1999
- [7] Software in Safety Critical Systems: Achievement and Prediction by Professor John McDermid & Dr Tim Kelly, Nuclear Future, Volume 02, No.03,
- [8] THE SAFETY INTEGRITY LEVELS OF IEC 61508 AND A REVISED PROPOSAL, M. A. Hennell¹, J. C. P. Woodcock² and M. R. Woodward³, 1 LDRA Ltd., Portside, Monks Ferry, Wirral CH41 5LH, U.K., E-mail: michael.hennell@ldra.com, Tel: +44 (0)151 649 9300, Fax: +44 (0)151 649 9666, 2 Department of Computer Science, University of York, Heslington, York YO1 5DD, U.K., 3 Department of Computer Science, University of Liverpool, Ashton Building, Ashton Street, Liverpool L69 3BX, U.K.
- [9] Kopetz, H., Ademaj, A., Grillinger, P., and Steinhammer, K. 2005. The Time-Triggered Ethernet (TTE) Design. In Proceedings of the Eighth IEEE international Symposium on Object-Oriented Real-Time Distributed Computing (May 18 - 20, 2005). ISORC. IEEE Computer Society, Washington, DC, 22-33

7 ANNEX A: Abbreviations

| | |
|---------|---|
| ACC | Adaptive Cruise Control |
| ASIL | Automotive Software Integrity Level |
| AUTOSAR | AUTomotive Open System ARchitecture |
| CAN | Controller Area Network |
| CNI | Communication Network Interface |
| CU | Central Unit |
| DAL | Design Assurance Level |
| ECU | Electronic Control Unit |
| E/E | Electrical/Electronic |
| EMC | Electro Magnetic Compatibility |
| FAA | Federal Aviation Administration |
| HW-COM | Hardware Communication Layer |
| FCL | Frame Copy Layer |
| IMA | Integrated Modular Avionics |
| MDA | Model-Driven Architecture |
| MDD | Model-Driven Development |
| MHL | Message Handling Layer |
| NAC | Network Access Controller |
| OMG | Object Management Group |
| OCS | One Channel Safe |
| PIM | Platform Specific Model |
| PSM | Passenger Service Unit |
| PSU | Passenger Service Unit |
| QoS | Quality of Service |
| RAM(S) | Reliability, Availability, Maintainability (and Safety) |
| RDC | Remote Data Concentrator |
| RTCA | Radio Technical Commission of Aeronautics |
| SIL | Safety Integrity Level |
| SSL | Secure Socket Layer |
| TD-COM | Table Driven Communication Layer |
| TTP | Time-Triggered Protocol |
| UML | Universal Modeling Language |