



Distributed Real-time Architecture for Mixed Criticality Systems

Architecture Conceptualization: Requirements, Terms and Principles

D 1.1.1

Project Acronym	DREAMS	Grant Agreement Number	FP7-ICT-2013.3.4-610640		
Document Version	2.0	Date	31.03.2015	Deliverable No.	1.1.1
Contact Person	Zaher Owda	Organisation	USIEGEN		
Phone	+49 271 740 3338	E-Mail	zaher.owda@uni-siegen.de		

Contributors

Name	Partner
Roman Obermaisser	USIEGEN
Zaher Owda	USIEGEN
Mohammad Abuteir	USIEGEN
Hamidreza Ahmadian	USIEGEN
Donatus Weber	USIEGEN
Obaid Ur-Rehman	USIEGEN
Thomas Koller	USIEGEN
Leire Rubio	IKL
Asier Larrucea	IKL
Cristina Cruces	IKL
David Gonzalez	IKL
Fernando Eizaguirre	IKL
Salvador Trujillo	IKL
Jon Pérez	IKL
Anton Trapman	ALSTOM
Daniel Gracia Pérez	TRT
Philippe Bonnot	TRT
Madeleine Faugère	TRT
Sylvain Girbal	TRT
Jimmy Le Rhun	TRT
Jingy Bin	TRT
Robert Heinen	TUV
Gernot Klaes	TUV
Gebhard Bouwer	TUV
Daniel S. Raho	VOSYS
Mian M. Hamayun	VOSYS
Gerhard Fohler	TUKL
Simara Perez	TUKL
Ankit Agrawal	TUKL
Jörn Migge	RTAW

Michael Soulie	ST
Marcello Coppola	ST
Simon Barner	FORTISS
Miltos Grammatikakis	TEI
Arjan Geven	TTT
Wilfried Steiner	TTT

Table of Contents

Contributors	2
Abstract	12
Contents of this Deliverable	13
1 Overview of Requirements Analysis Process	15
2 Completeness of Requirements	16
3 Coherence and Consistency of Requirements	16
4 Methodology for Identifying Gaps and Existing Building Blocks	18
5 Driving Role of Application Domains	20
5.1 Mixed-criticality in the avionics domain	20
5.2 Mixed-criticality in the wind power domain	23
5.3 Mixed-criticality in the Healthcare domain	26
1 Requirements for the Architecture	30
1.1 Safety (Measure of Success)	31
1.2 Timing requirements (Measure of Success)	36
1.3 Fault Detection and Health Monitoring	42
1.4 Fault-Tolerance	44
1.5 Domain-independence (Measure of Success)	46
1.6 Evolvability and Scalability	48
1.7 Technology independence	49
1.8 Exploitability	50
1.9 Complexity management for reduced development cost and effort	51
1.10 Heterogeneity	53
1.11 Efficiency	56
1.12 Energy and Power	57
1.13 Required Services	58
2 Requirements for Multicore Virtualization Technology	62
2.1 Gateways	62
2.2 Isolation	63
2.3 Memory Resources Efficiency	64
2.4 Monitoring and dynamic configuration of virtualized resources	66
2.5 Multi-Core virtualization	68
2.6 On-chip fault tolerance	71
2.7 Heterogeneity	72
2.8 Real-time	74

2.9	Reconfiguration support (measure of success).....	76
3	Requirements for Mixed-Criticality Network	79
3.1	Time and Space Partitioning in the Network.....	80
3.2	Safety and Fault Handling.....	83
3.3	Timing Requirements	90
3.4	Resource Management	91
3.5	Support for demonstrators	91
4	Requirements for Tooling, Scheduling and Analysis	93
4.1	Allocation of resources and DSE.....	94
4.2	Variability.....	97
4.3	Safety.....	98
4.4	Tools	99
4.5	Platform configuration	101
5	Requirements for Mixed-Criticality Certification	103
5.1	Mixed-criticality product lines with certification support.....	103
5.2	Certification Standard(s)	104
5.3	Cross-Domain Mixed-Criticality Patterns (IEC-61508).....	108
5.4	Modular safety-case for mixed-criticality systems.....	110
5.5	Overall Strategy	116
5.6	Test bed for validation, verification and evaluation of extra-functional properties.....	118
5.7	Integration in industrial (safety) engineering process	129
6	Requirements for Avionics Demonstrator.....	133
6.1	Domain-Specific Timing Requirements	133
6.2	Use Case	136
6.3	Assessment.....	139
6.4	Certification.....	140
6.5	Safety.....	140
7	Requirements for Wind-power Demonstrator.....	142
7.1	Assessment.....	142
7.2	Use Case	144
7.3	Certification	147
7.4	Timing and Safety	148
8	Requirements for Healthcare Demonstrator	150
8.1	Assessment.....	151
8.2	Certification	152
8.3	Full virtualization	153

8.4 Real time.....	153
8.5 Low power	156
8.6 Security.....	157
9 Requirements for Modeling and Development Process	160
9.1 Overall Organization of Meta-Models.....	161
9.2 Platform-independent Application Meta-Model	164
9.3 Platform Meta-Model.....	167
9.4 Platform-specific Meta-Model	168
9.5 Timing Requirements Meta-Model	169
9.6 Reliability / Safety Meta-Model	173
9.7 Energy / Power Analysis Model and Meta-Model	175
9.8 Security Meta-Model.....	177
9.9 Variability Meta-Model	180
9.10 Complexity Management	183
9.11 Resource Allocation and Design-Space Exploration.....	185
9.12 Overall Development Approach	188
9.13 Design, Development and Validation of MC Systems	190
9.14 Variability Binding	196
10 Requirements for Resource Management	198
10.1 Global Resource Management in Networked Multi-Core Chips	198
10.2 Resource Monitoring.....	200
10.3 Local Resource Management and Local Resource Scheduling.....	202
10.4 Timely Adaptation (Measure of Success)	204
10.5 Reliability and Safety	206
10.6 Abstract Service Levels of Resource States	207
10.7 Scalability.....	208
11 Requirements for Security.....	210
11.1 Top-level Security Architecture (off-chip and on-chip)	211
11.2 Chip-level security (logical and physical security)	212
11.3 Cluster-level Security.....	216
11.4 Security Properties and Validation.....	223
11.5 Application-level Security.....	225
11.6 Security in the Development Process.....	226
12 Building Blocks.....	226
12.1 Building Blocks for Architecture	227
12.1.1 GENESYS	227

12.2 Building Blocks for Multicore Virtualization Technology	230
12.2.1 TRESCCA.....	230
12.2.2 vRtical	230
12.3 Building Blocks for Mixed-criticality Network	231
12.3.1 ACROSS	231
12.3.2 SCARLETT	231
12.3.3 Internal Projects: TTT	231
12.4 Building Blocks for Architecture Tooling, Scheduling and Analysis.....	232
12.4.1 Internal Projects: ONERA.....	232
12.4.2 PEGASE	233
12.4.3 SCARLETT	234
12.4.4 Internal Projects: TTT	234
12.4.5 Internal Projects: TUKL	235
12.5 Building Blocks for Mixed-Criticality Certification.....	236
12.5.1 MultiPARTES	236
12.5.2 Internal Projects:FENTISS	239
12.5.3 TERESA.....	240
12.6 Building Blocks for Avionics Demonstrator	241
12.6.1 Internal Projects:TRT	241
12.7 Building Blocks for Wind-power Demonstrator	241
12.7.1 Internal Projects: ALSTOM	241
12.7.2 MultiPARTES	241
12.8 Building Blocks for Healthcare Demonstrator.....	242
12.8.1 TRESCCA.....	242
12.8.2 vRtical	243
12.9 Building Blocks for Modelling and Development Process.....	243
12.9.1 TIMMO-2-USE.....	243
12.9.2 ACROSS	246
12.9.3 RECOMP / ARAMIS	246
12.9.4 CESAR.....	247
12.9.5 MOSIS	248
12.9.6 VARIES	248
12.9.7 VERDE	248
12.10 Building Blocks for Resource Management.....	249
12.10.1 ACTORS	249
12.11 Building Blocks for Security	250

12.11.1	OVERSEE	250
12.11.2	TERESA	250
12.11.3	TRESCCA	250
12.11.4	virtical	251
12.11.5	MACsec	252
12.11.6	Internal Projects: TTT	252
12.11.7	D-MILS	253
12.11.8	IEEE 802.1X	253
13	Gaps	255
13.1	Architecture	255
13.2	Multicore Virtualization Technology	260
13.3	Mixed-criticality Network	264
13.4	Tooling, Scheduling and Analysis	266
13.5	Mixed-Criticality Certification	270
13.6	Modeling and Development Process	278
13.7	Resource Management	285
13.11	Security	287
	Terminology	296
1	Aperiodic Message	301
2	Application Service	301
3	Application Subsystem	301
4	Architectural Style	301
5	Architecture	301
6	Assurance Level	302
7	Behavior	302
8	Channel	302
9	End-to-End Channel	302
10	Error	302
11	Fail-operational System	302
12	Fail-safe System	303
13	Fault	303
14	Fault-Containment Region	303
15	Fault Hypothesis	303
16	Failure	303
17	Cluster	303
18	Component	303

19	Composability	304
20	Core Platform Service	304
21	Determinism	304
22	Development Methodology	304
23	Integration Level	304
24	Integration Level: Chip-Level	305
25	Integration Level: Cluster-Level	305
26	Integration Level: Core-Level	305
27	Gateway	305
28	Linking Interface	305
29	Linking Interface Specification	305
30	Message	306
31	Message Sent (Event)	306
32	Message Arrived (Event)	306
33	Mixed-Criticality System	306
34	Mixed-Criticality Architecture	307
35	Optional Platform Services	307
36	Partition	307
37	Periodic Message	307
38	Platform	307
39	Platform Services	307
40	Platform-Independent Model	308
41	Platform-Specific Model	308
42	Reliability	308
43	Replica Determinism	308
44	Service	308
45	Spatial Partitioning	308
46	Sporadic Message	308
47	State	309
48	State Recovery	309
49	Subsystem	309
50	System	309
51	Temporal Partitioning	309
52	Design Pattern	309
53	Dependability Patterns	309
54	Compliant Item	309

55	Safety manual for compliant items	310
56	Event.....	310
57	Timing Event.....	310
58	Task Activation (Event).....	310
59	Task Execution End (Event)	310
60	Frame Instantiation (Event).....	310
61	Frame Transmission Start (Event)	311
62	Frame Transmission End (Event).....	311
63	Timing (Event) Chain	311
64	Timing Constraint	311
65	Latency Constraint.....	311
66	Repetition Constraint	311
67	Synchronization Constraint	312
68	Worst Case Execution Time (WCET).....	312
69	Worst Case Response Time (WCRT).....	312
70	Worst Case Traversal Time (WCTT)	312
71	Secure End-to-End Channel.....	312
72	Secure Point-to-Point Channel	312
73	Security Mechanisms.....	312
74	Security Services.....	313
75	Confidentiality	313
76	Integrity	313
77	Authenticity.....	313
78	Authentication of data origin	313
79	Authentication of a communication partner	313
80	Availability	313
81	Access control.....	313
82	DoS attack.....	314
83	Man-in-the-middle attacks.....	314
84	Spoofing attack.....	314
85	Packet injection attack	314
86	Replay attack	314
87	Sniffing attack.....	314
88	Key Management	314
89	Cryptographic key.....	314
90	Key Exchange.....	315

Bibliography..... 316

Abstract

Deliverable *D1.1.1 – Architecture Conceptualization, Requirements, Terms and Principles* – defines the requirements for the DREAMS architecture, the models, the virtualization technologies at chip and cluster-level, the development and certification methods, and the demonstrators in the avionics, wind power and healthcare domains.

The overall set of requirements of different categories is complemented with the identification of already existing building blocks from other projects or existing technologies and techniques. The building blocks are provided as an input to the project by the partners in the DREAMS consortium. Requirements not covered or only partly covered by the building blocks are reflected as gaps, which must be closed in order to achieve the project goals.

Requirements and building blocks have been analyzed and consolidated in the following categories:

- Architecture
- Multicore virtualization
- Mixed-criticality network
- Tooling, scheduling and analysis
- Certification
- Avionics demonstrator
- Wind power demonstrator
- Healthcare demonstrator
- Modeling and development process
- Resource management
- Security

In addition, terms and principles are defined for a common view and understanding among all partners and as a common basis for all work packages. Relationships between the different terms of the terminology are established in diagrams.

Contents of this Deliverable

This deliverable is structured into three parts:

Part A introduces the process that was used for the identification of the requirements. We describe the measures for the achievement of completeness and consistency, the identification of building blocks and gaps, as well as the driving role of the application domains.

Part B describes the identified requirements, building blocks and gaps. The overall DREAMS architectural requirements for safety, fault tolerance and real-time performance with data, energy and system integrity are introduced in section 1. The DREAMS requirements contribute to the goal of building a cross-domain architectural framework for mixed-criticality integration on networked multi-core chips by introducing domain-independent components, core services and models that can be exploited in different application domains (e.g., avionics-WP6, wind power-WP7, healthcare-WP8).

The presented DREAMS services shall realize the virtualization of resources with timeliness, reliability, security and energy-efficiency at chip-level (see section 2) and at cluster-level (see section 3).

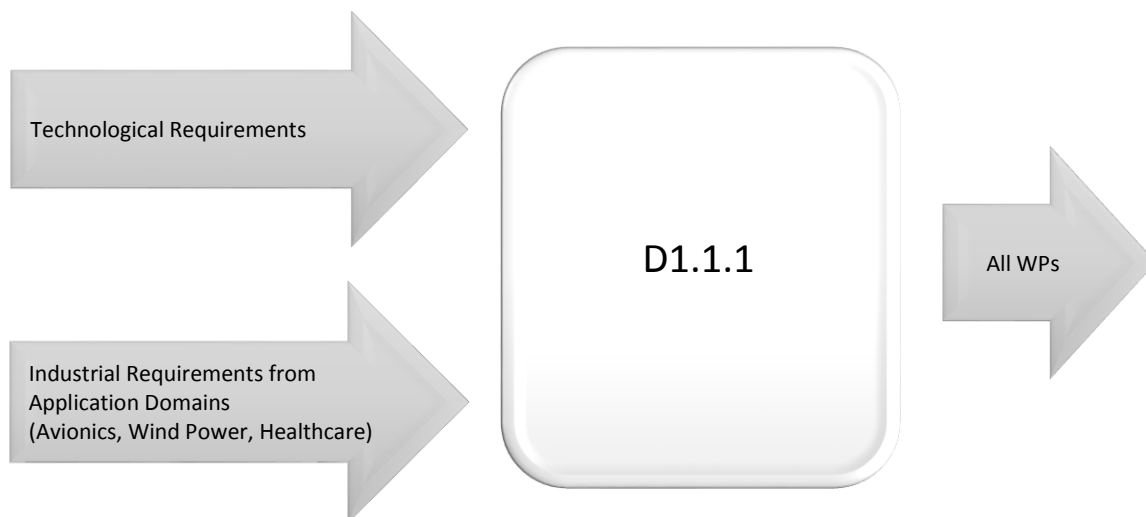


Figure 1: Relationship of the deliverable D1.1.1 to other WPs

Resource managers and development methods of the DREAMS architecture shall support the combination of offline scheduling (see section 4) with run-time adaptation. Moreover, requirements for certification, verification, validation and the DREAMS testbed are illustrated in section 5.

The requirements for the definition of the models and the development process for a model-driven development methodology are described in section 9. The models will serve for the development of analysis, scheduling, verification and configuration methods in WP4, and the realization of the demonstrator applications (WP6, WP7, WP8) using the platform developed in WP2 and WP3.

Suitable security mechanisms (see section 11) will protect the access to the virtualized communication and computational resources and secure the resource management.

Part C explains the common terminology for mixed-criticality systems that was converged from the different technological areas and application domains.

Part A

Requirements Analysis Process

1 Overview of Requirements Analysis Process

The requirements analysis was performed with a driving role of the application domains as well as active involvement of the executive board and all partners.

Working groups with experts for key topic areas were established to ensure completeness of requirements. Internal meetings and telephone conferences were organized in each working group.

At a global level, meetings with all working groups served to establish the coherence and consistency of the requirements, while also addressing the dependencies between working groups.

A three-phase analysis has been performed in order to define the deliverable's major parts as shown in Figure 2. The first phase was the requirements selection and consolidation, in which the requirements were identified and analyzed. In phase two, already existing relevant building blocks of input projects as well as intellectual property of the DREAMS consortium partners have been identified. In the last phase, gaps have been identified, denoting missing services needed to satisfy the DREAMS requirements based on the collected building blocks.

1. Top-down analysis was performed in order to identify the technological **requirements** (sections 01-11).
2. Bottom-up analysis was performed to identify the **building blocks** that have to be integrated into DREAMS architecture (section 12).
3. **Gaps** were identified for a meet-in-the-middle approach based on the requirements and gaps (section 13).

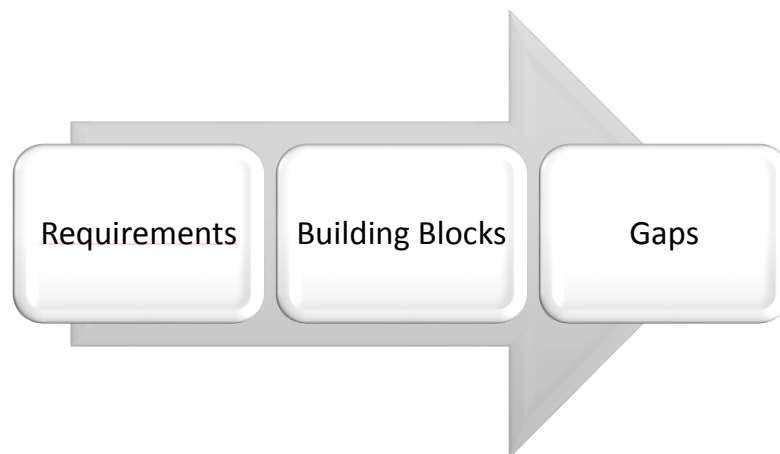


Figure 2: Three phase analysis

The selection process of requirements included existing sources from previous research and industrial projects and practices (e.g., GENESYS, MultiPARTES, ARTEMIS SRA), as well as new requirements emerging from the mixed-criticality scope of the targeted application domains (see Figure 1).

2 Completeness of Requirements

Completeness of requirements was ensured with respect to the technological areas of DREAMS and the targeted application domains. Working groups were established and lead by an expert of the respective area. The lead person and the working group ensured that the relevant requirements are covered for the respective technological area or application domain (cf. Table 2).

Working Group		Lead Person
Technological Areas	Architecture	Roman Obermaisser, USIEGEN
	Modeling & Dev. Process	Simon Barner, FORTISS
	Multicore Virtualization Technology	Michael Soulie, ST
	Mixed-Criticality Network	Arjan Geven, TTT
	Tooling, Scheduling and Analysis	Jörn Migge, RTAW
	Certification, V&V	Leire Rubio, IKL
	Security	Obaid Ur-Rehman, USIEGEN
	Resource Management	Gerhard Fohler, TUKL
Application Domains	Avionic Use Case & Dem.	Daniel Daniel Gracia Perez, TRT
	Wind power Use Case & Dem.	Anton Trapman, ALSTOM
	Healthcare Use Case & Dem.	Marcello Coppola, ST

Table 1: Working Groups for Requirements Analysis

In addition, completeness was established with respect to the measures of success and the DREAMS project objectives according to the description-of-work.

As a foundation for the seamless work in later project phases, completeness was also addressed from a managerial perspective. Each requirement was assigned a lead partner as well as contributing partners who make sure that the requirement is satisfied. The means for validation were defined in order to assess the satisfaction of the requirement in later project phases.

3 Coherence and Consistency of Requirements

The coherence and consistency of the requirements was the focus of several iterations of the requirements analysis. These iterations were performed with active participation of the working groups and their leaders, the executive board and the WP1 partners. The discussions and analysis of consistency and coherence lead to the following types of actions:

- Merging, splitting and alignment of requirements to avoid redundancy and prevent contradictions (e.g., types of required timing models at chip level and cluster level)
- Distinction between general requirements (e.g., time-space partitioning in the architecture) as well as specializations (e.g., time-space partitioning at chip level and cluster level)

- Alignment of terminology as a foundation for coherent requirements wording
- Consistency of requirements with respect to certification aspects based on reviews and recommendations of TUV
- Consistency between requirements and description-of-work

4 Methodology for Identifying Gaps and Existing Building Blocks

DREAMS is firmly based on 23 European and national projects in different industrial domains and technological areas. Therefore, we identified relevant existing results from these projects (called building blocks henceforth). In DREAMS the building blocks serve as the foundation for the consolidation, extension and closing of research gaps towards a converged European reference architecture for mixed-criticality systems.

In addition to these research projects, all partners identified relevant internal building blocks within their organizations. These building blocks represent Intellectual Property (IP) of the respective organizations that can be extended in DREAMS. Table 2 gives an overview of the input projects and the respective research areas.

Research Area	Input Project	DREAMS Partners
Cross-Domain Embedded System Architectures	GENESYS	USIEGEN, TTT
	ACROSS	FORTISS, TTT, TRT
	INDEXYS	TTT, TUKL
Security	OVERSEE	USIEGEN
	TERESA	IKL, USIEGEN
	TRESCCA	ST, TEI, VOSYS
Safety-Critical Multi-Core Architectures and Certification	ARAMIS	FORTISS
	RECOMP	TRT, FORTISS
	SAFECER	TTT
Development Methods	CESAR	SINTEF, TRT
	CRYSTAL	TRT, TTT
	VERDE	SINTEF, TRT
Resource Management and Dynamic Reconfiguration	FRESCOR	TUKL, UPV
	ACTORS	TUKL
	DIVA	SINTEF
	SCARLETT	TRT, ONERA, TTT
Mixed-Criticality at Chip Level	CERTAINTY	TRT
	MULTIPARTES	IKL, Alstom, UPV, FENTISS
	VIRTICAL	ST, TEI, VOSYS
Product Lines	MOSIS	SINTEF
	VARIES	SINTEF
Real-Time Modeling and Timing Analysis	PEGASE	RTAW, ONERA, TRT

	TIMMO-2-USE	RTAW
--	-------------	------

Table 2: Projects for Analysis of Building Blocks

Due to the complementarity of the partners in the DREAMS consortium and their active role in previous research projects, the coverage of the most relevant building blocks in the area of mixed-criticality systems is ensured.

The identified requirements and identified building blocks served as the input for the identification of gaps. Each requirement was checked against the building blocks in order to determine

- Parts of requirements that are already satisfied by building blocks (e.g., time/space partitioning of time-triggered networks)
- Parts of requirements with no solution in existing building blocks (e.g., end-to-end communication with TSP for periodic, sporadic and aperiodic communication)

The gaps resulted from the clustering of the requirements without coverage in the building blocks.

5 Driving Role of Application Domains

The application domains had a driving role in the definition of the requirements. The partners from the application domains have participated in the working groups, where their requirements were analyzed. Many technological requirements come from the application domains as indicated by the source field, while they are clustered according to their technological characteristics (e.g., safety, availability, timing, security) into D1.1.1

In the following, a summary of each domain is given with an overview of the boundary conditions and respective requirements:

- Explanation of mixed-criticality in the respective application domain
- Overview of types of systems, subsystems and their criticality
- Overview of present day architectures and platforms
- Overview of future plans for mixed-criticality systems in the domain
- Overview of boundary conditions and requirements in the domain (with link to WGs)

5.1 Mixed-criticality in the avionics domain

Avionic systems are complex networked systems on which applications with different criticalities levels (i.e., different safety levels like DAL A, B, C, D and E, with A being the most critical and E not critical) and kinds (i.e., safety and security) can be executed. To manage such complex systems some basic designs are recommended by the standardization bodies, the first one being to divide the whole system into different domains, each targeting a different purpose and limiting their criticality levels and kinds. Figure 3 shows the domain division proposed by the ARINC 664 standard [1] currently used in civil avionic systems.

The separation in domains allows the separate development of the systems composing each domain and limits the maximum safety level required in each domain. It also has the advantage of limiting the security requirements between the ACD domain and the other domains. Basically, there is a single connection between the ACD domain and the other domains (through the AISD domain), but this connection is unidirectional, from domain ACD to AISD, and thus no attack can be performed on systems running on the ACD domain.

Within ACD the domain multiple systems (also called functions) with different safety criticalities exist, but not DAL E. These systems might communicate with each other. The communication channels are statically defined so a system doesn't receive unintended communications. Strict communication rules are defined to avoid that the highest criticality systems are not impacted by lower ones. For example DAL A systems might send and receive data from DAL A systems, but only send data to lower criticality systems. This ensures that a high critical system is not impacted by an incorrect data received from a low critical system. Obviously a low critical system will not send an incorrect data on purpose, but as the system was developed with a low criticality in mind some failure verifications are not done and a send of incorrect data might happen.

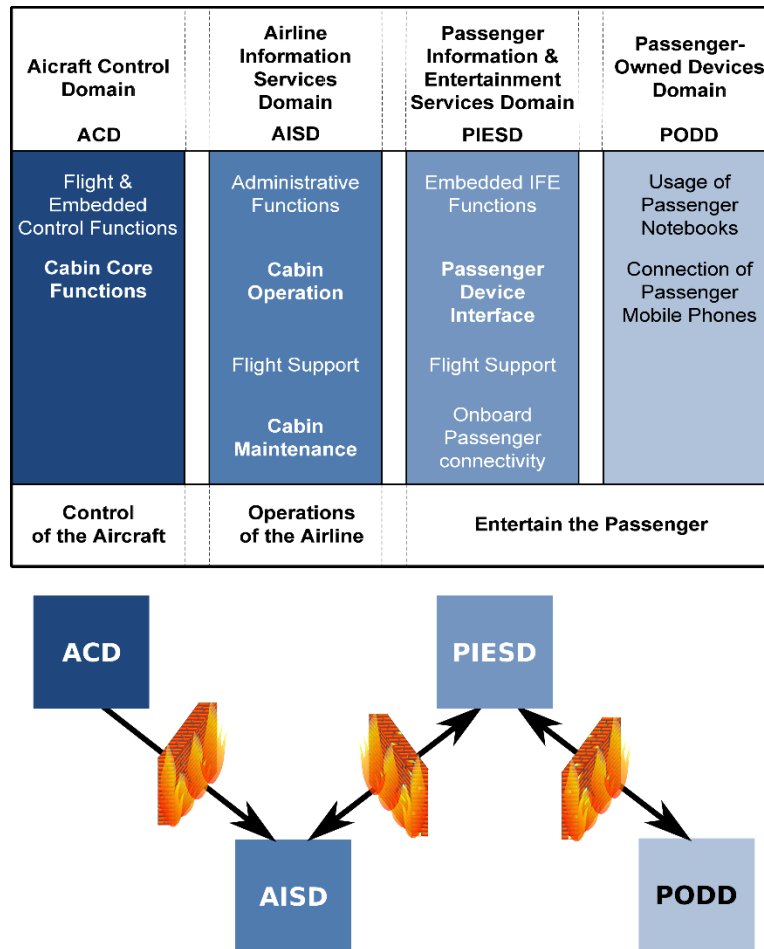


Figure 3 Avionics domains

Typically a system/function runs in a single computing resource (i.e., a single processor machine), but a single computing resource might run one or multiple functions. When a computing resource runs multiple functions it is called *Integrated Modular Avionics* (IMA) module [2]. IMA modules can run one or multiple functions, with different criticalities when running multiple functions, on a single computing resource.

5.1.1.1 Future plans for mixed-criticality systems

The limits of current single core computing resources are being reached on current IMA modules. There is a need for more computing intensive functions and a wish for further integration of functions within an IMA module. In the past the performance of single core processors was regularly improved, however this tendency has stopped and today processors with better performance are multi cores. This makes of multi cores processors the natural candidates for IMA modules.

However, before they can be used on avionic systems, multi core processors and/or their usage must respect the domain requirements, specially the time partitioning requirement introduced in the next section (or equivalent if these need to be redefined with the usage of multi cores). At the same time the proposed new multi cores and/or the usage of multi core COTS (Components Off The Shelves) processors must respect the initial market requirement: performance. For example, [3] develops a solution allowing the integration of functions with different criticality on multi cores at the OS level for the railway domain, but its applicability is limited and the performance goal only partially achieved. The

authors propose that when a critical function is executed only one core is used in the system, i.e., the other cores remain idle. However, this solution doesn't improve the integration of multiple critical functions (typically DAL A, B or C) on a multi core¹ when compared against an equivalent single core processor. Additionally, it doesn't improve the performance of a single critical function as it doesn't provide a solution to develop critical functions using multiple cores.

Finally, if the performance goal is achieved while respecting the domain requirements, the usage of shared resources that are not part of the processor must be considered. It is specially the case of the communication facilities, like the AFDX [4] or TTEthernet [5] connections. Multi cores would allow the integration of functions that make heavy use of the communication interfaces, and it must be studied if these interfaces can hold them.

5.1.1.2 Boundary conditions and requirements in the domain

The civil avionics domain is a highly regulated environment. When referring to the IT systems integrated on a plane two standards are currently considered (among others):

- DO254/ED-11 [6] refers to the development of hardware systems. Basically it specifies the development process of the different hardware systems on a plane, but not its functionality.
- D0178/ED-12 [7] deals with the safety of software used in critical functions. So unlike the D0254 it not only considers the development process of the software but also defines some safety related characteristics it must respect.

While the D0178 puts itself at the software level, it should be noted that the safety requirements described in it can be supplied by the hardware, but the analysis if they are respected or not are done at the software level. Aside of the development process requirements the main safety related characteristic that avionic software must respect are:

- Spatial partitioning: basically defines that data (code and data) and I/O resources of a function should not be modified by other functions.
- Time partitioning: a function should only be able to use the allocated resources during its scheduled period of execution.
- Failure partitioning: a malfunctioning function (which may be caused by a malfunctioning hardware allocated to the function) should not be propagated to other functions running on the system.

The first and the last requirements are fairly straightforward supported in current single core systems. Spatial partitioning is ensured by a combination of hardware mechanisms (multiple levels of execution like operating and application, and MMU) and the usage of a high privileged task (the operating system) that manages the MMU for the functions. Failure partitioning is typically ensured by the same mechanisms (hardware and operating system). These two requirements force that the resources required by the applications are now are design time. Time partitioning enforces that the worst case execution time

¹ Which is the typical scenario of an IMA system.

(WCET) of the applications is known to define the allocated time of the resources for the application. The requirement doesn't impede that two functions using two different resources could be run simultaneously, for example a function could be using the core of the processor while a second function could be using a DMA engine simultaneously. However, this is currently avoided because such situations cause *interferences* between the two functions that make the computation of their WCET difficult or impossible to compute, and even estimations are frequently extremely pessimistic. Thus the current solution on single core processors to respect the time partitioning requirement is to rewrite it with a stricter version: a function should only be able to use the allocated resources during its scheduled period of execution *and only a function is executed at a time in the processor*. While this solution slightly degrades the peak performance that can be achieved by using the single core processor, it is not scalable to multi core processors. It would only allow the usage of one core at a time, which clearly breaks the purpose of using multi core processors.

Solutions for effective use of multi core processors while respecting the previous requirements are needed, and then it must be validated that the solutions can support the development process defined by the DO178 standard.

5.2 Mixed-criticality in the wind power domain

The supervision and control system centralizes the intelligence of wind turbines, though many other computing platforms are deployed in the turbine to control local processes in subsystems.

The main purpose of this system is to supervise and control all the distributed subsystems that compose the wind turbine through a field bus interface. Strict real-time constraints apply to the core of the system, so it shall be a highly tested, robust and reliable embedded system based on a RTOS.

However, this system could be already considered as mixed-criticality since it combines these core real-time functionalities with some other less critical features. In this second group, the human-machine interface and communication abilities could be included. However, the interaction between these two criticality levels is usually not appropriately handled since mechanisms to guarantee independence are not available. The lack of those mechanisms avoids the addition of safety tasks to this platform, since in this case strict regulations apply and make it necessary (see section "Boundary conditions and requirements in the domain").

5.2.1.1 Future plans for mixed-criticality systems

A very appropriate scenario for integration of mixed-criticality systems in the wind power domain is the addition of safety or protection functionalities in the supervision and control system. If current limitations are overcome, the following system composition could be proposed.

The protection system is in charge of maintaining the wind turbine in a safe state. The main functionality of the protection system is to assure that the design limits of the wind turbine are not exceeded. The protection functions shall be activated as a result of a failure of the control function (running in the supervisory system) or of the effects of an internal or external failure or dangerous event.

Currently, the protection system is usually implemented in an independent module integrated in the field bus. This solution lacks flexibility due to the inherent limitations of a commercial hardware based system. Therefore it is not possible to program complex logics, but only basic operations with digital inputs and outputs.

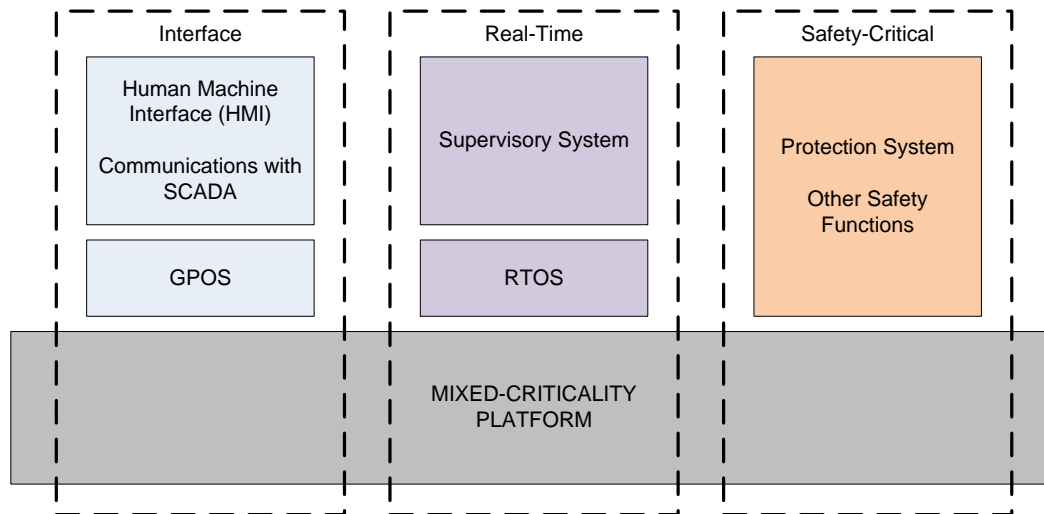


Figure 4: System Overview

The realization of this mixed-criticality architecture would bring important benefits and allow overcoming the limitations of federated architectures, in terms of complexity, scalability, reliability and cost-size-weight factors. The challenge is to provide sufficient evidence of isolation, separation and independence among safety and non-safety related functions.

5.2.1.2 *Boundary conditions and requirements in the domain*

Wind turbines are machines according to the definition of the EU Machinery Directive [1]. Therefore they must be compliant with directive 2006/42/EC, and meet functional safety specifications.

IEC 61400 standard [2] is especially relevant as it focuses on wind turbine systems. However, currently it does not include any instruction on how to design safety-relevant parts of the control system although proposals in the latest draft version start to refer to the process of the established standards of the ISO 13849 [3] and IEC 62061 [4] for this task, since they are appropriately harmonized. They both reference IEC 61508 [5] domain independent standard.

The Guideline 2010 “Guideline for the Certification of Wind Turbine” [6] shall also be mentioned. This document, elaborated by GL Renewables Certification provides in Chapter 2 the safety requirements necessary to ensure that the components of the wind turbine are always kept within their operation limits. This is achieved by the implementation of protection functions, such as:

- Protection against excessive rotor speed
- Protection against excessive power production
- Protection against short circuit
- Etc.

All these functions shall be performed by the safety (protection) system. The guideline recommends performance level “PL d” for most of the protection functions, and that shall be also the performance level met by the safety system.

The proposed architecture includes different safety and control systems, as shown in the following figure, extracted from [6]. If they are deployed on the same platform, thus conforming a mixed-criticality system, appropriate measures shall be taken in order to guarantee independence.

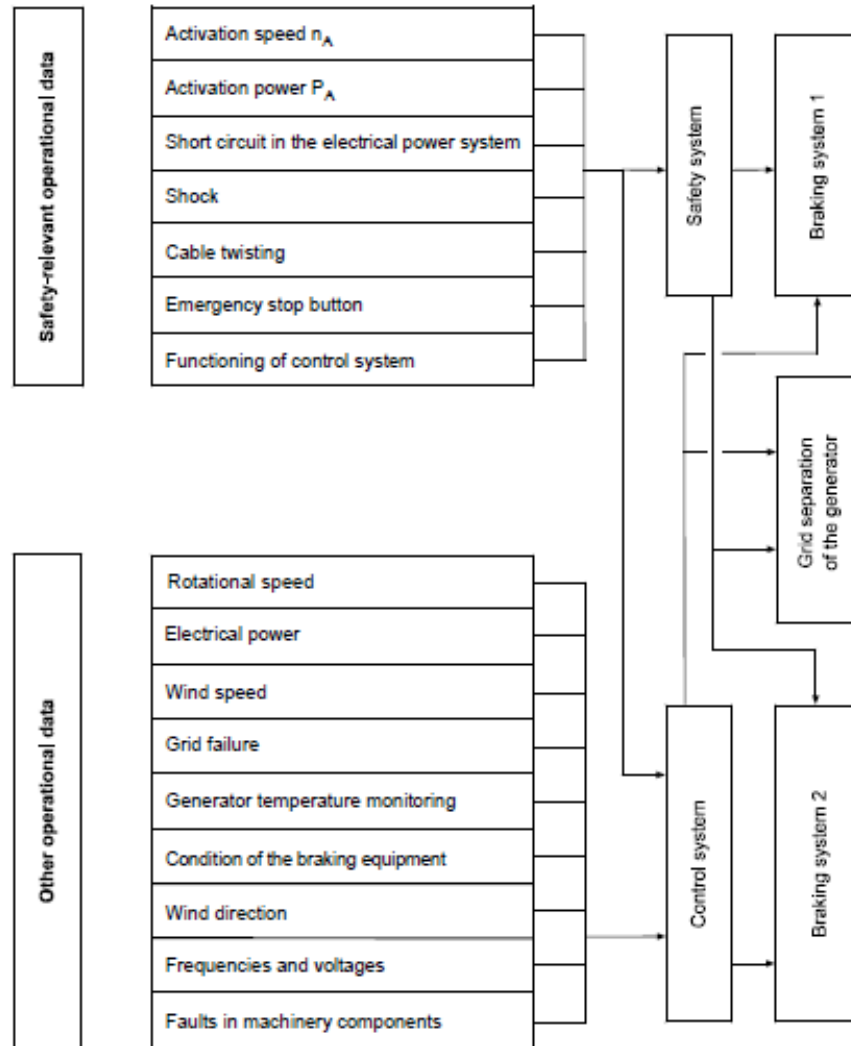


Figure 5: Control Systems

5.3 Mixed-criticality in the healthcare domain

The merging of Information Technology and Systems with Healthcare and medicine introduces new opportunities in the Clinical Healthcare sector. The constantly changing landscape of Healthcare technologies makes a fast growing area of research and development. As Healthcare monitoring devices getting smaller and gateways more powerful it is possible to think about new cost effective Healthcare services. Monitoring uses the latest telecommunication technologies to exchange patient information and provide health care services across geographic, time social and architectural barriers [Chio06]. Although, Health monitoring may be delivered not only in a hospital environment but also at home, in DREAMS we targets the Hospital patient-monitoring.

The DREAMS patient-monitoring is bare networked systems based on a combination of monitoring devices, Hospital Gateways, and telecommunication technology with both critical (those that perform critical computations) and non-critical components. The interactions among critical and non-critical components have to be carefully considered in order to ensure the safe operation of the overall healthcare system. Healthcare systems have to be aware of this difference and ensure that non critical functionality does not affect the operation of the critical aspects of the healthcare functions. Following the guidelines used in other domains such us (the civil avionics systems) it is important to divide the whole systems into different domains. Figure 3 shows the domain division as initially thought in WP8. As we can see we have identified 4 domains that are mapped to 3 different systems. The Remote Monitoring domain is the one mapped on the remote monitoring device while the user domain is the one mapped to the owned user devices such as tablets or mobile phones. The last and the most important one is the Healthcare and Entertainment domains mapped to the Hospital Gateway (HGW). The separation in domains allows the separate development of the overall system defining a unique way to communicate each other.

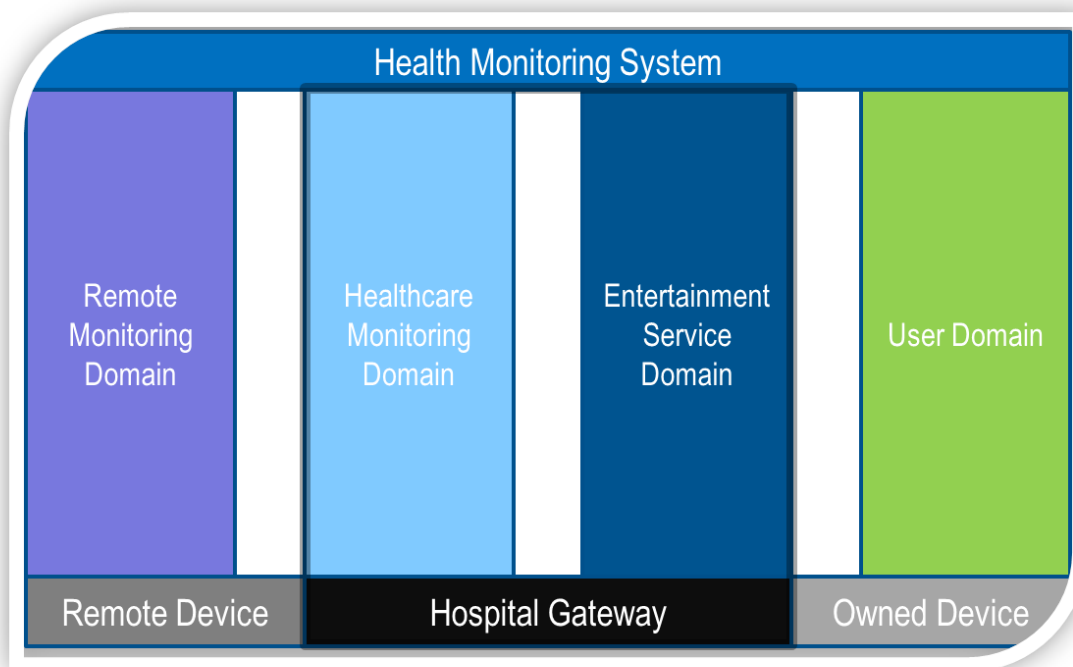


Figure 6 Healthcare domains

The HGW is the heart of the care system where different criticalities exist. The gateway utilizes several Body Monitoring devices for general inquiry and health monitoring services. The body monitoring device is a mobile, flexible cardiac monitoring technology that allows physicians to monitor important biometric patient data and helps to maintain a constant connection between patients and their care teams. The Data sent from the Body monitoring to the HGW can be analyzed and visualized locally. In parallel the HGW can provide an Entertainment service to patients. General information content can be visualized to the room TV or visualized remotely in the owned user device such as tablets or smart phones. The Entertainment Service domain includes several functionalities with several criticalities in term of bandwidth and latency toward the common and shared resource that is the external DDR memory. Although in modern systems we can have different DDR banks and channels multiple tasks belonging to different domains running concurrently are competing for DDR bandwidth and performance. DDR bandwidth and performance can vary significantly due to access contention as already shown by [Yun13] DDR contentions can cause up to 41% of WCET increases resulting in 28% of deadline violations. This means that the Healthcare monitoring domain can be impacted by the memory interferences.

5.3.1.1 Future plans for mixed-criticality systems

Thirty years ago, health care technologists realized a simple truth: monitoring patients improves outcomes. Today Hospital's rooms are populated with dozens of devices: pulse oximeters, multi-parameter monitors, ECG monitors, Holter monitors and more. However, hospital error is still a leading cause of preventable death. Thousands of errors occur in hospitals every day. Many of these errors are caused by false alarms, slow responses, and inaccurate treatment delivery. Today, a new technology disruption is spreading through patient care. Advanced device connectivity will change medical practice, lower costs, and improve patient outcomes. The healthcare industry is evolving from standalone systems to networked systems, connecting devices to improve patient outcomes and replace dedicated wiring with wireless/wired networks. By networking devices, alarms can become smart, only sounding when multiple devices indicate errant physiological parameters. By connecting measurements to treatment, smart drug delivery systems can react to patient conditions much faster and more reliably than busy hospital staff. By tracking patients around the hospital and connecting them to cloud resources, efficiency of care can be dramatically improved.

The core of the Healthcare networked system is the Hospital Gateway (HGW) integrating thousands of devices with mixed criticalities, in a large hospital environment. Since modern hospitals use hundreds of devices for patient care and monitoring presents scalability, performance and data discovery challenges. The HGW will be based on a shared memory powerful multicore processor with a particular focus on network connectivity. However, before to be used on Hospital environments, HGW must respect the specific requirements, especially in term of bandwidth (Time) partitioning, real-time, spatial and failure requirements introduced in the next section. As already presented by [Tang11] the performance of an application depends not only by the application running within a core but also by the co-runners. Today, multicore architectures are showing poor performance Isolation that is inherently acquired from the shared resources. The poor performance isolation implies high variability since OS scheduler becomes non-deterministic and unpredictable and that the QoS reserved resources are not as effective.

In order to meet the requirements of HGW we need to deal with parallelism and shared resources. In other words it is necessary to address the shared resource management capabilities that modern multicore platforms offer. Shared resource contention remains an unsolved problem in existing OS scheduling. Previous solutions focus primarily on cache contention that is not the dominant cause of performance degradation.

5.3.1.2 *Boundary conditions and requirements in the domain*

The most important technical requirements that should be addressed in the Healthcare domain focusing on patient monitoring use case are

- Spatial partitioning: basically defines that data (code and data) and I/O resources assigned to Healthcare Monitoring Domain should not be impacted by other tasks running in Entertainment Service Domain
- Failure partitioning: a malfunctioning function in Entertainment Service Domain should not be propagated to other functions running on the system.
- Bandwidth partitioning: some tasks with less stringent determinism and real-time requirements should only be able to use the predefined bandwidth assigned during its scheduled period of execution.
- Real-time Behavior: some functions must react in real-time, therefore they have to provide a time predictable computation and communication among different networked devices.

Spatial and failure partitioning are ensured by a combination of hardware mechanisms (MMU, IOMMU, NoC firewalls and redundancy) and software such as system virtualization. Considering the current shared-memory multicore architectures of HGW the consolidation of Bandwidth driven tasks together with critical Real-time tasks, poses significant challenges due to interferences on shared resources such as system bus and memory. As shown [Pelizzoni 10] that a task can suffer 300% WCET increase due to memory interference even when tasks spend only 10% of their time on fetching memory in an eight core system. In fact, a typical shared memory multi-core HGW system includes shared system bus and memory controller that arbitrates memory read write requests among cores and hardware accelerators. Today arbitration scheme in the bus and memory controllers tries to maximize the bus and memory bandwidths without considering the different criticalities of memory requests. As a consequence, individual request can be delayed thus missing the related application requirements. In particular memory performance in multicore HGW platforms can vary significantly depending on how data are located in the DDR banks and how the banks are shared among the cores and hardware accelerators at a given time. When all cores or accelerators are accessing data located in different memory banks, requests can be processed in parallel. On the other hand, when all cores are accessing data located in the same memory bank at the same time, requests would be delayed due to contention in the bank. Unfortunately, today's operating systems and middleware view DDR as a single resource and do not consider banks when allocating memory. Therefore, the exact locations of the allocated memory over the banks are unpredictable. Moreover, memory controllers are typically configured to interleave the banks in order to improve bank-level parallelism. This further exacerbates the problem because multiple programs running on different cores at a given time are likely to share banks, even though they do not share memory space.

Effective and cost driven solutions to use shared memory multi-core system for HGW while respecting the previous requirements are needed, and then it must be validated on the field.

Part B Requirements

1 Requirements for the Architecture

This section describes the requirements of the DREAMS architecture for mixed-criticality systems (herein called the architecture for short). The architectural requirements combine safety, fault tolerance and real-time performance with data, energy and system integrity considerations. In the following discussion a description of the main sections of this chapter and their relationship to other sections is presented.

Section 1.1 presents the requirements related to the safety aspects. Safety is a primary measure of success thus it has a significant impact on the overall DREAMS architecture. The stability and integrity of the services as well as temporal and spatial partitioning shall be supported and provided at on-/off-chip levels. Temporal order and synchronization requirements as part of the timing constraint are presented in section 1.2. Requirements for fault detection and health monitoring are illustrated in section 1.3, in which the availability of fault information and fault models with adequate detection strategies is demanded. Section 1.4 includes fault-tolerance requirements like determinism of replicated components and support of active redundancy.

The required core architectural services are introduced in section 1.5. Furthermore, customization and refinement of the core services using domain-specific higher services are introduced. Evolvability and scalability requirements provided in section 1.6, which emphasize the DREAMS architecture need to leverage multi-core platforms for a system perspective of mixed-criticality applications combining the chip-level and cluster-level. This shall be achieved using gateways and remote access to virtualized resources.

Requirements for technology independence are introduced in section 1.7. The technology independence supports different underlying implementation options for each of the core platform services. Exploitability requirements are presented in section 1.8, allowing to combine top-down and bottom-up design styles. Reduced development cost, effort and complexity management are illustrated in section 1.9.

DREAMS proposes a distributed architecture that requires a uniform namespace to successfully identify subsystems. The coexistence of different criticalities and computational models in the DREAMS architecture requires the support of heterogeneity as shown in section 1.10. This will be provided using explicit message-based communication primitives in combination with shared memories as well as different timing models for computation and communication.

Another key point of the DREAMS architecture is the support for virtualization to improve the performance as provided in section 1.11. Low power management is addressed in section 1.12 and required services for the DREAMS architecture are introduced in section 1.13.

The architecture requirements are strongly related to the other requirements categories in this deliverable. For example, more detailed requirements at chip level and cluster level are defined in sections 2 and 3. In particular, the architecture requirements are the baseline for the safety concept, where detailed safety requirements address system safety as presented in section 5. Also, the identified services and extra-functional properties of the architecture are the baseline for the reconfiguration and adaptation upon foreseen and unforeseen changes in the operational and environmental conditions (see section 10).

1.1 Safety (Measure of Success)

ID	R 1.1.1	
Topic	Architecture	
Subtopic	Safety (Measure of Success)	
Name	Stability of Prior Services upon Integration of Mixed-Criticality	
Responsibility	WP	1,2,3
	Lead partner	USIEGEN
	Participating partners	USIEGEN
Description	The architecture shall assure that the behaviour of a subsystem in the value and time domain before integration into a larger systems equals the behaviour after integration.	
Rationale	The stability-of-prior-services principle is essential to allow modular certification, independent design, development, and verification of subsystems.	
Significance	High	
Means for validation/verification	TRT, ALSTOM and ST will perform a validation of the stability of priori services as part of the demonstrators (T6.3, T7.3, T8.3). The incremental integration and implementation of the application subsystems for the demonstrators on the DREAMS platform will serve as an experimental evaluation of the stability of prior services.	
Source	ARTEMIS SRA	
Additional Information		

ID	R 1.1.2	
Topic	Architecture	
Subtopic	Safety (Measure of Success)	
Name	Temporal partitioning	
Responsibility	WP	1,2,3
	Lead partner	USIEGEN
	Participating partners	USIEGEN, UPV, FENTISS, RTAW, ST, TEI,TTT,TRT
Description	The architecture shall implement temporal partitioning and each partition shall be able to access shared resources (e.g., network, shared processor) with a priori defined temporal constraints. The	

	temporal constraints include the latency, the jitter and the duration of availability during a scheduled access.
Rationale	<p>A partition cannot affect the ability of other partitions access shared resources, such as the network or a shared CPU. This includes the temporal behaviour of the services provided by resources (latency, jitter, duration of availability during a scheduled access).</p> <p>Temporal partitioning shall be established by the network, by hypervisors and through physical separation.</p>
Significance	High
Means for validation/verification	<p>Temporal partitioning of the core platform services will be validated at module-test level and integration-test level in WPs 2, 3 and 4.</p> <ul style="list-style-type: none"> • WP leaders will lead the experimental evaluation of temporal partitioning as part of the integration in tasks in the respective WPs <ul style="list-style-type: none"> ○ ST: T2.4 ○ TTT: T3.4 ○ RTAW: T4.4 • Technology partners will contribute to the experimental validation of temporal partitioning for their technological building blocks <ul style="list-style-type: none"> ○ Message-based communication (USIEGEN in T2.1) ○ Shared memory (ST, TEI in T2.1) ○ Hypervisor (UPV, VOSYS in T2.2) ○ Cluster-Level Communication (TTT in T3.1) ○ Resource Management (TUKL in T3.2) <p>USIEGEN will participate in the experimental evaluation of temporal partitioning of the time-triggered extension of STNOC and the end-to-end channels in Healthcare and Avionic Demonstrators (Task T2.4)</p>
Source	WG Avionics, DOW, Genesys, ACROSS, standards (DO-178B/C)
Additional Information	

ID	R 1.1.3
Topic	Architecture
Subtopic	Safety (Measure of Success)
Name	Spatial partitioning

Responsibility	WP	1,2,3
	Lead partner	USIEGEN
	Participating partners	USIEGEN, UPV, FENTISS, ST, TEI,TTT,TRT
Description	The architecture shall implement spatial partitioning for shared resources.	
Rationale	<p>Spatial partitioning does not only consider the memory, but also the attached devices, network , etc. Different resources shall be supported including on-chip/off-chip communication, computational resources of processor cores, I/O and memory.</p> <p>Spatial partitioning is a foundation for mixed-criticality integration in order to establish fault containment and the absence of unintended side-effects between functions.</p> <p>Partitioning shall be established by the network, by hypervisors and through physical separation.</p>	
Significance	High	
Means for validation/verification	<p>Spatial partitioning of the core platform services will be validated at module-test level and integration-test level in WPs 2, 3, 4.</p> <ul style="list-style-type: none"> • WP leaders will lead the experimental validation of spatial partitioning as part of the integration tasks in the respective WPs <ul style="list-style-type: none"> ○ ST: T2.4 ○ TTT: T3.4 ○ RTAW: T4.4 • Technology partners will contribute to the experimental validation of spatial partitioning for their technological building blocks <ul style="list-style-type: none"> ○ Message-based communication (USIEGEN in T2.1) ○ Shared memory (ST, TEI in T2.1) ○ Hypervisor (UPV, VOSYS in T2.2) ○ Cluster-Level Communication (TTT in T3.1) ○ Resource Management (TUKL in T3.2) <p>USIEGEN will participate in the experimental evaluation of spatial partitioning of the time-triggered extension of STNOC and the end-to-end channels in Healthcare and Avionic Demonstrators (Task T2.4)</p>	
Source	WG Avionics, DOW, Genesys, ACROSS, standards (DO-178B/C)	

Additional Information	
------------------------	--

ID	R 1.1.4	
Topic	Architecture	
Subtopic	Safety (Measure of Success)	
Name	Integrity of partitions	
Responsibility	WP	1,2,3
	Lead partner	UPV
	Participating partners	UPV, FENTISS, RTAW, ST, TEI,TRT
Description	A service in one partition shall only alter code and/or private data located in its own partition.	
Rationale		
Significance	High	
Means for validation/verification	The experimental validation of memory protection and spatial partitioning of the hypervisor in T2.4 will be performed within the three demonstrators.	
Source	WG Avionics, DOW, Genesys, ACROSS, standards (DO-178B/C)	
Additional Information		

ID	R 1.1.5	
Topic	Architecture	
Subtopic	Safety (Measure of Success)	
Name	Partitioning of input/output resources	
Responsibility	WP	1,2,3
	Lead partner	USIEGEN
	Participating partners	USIEGEN, UPV, FENTISS, RTAW, TTT,TRT
Description	The architecture shall support partitioning of input/output resources.	
Rationale	A partition shall only interfere with control of external devices (e.g., actuators) of its own partition.	
Significance	High	

Means for validation/verification	Same means for validation as R 1.1.2
Source	WG Avionics, DOW, Genesys, ACROSS, standards (DO-178B/C)
Additional Information	Protection of I/O can be realized by mapping I/O access to time-triggered messages on the NoC and off-chip networks.

ID	R 1.1.6	
Topic	Architecture	
Subtopic	Safety (Measure of Success)	
Name	Fail operational support up to highest criticality levels (10^{-9} failures per hour)	
Responsibility	WP	1,2,3
	Lead partner	ONERA and TRT
	Participating partners	USIEGEN, ONERA, TRT, TUKL
Description	<p>In order to support safety up to highest criticality levels, the architecture shall support <i>fault-tolerance strategies</i> that enable the continued operation of the system in the presence of component failures due to transient and permanent physical faults. The fault-tolerance strategies shall support the highest criticality levels (10^{-9} failures per hour) based on typical component failure rates.</p> <p>Different target safety levels with corresponding different fault-tolerance strategies shall be supported in the same system to achieve a balanced trade-off between cost and fault-tolerance of individual subsystems.</p> <p>The platform and system should be able to integrate multiple applications with different criticality levels (e.g., in avionics from DAL A to DAL E, that is from very critical to not critical at all).</p>	
Rationale	<p>Since component failure rates are usually in the order of 10^{-5} to 10^{-6}, the highest criticality levels (10^{-9} failures per hour) require the system as a whole to be more reliable than any one of its constituent components.</p>	
Significance	High	
Means for validation/verification	The requirement will be assessed in the three demonstrators (T6.3, T7.3, T8.3, T1.8) with the support of the technology providers (WP1, WP2, WP3 and WP4).	
Source	WG Avionics, DOW	

Additional Information	
------------------------	--

ID	R 1.1.7	
Topic	Architecture	
Subtopic	Safety (Measure of Success)	
Name	Fault hypothesis	
Responsibility	WP	1,2,3
	Lead partner	USIEGEN
	Participating partners	USIEGEN,ST,TEI,TTT,RTAW,UPV,VOSYS,IKL,TRT, ST, ALSTOM
Description	<p>The architecture shall be based on a fault hypothesis which identifies the assumptions regarding the units of failure and the types of faults that the resulting system is supposed to handle.</p> <p>The fault hypothesis must define the fault-containment regions for design faults, transient hardware faults and permanent hardware faults.</p> <p>The fault hypothesis shall support the hierarchical system structure of a DREAMS system with different containment coverage at chip and cluster level for physical faults.</p>	
Rationale	<p>The justification for building safety-critical systems from replicated resources rests on an assumption of failure independence among redundant units. For this reason the independence of fault-containment regions is of critical importance. The independence of fault-containment regions can be compromised by shared physical resources (e.g., power supply, timing source), external faults (e.g., EMI, spatial proximity) and design.</p>	
Significance	High	
Means for validation/verification	USIEGEN will evaluate the fault hypothesis as part of the consolidation of assessment (T1.8).	
Source	DOW	
Additional Information		

1.2 Timing requirements (Measure of Success)

ID	R 1.2.1
----	---------

Topic	Architecture	
Subtopic	Timing requirements (Measure of Success)	
Name	Predictability of Services	
Responsibility	WP	1,2,3,4
	Lead partner	USIEGEN
	Participating partners	USIEGEN,UPV,TUKL,TTT,VOSYS,RTAW
Description	<p>The architecture shall support reasoning about the temporal properties of an application and to provide guarantees for the timely execution of an application service for safety-critical applications.</p> <p>In order to analyse application-specific temporal constraints, the architecture shall enable the development of software for which the calculation of tight bounds on the WCET is possible with feasible effort. The key to calculate tight bounds on the WCET with feasible effort is deterministic behaviour in each layer of the system that is to be analysed.</p> <p>Temporal predictability shall be established for:</p> <ul style="list-style-type: none"> • <i>Communication services</i> between partitions including end-to-end communication channels in a system with networked multi-core chips • <i>Execution services</i> including partition scheduling , hypervisors and memory access • <i>Resource management services</i> including reconfiguration within predictable time and predictable reconfiguration results 	
Rationale	Timeliness is vital property for the majority of safety-critical embedded applications addressed by the DREAMS architecture	
Significance	High	
Means for validation/verification	<p>USIEGEN will experimentally and analytically evaluate the predictability of the end-to-end communication channels (T2.4).</p> <p>UPV and VOSYS will experimentally and analytically evaluate the predictability of the execution services (T2.4).</p> <p>TTT will experimentally and analytically evaluate the predictability of the cluster-level communication (T3.4).</p> <p>TUKL will experimentally and analytically evaluate the predictability of the resource management services (T3.4)).</p>	

Source	GENESYS (partly)
Additional Information	

ID	R 1.2.2	
Topic	Architecture	
Subtopic	Timing requirements (Measure of Success)	
Name	Temporal order	
Responsibility	WP	1,2,3
	Lead partner	USIEGEN
	Participating partners	USIEGEN,TTT,TUKL,RTAW
Description	The architecture shall ensure that all safety-critical subsystems of a system see a sequence of critical events (e.g. reception of messages) in the same order or can re-establish the temporal order.	
Rationale	Since the temporal order of incoming events can have an influence of the internal state of a subsystem, a consistent view on the temporal order of critical events is generally necessary to achieve replica determinism.	
Significance	High	
Means for validation/verification	USIEGEN will analytically evaluate this property for the end-to-end communication channels (T2.4, T1.8).	
Source	GENESYS	
Additional Information		

ID	R 1.2.3	
Topic	Architecture	
Subtopic	Timing requirements (Measure of Success)	
Name	Bounded delays and jitter	
Responsibility	WP	2,3
	Lead partner	USIEGEN

	Participating partners	USIEGEN,UPV,TUKL,TTT,ST,TEI,RTAW
Description	The communication service shall support the transfer of messages with known and bounded latency and bounded jitter.	
Rationale	The communication service is a core platform service of the architectural style. Known and bounded latencies of the communication channels are required to guarantee upper bounds on the response time of distributed services and on error detection times. Bounded jitter of the communication channels is required when the system has to react at a specific point in time. This is especially important if multiple distributed actions have to be temporally coordinated.	
Significance	High	
Means for validation/verification	USIEGEN will analytically evaluate this property for the end-to-end communication channels (T2.4, T1.8). TTT will analytically and experimentally evaluate the timing requirements for the cluster-level communication services (T3.1, T3.4).	
Source	DOW / GENESYS	
Additional Information		

ID	R 1.2.4	
Topic	Architecture	
Subtopic	Timing requirements (Measure of Success)	
Name	Synchronized activities	
Responsibility	WP	2, 3
	Lead partner	TTT
	Participating partners	ALSTOM, USIEGEN, FENTISS, IKL
Description	The time services of the architecture should support the synchronization of the application services and platform services of the system via a global synchronization mechanism.	
Rationale	In a distributed system, the architecture must be prepared to provide a mechanism to synchronise the different subsystems and processing units.	
Significance	Medium	

Means for validation/verification	The requirement will be validated within module-tests of the implementation in WP2, 3 and integration test in T1.7, T2.4, T3.4.
Source	ALSTOM / GENESYS
Additional Information	

ID	R 1.2.5	
Topic	Architecture	
Subtopic	Timing requirements (Measure of Success)	
Name	Bandwidth/throughput definition, verification and monitoring.	
Responsibility	WP	1,2,3
	Lead partner	TTT, ST
	Participating partners	ONERA, RTAW, TTT, FENTISS, TUKL
Description	The architecture shall provide means for resources and resource utilization needs definition, verification and monitoring.	
Rationale	Those properties are needed to be able to compute execution times and runtime management.	
Significance	High	
Means for validation/verification	The requirement will be validated by TTT and ST within module-tests of the implementation in WP2, 3 and integration test in T1.7, T2.4, T3.4.	
Source	DoW, Standards	
Additional Information		

ID	R 1.2.6	
Topic	Architecture	
Subtopic	Timing requirements (Measure of Success)	
Name	Network analyzability (on-chip, off-chip, end-to-end)	
Responsibility	WP	1,3,4
	Lead partner	TTT
	Participating partners	TTT,RTAW

Description	For high- and medium-level critical messages, the communication services shall allow that deterministic bounds on worst-case communication latency and communication jitter can be calculated.
Rationale	The communication services are designed to transport messages of application subsystems with differing criticality levels. There will be at least three levels of criticality: high, medium, and low. The criticality of a message is the criticality of the respective application subsystem. For high and medium critical messages the network allows that analytical methods can be applied to derive deterministic bounds on latency and jitter.
Significance	High
Means for validation/verification	This property of cluster-level communication will be analytically evaluated by TTT and RTAW in the context of T1.8 and T4.4.
Source	ARTEMIS SRA
Additional Information	

ID	R 1.2.7	
Topic	Architecture	
Subtopic	Timing requirements (Measure of Success)	
Name	WC*T computation	
Responsibility	WP	4, 1
	Lead partner	USIEGEN
	Participating partners	ONERA, RTAW, TTT, UPV, TUKL, USIEGEN, ST, TEI
Description	The execution time of the core platform services of the architecture shall be time-bounded, that is the WC*T of the core platform services shall be possible to obtain.	
Rationale	The WC*T computation is a hard requirement of time-dependent safety-critical systems.	
Significance	High	
Means for validation/verification	<p>The technology partners shall provide information about WC*T of the developed core platform services.</p> <ul style="list-style-type: none"> • USIEGEN, ST, TEI, TTT: Determination of WC*T of on and off chip communication services in Task T2.4 and T3.4 • VOSYS, UPV, FENTISS: Determination of WC*T of execution services in Task T2.4 	

	<ul style="list-style-type: none"> • TUKL: Determination of WC*T of resource management services in Task T4.4 • ONERA: Determination of WC*T of fault recovery in Task T2.4 and T4.4
Source	WG Avionics, DoW, Standards, CERTAINTY,....
Additional Information	

ID	R 1.2.8	
Topic	Avionics	
Subtopic	Timing requirements (Measure of Success)	
Name	Minimum period cycle	
Responsibility	WP	1,2
	Lead partner	USIEGEN
	Participating partners	ONERA, RTAW, TTT, FENTISS, TUKL
Description	The architecture shall support a minimum period cycle (and associated deadline detection support) less than or equal to 50ms.	
Rationale	Due to the typical implementation of safety critical systems the minimum period cycle is one of the parameters at the heart of the application definition.	
Significance	High	
Means for validation/verification	The requirement will be experimentally evaluated by TRT in the avionic demonstrator (T6.3).	
Source	WG Avionics, TRT	
Additional Information	This limit is due to the current implementation of avionic solutions, but support for smaller deadlines is welcome.	

1.3 Fault Detection and Health Monitoring

ID	R 1.3.1	
Topic	Architecture	
Subtopic	Fault Detection and Health Monitoring	
Name	Health monitoring	
Responsibility	WP	1, 2, 3

	Lead partner	ONERA
	Participating partners	ONERA, TTT, FENTISS, TUKL,ST, TEI, TRT
Description	The architecture shall provide means for health monitoring. A fault model shall be defined. For each fault, adequate detection strategies shall be defined (e.g., checking the correctness of services provided in partitions, checking the availability of resources).	
Rationale	Faults such as overuse of shared resources, deadline exceeding and rules violation need to be monitored in order to react accordingly in a safety-critical system.	
Significance	High	
Means for validation/verification	The requirement will be assessed using the demonstrators (T6.3, T7.3, T8.3, T1.8) and validated at module-test level and integration-test level in WPs 1, 2, 3.	
Source	Windpower WG, Avionics WG, DOW, Genesys, ACROSS, standards, ONERA	
Additional Information		

ID	R 1.3.2	
Topic	Architecture	
Subtopic	Fault Detection and Health Monitoring	
Name	System faults information	
Responsibility	WP	1, 2, 3
	Lead partner	ONERA
	Participating partners	ONERA, TTT, FENTISS,TUKL
Description	The architecture shall provide means to provide fault information to the applications (i.e., lost messages) and system.	
Rationale	The applications receive information about faults occurring at the lower levels in order to take correction actions.	
Significance	High (in some designs the decisions are only taken by the OS/task scheduler and the significance becomes low)	
Means for validation/verification	The requirement will be assessed using the demonstrators (T6.3, T7.3, T8.3, T1.8) and validated at module-test level and integration-test level in WPs 1, 2, 3.	

Source	WG Avionics, DOW(precision), standards, industrial development (TRT), ...
Additional Information	WP3 is included to indicate that network faults should also be considered.

ID	R 1.3.3	
Topic	Architecture	
Subtopic	Fault Detection and Health Monitoring	
Name	Support for fault detection by the application	
Responsibility	WP	1
	Lead partner	ONERA
	Participating partners	ONERA, TTT, FENTISS, TUKL
Description	The architecture shall provide the applications with means to inform the GRM that faults have been detected.	
Rationale	The applications should be able to provide fault information (unexpected behavior) so the system can take high level decisions (e.g., for fault containment).	
Significance	High	
Means for validation/verification	The requirement will be assessed using the demonstrators (T6.3, T7.3, T8.3, T1.8) and validated at module-test level and integration-test level in WPs 1, 2, 3.	
Source	WG Avionics, DOW(precision), standards, industrial development (TRT)	
Additional Information		

1.4 Fault-Tolerance

ID	R 1.4.1	
Topic	Architecture	
Subtopic	Fault-Tolerance	
Name	Replica Determinism and Non-Deterministic Subsystems	
Responsibility	WP	1
	Lead partner	USIEGEN

	Participating partners	USIEGEN,TTT
Description	<p>The architecture has to ensure replica determinism of replicated components of safety-critical subsystems.</p> <p>Indeterminism must be supported for non safety-critical subsystems without fault-tolerance requirements.</p> <p>Non-deterministic components of non safety-critical subsystems shall not affect replica determinism of safety-critical subsystems.</p>	
Rationale	<p>A set of replicated components is replica determinate if all the members of this set have the same encapsulated state, and produce the same output messages at points in time that are at most an interval of d time units apart (as seen by an omniscient external observer). Replicated components have to be replica determinate if voting should be done by bit-by-bit comparison of the component's output messages. The advantage of bit-by-bit comparison is that it can be systematically applied, and does not require any application specific knowledge about the values that have to be compared.</p>	
Significance	High	
Means for validation/verification	The requirement will be analytically evaluated by USIEGEN in the architecture assessment (T1.8).	
Source	USIEGEN	
Additional Information		

ID	R 1.4.2	
Topic	Architecture	
Subtopic	Fault-Tolerance	
Name	Support for active redundancy with replication involving multiple DREAMS chips (i.e., networked multi-core chips)	
Responsibility	WP	1,2,3,4
	Lead partner	USIEGEN
	Participating partners	USIEGEN,FORTISS
Description	<p>The architecture has to support error detection and error masking by the use of replicas and voting mechanisms (e.g. self-checking pair, triple modular redundancy).</p> <p>The architecture shall support replicas (1) as partitions on the same processor core, (2) as partitions on different processor cores of the same chip or (3) as partitions on different chips.</p>	

Rationale	In a system with networked multi-core chips, the prerequisites for active redundancy (e.g., replica determinism) shall be realized independently of the location of the replicas.
Significance	
Means for validation/verification	The requirement will be analytically evaluated by USIEGEN in the architecture assessment (T1.8).
Source	DOW
Additional Information	

ID	R 1.4.3	
Topic	Architecture	
Subtopic	Fault-Tolerance	
Name	Real-time fault recovery strategies	
Responsibility	WP	WP1, WP4
	Lead partner	ONERA
	Participating partners	TRT, TUKL
Description	For each fault identified based on R 1.3.1 adequate recovery strategies shall be defined such as the switching between off-line scheduling tables at runtime.	
Rationale	Increase of reliability, management of mixed-criticality	
Significance	High	
Means for validation/verification	The requirement will be assessed using the demonstrators (T6.3, T1.8) and validated at module-test level and integration-test level in WPs 1, 4.	
Source	DoW or similar	
Additional Information	Refinement of R 10.1.1 (GRM) in order to recover from faults	

1.5 Domain-independence (Measure of Success)

ID	R 1.5.1	
Topic	Architecture	
Subtopic	Domain-independence (Measure of Success)	
Name	Domain-independent core services	
Responsibility	WP	1

	Lead partner	USIEGEN
	Participating partners	USIEGEN, TTT, TUKL, UPV, ST, TEI
Description	The core platform services of DREAMS shall encompass communication services, time services, execution services and resource management services.	
Rationale	<p>The architecture shall introduce a minimal set of mandatory architectural services called core services for ensuring the required properties of the DREAMS architecture (e.g., real-time, safety, reliability, isolation, security). The core services shall serve as the foundation for the construction of higher platform services and application services in all considered application domains.</p> <p>These core services also lay the foundation for exploiting the economies of scale as they can be implemented in a space and energy efficient way in hardware for a multitude of application domains.</p>	
Significance	High	
Means for validation/verification	The requirement will be experimentally evaluated by TRT, ALSTOM and ST through the use of the core platform services in demonstrators from three different domains (T6.3, T7.3, T8.3).	
Source	DOW	
Additional Information		

ID	R 1.5.2	
Topic	Architecture	
Subtopic	Domain-independence (Measure of Success)	
Name	Modularity and Intellectual-Property Software-Modules	
Responsibility	WP	1,2,3
	Lead partner	USIEGEN
	Participating partners	USIEGEN, UPV, ST, TEI, FORTISS,VOSYS,UPV,FENTISS,TUKL
Description	<p>The architecture shall be a modular architecture supporting the customization and refinement of the core services using domain-specific higher services.</p> <p>Domain-specific higher services shall be supported via middleware layers within application cores or as system cores.</p>	

	The architecture shall support the integration of components based on the interface specifications, i.e., without having to understand the internals of the components (e.g., integration of precompiled software modules).
Rationale	Different domains have different requirements with respect to functionality and extra-functional constraints. Domain-specific architectural services can refine the core services to provide a platform for specific application domains, while enabling the exploitation of the economies of scale through core services. Third party suppliers may wish to deliver only precompiled code in order to protect their intellectual property (i.e. the source code).
Significance	High
Means for validation/verification	The requirement will be experimentally evaluated by TRT, ALSTOM and ST through the use of the core platform services in demonstrators from three different domains and the realization of domain-specific services on top of the core platform services (T6.3, T7.3, T8.3).
Source	DOW and ARTEMIS SRA
Additional Information	

1.6 Evolvability and Scalability

ID	R 1.6.1	
Topic	Architecture	
Subtopic	Evolvability and Scalability	
Name	Networked multi-core chips for resource requirements beyond a single DREAMS chip	
Responsibility	WP	1,2,3
	Lead partner	TTT
	Participating partners	USIEGEN,TTT,ST,TEI,VOSYS,UPV
Description	<p>The architecture shall leverage multi-core platforms for a system perspective of mixed-criticality applications combining the chip-level and cluster-level. Using gateways, access to virtualized resources shall become location transparent.</p> <p>For example, the access to a remote I/O resource located on another chip shall be relayed via gateways involving gateways between on-chip and off-chip networks (i.e., vertical integration)</p>	

	<p>and possibly gateways between different types of off-chip networks (i.e., horizontal integration).</p> <p>Different resources (e.g., I/O, shared memories) shall be mapped to the message-based communication.</p>
Rationale	In many mixed-criticality systems, platforms encompassing networked multi-core chips will be required. In addition to requirements exceeding the resources of a single chip, today's technology does not support the manufacturing of electronic devices with failure rates low enough to meet the reliability requirements of ultra-dependable systems. This can only be achieved by utilizing fault-tolerance strategies that enable the continued operation of the system in the presence of component failures.
Significance	High
Means for validation/verification	The requirement will be evaluated at the cluster-level by TTT in Task T3.4. TTT will support the evaluation in the demonstrators (T6.3, T7.3, T8.3).
Source	DOW
Additional Information	

1.7 Technology independence

ID	R 1.7.1	
Topic	Architecture	
Subtopic	Technology independenceTechnology independence	
Name	Technology independence	
Responsibility	WP	1,2,3,4
	Lead partner	USIEGEN
	Participating partners	USIEGEN,TTT,IKL,ST,TEI,UPV,VOSYS,FENTISS, TRT
Description	<p>Technology independence should be provided by supporting different underlying implementation options for each of the core platform services.</p> <p>The architecture should be able to address different hardware and software solutions than those proposed in the project. These can be done for example in hardware by addressing multi-cores, but not just those provided by a single provider; another solution includes clear hardware/software interfaces/API that might allow</p>	

	<p>other technology providers to integrate their solutions in the architecture.</p> <p>For example, the core communication services should be realizable using different protocols in NoCs or off-chip networks. DREAMS should not be restricted to specific protocols, but any protocol providing the core services will be a suitable foundation for the DREAMS architecture.</p> <p>Likewise, different hypervisors should be supported as the basis for the execution services of DREAMS.</p>
Rationale	<p>Separation of design and implementation is a key aspect of the evolvability of a system. The application design does not have to be changed when the system has to be ported to a new technology generation.</p> <p>The success of the DREAMS framework depends on its applicability on different hard (and soft) systems.</p>
Significance	Medium
Means for validation/verification	<p>The requirement will be analytically evaluated for each of the core platform services provided by WP2,3,4:</p> <ul style="list-style-type: none"> • TTT will evaluate the technology independence of the cluster-level communication • USIEGEN, ST and TEI will evaluate the technology independence of the chip-level communication • UPV, FENTISS and VOSYS will evaluate the technology independence of the execution services • TUKL will evaluate the technology independence of the resource management
Source	DOW, WG Avionics, Industrial
Additional Information	

1.8 Exploitability

ID	R 1.8.1	
Topic	Architecture	
Subtopic	Exploitability	
Name	Meet-in-the-middle(top-down and bottom-up design styles are combined)	
Responsibility	WP	1,5
	Lead partner	IKL

	Participating partners	USIEGEN, SINTEF, FORTISS
Description	The architecture should allow design methodologies where top-down and bottom-up design styles are combined	
Rationale	Many real product developments follow neither a strict top-down approach nor a strict bottom-up approach.	
Significance	Medium	
Means for validation/verification	IKL will support the use of Meet-in-the-middle development approach in the Wind-Power demonstrator.	
Source	ARTEMIS SRA	
Additional Information		

1.9 Complexity management for reduced development cost and effort

ID	R 1.9.1	
Topic	Architecture	
Subtopic	Complexity management for reduced development cost and effort	
Name	End-to-end channels in networked multi-core chips with transparency through gateways	
Responsibility	WP	1,2,3
	Lead partner	TTT
	Participating partners	USIEGEN,TTT,IKL,ST,TEI
Description	<p>The architecture shall support end-to-end channels over hierarchical, heterogeneous and mixed-criticality networks. DREAMS shall introduce gateways for a system perspective of mixed-criticality applications combining the chip-level and cluster-level.</p> <p>On the one hand side, DREAMS shall support uniform communication between heterogeneous and mixed-criticality networks. Gateway services shall enable this horizontal integration at the cluster-level across different off-chip communication networks with different protocols (e.g., TTEthernet, EtherCAT, etc.), different reliabilities (e.g., fault-tolerant networks with media redundancy and active star couplers, low-cost fieldbus networks).</p>	

	<p>In addition, gateway services between NoCs and off-chip networks shall enable vertical integration through the seamless communication in hierarchical networks respecting mixed-criticality safety and security requirements.</p> <p>The gateways shall support location transparency with uniform communication between partitions in the same processor core, the same chip or on different chips.</p>
Rationale	Gateways allow transparency for the communication partners communicating through end-to-end channels.
Significance	High
Means for validation/verification	The validation of the end-to-end location-independence properties will be validated in the T6.3, T7.3 and T8.3 demonstrator assessment tasks performed by the demonstrator partners for different gateways and aspects of the heterogeneous mixed-criticality networks.
Source	DOW
Additional Information	

ID	R 1.9.2	
Topic	Architecture	
Subtopic	Complexity management for reduced development cost and effort	
Name	Namespace for uniform identification of subsystems	
Responsibility	WP	1,2,3
	Lead partner	USIEGEN
	Participating partners	USIEGEN, ST, TETTT,TUKL,FORTISS
Description	<p>The architecture shall support a unique, but uniform identification of subsystems. Therefore, a logical namespace and a physical namespace shall be distinguished.</p> <p>The physical namespace shall identify the resources of networked multi-core chips including off-chip clusters, nodes, processor cores and partitions.</p> <p>The logical namespace shall support the identification of components independently from their physical location and</p>	

	enable the provision of dedicated namespaces for subsystems, even if multiple subsystems share the same physical resources.
Rationale	Separate namespaces simplify independent development of distributed subsystems.
Significance	High
Means for validation/verification	The requirement will be analytically evaluated by USIEGEN in the architecture assessment (T1.7).
Source	DOW
Additional Information	

ID	R 1.9.3	
Topic	Architecture	
Subtopic	Complexity management for reduced development cost and effort	
Name	Correctness-by-Construction	
Responsibility	WP	1
	Lead partner	USIEGEN
	Participating partners	USIEGEN
Description	The architecture shall support predictable engineering practices that simplify or evade verification challenges.	
Rationale	The objective of correctness-by-construction is to transform a specification step by step into a correct design by applying provably correct design methods.	
Significance	High	
Means for validation/verification	The requirement will be analytically evaluated by USIEGEN in the architecture assessment (T1.7).	
Source	ARTEMIS SRA	
Additional Information		

1.10 Heterogeneity

ID	R 1.10.1
Topic	Architecture

Subtopic	Heterogeneity	
Name	Coexistence of different models computation	
Responsibility	WP	1,2
	Lead partner	USIEGEN
	Participating partners	USIEGEN, ST, TEI, TTT, FORTISS
Description	<p>The architecture shall support different models of computation with corresponding interaction mechanisms on on-chip and off-chip networks: predictable time-triggered communication, event-triggered communication with dynamic arbitration and shared memories.</p> <p>Supported models of computation shall include dataflow, time-triggered messaging and distributed shared memory.</p>	
Rationale	Mixed-criticality systems require support for application subsystems that differ not only in their criticality, but also exhibit dissimilar requirements in terms of timing (e.g., firm, soft, hard, non real-time) and different models of computation (e.g., dataflow, time-triggered messaging, distributed shared memory).	
Significance	High	
Means for validation/verification	USIEGEN will support the experimental evaluation of the requirement in the avionic and healthcare demonstrators (T1.6, T2.4).	
Source	DOW	
Additional Information		

ID	R 1.10.2	
Topic	Architecture	
Subtopic	Heterogeneity	
Name	Explicit message-based communication	
Responsibility	WP	1, 2,3 ,4
	Lead partner	USIEGEN
	Participating partners	TTT, ST, TEI, VOSYS, FENTISS

Description	Communication between components in an application subsystem should be performed explicitly using communication primitives provided by the architecture.
Rationale	Coding practice allowing re-usage of code, analysis, architecture independence, ...
Significance	Medium
Means for validation/verification	<p>The requirement will be evaluated at the chip-level by ST, TEI and USIEGEN in Task T2.4.</p> <p>The requirement will be evaluated at the cluster-level by TTT in Task T3.4.</p> <p>USIEGEN and TTT will support the evaluation in the demonstrators (T6.3, T7.3, T8.3).</p>
Source	Avionics WG, Standards, TRT
Additional Information	WP4 is included because the impact of such explicit communication means on tooling/scheduling/analysis

ID	R 1.10.3	
Topic	Architecture	
Subtopic	Heterogeneity	
Name	Support for multiple types of timing models for computation and communication	
Responsibility	WP	1, 4
	Lead partner	USIEGEN
	Participating partners	TRT, ONERA, RTAW, TTT, FENTISS, TUKL, UPV
Description	<p>The architectural style and the development methodology shall consider multiple types of communication and computational activities. At least:</p> <ul style="list-style-type: none"> • Periodic activities: Such an activity realizes a synchronous repetitive process with an activation period and phase set at the process creation. The period and phase can be specified with respect to a system-wide synchronized global time base. <p>In case of computations, the task is only restarted after it is over at the next activation period, thanks to a wait instruction. In case of communication, the instants of</p>	

	<p>periodic message transmissions are specified by an a priori planned conflict-free communication schedule to ensure determinism and temporal properties such as latency, latency jitter, bandwidth, and message order.</p> <ul style="list-style-type: none"> • Sporadic activities: Such an activity is pseudo aperiodic with a minimum interarrival time in between successive activations. It realizes an asynchronous repetitive process. <p>In case of computations, the trigger is typically set from inside the process as a fixed time delay or timeout. In case of communication, sporadic messages establish rate-constrained data-flows with maximum bandwidth use to provide bounded latency.</p> <ul style="list-style-type: none"> • Aperiodic activities: This type of activity is aperiodic and realizes an asynchronous non-repetitive process triggered by some external events. <p>A restartable aperiodic activity is an asynchronous process that is used for time consuming services. Those processes are associated with a manager process in charge of the activation and cancellation of the service.</p>
Rationale	These are a representative set of task types used in the development of critical applications.
Significance	High
Means for validation/verification	<p>The requirement will be evaluated for the hypervisors in Task T2.4 and the development methodology in Task T1.6.</p> <p>The validation as part of the avionic demonstrator will be performed experimentally by TRT in T6.3.</p>
Source	WG Avionics, Industrial practices (avionics), ...
Additional Information	

1.11 Efficiency

ID	R 1.11.1	
Topic	Architecture	
Subtopic	Efficiency	
Name	Virtualization Efficiency	
Responsibility	WP	2
	Lead partner	ST

	Participating partners	ST, TEI, VOSYS, FENTISS
Description	<p>The virtualization overhead of the platform services shall be less than 10%</p> <p>Hardware-assisted virtualization provides hardware support that facilitates and improves performances when building hypervisors and allows guest Operating Systems to be run in isolation for a security</p>	
Rationale	Improved performance by using hardware assisted virtualization technology	
Significance	High	
Means for validation/verification	Validation consists in executing a benchmark code in the virtualized environment and in bare metal environment (non-virtualized). The execution is performed on the FPGA board.	
Source	Exploitation in real products	
Additional Information		

1.12 Energy and Power

ID	R 1.12.1	
Topic	Architecture	
Subtopic	Energy and Power	
Name	Low Power	
Responsibility	WP	1,2,3
	Lead partner	ST
	Participating partners	ST
Description	<p>The main objective is rather the possibility to roughly compare different NoC configurations with each other's to identify the low power on while preserving fidelity.</p>	
Rationale	<p>A NoC energy characterization is fundamental for entry system-level power characterization. It is provided as follows:</p> <ul style="list-style-type: none"> System-level NoC static/dynamic power consumption analysis model that can be used to evaluate the power consumption of the NoC for a given configuration (see R 9.7.1). 	

	<ul style="list-style-type: none"> System-level energy / Power requirements meta-model used to specify demands of application subsystems (see R 9.7.2). From the resource management point of view, the power model of the resources (computational, memory, I/O..) shall be available, in order to assess the power consumption of different configurations and perform reconfiguration at the system-level (see R10.1.1)
Significance	HIGH
Means for validation/verification	System-level energy and power characterization can be based on regression models obtained from low-level RTL simulation models and synthesis results.
Source	ST input and open source software
Additional Information	

1.13 Required Services

ID	R 1.13.1	
Topic	Architecture	
Subtopic	Required Services	
Name	Message-based communication service hiding implementation details and implementation technology	
Responsibility	WP	1,2,3
	Lead partner	USIEGEN
	Participating partners	USIEGEN, ST,TEI,TTT
Description	<p>The architecture shall support message-based end-to-end communication between partitions (on the same or different multi-core chips) with the following communication modes (cf. concept definition in Section 30)</p> <ul style="list-style-type: none"> Time-triggered communication for periodic messages: Periodic messages shall be supported by time-triggered communication defined by a period and phase with respect to a global time base Rate-constrained communication for sporadic messages: Rate-constrained communication shall establish the transport of sporadic messages with minimum interarrival times and priorities. Rate-constrained communication shall 	

	<p>guarantee sufficient bandwidth allocation for each transmission, with defined limits for delays and temporal deviations.</p> <ul style="list-style-type: none"> • Best-effort communication for aperiodic messages: Aperiodic messages have no timing constraints on successive messages and no guarantees with respect to delivery and incurred delays
Rationale	The message-based service of components shall hide the implementation details of components. For example, due to the message-based interfaces of components, it is possible to replace a component with a general purpose CPU and an operating system by a state machine in hardware without any impact on the message-based interface of the component.
Significance	High
Means for validation/verification	The requirement will be experimentally evaluated by TRT, ALSTOM and ST through the use of the core platform services in demonstrators from three different domains (T6.3, T7.3, T8.3).
Source	DOW
Additional Information	

ID	R 1.13.2	
Topic	Architecture	
Subtopic	Required Services	
Name	Core service : Global time base	
Responsibility	WP	1,3
	Lead partner	USIEGEN
	Participating partners	USIEGEN,TTT
Description	The architecture shall provide a consistent global time service for the system of networked multi-core chips with bounded precision, bounded accuracy, sufficient fine granularity and a sufficient wide horizon.	
Rationale	A global time service enables the coordination of cooperative timed actions of distributed subsystems in the temporal domain. Global time is also needed to check the temporal validity of real-time data that originate from distributed subsystems.	
Significance	High	

Means for validation/verification	The requirement will be experimentally evaluated by TRT, ALSTOM and ST through the use of the core platform services in demonstrators from three different domains (T6.3, T7.3, T8.3).
Source	ARTEMIS SRA
Additional Information	

ID	R 1.13.3	
Topic	Architecture	
Subtopic	Required Services	
Name	Core service : Reconfiguration	
Responsibility	WP	3,4
	Lead partner	TUKL
	Participating partners	TUKL,ONERA,TRT,UPV,FENTISS,VOSYS,RTAW, TTT
Description	The system can be reconfigured upon foreseen and unforeseen changes in its operational and environmental conditions within in a predictable time span.	
Rationale	Basic goal of DREAMS.	
Significance	High	
Means for validation/verification	The requirement will be experimentally evaluated by TRT, ALSTOM and ST through the use of the core platform services in demonstrators from three different domains (T6.3, T7.3, T8.3).	
Source	DOW	
Additional Information		

ID	R 1.13.4	
Topic	Architecture	
Subtopic	Required Services	
Name	Core service : Execution	
Responsibility	WP	2
	Lead partner	VOSYS
	Participating partners	UPV, FENTISS , VOSYS

Description	<p>For the sharing of processor cores among mixed criticality applications, including safety-critical ones, partitioning OSes (e.g., XtratuM) shall be used, which ensure time and space partitioning for the computational resources. The scheduling of computational resources (e.g., processor, memory) in DREAMS will ensure that each task obtains not only a predefined portion of the computation power the processor core, but also that execution occurs at the right time and with a high level of temporal predictability.</p> <p>On one hand, the architecture will support static scheduling, where an offline tool creates a schedule with pre-computed scheduling decisions for each point in time. In addition, we will support dynamic scheduling by employing a quota system in the scheduling of tasks in order to limit the consequences of faults. Safety-critical partitions will establish execution environments that are amenable to certification and worst-case execution time analysis, whereas partitions for non safety-critical partitions will provide more intricate execution environments (e.g., based on Linux). In addition, the separation between safety-critical and non safety-critical application subsystems will be supported using dedicated on-chip processor clusters with respective OSes.</p>
Rationale	Safety critical and non-safety critical applications have different requirements, thus they cannot be treated in the same fashion. DREAMS will provide a mixed criticality environment so that safety critical partitions respect their constraints while co-existing with non-safety critical partitions
Significance	High
Means for validation/verification	The requirement will be experimentally evaluated by TRT, ALSTOM and ST through the use of the core platform services in demonstrators from three different domains (T6.3, T7.3, T8.3).
Source	DOW
Additional Information	

2 Requirements for Multicore Virtualization Technology

The following requirements for the multi core virtualization technology gather all aspects needed to successfully design an isolated and safety-critical on-chip device relying on the ST Network-on-Chip (STNoC) communication infrastructure enhanced with memory interleaving capability, time-triggered message delivery and virtualization layer. Detailed requirements regarding isolation aspects are stated in section 2.2

The on-chip device will run temporally and spatially isolated operating systems using XtratuM for safety-critical tasks or KVM (Linux) for non-safety critical tasks. Requirements towards multi core virtualization and memory resources efficiency are stated in sections 2.3 and 2.5. Connection to the off-chip system is established through a jointly defined gateway, the corresponding requirements are defined in section 2.1. The virtualized approach of the resource management is controlled through local resource monitors and schedulers that dynamically adapt the resource allocation to the run-time conditions of the device (requirements in section 2.4).

Furthermore, the multicore virtualized on-chip device will be driven to support traffic paradigm (time triggered or message passing) heterogeneity, making possible the coexistence on the same platform of hard real time traffics and best effort traffics, while not impacting significantly, and only in a bounded measure the performances for each traffic class. Relevant requirements are provided in sections 2.7, 2.8 and 2.9

The on-chip fault tolerance with requirements in section 2.6, low-level security support and safe run-time reprogramming schemes complete the list of embedded features in this ambitious DREAMS on-chip device.

2.1 Gateways

ID	R 2.1.1	
Topic	Virtualization-Chip	
Subtopic	Gateways	
Name	On-/off-chip communication interfacing	
Responsibility	WP	2,3
	Lead partner	USIEGEN
	Participating partners	TEI,ST,TTT
Description	<p>The architecture shall provide I/O interfaces between on-chip and off-chip networks in order to realize end-to-end communication channels between partitions on networked multi-core chips.</p> <ul style="list-style-type: none"> • Relaying of communication modes (periodic, sporadic, aperiodic, memory access): The gateway shall support relaying periodic messages (defined by period and phase), sporadic messages (defined by rate-constraints and priorities), aperiodic messages and access to shared memory. 	

	<ul style="list-style-type: none"> • Clock synchronization: The gateway must provide external clock synchronization for the on-chip network based on the off-chip clock source. • Selective redirection of messages using filtering of messages must be implemented in the gateway in order to address significant differences in on-chip/off-chip bandwidths. • Resolving of property mismatches: In the gateway property mismatches (e.g., packet size, name space, time base) between on-chip and off-chip networks shall be resolved. • Routing and mapping of namespaces: The mapping between virtual channels shall be done by the gateway (i.e., correspondence between physical links and TDMA slots of NoC and off-chip networks) • Temporal and spatial partitioning: The gateway shall provide spatial and temporal partitioning based on a priori of the permitted communication behavior for the different traffic types (e.g., time-triggered schedule and rate-constraints).
Rationale	To allow the abstraction of communication between multiple chips in a transparent and efficient manner and provide virtualization techniques for transparent network platforms.
Significance	HIGH
Means for validation/verification	<p>The requirements will be validated by USIEGEN, TTT, ST and TEI at module-test level in Task T2.1 and T2.3, as well as at integration-test level in Task T2.4 and Task T1.8.</p> <p>USIEGEN, TTT, ST and TEI will support the use and validation of the gateways in the avionics, wind power and healthcare demonstrators.</p>
Source	T2.1 T2.3
Additional Information	

2.2 Isolation

ID	R 2.2.1	
Topic	Virtualization-Chip	
Subtopic	Isolation	
Name	On chip time and space partitioning	
Responsibility	WP	WP2, WP3

	Lead partner	USIEGEN
	Participating partners	VOSYS, UPV, TEI, ST
Description	<p>The on-chip network shall ensure time and space partitioning based on a priori knowledge of the permitted behavior of cores.</p> <p>The on-chip network must ensure that a core cannot affect time-triggered messages from other cores in either the value or time domain.</p> <p>The on-chip network must ensure that a core can only impose a bounded delay on the timing of rate-constrained message from other cores.</p> <p>The a priori knowledge about the permitted behavior of cores shall be configurable via the GRM using the LRS at the network interfaces.</p>	
Rationale	The foundations for the integration of mixed-criticality systems is the time and space partitioning, which establish fault containment and the absence of unintended side-effects between functions.	
Significance	High	
Means for validation/verification	The requirements will be validated by USIEGEN, VOSYS, UPV, TEI and ST at module-test level in Task T2.1 and T2.3, as well as at integration-test level in Task T2.4 and Task T1.8.	
Source	T2.2	
Additional Information	Refinement of R 1.1.3 from WP1	

2.3 Memory Resources Efficiency

ID	R 2.3.1	
Topic	Virtualization-Chip	
Subtopic	Memory Resources Efficiency	
Name	Memory subsystem	
Responsibility	WP	2
	Lead partner	ST
	Participating partners	ST, TEI
Description	The architecture shall support memory interleaving.	

Rationale	Improved isolation by connecting to multi-bank shared memory hierarchies (several DDR memory controllers on chip) Increasing the memory contention implies that WCET exceeds ACET. Since critical and noncritical applications are scheduled according to WCET and ACET respectively, we need to allocate more time than needed, resulting in significantly degraded the processor utilization. One way to address this problem is to implement in the NOC memory interleaving that supports memory partitioning, so enabling developers to better upper bound interference in a way that reduces WCETs, thereby maximizing the memory efficiency without compromising applications with mixed criticality levels
Significance	HIGH
Means for validation/verification	ST will perform the validation using the demonstrator in WP8 (Task T8.3)
Source	T2.1
Additional Information	

ID	R 2.3.2	
Topic	Virtualization-Chip	
Subtopic	Memory Resources Efficiency	
Name	Cache	
Responsibility	WP	2
	Lead partner	ST
	Participating partners	ST, TEI, VOSYS
Description	Partitioning mechanisms for a shared L2 Cache shall be provided.	
Rationale	Sharing caches in multicore architectures implies greater variance in execution times with severe degradation of WCETs. Since mixed critical systems must be designed for WCET behaviour, the applications will be executed much more slowly since each application has to be guaranteed by its time budget. One way to address this problem is to implement a local resource manager that manages memory bandwidth by software/hypervisor	
Significance	HIGH	

Means for validation/verification	ST and TEI will perform the validation by simulation in Task T2.1 on a Gem5 model of the Spidergon STNoC backbone with a configurable network interface
Source	DOW (T2.1)
Additional Information	

ID	R 2.3.3	
Topic	Virtualization-Chip	
Subtopic	Memory Resources Efficiency	
Name	Avoid NoC memory traffic interferences	
Responsibility	WP	2, 4
	Lead partner	ST
	Participating partners	TEI, TRT, ONERA, RTAW, FENTISS, TUKL
Description	Memory traffic interferences shall be bounded (or avoided).	
Rationale	Impact of memory traffic and interferences (in multi-cores) should be bounded and computable in order to be able to provide WC*T. It is an aspect to be considered for timing partitioning.	
Significance	High	
Means for validation/verification	ST will validate the requirement as part of the assessment (T6.3, T7.3, T8.3, T1.8), processor virtualization (WP2) and tools (WP4)	
Source	WG Avionics, Standards, Genesys, ACROSS, CERTAINTY, ...	
Additional Information		

2.4 Monitoring and dynamic configuration of virtualized resources

ID	R 2.4.1	
Topic	Virtualization-Chip	
Subtopic	Monitoring and dynamic configuration of virtualized resources	
Name	On-chip monitoring and dynamic configuration of virtualized resources	
Responsibility	WP	2
	Lead partner	TRT

	Participating partners	TUKL, ONERA, UPV, FENTISS, VOSYS, ST
Description	Algorithms shall be implemented to monitor and dynamically configure virtualized chip-level resources using local resource monitors (MON) and local resource schedulers (LRS) to implement the decisions from the global resource management (GRM) such as resource-specific configurations, as well as monitoring their behaviour with feedback to the GRM.	
Rationale	The separation of system wide decisions of global resource management and their local execution depends on LRS and LRM.	
Significance	High	
Means for validation/verification	ST will experimentally validate the requirement based on the PGA board.	
Source	<p>DoW:</p> <p>Local resource scheduling and monitoring services for the integration of offline and online scheduling services.</p> <p>Adaptation engine as part of the LRM, coupling the monitored information to reconfiguration decisions.</p> <p>Novel distributed real-time scheduling heuristics at the network interface layer.</p>	
Additional Information	Includes refinement of R 10.1.1, R 10.2.1, R 10.3.2 for the chip-level	

ID	R 2.4.2	
Topic	Virtualization-Chip	
Subtopic	Monitoring and dynamic configuration of virtualized resources	
Name	Deadline monitoring	
Responsibility	WP	2
	Lead partner	UPV
	Participating partners	UPV, FENTISS, VOSYS, ONERA, TRT, TUKL
Description	The real-time critical tasks of the system should be monitored in order to check that they are executed in their assigned time (MAX_DEADLINE).	
Rationale	If a real-time critical task is not executed within the deadline, corrective action must be taken (e.g. bring the wind turbine to the safe state).	

	<p>This requirement impacts in the two levels of the architecture:</p> <ul style="list-style-type: none"> - Hypervisor: A partition with the real-time critical tasks shall require the specified amount of computation in a interval. - guestOS: The guestOS is in charge of execute and monitor the timing constraints of the real-time tasks <p>This is a special case of fault monitoring.</p>
Significance	Medium
Means for validation/verification	UPV will perform an integration test of this requirement in task 2.4
Source	ALSTOM, TRT
Additional Information	Refinement of R 10.2.1 for the chip-level

2.5 Multi-Core virtualization

ID	R 2.5.1	
Topic	Virtualization-Chip	
Subtopic	Multi-Core virtualization	
Name	Full virtualization	
Responsibility	WP	2
	Lead partner	VOSYS,TEI
	Participating partners	ST,UPV
Description	Full virtualization support of asymmetrical multicore processors shall be provided.	
Rationale	the hypervisor must manage efficiently connected resources	
Significance	HIGH	
Means for validation/verification	VOSYS and TEI will perform the validation of hardware virtualization extensions at the NoC network interface layer, as well as validation of system drivers, software configuration mechanisms and specialized VMs	
Source	T2.2, T2.3	
Additional Information		

ID	R 2.5.2	
Topic	Virtualization-Chip	
Subtopic	Multi-Core virtualization	
Name	High performance	
Responsibility	WP	2
	Lead partner	ST
	Participating partners	TEI, VOSYS, FENTISS
Description	The hypervisor shall use system drivers, software configuration mechanisms and specialized VMs to manage efficiently and securely different types of connected resources.	
Rationale	Improved performance by using hardware assisted virtualization technology. Targeted virtualization overhead less than 10%	
Significance	High	
Means for validation/verification	ST, VOSYS and FENTISS will validate the requirement in the healthcare demonstrator in Task T8.3.	
Source	Exploitation in real products	
Additional Information		

ID	R 2.5.3	
Topic	Virtualization-Chip	
Subtopic	Multi-Core virtualization	
Name	Legacy support	
Responsibility	WP	2
	Lead partner	VOSYS
	Participating partners	VOSYS, TEI
Description	The architecture shall support 32 bits and 64 bits guests.	
Rationale	This requirements ensures the possibility to run 64-Bit Operating Systems completely unmodified in parallel to other operating systems manage 32 bits or 64 bits guests.	
Significance	HIGH	
Means for validation/verification	VOSYS will perform validation of hardware virtualization extensions at the NoC network interface layer, as well as validation of system drivers, software configuration mechanisms and specialized VMs within Tasks T2.2 and T2.3.	

Source	T2.2, T2.3
Additional Information	

ID	R 2.5.4	
Topic	Virtualization-Chip	
Subtopic	Multi-Core virtualization	
Name	Soft real-time virtualization	
Responsibility	WP	2
	Lead partner	VOSYS
	Participating partners	VOSYS
Description	The execution services shall support soft real-time requirements.	
Rationale	Combine general purpose and real-time operating systems on one multi-core platform to simultaneously meet requirements for real-time and non real-time performance	
Significance	HIGH	
Means for validation/verification	VOSYS will experimentally validate this requirement in collaboration with ST within the WP8 Healthcare demonstrator.	
Source	T2.2	
Additional Information	Refinement of R 1.10.1	

ID	R 2.5.5	
Topic	Virtualization-Chip	
Subtopic	Multi-Core virtualization	
Name	Distributed I/O Virtualization – New Hardware Extensions	
Responsibility	WP	2
	Lead partner	VOSYS,TEI
	Participating partners	ST,UPV
Description	The hypervisor shall efficiently manage the I/O virtualization via the distributed IOMMU in mixed criticality systems.	
Rationale	The requirement enables full virtualization support of multiple cores.	
Significance	HIGH	

Means for validation/verification	<ul style="list-style-type: none"> VOSYS will experimentally validate the Hypervisor support for IOMMU virtualization in the WP8 Healthcare demonstrator. TEI and ST will validate the hardware extensions for IOMMU virtualization in the DREAMS platform, if necessary. ST will demonstrate it in the WP8 (healthcare) demonstrator.
Source	T 2.1, T2.3
Additional Information	

ID	R 2.5.6	
Topic	Virtualization-Chip	
Subtopic	Multi-Core virtualization	
Name	ARM 64bits and Spidergon STNoC interfacing	
Responsibility	WP	2
	Lead partner	ST
	Participating partners	TEI, USIEGEN
Description	The hardware integration between ARM and Spidergon STNoC shall performed, as well as the integration of system drivers, software configuration mechanisms and specialized VMs to manage efficiently and securely different types of connected resources.	
Rationale	Develop an efficient hardware assisted full virtualization system	
Significance	High	
Means for validation/verification	ST, TEI and USIEGEN will perform system-level, RTL simulation and/co-simulation and FPGA Platform prototyping in WP2 (Task T2.1, T2.2, T2.3) and assessment in the healthcare demonstrator (T8.3).	
Source	DoW	
Additional Information		

2.6 On-chip fault tolerance

ID	R 2.6.1
----	---------

Topic	Virtualization-Chip	
Subtopic	On-chip fault tolerance	
Name	On-Chip Redundancy	
Responsibility	WP	2,9
	Lead partner	IKL
	Participating partners	TUV
Description	<p>Safety requirements for multiple core application e.g. HFT=1 (On-Chip redundancy)</p> <p>If HFT=1 is required, the on-chip redundancy has to be in accordance with IEC 61508-2, Annex E, F</p>	
Rationale	<p>Justification here is to fulfil the requirements of IEC 61508 in case On-Chip redundancy shall be used.</p> <p>In case redundancy is implemented using separate chips this requirement is not applicable.</p>	
Significance	High	
Means for validation/verification	<p>Validation partner: TÜV/IKL</p> <p>Validation activity: Inspection of the adequacy of the techniques regarding Annex E, F of IEC 61508-2</p> <p>Results from the task will be reported in: D9.2.1 Standardization Report at the end of the project</p> <p>Task for Validation: T9.2</p>	
Source	TUV	
Additional Information		

2.7 Heterogeneity

ID	R 2.7.1	
Topic	Virtualization-Chip	
Subtopic	Heterogeneity	
Name	Heterogeneity of communication paradigm	
Responsibility	WP	WP2
	Lead partner	USIEGEN
	Participating partners	ST, TEI

Description	<p>The on-chip network shall provide the following interaction mechanisms required for different models of computation:</p> <ul style="list-style-type: none"> • Time-Triggered communication: periodic messages are to be sent according to a priori defined communication schedule. The time-triggered communication system guaranties the timely arrival of the messages. • Rate-Constrained communication: sporadic messages shall realize a communication paradigm, in which successive messages belonging to the same rate-constrained dataflow are guaranteed to be offset by a minimum duration as configured. • Best-Effort communication: this communication offers no guarantee whether or when aperiodic messages are transmitted, what delays occur and whether messages arrive at the recipient. Best-effort shall exploit the remaining bandwidth of the network and have principally the lowest priority. • Shared memory access
Rationale	The proposed architecture should not be bounded to a specific communication paradigm, in order to be able to interconnect heterogeneous networks, each of which invokes possibly different communication paradigm.
Significance	High
Means for validation/verification	<p>The requirements will be validated by USIEGEN, ST and TEI at module-test level in Task T2.1 and T2.3, as well as at integration-test level in Task T2.4 and Task T1.8.</p> <p>USIEGEN, ST and TEI will support the use and validation of the gateways in the avionics, wind power and healthcare demonstrators.</p>
Source	T2.1
Additional Information	

ID	R 2.7.2	
Topic	Virtualization-Chip	
Subtopic	Heterogeneity	
Name	Heterogenity of processors	
Responsibility	WP	2
	Lead partner	ST

	Participating partners	VOSYS, TEI
Description	The architecture shall provide support of different processors and/or hardware accelerators with shared memory access	
Rationale	The proposed architecture should not be bounded to a specific architecture paradigm, in order to be able to map different architecture templates.	
Significance	High	
Means for validation/verification	ST, VOSYS and TEI will validate the requirement through the demonstrators that will be based on a(common) heterogeneous architecture	
Source	T2.1	
Additional Information	Refinement of R 1.10.1	

2.8 Real-time

ID	R 2.8.1	
Topic	Virtualization-Chip	
Subtopic	Real-time	
Name	Preemptive scheduling	
Responsibility	WP	2,4
	Lead partner	UPV
	Participating partners	UPV, FENTISS, VOSYS, TUKL, RTAW, ONERA
Description	<p>The virtualization layer (hypervisor) should support the following scheduling policies:</p> <ul style="list-style-type: none"> - Cyclic scheduling - Fixed-priority preemptive scheduling <p>Each core shall have assigned one of the supported policies.</p>	
Rationale	<p>A common practice in Avionics is to use static cyclic scheduling for real-time critical applications.</p> <p>Non critical application can require to be executed under priority based scheme.</p> <p>Both scheduling policies can be used in a multi-core system by using different policies to the cores.</p>	

Significance	Medium
Means for validation/verification	UPV will perform an integration test of this requirement in task 2.4
Source	ALSTOM
Additional Information	

ID	R 2.8.2	
Topic	Virtualization-Chip	
Subtopic	Real-time	
Name	Realtime virtualization support	
Responsibility	WP	2
	Lead partner	VOSYS
	Participating partners	VOSYS, UPV, FENTISS
Description	To combine the real-time with non- real-time applications	
Rationale	An embedded hypervisor that combine the benefits of the hardware assisted virtualization with the requirements of embedded applications such as memory and code size and real - time deterministic performance. Deterministic performance would mean that a RTOS would respect its deadlines and low-latency processing in response to external events.	
Significance	High	
Means for validation/verification	<ul style="list-style-type: none"> VOSYS will experimentally demonstrate the co-existence of real-time and non real-time applications using the WP8 Healthcare demonstrator. ST will demonstrate it in WP8 	
Source	Optimized hierarchical real-time scheduling heuristics at the network interface layer (DOW)	
Additional Information		

ID	R 2.8.3	
Topic	Virtualization-Chip	
Subtopic	Real-time	

Name	Bounded jitter/latency	
Responsibility	WP	2
	Lead partner	USIEGEN
	Participating partners	ST, TEI, VOSYS, USIEGEN
Description	The architecture shall provide means to bound the latency and jitter of safety-critical communication (i.e., periodic and sporadic messages) at chip-level.	
Rationale	Required to compute schedules, WC*T	
Significance	High	
Means for validation/verification	The requirements will be validated by USIEGEN at module-test level in Task T2.1 and T2.3, as well as at integration-test level in Task T2.4.	
Source	Avionics WG, DoW, Standards, ...	
Additional Information		

2.9 Reconfiguration support (measure of success)

ID	R 2.9.1	
Topic	Virtualization-Chip	
Subtopic	Reconfiguration support (measure of success)	
Name	LRM: Reconfiguration and resource management support	
Responsibility	WP	WP2, WP3, WP4
	Lead partner	USIEGEN
	Participating partners	TTT, TRT, ONERA , UPV, FENTISS, VOSYS, ST
Description	The gateway and the network-on-a-chip shall support (at runtime) the update of the configuration including the time-triggered schedule, the rate-constraints and the filtering specification of the gateway. Therefore, a secure configuration channel from the GRM to the LRMs of the network interfaces and the gateways shall be available for runtime configuration and resource management.	
Rationale	Reconfigurability of gateways and NoC is required for dynamic resource management.	
Significance	High	

Means for validation/verification	The requirements will be validated by USIEGEN and TTT at module-test level in Task T2.1 and T2.3, as well as at integration-test level in Task T2.4 and Task T1.8.
Source	T2.2
Additional Information	Refinement of R 10.3.1

ID	R 2.9.2	
Topic	Virtualization-Chip	
Subtopic	Reconfiguration support (measure of success)	
Name	Bounded reconfiguration time of NoC and Gateways	
Responsibility	WP	2, 3, 4
	Lead partner	USIEGEN
	Participating partners	TTT, TRT, ONERA,
Description	The reconfiguration time of the NoC and the gateways shall be bounded.	
Rationale	Stalling the system in reconfiguration state, could cause failure in high-critical systems.	
Significance	High	
Means for validation/verification	The requirements will be validated by USIEGEN and TTT at module-test level in Task T2.1 and T2.3, as well as at integration-test level in Task T2.4 and Task T1.8.	
Source	DoW	
Additional Information	Refinement of 10.4.1	

ID	R 2.9.3	
Topic	Virtualization-Chip	
Subtopic	Reconfiguration support (measure of success)	
Name	Switching the existing static schedules	
Responsibility	WP	2,4
	Lead partner	UPV
	Participating partners	TRT, TUKL, ONERA

Description	The resource allocation strategies should allow for online changes of the allocation plans in order to permit adaptation to foreseen (and possible unforeseen) changes in the availability of the resources. Such changes must complete within in a predictable time span.
Rationale	Necessary for timeliness in system for different criticalities.
Significance	
Means for validation/verification	UPV and TUKL will validate the requirement in task 2.4
Source	DoW
Additional Information	

3 Requirements for Mixed-Criticality Network

In this section, the requirements on the DREAMS networking approach are presented. The requirements are linked to the DREAMS DoW and as such provide the foundation for the mixed-criticality network and its associated components, particularly with regards to the gateways for end-to-end segregation as means for integration of mixed criticalities at the network-level.

- **On-chip and off-chip communication systems with Time and Space Partitioning:** The architectural time- and space partitioning requirements are translated to the communication system to exploit knowledge about the permitted communication behaviour of components for TSP between components. This communication system encompasses different on-chip and off-chip networks. For example, the requirements cover configuration with periods and phases for time-triggered communication activities and minimum inter-arrival times for event-triggered communication activities. Different resources (e.g., I/O, shared memories) are mapped to the message-based communication. The communication system is required to establish temporal and spatial partitioning of the resources. Partitioning is preserved when accessing resources across different multi-core chips.
- **Virtualization for off-chip resources:** The requirements necessary to leverage multi-core platforms for a system perspective of mixed-criticality applications combining the chip-level and cluster-level is defined on the basis of gateway requirements. Using gateways, access to virtualized resources are to become location transparent. For example, the access to a remote I/O resource located on another chip is relayed via gateways involving gateways between on-chip and off-chip networks (i.e., vertical integration) and possibly gateways between different types of off-chip networks (i.e., horizontal integration).
- **Support for message-based communication:** The requirements cover the basis for message-based communication in order to support the time-and space partitioning requirements on the system level. In time intervals not used for time-triggered communication, the network will be used for accessing and virtualizing the global shared memory in a distributed manner. Detailed requirements are provided in section 3.1.
- **Network gateways:** On the cluster-level, DREAMS will bridge between several network types and bridge chip-level with network scheduling. The requirements on the necessary gateways to be developed in order to allow the abstraction of communication between multiple chips in a transparent manner and provide virtualization techniques for transparent network platform taking into account complete segregation to establish mixed criticality support on the network level, are defined in this section. Support for heterogeneous wired and wireless networks of mixed criticality and protocols (i.e. TTEthernet, AFDX, EtherCAT) is also covered by these requirements.

The related work package that mainly targets the implementation of the network-level requirements is WP3. In the foreground of this chapter are those activities that relate to the fundamental concepts related to safety up to the highest levels, real-time support satisfying the timing requirements from the three application domains, and fault containment and encapsulation at the network level to ensure data and system integrity also in the case of system faults.

The network level requirements related to resource management on the global level and the support for timely adaptation, as well as security (integrity, authenticity and availability) based on security models for the network side are not covered in this chapter. These two topics (although part of DREAMS WP3) are covered in separate chapters focusing on these specific (cross-WP) requirements.

3.1 Time and Space Partitioning in the Network

ID	R 3.1.1	
Topic	Networking	
Subtopic	Time and Space Partitioning in the Network	
Name	Mixed-critical traffic	
Responsibility	WP	WP3
	Lead partner	TTT
	Participating partners	ONERA, RTAW, TTT, FENTISS, TUKL
Description	The core platform services for off-chip communication (called the network for short) shall provide support for mixed-critical traffic.	
Rationale	The solutions (demos) built with the framework will mix transfers of different function with different criticalities.	
Significance	High	
Means for validation/verification	Implementation (WP3.1), Tooling (WP4.1), Assessment (WP6.3, WP7.3, WP8.3)	
Source	DoW, Standards	
Additional Information		

ID	R 3.1.2	
Topic	Networking	
Subtopic	Time and Space Partitioning in the Network	
Name	Time Partitioning	
Responsibility	WP	3
	Lead partner	TTT
	Participating partners	
Description	The network shall ensure bounds on the temporal impact of any one message communicated in the network on a defined set or several defined sets of message in the network.	

Rationale	In a mixed-criticality network messages have different criticality levels. Such a network needs to ensure that the impact of messages on each other is well understood. Note that sometimes it is not possible or too expensive to completely eliminate the effects of a message on other messages. Hence, instead of requiring “no” impact of a message on other messages, it is sufficient to understand the quantity of impact. Additional detailed requirements will then define more detailed constraints on partitioning, in particular on the quantity of impact.
Significance	High
Means for validation/verification	Assurance is achieved through implementation (T3.1), Tooling (T4.1), and Assessment (T6.3, T7.3, T8.3)
Source	DoW
Additional Information	Refinement of R 1.1.3

ID	R 3.1.3	
Topic	Networking	
Subtopic	Time and Space Partitioning in the Network	
Name	Time Partitioning – highest criticality messages	
Responsibility	WP	3
	Lead partner	TTT
	Participating partners	
Description	The network shall ensure that there is no temporal impact of any message on messages of highest criticality.	
Rationale	There is a highest criticality message class. Using the time-triggered communication paradigm it can be guaranteed that the time-triggered messages (or a subset of them) will not be subject of temporal impact from any other message.	
Significance	High	
Means for validation/verification	Temporal properties of highest criticality messages will be validated in implementation in Task 3.1 and using available tools. These tooling aspects are handled in task 4.1. This way assuring that TT messages will not be subject of temporal impact from any other messages. Further assessment is done in the demonstrator work packages (WP6.3, WP7.3, WP8.3).	
Source	DoW	
Additional Information	Refinement of R 1.1.3	

ID	R 3.1.4	
Topic	Networking	
Subtopic	Time and Space Partitioning in the Network	
Name	Time Partitioning – lowest criticality messages	
Responsibility	WP	3
	Lead partner	TTT
	Participating partners	
Description	The network may transport a set of lowest criticality messages without bounds of temporal impact from other messages.	
Rationale	For lowest criticality messages, i.e., best-effort messages, the network does not give any transmission guarantees.	
Significance	High	
Means for validation/verification	Will be assured using available tools that lowest criticality messages will not have a temporal impact in other messages. Assurance is achieved through implementation (T3.1), Tooling (T4.1), and Assessment (T6.3, T7.3, T8.3)	
Source	DoW	
Additional Information	Refinement of R 1.1.3	

ID	R 3.1.5	
Topic	Networking	
Subtopic	Time and Space Partitioning in the Network	
Name	Time Partitioning – medium criticality messages	
Responsibility	WP	3
	Lead partner	TTT
	Participating partners	
Description	The network should transport a set or several sets of medium-critical messages for which bounds on their temporal impact from other messages are specified.	
Rationale	Medium-critical messages have a known bound in the temporal impact on them.	
Significance	Medium	

Means for validation/verification	Assurance is achieved through implementation (T3.1), Tooling (T4.1), and Assessment (T6.3, T7.3, T8.3)
Source	DOW
Additional Information	Refinement of R 1.1.3

ID	R 3.1.6	
Topic	Networking	
Subtopic	Time and Space Partitioning in the Network	
Name	Space Partitioning	
Responsibility	WP	3
	Lead partner	TTT
	Participating partners	
Description	The network elements, i.e., switches, shall provide means for separate memory space for messages of different criticality.	
Rationale	Switches integrate messages of different criticality. In today's networks this integration is typically done in a store and forward fashion in which all messages (or most messages) are locally stored in a switch's memory, prioritized, and forwarded. To ease the verification, validation, and certification process, as well as to reduce design risks, the memory structure of a switch for mixed-criticality networks allows configuring different memory locations for different messages.	
Significance	High	
Means for validation/verification	TTT will participate in the research, development and validation process of space partitioning.	
Source	DoW	
Additional Information	Refinement of R 1.1.3	

3.2 Safety and Fault Handling

ID	R 3.2.1	
Topic	Networking	
Subtopic	Safety and Fault Handling	

Name	Tolerable number of faults	
Responsibility	WP	3
	Lead partner	ALSTOM
	Participating partners	TTT, IKL
Description	The tolerable failure rate of the network shall be PFH (probability of dangerous failure per hour) $10^{-9}/h$ to $10^{-8}/h$	
Rationale	The safety critical functions in the Wind Power domain must be certified according to ISO-13849, Performance Level d (PLd). This is equivalent to IEC-61508 SIL 2, which defines a PFH of $10^{-7}/h$ to $10^{-6}/h$. The share fraction for the network is 1%, therefore the network PFH must be $10^{-9}/h$ to $10^{-8}/h$.	
Significance	High	
Means for validation/verification	D7.2.1: Wind power demonstrator	
Source	ALSTOM	
Additional Information	Refinement of the R 1.1.7 for the networks	

ID	R 3.2.2	
Topic	Networking	
Subtopic	Safety and Fault Handling	
Name	Fault Detection in the Network	
Responsibility	WP	3
	Lead partner	TTT
	Participating partners	
Description	The network shall implement diagnosis functionality to detect faulty components.	
Rationale	A network can detect faulty components by monitoring the number of faulty messages received and correct messages not received. These two main categories can be further refined in several sub-categories that define what constitutes the failure in a faulty message as well as what are the expectation criteria that a correct message should have been arrived.	
Significance	High	

Means for validation/verification	TTT will lead the validation of resource monitor (MON). MON shall monitor the resource associated such as network interfaces WP2 or network switches WP3 and must detect the network failures. Assurance aspects according to IEC 61508 for mixed criticality networks will be taken into account.
Source	DoW
Additional Information	Refinement of R 10.2.1

ID	R 3.2.3	
Topic	Networking	
Subtopic	Safety and Fault Handling	
Name	Fault Detection in the Network – Switch Counters	
Responsibility	WP	3
	Lead partner	TTT
	Participating partners	
Description	Switches shall implement a set of failure counters for monitoring faulty behavior.	
Rationale	The failure counters will be increased each time the switch perceives a faulty behaviour. Detailed description of type and number of failure counters will be done in WP3.	
Significance	High	
Means for validation/verification	TTT will lead the validation of resource monitor (MON). MON shall monitor the resource associated, in this case, the detected failures by network switches.	
Source	DoW	
Additional Information	Refinement of R 10.2.1	

ID	R 3.2.4	
Topic	Networking	
Subtopic	Safety and Fault Handling	
Name	Fault Detection in the Network – Switch accessible	
Responsibility	WP	3
	Lead partner	TTT

	Participating partners	
Description	The failure counters of a switch should be remotely accessible over the network.	
Rationale	The failure counters of a switch ideally can be read from end systems over the network.	
Significance	Medium	
Means for validation/verification	TTT will lead the validation of resource monitor (MON). MON shall monitor the resource associated, in this case, the detected failures by network switches and shall report the significant changes to the GRM.	
Source	DoW	
Additional Information	Refinement of R 10.2.1	

ID	R 3.2.5	
Topic	Networking	
Subtopic	Safety and Fault Handling	
Name	Fault Isolation in the Network	
Responsibility	WP	3
	Lead partner	TTT
	Participating partners	
Description	The network shall implement error-containment regions.	
Rationale	Error-containment regions guarantee that a failure in one part of the network will not result in an error that will propagate widely through the network. Examples of error-containment regions are: self-checking pair, commander/monitor, local guardian, central guardian concepts.	
Significance	High	
Means for validation/verification	Definition of the mechanism for fault isolation in the network. Assurance according to IEC 61508 for mixed criticality networks.	
Source	DoW	
Additional Information		

ID	R 3.2.6	
Topic	Networking	
Subtopic	Safety and Fault Handling	
Name	Fault Recovery in the Network	
Responsibility	WP	3
	Lead partner	TTT
	Participating partners	
Description	The network shall provide means for recovery after transient upsets.	
Rationale	When the failure of a component has been detected (and potentially isolated) a recovery routine can attempt to get the component operational again. Typically recovery involves a restart of a component or a part of the component.	
Significance	High	
Means for validation/verification	Network recovery strategies based on R 1.3.1 requirement will be assessed in demonstrators and validated at module test level and integration test level. Assurance according to IEC 61508 for mixed criticality networks.	
Source	Functional safety standards (e.g. IEC 61508)	
Additional Information		

ID	R 3.2.7	
Topic	Networking	
Subtopic	Safety and Fault Handling	
Name	E2E Fault Model	
Responsibility	WP	3
	Lead partner	IKL
	Participating partners	USIEGEN
Description	Faults of interest for the DREAMS architecture should be identified. When end to end communication takes place, several faults might occur during communication such as transmission errors, repetitions, deletion, insertion, re-sequencing, corruption, delay and masquerade.	

Rationale	According to IEC-61508-2-2 7.4.11, when data communication is used in the implementation of a safety function then the failure measure (such as the residual error rate) of the communication process shall be estimated taking into account transmission errors, repetitions, deletion, insertion, re-sequencing, corruption, delay and masquerade. This failure measure shall be taken into account when estimating the failure measure of the safety function due to random failures.
Significance	Medium
Means for validation/verification	D1.2.1 architectural style of DREAMS and D3.3.1 High-level Design of Cluster-level Safety and Security services
Source	DoW (T 1.2 Definition of Cross-Domain Architectural Style for Mixed-Criticality Systems and T3.3 Cluster-level Safety and Security)
Additional Information	Refinement of Fault Hypothesis R 1.10.1

ID	R 3.2.8	
Topic	Networking	
Subtopic	Safety and Fault Handling	
Name	Network Fault Detection at the Application Level	
Responsibility	WP	1, 3
	Lead partner	IKL
	Participating partners	
Description	For the possible faults in end to end communication previously detected and collected a SCL (Safety Communication Layer) shall be developed with the intent of considering the communication channel as a black channel and leaving all detection mechanisms in charge of this layer that will isolate the end application from the communication channel faults.	
Rationale	<p>According to IEC-61508-2 7.4.11, the techniques and measures necessary to ensure the required failure measure (such as the residual error rate) of the communication process (see 7.4.11.1) shall be implemented according to the requirements of this standard and IEC 61508-3. This allows two possible approaches:</p> <ul style="list-style-type: none"> – The entire communication channel shall be designed, implemented and validated according to the IEC 61508 series and IEC 61784-3 or IEC 62280 series. This a so-called ‘white channel’ (see Figure 7 a); <p>or</p>	

	– Parts of the communication channel are not designed or validated according to the IEC 61508 series. This is a so-called 'black channel' (see Figure 7 b). In this case, the measures necessary to ensure the failure performance of the communication process shall be implemented in the E/E/PE safety-related subsystems or elements that interface with the communication channel in accordance with the IEC 61784-3 or IEC 62280 series as appropriate.
Significance	Medium
Means for validation/verification	D1.2.1 architectural style of DREAMS and D3.3.2 First Implementation of Cluster-level Safety and Security services D3.3.3 Final Implementation of Cluster-level Safety and Security services
Source	DoW (T 1.2 Definition of Cross-Domain Architectural Style for Mixed-Criticality Systems and T3.3 Cluster-level Safety and Security)
Additional Information	

ID	R 3.2.9	
Topic	Networking	
Subtopic	Safety and Fault Handling	
Name	Network Fault Recovery at the Application Level	
Responsibility	WP	1, 3, 7
	Lead partner	IKL
	Participating partners	
Description	Mechanism should be defined on how the end application should react to faults detected in the non-trusted communication channel by the Safety Communication Layer.	
Rationale	Safety Communication Layer will not implement a full application but only a layer for the safety assurance of the communication aspects of the black channel.	
Significance	Low	
Means for validation/verification	D1.2.1 architectural style of DREAMS and D3.3.1 High-level Design of Cluster-level Safety and Security services	
Source	DoW (T 1.2 Definition of Cross-Domain Architectural Style for Mixed-Criticality Systems and T3.3 Cluster-level Safety and Security)	
Additional Information		

ID	R 3.2.10	
Topic	Networking	
Subtopic	Safety and Fault Handling	
Name	Fault tolerance and redundancy	
Responsibility	WP	3, 6
	Lead partner	ONERA
	Participating partners	ONERA, RTAW, TTT, FENTISS, TUKL
Description	The network shall provide support for fault tolerance and redundancy.	
Rationale	A specific case of fault tolerance applied to networks.	
Significance	High	
Means for validation/verification	<p>TTT and TUKL will implement fault tolerant communication at the GRM level. (T3.2)</p> <p>ONERA will provide some scenarios involving errors that should be supported by the network (T4.1).</p> <p>The requirement will be experimentally evaluated by ONERA, TTT, RTAW, FENTISS and TUKL through the use of the fault injection in the avionic demonstrator (T6.3).</p>	
Source	WG Avionics, DoW, Standards, ...	
Additional Information		

3.3 Timing Requirements

ID	R 3.3.1	
Topic	Networking	
Subtopic	Timing Requirements	
Name	Bounded jitter/latency	
Responsibility	WP	3, 4
	Lead partner	TTT
	Participating partners	RTAW, TTT, FENTISS, TUKL
Description	The network shall provide means to bound latency and jitter at cluster level.	

Rationale	Required to compute schedules, WC*T
Significance	High
Means for validation/verification	This property of cluster-level communication will be analytically evaluated by TTT and RTAW in the context of T1.8 and T4.4.
Source	WG Avionics, DoW, Standards, ...
Additional Information	Refinement of R 1.2.3

3.4 Resource Management

ID	R 3.4.1	
Topic	Networking	
Subtopic	Resource Management	
Name	Cluster level monitoring and dynamic configuration of virtualized resources	
Responsibility	WP	3
	Lead partner	TTT
	Participating partners	TUKL, ONERA
Description	Algorithms shall be developed to monitor and dynamically configure virtualized cluster-level resources using local resource monitors (MON) and local resource schedulers (LRS) to implement the decisions from the global resource management (GRM) such as resource-specific configurations, as well as monitoring their behaviour with feedback to the GRM.	
Rationale	The separation of system wide decisions of global resource management and their local execution depends on LRS and LRM.	
Significance	High	
Means for validation/verification	Assurance is achieved through implementation (T3.1 and T3.2) and assessment (T6.3, T7.3, T8.3)	
Source	DoW	
Additional Information	Refinement of R 10.1.1	

3.5 Support for demonstrators

ID	R 3.5.1
Topic	Networking

Subtopic	Support for demonstrators	
Name	EtherCAT Data logger	
Responsibility	WP	3,7
	Lead partner	IKL
	Participating partners	ALSTOM
Description	<p>The EtherCAT data acquisition system should log the information of the EtherCAT I/O variables, including the variables related to Safety Over EtherCAT (SoE).</p> <p>The input of the system will be an ENI (EtherCAT Network Information) file as defined in the document ETG.2100 S (R) V1.0.0 of the EtherCAT Technology Group.</p> <p>The acquisition system will capture the EtherCAT frames from the bus and will extract the value of the variables using the information extracted from the ENI file.</p> <p>The information of the value of the variables will be stored in a binary file format for measurement data.</p>	
Rationale	<p>Taking into account the demonstration of DREAMS in wind power, an EtherCAT Data logger tool allows the capture and storage of the information exchanged between the master and the slave modules of an EtherCAT bus in an indeterminate period of time.</p>	
Significance	Medium	
Means for validation/verification	IKL will support the use and validation of the EtherCAT Data logger in the wind power demonstrator.	
Source	ALSTOM, WP7	
Additional Information		

4 Requirements for Tooling, Scheduling and Analysis

This section provides all requirements related to WP4 “Architecture Tooling, Scheduling and Analysis”. WP4 is concerned with the algorithms underlying DREAMS model driven development approach (see Figure 7) and the implementation of these algorithms into tools, so that they can concretely be applied to use cases.

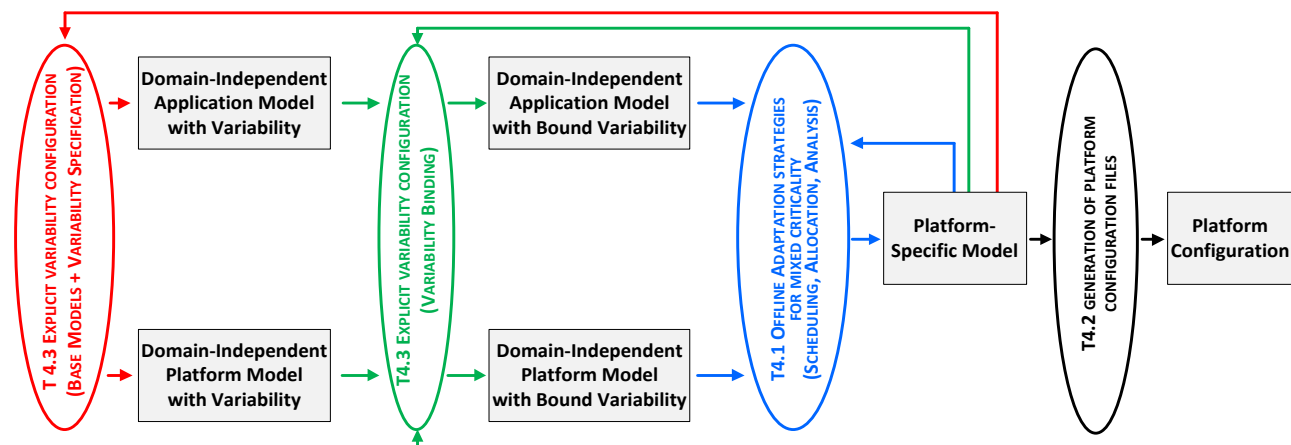


Figure 7 : Overview of the DREAMS Model Driven Development Approach

In Figure 7 models (see WP1) are represented by rectangles whereas the activities that perform the transformations between models are shown as ovals. Arrows indicate input and output relations of forward steps or loops. The later represent iterative revision for optimization or widening or changing

The blue oval represents the activities of defining allocation of resources and scheduling, needed for building a platform specific model, for given couple of an application and a platform model. These activities are covered by T4.1 and the corresponding requirements are presented in Section 4.1. The requirements specifically related to safety aspects of real time faults have been put into Section 4.3. The red and green ovals represent activities concerned with the consideration of variability of the considered family of systems. These are covered by T4.2 and the related requirements are listed here in 4.2.

The black oval represents the activity of generating configuration files of DREAMS platform services automatically out of a platform specific model, to avoid error prone manual editing. This is covered by T4.2 and the corresponding requirements can be found here in 4.5.

Last but not least, Section 4.4 states requirement about the comprehensive set of tools that implement the algorithms (tool chain) and in particular those related to tool inter-operability.

4.1 Allocation of resources and DSE

ID	R 4.1.1	
Topic	Tools & Scheduling	
Subtopic	Allocation of resources and DSE	
Name	Generation algorithms of resource allocation configurations	
Responsibility	WP	4
	Lead partner	TUKL
	Participating partners	TUKL, ONERA, RTaW
Description	A configuration algorithm shall be designed for a resource allocation strategy based on the decomposition of global constraints into local constraints of virtualized resources that are guaranteed by local and global resource managers.	
Rationale	System wide decisions are taken by global resource management and communicated via resource allocation configuration throughout the resources in the system.	
Significance	High	
Means for validation/verification	The participating partners analytically experimentally validate the resource allocation algorithms (D4.1.3) implemented in the Avionics Demonstrator.	
Source	DOW (T4.1)	
Additional Information	Refinement of R 10.1.1, R 10.3.1	

ID	R 4.1.2	
Topic	Tools & Scheduling	
Subtopic	Allocation of resources and DSE	
Name	Criticality spectrum: combination of offline and online scheduling	
Responsibility	WP	4
	Lead partner	TUKL
	Participating partners	TUKL
Description	The scheduling algorithms on which the resource allocation strategy is based shall allow the segregation of activities with different criticality levels. It shall in particular be possible to schedule activities with low criticality online without interference	

	on activities with high criticality, for which scheduling tables have been designed offline.
Rationale	Specific case of mixed criticality considered in DREAMS.
Significance	High
Means for validation/verification	TUKL performs the analytical validation of the online/offline scheduling methods included in D4.1.2 within WP4.
Source	DoW (T4.1)
Additional Information	

ID	R 4.1.3	
Topic	Tools & Scheduling	
Subtopic	Allocation of resources and DSE	
Name	Response time analysis algorithms	
Responsibility	WP	4
	Lead partner	TUKL
	Participating partners	TUKL, ONERA, RTAW, UPV
Description	End-to-end response time analysis algorithm shall be developed which are able to account for scheduling algorithms considered by the resource allocation strategy.	
Rationale	The outcomes of the response time analysis algorithm are needed to verify the end-to-end latency constraints.	
Significance	High	
Means for validation/verification	RTAW participates in the experimental evaluation of the usage of the timing analysis tool in the demonstrators (T6.3, T7.3, T8.3). Validation will also be accomplished through the integration test in WP4 (Task 4.4 Tool integration and demonstrator support).	
Source	DoW (T4.1)	
Additional Information		

ID	R 4.1.4	
Topic	Tools & Scheduling	
Subtopic	Allocation of resources and DSE	

Name	Design-space exploration	
Responsibility	WP	4
	Lead partner	FORTISS
	Participating partners	FORTISS, SINTEF
Description	<p>A prototype of a design space exploration (DSE) tool should be provided that should use the initial system model as well as and design objective specifications (e.g., timing, energy, reliability) as an input, and computes a set of pareto-optimal variants of the initial design. It should provide a model-to-model transformation from the original design into a fault-tolerant version in order to meet reliability goals. It should provide/use the following models and analyses:</p> <ul style="list-style-type: none"> • Reliability analysis (internal, see R 9.13.6) • Offline mapping and real-time scheduling (external, from R 4.1.1) • NoC energy analysis model (external, from R9.7.1) <p>In addition to that, an enhancement of product line testing technology (such as the tools from R4.2.1) should be provided that features architectural exploration to supports the designer in obtaining an optimized system configuration.</p>	
Rationale	Tool support is required in order to perform the complex task of the design space exploration and configuration optimization.	
Significance	medium (design optimization could also be based on experience)	
Means for validation/verification	See R9.11.3.	
Source	DoW (T4.1)	
Additional Information		

ID	R 4.1.5	
Topic	Tools & Scheduling	
Subtopic	Allocation of resources and DSE	
Name	Performance	
Responsibility	WP	4
	Lead partner	TUKL

	Participating partners	ONERA, RTAW, TTT, FENTISS, TUKL
Description	Computed schedules shall achieve most efficient use (performance, power, ...) of the resources, e.g., as much as possible spare time should be available for evolutions.	
Rationale	No use of providing a solution to use a multi-core as a mono-core, we need to achieve high performance provided by a multi-core.	
Significance	High	
Means for validation/verification	Each participating partner performs the experimental validation of the corresponding tool within WP4 Assessment in the demonstrators (WP6)	
Source	WG Avionics, TRT	
Additional Information		

4.2 Variability

ID	R 4.2.1	
Topic	Tools & Scheduling	
Subtopic	Variability	
Name	Automation of configuring DREAMS systems	
Responsibility	WP	4,5
	Lead partner	SINTEF
	Participating partners	IKL
Description	Variability modelling and analysis tools shall be enhanced to achieve by automatic means as well as guided manual means an optimal or best effort configuration of DREAMS platforms and DREAMS systems.	
Rationale	To facilitate the configuring of a DREAMS system is a major goal for DREAMS	
Significance	High	
Means for validation/verification	<p>SINTEF will validate this requirement by applying the optimization tool to examples developed in T4.1, T4.3 and T5.5. Furthermore the optimization tool will be applied to examples from appropriate pilot cases.</p> <p>The result of the experiments will be reported in deliverables D4.4."</p>	

Source	DoW (All 4.x tasks, but in particular T4.3)
Additional Information	Closely associated with other requirements on Variability

ID	R 4.2.2	
Topic	Tools & Scheduling	
Subtopic	Variability	
Name	Explicit configuration definition	
Responsibility	WP	4,5
	Lead partner	SINTEF
	Participating partners	IKL
Description	The platform configuration shall be explicitly defined	
Rationale	Configuring of DREAMS platforms and DREAMS systems shall be made as explicit as possible such that both formal and heuristic techniques can be jointly applied.	
Significance	High	
Means for validation/verification	SINTEF will validate this requirement by applying the product line tool bundle to examples developed in T4.1, T4.3 and T5.5. Furthermore the product line tool bundle will be applied to examples from appropriate pilot cases. The result of the experiments will be reported in deliverables D4.4	
Source	DoW (All tasks, but in particular T4.3)	
Additional Information	Closely associated with R5.2.2 and other variability requirements	

4.3 Safety

ID	R 4.3.1	
Topic	Tools & Scheduling	
Subtopic	Safety	
Name	Formal definition of Real-time faults detection and recovery strategies	
Responsibility	WP	4
	Lead partner	ONERA
	Participating partners	TRT, TUKL

Description	A fault model shall be defined including the overuse of shared resources, deadline exceeding, rules violation, ... For each fault, determination of adequate detection and recovery strategies shall be performed, e.g. switching between off-line scheduling tables at runtime.
Rationale	Increase of performance, management of mixed-criticality
Significance	High
Means for validation/verification	The requirement will be verified by ONERA and TUKL by analysing the proposed models and strategies in T4.1 and validated by TRT during the assessment on the avionic demonstrator in T6.3.
Source	DOW, ONERA
Additional Information	Refinement of R 1.1.7

4.4 Tools

ID	R 4.4.1	
Topic	Tools & Scheduling	
Subtopic	Tools	
Name	Tool chain	
Responsibility	WP	4
	Lead partner	RTaW
	Participating partners	RTaW, FORTISS, IKL, TTT, SINTEF, ONERA, UPV
Description	Design activities of the DREAMS development process shall be supported by a tool chain.	
Rationale	Many design activities are too complex to allow a repeatedly correct “manual” execution, either because the used algorithms are complex or because the amount of design data is high. The appropriateness of algorithms can only be assessed by applying them to real-world examples. To allow the actual application, tool support is often the indispensable enabler.	
Significance	High	
Means for validation/verification	This requirement will be verified by RTaW in T4.4 by analysing the features of the tools with respect to the algorithms developed in the DREAMS project.	
Source	DOW (measurement of success)	

Additional Information	In the context of prototypical implementation of algorithms, spreadsheet functions can also be considered to be a tool support.
------------------------	---

ID	R 4.4.2	
Topic	Tools & Scheduling	
Subtopic	Tools	
Name	Continuous data flow through the tool chain	
Responsibility	WP	WP4
	Lead partner	RTaW
	Participating partners	RTAW, FORTISS IKL, TTT, SINTEF, ONERA, UPV, TUKL, USIEGEN
Description	The exchange of data between consecutive tools in the DREAMS development process shall be automated so that it can be performed without “manual” recopying or reworking of the data.	
Rationale	Manual recopying or reworking is error prone and therefore not acceptable for safety critical systems, whereas dedicated software function for data transfer can be tested and always repeated in exactly the same way.	
Significance	High	
Means for validation/verification	This requirement will be verified by RTaW in T4.4 by analysing the documented interoperability of the tools and will be validated in T6.3, T7.3, T8.3, with the help of the demonstrator partner by using all applicable tools.	
Source	DOW (T4.4)	
Additional Information	In order to prove the appropriateness and feasibility of the DREAMS architecture it is not necessary that all data exchange between tools used by the demonstrators are actually performed by dedicated functionalities, because in principle these functionalities can always be developed and given the limited resources, newly developed tools can only reach the state of prototype and existing (external) tools may not be adaptable in the context of the project.	

ID	R 4.4.3	
Topic	Tools & Scheduling	

Subtopic	Tools	
Name	Cross-domain applicability of methods and tool	
Responsibility	WP	4, 6, 7, 8
	Lead partner	RTAW
	Participating partners	RTaW, ALSTOM, ST, TRT
Description	Methods and tool should at least be suitable for all application domains represented by the demonstrators.	
Rationale	Cross-domain applicability is major goal of the DREAMS project.	
Significance	Medium	
Means for validation/verification	This requirement will be validated by RTaW in T4.4, T6.3, T7.3, T8.3, with the help of the demonstrator partners, by analysing and reporting the actual usage of the methods and tool by the demonstrators.	
Source	DoW (cross domain applicability)	
Additional Information		

4.5 Platform configuration

ID	R 4.5.1	
Topic	Tools & Scheduling	
Subtopic	Platform configuration	
Name	Configuration file generators	
Responsibility	WP	4
	Lead partner	RTaW
	Participating partners	RTaW, FORTISS, IKL, TTT, SINTEF, ONERA, UPV, USIEGEN
Description	<p>The generation of the configuration files of the DREAMS platform for an instance of the DREAMS architecture, shall be supported by tools that use the system model as input.</p> <p>The format of the platform service configuration files shall be service provider independent.</p>	
Rationale	Manual editing or semi-automatic generation of configuration files without automated link to the system model is time consuming and error prone.	

	The usage of service provider independent configuration file formats makes the generators and the generated files reusable with modules from different providers and also with the simulation framework. Furthermore, adapted off-the-shelf components often require specific additional configurations parameters that are not covered by the DREAMS meta-model.
Significance	Medium
Means for validation/verification	This requirement will be verified by RTaW in T4.2 and T5.6 by analysing the documented generators and their usage. The requirement will be validated in T6.3, T7.3, T8.3, with the help of the demonstrator partners by using all applicable tools.
Source	DOW (T4.2)
Additional Information	

5 Requirements for Mixed-Criticality Certification

With the intention of moving towards the future needs and considerations of next generation systems' certification, several categories have been identified susceptible to be of interest.

- Mixed-Criticality as the integration on a single platform of safety relevant and non-safety relevant functions where non-safety might be maintained and modified by non-safety design teams that cannot affect the safety part.
- Modular safety cases or modular certification of compliant items as basic building blocks of a system. In such a way that modular safety cases for generic cases can help in the generation of specific safety cases.
- Cross-domain dependable patterns as reference designs for multiple domains that have similar problems to solve so their solutions are similar too. Dependable patterns will show considerations to be taken and possible solutions.
- Test beds and frameworks for the simulation and verification of mixed-criticality systems with fault injection mechanisms.
- Product line certification where there is variability in the components of the product to generate a subset of them (family of products).

To address the above mentioned issues, a detailed list of requirements is provided hereafter. Section 5.1 addresses requirements related to mixed criticality product lines with certification support whereas all certification standard related requirements are consolidated in section 5.2.

A dedicated section (5.3) provides requirements on cross-domain mixed criticality patterns linked to IEC-61508 standard. The overall DREAMS certification strategy and the modular safety case for mixed criticality systems are addressed by requirements of sections 5.4 and 5.5.

Requirements in section 5.6 and 5.7 are related to the validation, verification and evaluation test bed for extra functional properties and the integration of the DREAMS development methodology into industrial safety engineering processes.

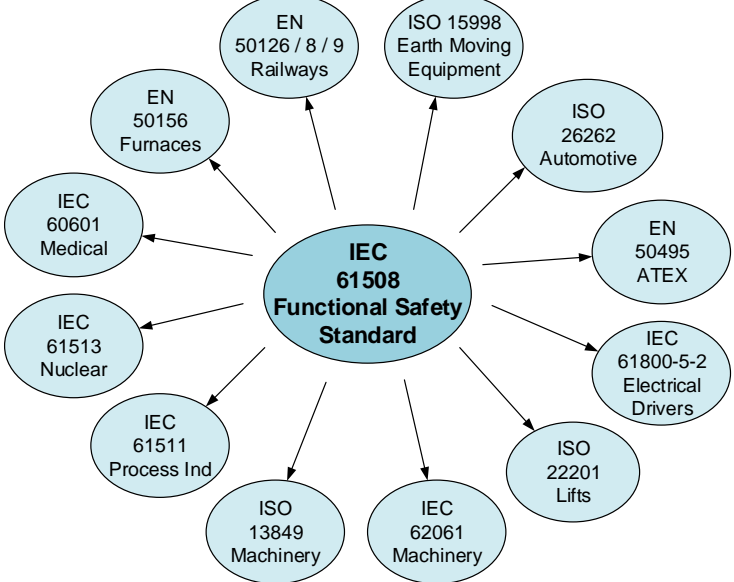
5.1 Mixed-criticality product lines with certification support

ID	R 5.1.1	
Topic	Certification V&V	
Subtopic	Mixed-criticality product lines with certification support	
Name	Mixed-criticality product line: enable certification of product-lines with variability management support	
Responsibility	WP	1, 4, 5
	Lead partner	SINTEF
	Participating partners	TUV, IKL, ALSTOM, FENTISS

Description	Mixed-criticality product line shall be supported to enable certification of product-lines with variability management.
Rationale	Mixed criticality systems have a little value if they are not certificated. Therefore, is mandatory to certificate the mixed-criticality systems. To make certification less laborious without loss of trust, Product Line techniques can be used to help certification of systems with high variability.
Significance	Certification is a must, and efficiency gain in certification will be a significant gain.
Means for validation/verification	SINTEF will participate in the analysis of how certification will affect and safely benefit, applying existing techniques based on variability models and how be applied in most optimal way. All analysis will be included in deliverable D5.5.2.
Source	DOW/Kick-off
Additional Information	

5.2 Certification Standard(s)

ID	R 5.2.1	
Topic	Certification V&V	
Subtopic	Certification Standard(s)	
Name	Meet compliance with certification standards with the application domains	
Responsibility	WP	5
	Lead partner	IKL
	Participating partners	TUV, ALSTOM, FENTISS, ST, TRT
Description	The requirements of the DREAMS project for the demonstrator use cases shall be compliant with appropriate standards.	
Rationale	IEC-61508 is the standard for functional safety of electrical/electronic/programmable electronic safety-related systems and is the parent of derivative standards of machinery (ISO-13849).	

	 <p>Avionic standards “Software Considerations in Airborne Systems and Equipment Certification” (DO-178B(or C)), “Design Assurance Guidance for Airborne Electronic Hardware” (DO-254) and “Integrated Modular Avionics (IMA) Development Guidance and Certification Considerations” (DO-297) (and higher level process requirements “Guidelines for Development of Civil Aircraft and Systems” (ARP 4754A) and “Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment” (ARP 4761)) should be considered and respected in the DREAMS framework/platform. Evolutions (new solutions) should be provided if needed (*note*: they should be in order to consider multi-cores. Also a certification authority should follow the proposition of the evolution, but this is ensured in the DREAMS project by TUV).</p>
Significance	High
Means for validation/verification	<p>IEC-61508: Inspection of Use Case Requirements and assessment by the certification authority TUV Rheinland.</p> <p>Other standards: Partners interested in compliance of the demonstrators in other standards should provide their own mean for Validation/Verification by the certification authority external to the DREAMS project.</p>
Source	DoW (pages 78, 84, 87 T6.1/T7.1/T8.1 Use case specification and evaluation methodology & pages 10 1.1.1.6 Obj. 6: Feasibility of DREAMS Architecture in Real-World Scenarios), WG Avionics
Additional Information	This requirement doesn't mean that the demonstrators and the used DREAMS framework will be developed using these standards, but that the development of the framework considers them so that the solution can be applied to real world critical systems in the future.

ID	R 5.2.2	
Topic	Certification V&V	
Subtopic	Safety	
Name	Qualification-related requirements on tools developed within DREAMS	
Responsibility	WP	5
	Lead partner	TUV
	Participating partners	TUV
Description	T3 / T2 tools developed during DREAMS project have to consider the requirements of IEC 61508 regarding tool qualifications.	
Rationale	<p>The degree of verification needed for tools depends on the possibility and simplicity to verify the output of tools against its input. Due to this fact the IEC61508 classifies the tools into 3 categories:</p> <p>T3: generates outputs which can directly or indirectly contribute to the executable code of the safety related system Example: translator, compiler, linker, assembler...</p> <p>T2: supports the test or verification of the design or executable code, where errors in the tool can fail to reveal defects but cannot directly create errors in the executable software; Example: static code analysis, emulator, simulator, test tools ...</p> <p>T1: generates no outputs which can directly or indirectly contribute to the executable code (including data) of the safety related system; Example: text editor</p>	
Significance		
Means for validation/verification	<ul style="list-style-type: none"> - A tool manual or tool specification must be available. - Errata sheet / bug list / release notes must be available - For each requirement described in the manual or specification there shall be at least one corresponding test case available. - Revisions have to managed a with configuration-management-tool - Libraries / libraries elements have to be developed according IEC 61508-3 (safety code) 	

	Validation partner are all partners who will develop new tools. They have to provide/collect the above mentioned information.
Source	IEC 61508-3, chapter 7.4
Additional Information	During the DREAMS project the tools will not be qualified, but it will be assessed whether the basics (as specified in "Means of validation/verification", here above) are fulfilled.

ID	R 5.2.3	
Topic	Certification V&V	
Subtopic	Safety	
Name	Qualification-related requirements on existing tools	
Responsibility	WP	5
	Lead partner	TUV
	Participating partners	TUV
Description	Existing required tools need to be checked if minimal requirements as specified in the "Means for Validation/Verification" are met. Furthermore the Tool name, Version, Release Date and a short description of the use within the DREAMS project shall be gathered	
Rationale	In case the minimal requirements are not met for existing tools it is possible to assess, if this is critical for certification of future DREAMS based products	
Significance		
Means for validation/verification	Checking of the following: <ul style="list-style-type: none"> • A tool manual or tool specification must be available. • Errata sheet / bug list / release notes must be available • Revisions have to managed a with configuration-management-tool 	
Source	IEC 61508-3, chapter 7.4	
Additional Information	T4.4 could, as part of the tool map, identify those that fall in this category (D 4.4.1).	

5.3 Cross-Domain Mixed-Criticality Patterns (IEC-61508)

ID	R 5.3.1	
Topic	Certification V&V	
Subtopic	Cross-Domain Mixed-Criticality Patterns (IEC-61508)	
Name	Cross-domain pattern examples (e.g. I/O server, communication server)	
Responsibility	WP	5
	Lead partner	UPV
	Participating partners	TUV, ALSTOM, FENTISS,IKL
Description	A set of cross-domain mixed-criticality patterns shall be identified and collected based on the previous experience and on user needs	
Rationale	Previous projects' and field experience can be used to identify common patterns present in mixed-criticality systems and partitioned architectures specifically. The DREAMS project can benefit from the partners' knowledge and experience on the topic.	
Significance	High	
Means for validation/verification	UPV will provide a list of the identified patterns produced along with the detailed description of each of them. This output should also contain a reference to the origin of each pattern. Part of D 5.3.1	
Source	DoW (T5.3 Cross-Domain Mixed-Criticality Patterns)	
Additional Information		

ID	R 5.3.2	
Topic	Certification V&V	
Subtopic	Cross-Domain Mixed-Criticality Patterns (IEC-61508)	
Name	Design of dependability patterns	
Responsibility	WP	5
	Lead partner	UPV
	Participating partners	TUV, ALSTOM, FENTISS,IKL
Description	A set of design and architectural patterns shall be defined contributing to the dependability and safety of a system	
Rationale	The contribution of this WP should include providing engineers with already consolidated patterns to ease the development of dependable and safe systems based on the DREAMS architecture.	

	These patterns should be reflected in a reduction of development costs when dependability and safety concerns are present.
Significance	High
Means for validation/verification	UPV will provide a list of the identified patterns shall be produced along with the detailed description of each of them. This output should also contain a reference to the origin of each pattern. Part of deliverable D 5.3.1.
Source	DoW (T5.3 Cross-Domain Mixed-Criticality Patterns)
Additional Information	

ID	R 5.3.3	
Topic	Certification V&V	
Subtopic	Cross-Domain Mixed-Criticality Patterns (IEC-61508)	
Name	Link to modular safety-cases	
Responsibility	WP	5
	Lead partner	UPV
	Participating partners	TUV, ALSTOM, FENTISS,IKL
Description	It shall be established how mixed-criticality patterns may be used when building a safe system based on modular safety-cases	
Rationale	This shall provide a guide on how to apply the patterns when a system integrates the modular elements of the DREAMS architecture which contribute to its overall safety	
Significance	High	
Means for validation/verification	UPV will detail, for each identified pattern, its contribution to the overall safety of the system and its relation to modular safety-cases. Part of deliverable D 5.3.1.	
Source	DoW (T5.3 Cross-Domain Mixed-Criticality Patterns)	
Additional Information		

ID	R 5.3.4	
Topic	Certification V&V	
Subtopic	Cross-Domain Mixed-Criticality Patterns (IEC-61508)	
Name	Definition of V&V strategy	
Responsibility	WP	5

	Lead partner	IKL
	Participating partners	TUV, ALSTOM, FENTISS,UPV
Description	A set of cross domain mixed-criticality patterns shall be collected and described, taking into account previous results from previous projects, the state of the art of mixed criticality systems V&V and certification, the modular safety cases from T 5.1, results from FP7 Teresa and input from WP1-T1.2.	
Rationale	Cross-domain mixed-criticality patterns will guide and support engineers towards solutions that solve commonly occurring problems in the development of mixed-criticality products (from design to verification & validation).	
Significance	High	
Means for validation/verification	IKL will collect cross-domain mixed criticality patterns including the development of relevant patterns. Part of deliverable D 5.3.1. ALSTOM will validate the patterns in the Wind Power demonstrator by integration.	
Source	DoW (T 5.3 Cross-Domain Mixed-Criticality Patterns)	
Additional Information		

5.4 Modular safety-case for mixed-criticality systems

ID	R 5.4.1	
Topic	Certification V&V	
Subtopic	Modular safety-case for mixed-criticality systems	
Name	Independent Certification of the Platform	
Responsibility	WP	5
	Lead partner	IKL
	Participating partners	TUV, FENTISS, ALSTOM
Description	<p>The DREAMS platform shall provide means for modular certification at different levels. Modular safety-cases for hypervisor, COTS and networks must be identified.</p> <p>Example: application/communication scheduling solutions that facilitate the introduction of new applications in a system ensuring that previous systems requirements are not impacted. This could be ensure by modeling solutions, hardware capabilities, network approaches, etc.</p>	

Rationale	Modular safety cases will provide an easier certification by re-use possibilities of available evidence. This is required in order to reduce certification cost/time.
Significance	High
Means for validation/verification	IKL will participate in the completion of deliverables: <ul style="list-style-type: none"> • D5.1.1 Modular safety-case for hypervisor • D5.1.2 Modular safety-case for selected COTS multicore processors • D5.1.3 Modular safety-case for selected mixed-criticality networks • D7.2.1: Wind power demonstrator Inspection by TÜV Rheinland of the correctness of the modular safety cases
Source	Windpower WG, ARTEMIS, DoW (pg 49, T5.1 Modular Safety Case)
Additional Information	Note: In the context of the DREAMS project the independent certification body will be TUV-Rheinland (member of the consortium)

ID	R 5.4.2	
Topic	Certification V&V	
Subtopic	Modular safety-case for mixed-criticality systems	
Name	Modular Certification of Subsystems	
Responsibility	WP	7
	Lead partner	IKL
	Participating partners	TUV, FENTISS, ALSTOM
Description	The wind power demonstrator must be built using the modular safety cases defined in R 5.4.1.	
Rationale	Modular safety cases will provide an easier certification by re-use possibilities or available evidence.	
Significance	High	
Means for validation/verification	IKL will participate in the identification of modular subsystems in the Wind power demonstrator (D7.2.1). Inspection by TÜV Rheinland of the correct use of the modular safety cases in the Wind Power demonstrator	
Source	ARTEMIS, DoW (T 7.2 Implementation of wind power use case)	
Additional Information		

ID	R 5.4.3	
Topic	Certification V&V	
Subtopic	Modular safety-case for mixed-criticality systems	
Name	Incremental/modular certification	
Responsibility	WP	1, 5
	Lead partner	IKL
	Participating partners	TRT, ONERA, RTAW, TTT, FENTISS, TUKL
Description	<p>The DREAMS architecture shall provide means for incremental/modular certification at different levels.</p> <p>Example: application/communication scheduling solutions that facilitate the introduction of new applications in a system ensuring that previous systems requirements are not impacted. This could be ensured by modeling solutions, hardware capabilities, network approaches, etc.</p>	
Rationale	This is required in order to reduce certification cost/time.	
Significance	High (from industrial point of view)	
Means for validation/verification	<p>Development for certification (WP5)</p> <p>TRT must validate this requirement in the avionics demonstrator</p>	
Source	WG Avionics, Industrial practices (avionics)	
Additional Information		

ID	R 5.4.4	
Topic	Certification V&V	
Subtopic	Modular safety-case for mixed-criticality systems	
Name	Modular certification of XtratuM (Hypervisor)	
Responsibility	WP	5
	Lead partner	FENTISS
	Participating partners	TUV, IKL, ALSTOM
Description	The independent certification of a hypervisor shall be established.	
Rationale	It is key for modular safety cases to rely in a certified hypervisor providing isolation of the modules. The hypervisor itself is considered part of the modular architecture and therefore the	

	approach to its certification must make special emphasis in the properties that enable this modularity, that is, app/HW dependant configuration and the provided API (XM API)
Significance	High
Means for validation/verification	FENTISS will participate in the definition and generation of the following outputs: <ul style="list-style-type: none"> – Safety concept, – IEC61508 tailoring, – list of requirements (specification) for a tool for validation of XMCF parsing output (XM configuration), – list of safety requirements allocated to the hypervisor contributing to the overall system safety – Modular safety case (D 5.1.1) for a hypervisor (XtratuM).
Source	DoW (T5.1 Modular Safety Case)
Additional Information	

ID	R 5.4.5	
Topic	Certification V&V	
Subtopic	Modular safety-case for mixed-criticality systems	
Name	Modular certification of multicore processor	
Responsibility	WP	5
	Lead partner	IKL
	Participating partners	TUV, FENTISS, ALSTOM
Description	The COTS shall be considered as an independent safety unit and its safety case must be defined.	
Rationale	The modular certification creates smaller safety items of the system that help in the management of the complexity of certification. Also the reusability is increased among safety projects reducing the overall cost.	
Significance	High	
Means for validation/verification	IKL will participate in the definition of a modular safety-case for selected COTS multicore processors (D 5.4.2). Validating Partner will be ALSTOM by including the COTS safety-case in the Wind Power demonstrator.	
Source	DoW (T5.1 Modular Safety-Case)	

Additional Information	For safety applications please see also R2.6.1
------------------------	--

ID	R 5.4.6	
Topic	Certification V&V	
Subtopic	Modular safety-case for mixed-criticality systems	
Name	Modular certification of the mixed-criticality network	
Responsibility	WP	5
	Lead partner	IKL/TTT
	Participating partners	TUV, ALSTOM
Description	DREAMS shall explore how the mixed-criticality network can contribute to the modular certification of a system.	
Rationale	Starting from an existing mixed-criticality network, like TTEthernet, DREAMS will explore how the existing network contributes to modular certification. Likewise, DREAMS will study modifications and improvements of existing mixed-criticality networks towards modular certification.	
Significance	medium	
Means for validation/verification	IKL/TTT will participate in the definition of modular safety case of selected mixed criticality networks (D5.4.3). The validation of the modular safety case will be done by means of use case-based analysis in the avionics demonstrator in which the mixed-criticality network is used in T 6.3 Project technologies assessment.	
Source	DoW (T5.1 Modular Safety Case)	
Additional Information	For safety networks see also R3.2.8	

ID	R 5.4.7	
Topic	Certification V&V	
Subtopic	Modular safety-case for mixed-criticality systems	
Name	Modular certification of NoC	
Responsibility	WP	5
	Lead partner	IKL/USIEGEN
	Participating partners	USIEGEN, IKL, ST, TTT

Description	The NoC shall be considered as an independent safety unit and its safety case must be defined.
Rationale	The modular certification creates smaller safety items of the system that help in the management of the complexity of certification. Also the reusability is increased among safety projects reducing the overall cost.
Significance	High
Means for validation/verification	ST will validate the inclusion of the safety-case for the NoC in the HealthCare demonstrator.
Source	DoW (T5.1 Modular Safety Case)
Additional Information	

ID	R 5.4.8	
Topic	Certification V&V	
Subtopic	Modular safety-case for mixed-criticality systems	
Name	Modular certification of Gateways	
Responsibility	WP	5
	Lead partner	IKL/USIEGEN
	Participating partners	USIEGEN, IKL, TTT , ST
Description	The gateway shall be considered as an independent safety unit and its safety case must be defined.	
Rationale	The modular certification creates smaller safety items of the system that help in the management of the complexity of certification. Also the reusability is increased among safety projects reducing the overall cost.	
Significance	High	
Means for validation/verification	ST will validate the inclusion of the safety-case for the NoC (gateway) in the HealthCare demonstrator.	
Source	DoW (T5.1 Modular Safety Case)	
Additional Information		

5.5 Overall Strategy

ID	R 5.5.1	
Topic	Certification V&V	
Subtopic	Overall Strategy	
Name	Subset of WP1, WP2, WP3 meet safety constraints and support it	
Responsibility	WP	5
	Lead partner	IKL
	Participating partners	USIEGEN, IKL, TRT, ONERA, RTAW,TTT,TUKL,FORTISS, SINTEF, ALSTOM, ST, TEI, UPV, FENTISS, TUV
Description	Requirements from WP1, WP2 and WP3 should build the modular safety case.	
Rationale	Modular safety cases will be defined based on the services and properties defined in the architectural style (WP1), selecting services and attributes to be linked with safety and certification standards.	
Significance	Medium	
Means for validation/verification	IKL will participate in the completion of deliverables: <ul style="list-style-type: none"> • D5.1.1 Modular safety-case for hypervisor • D5.1.2 Modular safety-case for selected COTS multicore processors • D5.1.3 Modular safety-case for selected mixed-criticality networks Inspection of the Requirements and the DREAMS architecture by all partners.	
Source	DoW (pg 49, T5.1 Modular Safety Case)	
Additional Information		

ID	R 5.5.2	
Topic	Certification V&V	
Subtopic	Overall Strategy	
Name	Certification process of DREAMS with limitations	
Responsibility	WP	5
	Lead partner	IKL
	Participating partners	TUV

Description	The safety requirements should be met taking into considerations that the scope is research (not certification of the outcome): - Pave the way: show that safety has been considered and it is feasible. - An industrial project could implement DREAMS (or a subset) and provide a certified solution: system or compliant item
Rationale	The real certification of the DREAMS architecture is out of the scope of the project.
Significance	Medium
Means for validation/verification	Based on this get positive assessment from TUV Rheinland with comments and limitations
Source	DREAMS Kick-of-Meeting, WP5 Parallel Session
Additional Information	Note: Not everything must be compliant with safety standards, only the subset that is representative and at least the subset used in WP7 demonstrator

ID	R 5.5.3	
Topic	Certification V&V	
Subtopic	Overall Strategy	
Name	Reusability of safety items	
Responsibility	WP	5
	Lead partner	IKL
	Participating partners	USIEGEN, IKL, TRT, ONERA, RTAW,TTT,TUKL,FORTISS, SINTEF, ALSTOM, ST, TEI, UPV, FENTISS, TUV
Description	The results of the project should help industrial projects based on DREAMS (or subset) to be certified (feasible)	
Rationale	An industrial project could implement DREAMS (or a subset) and provide a certified solution: system or compliant item	
Significance	Medium	
Means for validation/verification	All demonstrator partners TRT, ALSTOM, ST must validate this requirement by providing the use of the results of the project in the Wind Power (D7.2.1), Avionics (D6.2.2) and Healthcare (D 8.1.1) demonstrators.	
Source	DREAMS Kick-of-Meeting, WP5 Parallel Session	
Additional Information		

5.6 Test bed for validation, verification and evaluation of extra-functional properties

ID	R 5.6.1	
Topic	Certification V&V	
Subtopic	Test bed for validation, verification and evaluation of extra-functional properties	
Name	Gateway simulation building block between off-chip and on-chip networks	
Responsibility	WP	5, 3, 2
	Lead partner	USIEGEN
	Participating partners	USIEGEN, IKL,RTAW,ST,TTT,TEI
Description	<p>A gateway simulation building block shall be developed that couples on-chip and off-chip network simulations. Interfaces shall be provided to off-chip network simulations (e.g., TTEthernet). In addition, an interface of the simulation on-chip network shall be provided.</p> <p>The gateway simulation building block shall simulate the temporal alignment between off-chip and on-chip level simulation tools. Mapping between on-chip transactions and off-chip behaviour shall be supported.</p> <p>Relaying of different types of communication will be supported:</p> <ul style="list-style-type: none"> • Periodic time-triggered messages • Sporadic rate-constrained messages • Best-effort messages • Access to shared memory <p>Simulation of fault isolation mechanisms has to be performed by enforcing message timing and blocking faulty messages (e.g., minimum interarrival times, message period and phase).</p> <p>The gateway simulation building block shall support the simulation of end-to-end channels with message filtering, selective redirection and mapping of virtual channels.</p> <p>The gateway simulation building block shall be modular to support the replacement of individual interfaces while retaining the unchanged interfaces and elements of the gateway.</p>	
Rationale	A simulation building block for gateways must be available to simulate networked multi-core chips.	
Significance	High	
Means for validation/verification	USIEGEN will validate the gateway simulation building block at module-test level and integration-test level in Task T5.2.	

	In Task T5.6, RTaW and USIEGEN will support the use of the testbed in the demonstrators, thereby enabling the validation of the simulation building blocks using the demonstrators.
Source	DoW
Additional Information	Support for configuration Support for fault injection Support for modularity

ID	R 5.6.2	
Topic	Certification V&V	
Subtopic	Test bed for validation, verification and evaluation of extra-functional properties	
Name	Simulation building blocks for off-chip networks	
Responsibility	WP	2, 3, 5
	Lead partner	IKL/USIEGEN
	Participating partners	USIEGEN,IKL,RTAW
Description	<p>Network building blocks for switches shall be provided. Different topologies (e.g., multiple stars, rings) will be supported.</p> <p>Network building blocks for the protocols TTEthernet and EtherCAT shall be provided.</p> <p>Different mechanisms for handling contention between traffic types will be simulated (e.g., shuffling, timely block).</p> <p>The network building blocks will simulate the fault isolation mechanisms of the protocols. The containment of timing messages failures based on time-triggered communication schedules and the specification of rate-constraints has to be simulated.</p> <p>End system models will be provided containing network interfaces. The end system models shall be extendable with end system's specification of application behaviour.</p>	
Rationale	Simulation building blocks for off-chip networks is essential to decrease development time and cost.	
Significance	High	
Means for validation/verification	IKL, RTaW and USIEGEN will validate the network simulation building block at module-test level and integration-test level in Task T5.2.	

	In Task T5.6, IKL, RTaW and USIEGEN will support the use of the testbed in the demonstrators, thereby enabling the validation of the simulation building blocks using the demonstrators.
Source	DoW
Additional Information	Support for configuration Support for fault injection

ID	R 5.6.3	
Topic	Certification V&V	
Subtopic	Test bed for validation, verification and evaluation of extra-functional properties	
Name	Simulation building blocks for on-chip networks	
Responsibility	WP	2, 5
	Lead partner	USIEGEN
	Participating partners	USIEGEN,ST,RTAW,IKL, TEI
Description	<p>Simulation building blocks for the DREAMS network-on-a-chip shall be provided. The building blocks will offer services for rate-constrained event-triggered communication as well as time-triggered communication.</p> <p>Simulation building blocks for network interfaces will be provided as a foundation for simulating application components.</p> <p>The simulation blocks will allow to simulate the dynamic replacement of the NoC configuration (via local resource manager / local resource scheduler).</p> <p>The building blocks will allow to simulate the fault propagation/containment, the mechanisms for temporal and spatial partitioning, and the timing of the NoC.</p>	
Rationale	Simulation building blocks for on-chip networks are required to gain insights into design.	
Significance	High	
Means for validation/verification	<p>USIEGEN will validate the NoC simulation building block at module-test level and integration-test level in Task T5.2.</p> <p>In Task T5.6, USIEGEN and RTaW will support the use of the testbed in the demonstrators, thereby enabling the validation of the simulation building blocks using the demonstrators.</p>	
Source	DoW	
Additional Information	Support for configuration	

	Support for fault injection
--	-----------------------------

ID	R 5.6.4	
Topic	Certification V&V	
Subtopic	Test bed for validation, verification and evaluation of extra-functional properties	
Name	Simulation building for execution service	
Responsibility	WP	2, 5
	Lead partner	USIEGEN
	Participating partners	USIEGEN, IKL, RTAW, UPV
Description	<p>A simulation building block for the execution service shall be provided. The simulation building shall be a time-triggered dispatcher for the triggering of simulation tasks with a predefined period and phase.</p> <p>The simulation tasks can be provided by the user of the test bed.</p>	
Rationale	The simulation building block for the dispatcher will emulate the behaviour of the hypervisor at high abstraction level.	
Significance	High	
Means for validation/verification	<p>USIEGEN will validate the simulation building block at module-test level and integration-test level in Task T5.2.</p> <p>In Task T5.6, USIEGEN and RTaW will support the use of the testbed in the demonstrators, thereby enabling the validation of the simulation building blocks using the demonstrators.</p>	
Source	DoW	
Additional Information		

ID	R 5.6.5	
Topic	Certification V&V	
Subtopic	Test bed for validation, verification and evaluation of extra-functional properties	
Name	Configuration interfaces to integrate the simulation environment into the DREAMS development process	
Responsibility	WP	5, 4
	Lead partner	USIEGEN

	Participating partners	USIEGEN,RTAW
Description	<p>The configuration of the simulation building blocks (i.e., gateways, on-chip and off-chip networks, network interfaces, and dispatcher) shall be changeable runtime. An update of the configuration performed and/or triggered via the on-chip/off-chip network shall be simulated.</p> <p>The configuration format for the simulation building blocks has to be defined for the configuration tools in WP4.</p> <p>The configuration includes the message timing (e.g., time-triggered schedule, rate-constraints), the frame formats and the specification of the fault containment.</p>	
Rationale	Reduce the time and effort to integrate different simulation building block.	
Significance	High	
Means for validation/verification	USIEGEN and RTaW will validate the configuration of the simulation building blocks in Task T5.2.	
Source	DoW	
Additional Information		

ID	R 5.6.6	
Topic	Certification V&V	
Subtopic	Test bed for validation, verification and evaluation of extra-functional properties	
Name	Observability of timing and behavior	
Responsibility	WP	5, 4
	Lead partner	USIEGEN
	Participating partners	USIEGEN, IKL, RTAW
Description	<p>Sink simulation building block shall be provided, which support the observation of the following properties:</p> <ul style="list-style-type: none"> • End to end latency • Jitter • Throughput • Message failures • Application behavior 	

	<p>Support for filtering of observed data shall be supported.</p> <p>Monitoring of on-chip transactions and off-chip behaviour will be supported.</p>
Rationale	A simulation building block for Sink must be available to capture a simulation result during the simulation run time.
Significance	High
Means for validation/verification	<p>USIEGEN will validate the fault injection mechanisms in Task T5.2.</p> <p>In Task T5.6, USIEGEN and RTaW will support the use of the fault injection and simulation mechanisms in the demonstrators, thereby enabling the validation of the simulation building blocks using the demonstrators.</p>
Source	DoW
Additional Information	

ID	R 5.6.7	
Topic	Certification V&V	
Subtopic	Test bed for validation, verification and evaluation of extra-functional properties	
Name	Fault injection at communication networks	
Responsibility	WP	5
	Lead partner	ST
	Participating partners	ST,USIEGEN, IKL,RTAW
Description	<p>The simulation building block for fault injection at communication networks shall support to inject the following fault types:</p> <ul style="list-style-type: none"> • Omission • Corruption • Delay • Masquerading • Link failures • Component crash failures • Babbling idiot failures <p>The simulation building block for fault injection at communication networks will inject the fault according to the fault configuration (R 5.6.9)</p>	

Rationale	The fault injection is important to evaluating the dependability of System on Chip. Since in the operational environment it is difficult to identify the cause of failure, it is important to have a methodology and the tools that inject faults, create failure or errors and monitor the effects.
Significance	
Means for validation/verification	USIEGEN and RTaW will participate in the validation of the fault injection mechanisms in Task T5.2. In Task T5.6, USIEGEN and RTaW will support the use of the fault injection and simulation mechanisms in the demonstrators, thereby enabling the validation of the simulation building blocks using the demonstrators.
Source	DoW (Tasks 5.2 and 5.6)
Additional Information	

ID	R 5.6.8	
Topic	Certification V&V	
Subtopic	Test bed for validation, verification and evaluation of extra-functional properties	
Name	Simulation building blocks for fault injection at chip level	
Responsibility	WP	5
	Lead partner	RTAW/FENTISS
	Participating partners	RTAW,FENTISS, TT,USIEGEN,IKL
Description	Simulation building blocks for fault injection such as (fault injection at simulated on-chip network interface, fault injection in a simulated on-chip partition, fault injection in hypervisor and fault injection within a complete simulated node of the cluster) shall be provided. It will be possible to integrate the generic simulation building blocks with simulation models of the actual application behaviour in order to simulate systems based on the DREAMS platform and evaluate the timing (e.g., latencies, temporal partitioning, ...), reliability (e.g., probability of correct service based on faulty assumptions, fault containment coverage based on temporal and spatial partitioning).	

	The simulation building block for fault injection injection at communication networks will inject the fault according to the fault configuration (R 5.6.9)
Rationale	Simulation building blocks for fault injection allow to gain insights into design alternatives and design faults at early development stages, thus decreasing development time and cost.
Significance	High
Means for validation/verification	RTaW will participate in the validation of the fault injection mechanisms in Task T5.2. In Task T5.6, RTaW will support the use of the fault injection and simulation mechanisms in the demonstrators, thereby enabling the validation of the simulation building blocks using the demonstrators.
Source	DoW
Additional Information	

ID	R 5.6.9	
Topic	Certification V&V	
Subtopic	Test bed for validation, verification and evaluation of extra-functional properties	
Name	Configuration interface of fault injection mechanisms (e.g., fault containment units, failure modes, failure rates)	
Responsibility	WP	5
	Lead partner	USIEGEN
	Participating partners	USIEGEN,IKL, RTAW
Description	<p>The fault shall be injected in specific time intervals, and it shall be possible to inject the same fault with given frequencies. The fault type shall be defined based on the following criteria:</p> <ul style="list-style-type: none"> • Fault containment unit (e.g., node, processor core, NoC, gateway, off-chip communication link, off-chip switch) • Failure mode (e.g., babbling idiot failure, delay failure, masquerading failure, corruption, omission) • Frequency of failure • Persistence (e.g., permanent, transient) 	
Rationale	The configuration interface of fault injection mechanisms are used to adapt the behavior of the simulation models to DREAMS chips and off-chip communication	

Significance	High
Means for validation/verification	<p>USIEGEN and RTaW will participate in the validation of the fault injection mechanisms in Task T5.2.</p> <p>In Task T5.6, USIEGEN and RTaW will support the use of the fault injection and simulation mechanisms in the demonstrators, thereby enabling the validation of the simulation building blocks using the demonstrators.</p>
Source	DoW
Additional Information	

ID	R 5.6.10	
Topic	Certification V&V	
Subtopic	Test bed for validation, verification and evaluation of extra-functional properties	
Name	Analysis and evaluation of simulation results with respect to timing and reliability	
Responsibility	WP	5
	Lead partner	RTaW
	Participating partners	RTaW, IKL, USIEGEN
Description	<p>The simulation and fault-injection framework should provide functionalities that synthesise complex characteristics out of the simulation events, so that they can be visualized in plotting tools or integrated into reports. Examples of those complex characteristics are average bus loads or end-to-end response times.</p>	
Rationale	<p>Simulations are based on events that drive the simulated evolution of the systems. The isolated observation of these events does not allow drawing conclusion about complex properties such as average bus load or end-to-end response times.</p>	
Significance	High	
Means for validation/verification	<p>RTaW will participate in the validation of the simulation result analysis functionalities in Task T5.2.</p> <p>In Task T5.6, RTaW will support the use of the fault injection and simulation mechanisms in the demonstrators, thereby enabling the validation of the simulation building blocks using the demonstrators.</p>	

Source	DoW
Additional Information	The scope of this requirement is timing related reliability aspects.

ID	R 5.6.11	
Topic	Certification V&V	
Subtopic	Test bed for validation, verification and evaluation of extra-functional properties	
Name	Multi-source simulation building block with reusability of test cases	
Responsibility	WP	5
	Lead partner	USIEGEN
	Participating partners	USIEGEN,RTAW,IKL
Description	<p>A simulation building block for a multi-source core will be provided. The multi-source core will be configured based on a specification of the timing, contents and control information for time-triggered, rate-constrained and best-effort messages.</p> <p>The configuration will realize a test case that can be reused in different DREAMS system configurations.</p>	
Rationale	Multi-source simulation building block allows to test different versions of applications.	
Significance	High	
Means for validation/verification	<p>USIEGEN will validate the multi-source simulation building block at module-test level and integration-test level in Task T5.2.</p> <p>In Task T5.6, RTaW and USIEGEN will support the use of the testbed in the demonstrators, thereby enabling the validation of the simulation building blocks using the demonstrators.</p>	
Source	DoW	
Additional Information		

ID	R 5.6.12	
Topic	Certification V&V	
Subtopic	Test bed for validation, verification and evaluation of extra-functional properties	
Name	Transfer networking test results from simulation to physical systems	
Responsibility	WP	

	Lead partner	TTT
	Participating partners	USIEGEN, RTAW
Description	DREAMS shall develop a process describing how tests executed in simulation can be transferred and tested on a real physical target.	
Rationale	The process will cover test descriptions usable for both simulation and test on the physical target, as well as, description on how to relate the output of simulation to the execution of the test on the physical target. Ideally, the output of the execution on the real target will yield useful information to steer the tests in simulation, thereby enabling a feedback loop from the real target back to the simulation model.	
Significance		
Means for validation/verification	TTT will establish a framework for the transfer of test case results from the simulation environment to physical environment for the mixed-criticality network. Thereby, TTT will improve the test reusability to utilize this framework on the components of WP3. Transfer framework technique will be included in D 5.2.2.	
Source	DoW	
Additional Information		

ID	R 5.6.13	
Topic	Certification V&V	
Subtopic	Test bed for validation, verification and evaluation of extra-functional properties	
Name	Support for formal verification	
Responsibility	WP	5
	Lead partner	ST
	Participating partners	TEI
Description	The functional verification shall be implemented through a coverage-driven approach based on dynamic and static methodologies. In dynamic context automatic e-coded checkers shall be used to test functionalities while both code and functional coverage are used to verify the random stimuli generation. This methodology shall be applied to black-box functionalities. In static (or formal) context, white-box assertions shall be used in order to target sub-modules functionalities that cannot be simply addressed at top level. Same assertions should then be activated during dynamic simulation.	

Rationale	Formal approach is powerful when validating protocol on-chip / off-chip interfaces. In fact, Formal approach allows exhaustive verification in case of small size designs and when applicable, it allows to identify quickly and simply complex corner cases
Significance	
Means for validation/verification	ST will contribute to the development a formal verification framework via a process algebra language for modeling and a model checking tool for verification of temporal logic properties of the DREAMS architecture. Included in D 5.2.2 deliverable. To be Validated in the STNOC technology
Source	DoW (T 5.2 Simulation, Verification and fault-injection framework).
Additional Information	

5.7 Integration in industrial (safety) engineering process

ID	R 5.7.1	
Topic	Certification V&V	
Subtopic	Integration in industrial (safety) engineering process	
Name	Tool integration compatible with IEC 61508	
Responsibility	WP	4, 5
	Lead partner	RTaW
	Participating partners	IKL, SINTEF, TUV
Description	Verification steps shall be defined, which have to be done before and after the use of a tool.	
Rationale	For safety applications the solely use of the tool itself is not sufficient. In addition the use of a tool in the engineering process has to be considered regarding its pre- and post- conditions. Pre-conditions could be e.g. the definition of verification activities for specification, which have to be done before the tool is used. Post-condition could be e.g. manual checks whether the tool has been worked as expected.	
Significance		

Means for validation/verification	RTaW will validate this requirement in T5.4 by analysing the proposed integration of tools into the safety engineering process. RTaW will also generate a detailed mapping document of tools to a safety engineering process of reference, D 5.4.1.
Source	DoW (T5.4 Tool Integration in industrial Safety engineering process)
Additional Information	

ID	R 5.7.2	
Topic	Certification V&V	
Subtopic	Integration in industrial (safety) engineering process	
Name	Qualification-related requirements on tools developed within DREAMS	
Responsibility	WP	5
	Lead partner	TUV
	Participating partners	TUV
Description	T3 / T2 tools developed during DREAMS project have to consider the requirements of IEC 61508 regarding tool qualifications.	
Rationale	<p>The degree of verification needed for tools depends on the possibility and simplicity to verify the output of tools against its input. Due to this fact the IEC61508 classifies the tools into 3 categories:</p> <p>T3: generates outputs which can directly or indirectly contribute to the executable code of the safety related system Example: translator, compiler, linker, assembler...</p> <p>T2: supports the test or verification of the design or executable code, where errors in the tool can fail to reveal defects but cannot directly create errors in the executable software; Example: static code analysis, emulator, simulator, test tools ...</p> <p>T1: generates no outputs which can directly or indirectly contribute to the executable code (including data) of the safety related system; Example: text editor</p>	
Significance		
Means for validation/verification	<ul style="list-style-type: none"> - A tool manual or tool specification must be available. - Errata sheet / bug list / release notes must be available 	

	<ul style="list-style-type: none"> - For each requirement described in the manual or specification there shall be at least one corresponding test case available. - Revisions have to managed a with configuration-management-tool - Libraries / libraries elements have to be developed according IEC 61508-3 (safety code) <p>Validation partner are all partners who will develop new tools. They have to provide/collect the above mentioned information.</p>
Source	IEC 61508-3, chapter 7.4
Additional Information	During the DREAMS project the tools will not be qualified, but it will be assessed whether the basics (as specified in "Means of validation/verification", here above) are fulfilled.

ID	R 5.7.3	
Topic	Certification V&V	
Subtopic	Integration in industrial (safety) engineering process	
Name	Qualification-related requirements on existing tools	
Responsibility	WP	5
	Lead partner	TUV
	Participating partners	TUV
Description	Existing required tools need to be checked if minimal requirements as specified in the "Means for Validation/Verification" are met. Furthermore the Tool name, Version, Release Date and a short description of the use within the DREAMS project shall be gathered	
Rationale	In case the minimal requirements are not met for existing tools it is possible to assess, if this is critical for certification of future DREAMS based products	
Significance	Medium	
Means for validation/verification	Checking of the following: <ul style="list-style-type: none"> • A tool manual or tool specification must be available. • Errata sheet / bug list / release notes must be available • Revisions have to managed a with configuration-management-tool 	
Source	IEC 61508-3, chapter 7.4	
Additional Information	T4.4 could, as part of the tool map, identify those that fall in this category (D 4.4.1).	

ID	R 5.7.4	
Topic	Certification V&V	
Subtopic	Integration in industrial (safety) engineering process	
Name	Qualification Validation partner are all partners who will develop new tools. They have to provide/collect the above mentioned information.	
Responsibility	WP	5
	Lead partner	TUV
	Participating partners	TUV
Description	In safety applications "Engineering process" contains programming, parameterization, configuration and the download into the target system. For those tools the requirements of IEC 62061, chapter 6.11.2 "Software based parameterization" have to be observed.	
Rationale	The engineering process as described above must consider the integrity and identity of the downloaded parameter, configuration or any other safety related information.	
Significance		
Means for validation/verification	<p>Verification: based on document review;</p> <p>"Validation: Fault Injection Tests (FIT); Verification / Validation has to be done by all partners who are developing engineering tools. Examples for FIT could be:</p> <p>Is parameterization protected against unauthorized access (e.g. by password)</p> <p>Are measures implemented to:</p> <ul style="list-style-type: none"> • control the value range of inputs; • control corruption of data; • control the effects of errors from the process of transmitting parameters; • control the effects parameter transmission that was incomplete; and • control the effects of hardware or software faults or failures of parameterization tool. • confirmation of parameters <p>diverse functions for encoding and decoding of parameters</p>	

Source	IEC 62061, chapter 6.11.2
Additional Information	See R4.3.2 "Means for validation/verification" T4.4 could, as part of the tool map, identify those that fall in this category (D 4.4.1).

6 Requirements for Avionics Demonstrator

In this section, the requirements specific to the avionics demonstrator developed in the DREAMS project (Sec. 6.1 – 6.5) will be presented. The following discussion includes the organization of these requirements into subtopics and their relation to other requirement (sub-) topics in this document.

The avionics demonstrator, as the other demonstrators developed in DREAMS, as a safety critical solution using the DREAMS architecture includes the requirements described in the other requirements sections, from those concerning the DREAMS architecture (Sec. 1), solutions (Sec. 2, 3, 10, 11) and tools (Sec. 4) to those concerning the development process (Sec. 9) and certification (Sec. 5). However, some specific requirements concerning the avionics domain should also be considered in the development of the avionics demonstrator. Those are:

- Technical requirements concerning specially the timing requirements that are found in most avionics solutions (Sec. 6.1). Those apply to the demonstrator developed in the context of DREAMS but also apply to most of the networked solutions used on commercial solutions. Those are high level requirements that the DREAMS architecture should be able to support in order to be used in the avionics domain.
- Specific solutions (software and hardware) that must be used in the avionics demonstrator (Sec. 6.2), but that apply to most of the critical applications found in a plane. These concern mainly the platform which must be an embedded solution for the avionics domain and the network that must be able to provide some timing predictability.
- As an example of a safety critical avionics application the demonstrator must follow a couple of specific development and certification requirements (Sec. 6.3 and 6.4) in addition to those already defined in DREAMS.
- Finally, while most of the developments in DREAMS are not strictly speaking compulsory for an avionics application, the faults management and monitoring must be strictly respected in order to be able to apply the DREAMS architecture and tools in an avionics solution (Sec. 6.5). The DREAMS architecture and tools will enhance and positively impact the design and development of future solutions if this requirement is supported.

6.1 Domain-Specific Timing Requirements

ID	R 6.1.1
Topic	Avionics
Subtopic	Domain-Specific Timing Requirements
Name	Timing requirements for execution of tasks

Responsibility	WP	6
	Lead partner	TRT
	Participating partners	TRT, UPV, VOSYS, FENTISS, RTaW
Description	The minimum period time supported by the DREAMS architecture (including the HW/SW platform, methodology, models and tools) shall be 50ms, meaning that a cyclic task shall be able to start its execution 50ms after the start of the previous execution of the same task.	
Rationale	50ms is required to be the minimum period time supported by the DREAMS avionic demonstrator	
Significance	High	
Means for validation/verification	The requirement will be validated in the avionic demonstrator in Task T6.2/T6.3.	
Source	Avionics demonstrator applications.	
Additional Information		

ID	R 6.1.2	
Topic	Avionics	
Subtopic	Domain-Specific Timing Requirements	
Name	Intra-application end-to-end communication time	
Responsibility	WP	6
	Lead partner	TRT
	Participating partners	TRT, TTT, ST, USIEGEN, RTaW
Description	The end-to-end communication time between two tasks of an application subsystem on the DREAMS architecture shall be less than 50ms.	
Rationale	The communication time between two tasks of an application from the time the emitter sends a message to the time the receiver processes it should be less than 50ms for a typical avionic use case. This requirement can be ignored if the tasks are located in different subsystems and the communication is done through the network (AFDX, Ethernet ...); see R6.1.3.	
Significance	High	
Means for validation/verification	The requirement will be validated in the avionic demonstrator in Task T6.2/T6.3.	

Source	Avionics demonstrator applications.
Additional Information	

ID	R 6.1.3	
Topic	Avionics	
Subtopic	Domain-Specific Timing Requirements	
Name	End-to-end communication time between applications	
Responsibility	WP	6
	Lead partner	TRT
	Participating partners	TRT, TTT, ST, USIEGEN, RTaW
Description	The DREAMS architecture shall ensure that it is possible to design a solution where two subsystems can be connected in which the time between a data generation from one application subsystem and the time the data has been delivered to another application subsystem is less than 1s. This latency shall be ensured even when the two application subsystems are in different clusters.	
Rationale	This requirement is the complementary to R6.1.2 applied to networks. Required to support a large variety of avionics solutions. The actual communication time might depend on the number of hops between emitter and receiver, but the DREAMS architecture should be able to allow the design of solutions which respect and validate this time limits of 1s.	
Significance	High	
Means for validation/verification	The requirement will be validated in the avionic demonstrator in Task T6.2/T6.3.	
Source	Avionics demonstrator applications.	
Additional Information		

ID	R 6.1.4	
Topic	Avionics	
Subtopic	Domain-Specific Timing Requirements	
Name	Network bandwidth requirements	
Responsibility	WP	6
	Lead partner	TRT

	Participating partners	TRT
Description	The DREAMS architecture shall provide networks with a transmission rate of at least 100Mbps.	
Rationale	The avionics demonstrator requires a network technology allowing at least 100Mbps transmission rates.	
Significance	High	
Means for validation/verification	The requirement will be validated in the avionic demonstrator in Task T6.2/T6.3.	
Source	Avionics demonstrator applications.	
Additional Information		

6.2 Use Case

ID	R 6.2.1	
Topic	Avionics	
Subtopic	Use Case	
Name	Hardware platform for avionics demonstrator	
Responsibility	WP	6
	Lead partner	TRT
	Participating partners	TRT, FENTISS, TTT
Description	The major part of avionics demonstrator shall be based on the hardware solution used by TRT for the embedded real-time applications should be a Freescale PowerPC based multi-core.	
Rationale	TRT already designs and maintains the solutions using the Freescale PowerPC platforms. In order to increase the applicability of the DREAMS architecture in future TRT applications, the architecture must be suitable to extend this platform.	
Significance	High	
Means for validation/verification	The requirement will be validated in the wind power use case in Task T6.2.	
Source	TRT	

Additional Information	
------------------------	--

ID	R 6.2.2	
Topic	Avionics	
Subtopic	Use Case	
Name	Mixed criticality on Freescale PowerPC platform	
Responsibility	WP	6
	Lead partner	TRT
	Participating partners	TRT, ONERA, FENTISS
Description	The avionics demonstrator based on a Freescale multi-core PowerPC platform and the DREAMS architecture shall combine safety and non-safety functionalities by means of the hypervisor.	
Rationale	The Freescale PowerPC multi-core platforms support functionalities with no safety requirements such as the Human Machine Interface (non real-time) or the Supervisory System (real-time). These functionalities are to be combined with a new safety-critical application subsystem: the protection system in charge of maintaining the critical applications in a safe state. This functionality will be included into the same platform by using the DREAMS architecture.	
Significance	High	
Means for validation/verification	The requirement will be validated in the avionics use case in Task T6.2.	
Source	TRT	
Additional Information		

ID	R 6.2.3	
Topic	Avionics	
Subtopic	Use Case	
Name	Off-chip network protocol	
Responsibility	WP	6
	Lead partner	TRT
	Participating partners	TRT, TTT

Description	The off-chip network protocol used in the avionics demonstrator shall be based on highly reliable deterministic network ethernet, like AFDX or TTEthernet.
Rationale	In order to increase the applicability of the DREAMS architecture in future TRT applications, the architecture must support the currently used off-chip network solution.
Significance	High
Means for validation/verification	The requirement will be validated in the avionics use case in Task T6.2.
Source	TRT
Additional Information	Given the participation of TTTech in the DREAMS project the used network technology will be TTEthernet.

ID	R 6.2.4	
Topic	Avionics	
Subtopic	Use Case	
Name	XtratuM hypervisor	
Responsibility	WP	6
	Lead partner	TRT
	Participating partners	TRT, FENTISS, UPV, ONERA
Description	The hypervisor used to provide TSP in the avionics demonstrator shall be XtratuM.	
Rationale	In the development of the applications the portability is needed, and the hypervisor provides it. Given the critical/mixed-critical target of the applications time and space partitioning should be ensured by the hypervisor.	
Significance	High	
Means for validation/verification	The requirement will be validated in the wind power use case in Task T6.2.	
Source	TRT	
Additional Information		

6.3 Assessment

ID	R 6.3.1	
Topic	Avionics	
Subtopic	Assessment	
Name	Demonstrator assessment	
Responsibility	WP	6
	Lead partner	TRT
	Participating partners	TRT
Description	The assessment shall be conducted analytically and by experimental evaluation. Key performance indicators shall be measured and compared against initially defined values.	
Rationale	This is required in order to assess the innovation carried out in the DREAMS project.	
Significance	High	
Means for validation/verification	The requirement will be validated in the avionic demonstrator in Task T6.2/T6.3.	
Source	DoW	
Additional Information		

ID	R 6.3.2	
Topic	Avionics	
Subtopic	Assessment	
Name	Methods and tools must be applied for the Avionics demonstrator	
Responsibility	WP	6
	Lead partner	TRT
	Participating partners	TRT, RTaW
Description	The avionics demonstrator shall apply as many as possible of the methods and tools provided by the other work packages.	
Rationale	The suitability of methods and tools for an application domain can only be assessed by actually applying them to a real-world example of the domain and DREAMS has the declared goal of cross domain applicability.	
Significance	High	

Means for validation/verification	The requirement will be validated in the avionic demonstrator in Task T6.3.
Source	DoW (cross domain applicability)
Additional Information	It should however be tolerated that parts the DREAMS methods and tools are not applied, because in some cases the required information about the system cannot be obtained in a reasonable amount of time with respect to the limits of the project.

6.4 Certification

ID	R 6.4.1	
Topic	Avionics	
Subtopic	Certification	
Name	Certifiability	
Responsibility	WP	6
	Lead partner	TRT
	Participating partners	TRT
Description	The avionics demonstrator should be developed by taking into account avionics certifiability (DO-178C software,).	
Rationale	If the DREAMS architecture is designed with this requirement in mind, the applicability of the solution in the avionics domain is higher.	
Significance	Medium	
Means for validation/verification	The requirement will be validated in the avionic demonstrator in Task T6.2/T6.3.	
Source	DoW	
Additional Information		

6.5 Safety

ID	R 6.5.1	
Topic	Avionics	
Subtopic	Safety	

Name	Execution of real-time faults detection and recovery strategies	
Responsibility	WP	2
	Lead partner	TRT
	Participating partners	ONERA
Description	Faults run-time monitoring and recovery strategies as defined in requirement R 1.3.1 shall be implemented on the real target chosen by the avionic demonstrator. The implementation shall also be compliant with the executive layer.	
Rationale	The success of the DREAMS architecture depends on its applicability for systems in different application domains.	
Significance	High	
Means for validation/verification	The requirement will be validated in the avionic demonstrator in Task T6.2/T6.3.	
Source	DOW	
Additional Information		

7 Requirements for Wind-power Demonstrator

The Wind Power demonstrator is based on the supervision and control system of the off-shore wind turbines. The system is executed in the GALILEO platform, and requires several inputs and outputs that are connected through an EtherCAT fieldbus, including the protection system. This protection system is in charge of keeping the wind turbine within the design limits, for instance stopping the blades when their speed achieves a maximum value.

The demonstrator aims at including the protection system in GALILEO, and duplicates it in a Zynq™-7000 platform in order to increase the dependability of the system.

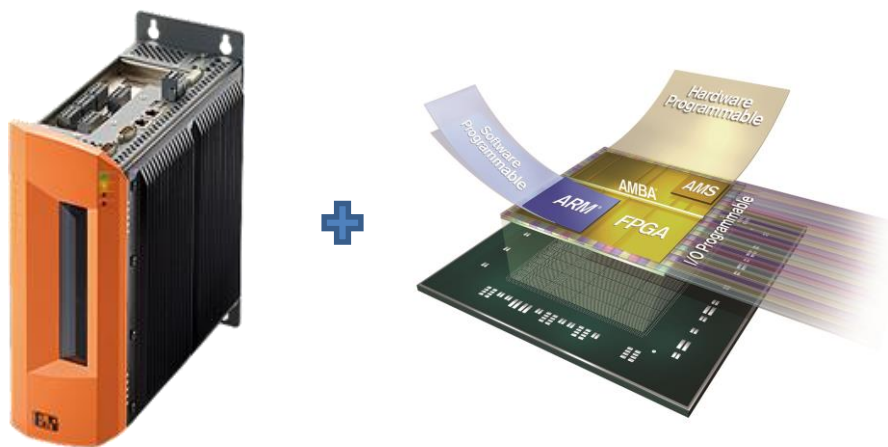


Figure 8: Wind Power Demonstrator Platform

To address the above mentioned issues, a detailed list of requirements is provided hereafter. Section 7.1 provides all requirements related to the assessment of the wind power demonstrator. Use case and certification related requirements are available in sections 7.2 and 7.3. Finally all timing and safety requirements are addressed in section 7.4.

7.1 Assessment

ID	R 7.1.1	
Topic	Wind Power	
Subtopic	Assessment	
Name	Demonstrator assessment	
Responsibility	WP	7
	Lead partner	FENTISS

	Participating partners	ALSTOM, RTAW
Description	The assessment shall be conducted analytically and by experimental evaluation. Key performance indicators shall be measured and compared against initially defined values.	
Rationale	This is required in order to assess the innovation carried out in the DREAMS project.	
Significance	High	
Means for validation/verification	The requirement will be validated in the wind power use case in Task T7.2/T7.3.	
Source	ALSTOM	
Additional Information		

ID	R 7.1.2	
Topic	Wind Power	
Subtopic	Assessment	
Name	Methods and tools must be applied for the wind power demonstrator	
Responsibility	WP	7
	Lead partner	RTAW
	Participating partners	RTAW, ALSTOM
Description	The wind power demonstrator shall apply as many as possible of the methods and tools provided by the other work packages.	
Rationale	The suitability of methods and tools for an application domain can only be assessed by actually applying them to a real-world example of the domain and DREAMS has the declared goal of cross domain applicability.	
Significance	High	
Means for validation/verification	The requirement will be validated in the wind power use case in Task T7.2/T7.3.	
Source	DoW (cross domain applicability)	
Additional Information	It should however be tolerated that parts the DREAMS methods and tools are not applied, because in some cases the required information about the system cannot be obtained in a reasonable amount of time with respect to the limit of the project.	

7.2 Use Case

ID	R 7.2.1	
Topic	Wind Power	
Subtopic	Use Case	
Name	Meet case-studies requirements regarding safety	
Responsibility	WP	7
	Lead partner	IKL
	Participating partners	TUV, ALSTOM, FENTISS
Description	The wind power demonstrator shall describe detailed requirements specification with regard to safety of the use case	
Rationale	Integration of requirements from different work packages and pre-evaluate of the impact with regard to safety on the wind power demonstrator. It will lead the development of the evaluation plan.	
Significance	High	
Means for validation/verification	The requirement will be validated in the wind power use case in Task T7.1. (The evaluation plan monitors that requirements defined are fulfilled in technical work)	
Source	DoW (7.1 Use case specification and evaluation methodology)	
Additional Information		

ID	R 7.2.2	
Topic	Wind Power	
Subtopic	Use Case	
Name	Hardware platform for wind power demonstrator	
Responsibility	WP	7
	Lead partner	ALSTOM
	Participating partners	IKL, FENTISS
Description	The major part of the wind power demonstrator shall be based on the hardware solution used by ALSTOM for the embedded real-time applications: GALILEO V4.	
Rationale	ALSTOM already designs and maintains the GALILEO platform. In order to increase the applicability of the DREAMS architecture in	

	<p>future ALSTOM applications, the architecture must be suitable to extend this platform. Additionally, if the demonstrator is based on the GALILEO platform, it could also be validated by using already available HIL tools.</p> <p>GALILEO platform will also be enhanced by the harmonized platform of the DREAMS project.</p>
Significance	High
Means for validation/verification	The requirement will be validated in the wind power use case in Task T7.2.
Source	ALSTOM
Additional Information	

ID	R 7.2.3	
Topic	Wind Power	
Subtopic	Use Case	
Name	Mixed criticality on GALILEO platform	
Responsibility	WP	7
	Lead partner	ALSTOM
	Participating partners	IKL, FENTISS
Description	The wind power demonstrator based on the GALILEO platform and the DREAMS architecture shall combine safety and non-safety functionalities by means of the hypervisor.	
Rationale	The multi-core GALILEO platform supports functionalities with no safety requirements such as the Human Machine Interface (non real-time) or the Supervisory System (real-time). These functionalities are to be combined with a new safety-critical application subsystem: the protection system in charge of maintaining the wind turbine in a safe state. This functionality will be included into the same platform by using the DREAMS architecture.	
Significance	High	
Means for validation/verification	The requirement will be validated in the wind power use case in Task T7.2.	
Source	ALSTOM	
Additional Information		

ID	R 7.2.4
----	---------

Topic	Wind Power	
Subtopic	Use Case	
Name	Field-bus protocol	
Responsibility	WP	7
	Lead partner	ALSTOM
	Participating partners	IKL
Description	The field-bus protocol used in the wind power demonstrator shall be EtherCAT.	
Rationale	In order to increase the applicability of the DREAMS architecture in future ALSTOM applications, the architecture must support the currently used field-bus solution. Additionally, if this field-bus is used, the demonstrator could also be validated by using already available HIL tools.	
Significance	High	
Means for validation/verification	The requirement will be validated in the wind power use case in Task T7.2.	
Source	ALSTOM	
Additional Information		

ID	R 7.2.5	
Topic	Wind Power	
Subtopic	Use Case	
Name	XtratuM hypervisor	
Responsibility	WP	7
	Lead partner	ALSTOM
	Participating partners	IKL, FENTISS, UPV
Description	The hypervisor used to provide TSP in the Wind Power demonstrator shall be XtratuM with support for Windows Embedded CE 6.0	
Rationale	In order to increase the applicability of the DREAMS architecture in future ALSTOM applications, the architecture must support the use of the current operating system solution. The development of the demonstrator could save some effort on the application software if it is reused from the real field software, which runs on	

	Windows Embedded CE 6.0. This way, the effort could be concentrated on the architectural approach.
Significance	High
Means for validation/verification	The requirement will be validated in the wind power use case in Task T7.2.
Source	ALSTOM
Additional Information	

7.3 Certification

ID	R 7.3.1	
Topic	Wind Power	
Subtopic	Certification	
Name	Protection system certifiability	
Responsibility	WP	7
	Lead partner	ALSTOM
	Participating partners	IKL, TUV
Description	The Wind Power demonstrator should be developed by taking into account the certifiability of the protection system according to ISO-13849, Performance Level PLd.	
Rationale	The protection system must be certified according to the machinery standard (ISO-13849). If the DREAMS architecture is designed with this requirement in mind, the applicability of the solution in the wind power domain would be higher.	
Significance	Medium	
Means for validation/verification	The requirement will be validated in the wind power use case in Task T7.2.	
Source	ALSTOM	
Additional Information	<p>According to ISO-13849 PL is defined as: “discrete level used to specify the ability of safety-related parts of control systems to perform a safety function under foreseeable conditions”.</p> <p>For the purposes of part ISO 13849-1, the ability of safety-related parts to perform a safety function is expressed through the determination of the performance level.</p> <p>Following table shows equivalences between PL and SIL levels.</p>	

		PL	SIL (IEC 61508-1, for information) high/continuous mode of operation
		a	No correspondence
		b	1
		c	1
		d	2
		e	3
		Also, See R 7.2.2	

ID	R 7.3.2	
Topic	Wind Power	
Subtopic	Certification	
Name	Safe state	
Responsibility	WP	7
	Lead partner	ALSTOM
	Participating partners	IKL
Description	The protection system shall be a fail-safe safety system.	
Rationale	The protection system must check operational values of the wind turbine and bring the system into the safe state (STOPPED) when these values exceed the design limits.	
Significance	High	
Means for validation/verification	The requirement will be validated in the wind power use case in Task T7.2.	
Source	ALSTOM	
Additional Information		

7.4 Timing and Safety

ID	R 7.4.1
Topic	Wind Power
Subtopic	Timing and Safety

Name	support for safe state	
Responsibility	WP	7
	Lead partner	ALSTOM
	Participating partners	ALSTOM, IKL, RTaW
Description	<p>The wind turbine shall achieve the stop state (safe state) when the speed of the blades is greater than or equal MAX_BLADE_SPEED.</p> <p>The wind turbine shall be in the safe state until a manual reset of the system.</p>	
Rationale	<p>IEC-61508-1 7.10.2.6: The E/E/PE system safety functions requirements specification shall contain:</p> <p>A description of all the safety functions necessary to achieve the required functional safety, which shall, for each safety function,</p> <ul style="list-style-type: none"> • provide comprehensive detailed requirements sufficient for the design and development of the E/E/PE safety-related systems, • include the manner in which the E/E/PE safety-related systems are intended to achieve or maintain a safe state for the EUC, • specify whether or not continuous control is required, and for what periods, in achieving or maintaining a safe state of the EUC, and • specify whether the safety function is applicable to E/E/PE safety-related systems operating in low demand, high demand or continuous modes of operation; 	
Significance	High	
Means for validation/verification	<p>In Task T7.3 the demonstrator system will be forced to a speed greater than MAX_BLADE_SPEED and the validator must check that the wind turbine is stopped.</p> <p>Based on the speed conditions, the validation engineer must check that the blades are still stopped.</p> <p>The validation engineer must perform a manual reset and the blades must leave the stopped condition and start moving.</p>	
Source	DoW (page 154, 3.2.2.2 Exploitation for Wind Power and Industrial Domain)	
Additional Information		

8 Requirements for Healthcare Demonstrator

The target platform for the Healthcare demonstrator is a distributed platform that includes 2 devices: the media home gateway (MHG) and the remote monitoring as shown in Figure 9.

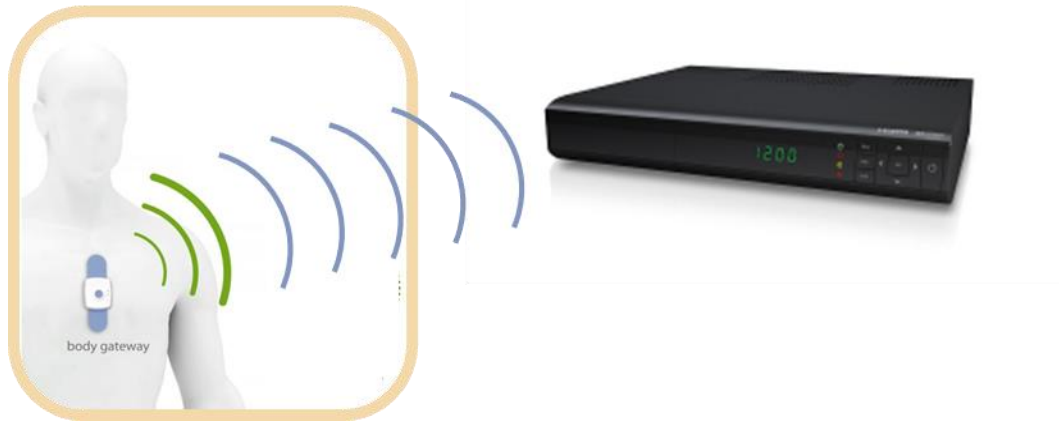


Figure 9: Healthcare Platform

The MHG is a typical shared memory multi-core system as shown in Figure 10. It includes a shared system NoC, some on demand accelerators, I/O components and one or two memory controllers that arbitrate memory read/write requests among cores and accelerators. In the MHG the processing time of a memory request is highly variable as it mainly depends on the sequence of memory accesses and the state of NoC, DRAM controllers and memory banks.

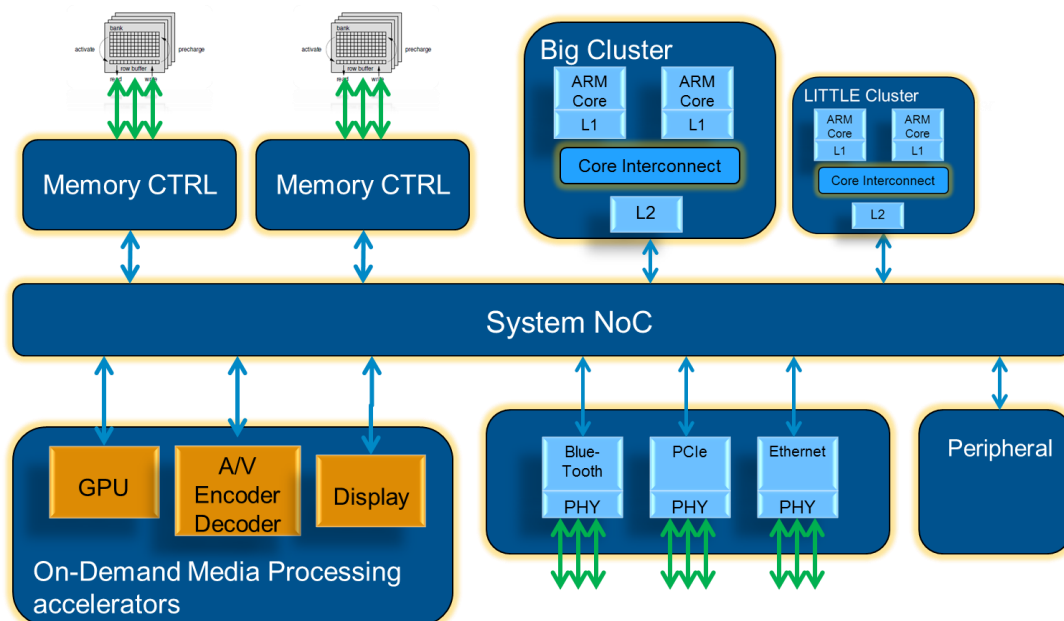


Figure 10: MHG System on Chip (SoC)

The arbitration scheme implemented in the NoC and in the memory controller usually tries to maximize the overall throughput and it is unaware of priorities of memory requests with different criticality levels.

Since throughput is the main target and the memory access can have high variance the system might experience severe timing degradation affecting the temporal predictability of real-time healthcare applications.

Last but not least, the virtualization in the MHG has to implement a strong isolation that is used to protect the medical information within a secure VM and to provide an efficient run-time support in a soft real-time system, where two types of VMs (real time and non-real-time) can coexist simultaneously. To address the above mentioned issues, a detailed list of requirements is provided hereafter.

Similar to the wind power demonstrator, assessment and certification requirements are available in sections 8.1 and 8.2. Real time and low power requirements are addressed in sections 8.4 8.5 while section 8.3 consolidates all requirements related to the full virtualization planned for the healthcare demonstrator. Finally all security requirements are provided in section 8.6

8.1 Assessment

ID	R 8.1.1	
Topic	Healthcare	
Subtopic	Assessment	
Name	Demonstrator assessment	
Responsibility	WP	8
	Lead partner	ST
	Participating partners	ST
Description	The assessment shall be conducted analytically and by experimental evaluation. Key performance indicators shall be measured and compared against initially defined values.	
Rationale	This is required in order to assess the innovation carried out in the DREAMS project.	
Significance	High	
Means for validation/verification	The requirement will be validated in the healthcare use case in Task T8.2.	
Source	DoW	
Additional Information		

ID	R 8.1.2	
Topic	Healthcare	
Subtopic	Assessment	

Name	Methods and tools must be applied for the healthcare demonstrator	
Responsibility	WP	8
	Lead partner	ST
	Participating partners	ST, RTaW
Description	The healthcare demonstrator shall apply as many as possible of the methods and tools provided by the other work packages.	
Rationale	The suitability of methods and tools for an application domain can only be assessed by actually applying them to a real-world example of the domain and DREAMS has the declared goal of cross domain applicability.	
Significance	High	
Means for validation/verification	The requirement will be validated in the healthcare use case in Task T8.3.	
Source	DoW (cross domain applicability)	
Additional Information	It should however be tolerated that parts the DREAMS methods and tools are not applied, because in some cases the required information about the system cannot be obtained in a reasonable amount of time with respect to the limit of the project.	

8.2 Certification

ID	R 8.2.1	
Topic	Healthcare	
Subtopic	Certification	
Name	Certifiability	
Responsibility	WP	8
	Lead partner	ST
	Participating partners	ST
Description	The healthcare demonstrator should be developed by taking into account healthcare certifiability	
Rationale	If the DREAMS architecture is designed with this requirement in mind, the applicability of the solution in the healthcare domain is higher.	
Significance	Medium	

Means for validation/verification	The requirement will be validated in the healthcare use case in Task T8.2.
Source	DoW
Additional Information	

8.3 Full virtualization

ID	R 8.3.1	
Topic	Healthcare	
Subtopic	Full virtualization	
Name	Interoperability of ARM-based platform	
Responsibility	WP	8
	Lead partner	ST
	Participating partners	TEI, VOSYS, FENTISS
Description	The architecture should support 64 and 32 bit interoperability.	
Rationale	Supporting different platform architectures is a viable market policy	
Significance	Medium	
Means for validation/verification	The requirement will be validated in the healthcare use case in Task T8.2. The FPGA development board will include a HW/SW integration plan.	
Source	DoW	
Additional Information		

8.4 Real time

ID	R 8.4.1	
Topic	Healthcare	
Subtopic	Real time	
Name	Soft real-time	
Responsibility	WP	8,2
	Lead partner	ST, RTaW
	Participating partners	TEI, VOSYS, FENTISS

Description	Soft real-time applications shall be combined with non real-time applications in a reliable manner. The focus is on reliable and predictable communication and synchronization between on- and off-chip domains.
Rationale	An embedded hypervisor is required that combines the benefits of the hardware-assisted virtualization with the requirements of embedded applications and real-time deterministic performance.
Significance	High
Means for validation/verification	The requirement will be assessed in the healthcare use case in Task T8.3.
Source	DoW
Additional Information	

ID	R 8.4.2	
Topic	Healthcare	
Subtopic	Real time	
Name	Robust real-time I/O management	
Responsibility	WP	8,2
	Lead partner	ST, RTaW
	Participating partners	TEI, VOSYS, FENTISS
Description	Different I/O flows shall be identified within the healthcare demonstrator and predictable bandwidth reservations shall be enforced for the real-time ones.	
Rationale	<p>Due to this requirement it is important that I/O device virtualization techniques are supported to enable proper scheduling of multiple I/O data transactions.</p> <p>Since the system is distributed and I/O devices are shared among different tasks it is important to identify the medical information that is coming from remote monitoring devices.</p>	
Significance	High	
Means for validation/verification	The requirement will be assessed in the healthcare use case in Task T8.3.	
Source	DoW	
Additional Information		

ID	R 8.4.3
----	---------

Topic	Healthcare	
Subtopic	Real time	
Name	Efficient memory performance isolation for real-time systems	
Responsibility	WP	2,8
	Lead partner	ST
	Participating partners	TEI, VOSYS
Description	Throughput-intensive workloads shall be integrated with critical real-time workloads using a shared memory subsystem.	
Rationale	<p>The integration poses a significant challenge due to the interference in accessing the external DDR. This affects the temporal predictability of memory-intensive real-time applications due to the high variance of their memory access time.</p> <p>A shared memory hierarchy that includes shared caches and external memory accesses is a big challenge in designing a predictable real-time systems for healthcare. Therefore, there is an increasing need for memory bandwidth management solutions that provide Quality of Service (QoS).</p>	
Significance	High	
Means for validation/verification	The requirement will be assessed in the healthcare use case in Task T8.3.	
Source	DoW	
Additional Information		

ID	R 8.4.4	
Topic	Healthcare	
Subtopic	Real time	
Name	STNoC for real-time systems	
Responsibility	WP	8,2
	Lead partner	ST, RTaW
	Participating partners	TEI, VOSYS, USIEGEN, RTAW
Description	Throughput-intensive workloads shall be integrated together with critical real-time workloads using a shared NoC.	

Rationale	Interference on traditional NoCs poses a significant challenge. Real-time applications can have a large WCET increase due to on-chip traffic interferences. Using shared interconnection technologies such as a NoC in real-time systems can pose a big challenge in designing predictable on-chip communication for healthcare. End-to-end delays can be difficultly bound and variance of worst-case execution time can be very large.
Significance	High
Means for validation/verification	The requirement will be assessed in the healthcare use case in Task T8.3.
Source	DoW
Additional Information	

8.5 Low power

ID	R 8.5.1	
Topic	Healthcare	
Subtopic	Low power	
Name	Low power (big.LITTLE) architecture	
Responsibility	WP	2,8
	Lead partner	ST
	Participating partners	ST, TEI
Description	A heterogeneous big.Little ARM v8 architecture should be supported in which critical tasks can be associated to specific cores.	
Rationale	Real-time and low power differentiation	
Significance	Medium	
Means for validation/verification	The requirement will be assessed by the FPGA platform prototyping in Task T2.1-2.4.	
Source	DoW	
Additional Information		

8.6 Security

ID	R 8.6.1	
Topic	Healthcare	
Subtopic	Security	
Name	Off-chip communications shall be confidential	
Responsibility	WP	3
	Lead partner	USIEGEN
	Participating partners	USIEGEN, TTT, ST, TEI, VOSYS
Description	The communications between the media home gateway (MHG) and the remote monitoring system shall be confidential.	
Rationale	In order to ensure the privacy of the patient, the communications between the media home gateway (MHG) and the remote monitoring system shall be kept confidential. This can be done by encrypting all the traffic leaving the MHG.	
Significance	High	
Means for validation/verification	Packet sniffing and inspection	
Source	WP3	
Additional Information	Privacy of medical information is of prime concern.	

ID	R 8.6.2	
Topic	Healthcare	
Subtopic	Security	
Name	Off-chip communications shall ensure data integrity	
Responsibility	WP	3
	Lead partner	USIEGEN
	Participating partners	TTT, ST, TEI, VOSYS
Description	The communications between the media home gateway (MHG) and the remote monitoring system shall ensure the integrity of the data being transmitted.	
Rationale	Data integrity shall be ensured in order to avoid errors or intentional modification to the data being transmitted between the MHG and the remote monitoring system.	

Significance	HIGH
Means for validation/verification	By construction in the underline physical medium
Source	WP3
Additional Information	Invalid or modified data should be dropped to avoid false measurements.

ID	R 8.6.3	
Topic	Healthcare	
Subtopic	Security	
Name	Data spoofing shall not be possible for off chip communications	
Responsibility	WP	3
	Lead partner	USIEGEN
	Participating partners	USIEGEN
Description	It should not be possible for an adversary to insert false measurements for the communications between the media home gateway (MHG) and the remote monitoring system.	
Rationale	In order to ensure the correct measurements are reported to the remote monitoring device, it shall be ensured that false measurements cannot be injected by the adversary using packet spoofing attacks.	
Significance	HIGH	
Means for validation/verification	Packet spoofing attacks	
Source	WP3	
Additional Information	False measurements about the health of a patient can be reported through packet spoofing by an adversary.	

ID	R 8.6.4	
Topic	Healthcare	
Subtopic	Security	
Name	Denial of Service attacks should be avoided as much as possible	
Responsibility	WP	3

	Lead partner	USIEGEN
	Participating partners	USIEGEN, TTT
Description	It might not be possible to avoid the Denial of Service (DoS) attacks totally, but it should be possible to reduce them in numbers or reduce their impact.	
Rationale	An adversary may bring down either end of the communicating parties, i.e., the MHG and the remote monitoring system via DoS attacks.	
Significance	HIGH	
Means for validation/verification	Active DoS attacks	
Source	WP3	
Additional Information	DoS attack might bring the patient monitoring to a halt.	

9 Requirements for Modeling and Development Process

In this section, the requirements on the DREAMS modeling approach (Sec. 9.1 - 9.9) and a corresponding model-driven development process (Sec. 9.10 - 9.14) will be presented. The following discussion includes the organization of these requirements into subtopics and their relation to other requirement (sub-) topics in this document.

The overall objective of DREAMS of developing a cross-domain architecture and design tools for networked mixed-criticality systems has a significant impact on the requirements on the development process. First of all, high-level requirements on the development process are described in the *Overall development approach* (Sec. 9.12). It is refined in the remaining requirements subtopics on the development process:

- The construction of systems with mixed-criticality requirements based on the DREAMS-architecture requires the application of development processes for safety-critical systems as defined in certification standards. The requirements on *Design, Development and validation of mixed-criticality systems* (Sec.9.13) demand the application of the V-model shape which is typically employed since it covers the entire development progress including requirements engineering, specification, implementation and integration. Validation, verification and certification activities for mixed-criticality systems are covered in a dedicated requirement topic (see Sec.5.5)Section5)
- The efficient implementation of the aforementioned systems depends on the adequate dimensioning of the execution platform (i.e., instances of the DREAMS architecture), the identification of appropriate deployments of application subsystems onto it, as well as the corresponding scheduling of shared resources. While the requirements on the algorithms and the provision of appropriate tools are provided in dedicated requirement topics (see Sec. 10 and Sec. 4, respectively), the integration of these methods into the development process is covered in the subtopic on *Resource Allocation and Design-Space Exploration* (Sec. 9.11).
- Requirements on *Variability Binding* (Sec. 9.14) generalize the aspect of appropriately dimensioning and configuring the system under design with methods considering entire product-line families.
- Finally, the requirements on the use of model-based tools in the development process in order to *manage the complexity* of developing DREAMS-based application systems (Sec. 9.10) represent the interface to the requirements on the DREAMS modeling approach.

As pointed out above, in DREAMS a model-driven development process will be defined. With the exception of (descriptive) analysis models used to estimate specific properties of DREAMS (sub-) systems, the majority of models are defined using meta-models (i.e., the rules and constructs according to which a model is created for a particular aspect or domain). In the following, the requirements on these meta-models will be discussed.

- The subtopic on the *Overall Organization of Meta-Models* (Sec. 9.1) contains high-level requirements on the design of the DREAMS meta-models. They are indented to ensure their applicability in the tool-supported development process and will serve as evaluation criteria in the assessment phase of the project.
- The first three subtopics of the DREAMS meta-model (Sec. 9.2, 9.3, and 7.4) follow the established categorization into meta-models for a platform-independent (application) model (PIM), a platform-model (PM) and a platform-specific model (PSM), as suggested in the Model-Driven Architecture (MDA).

PIM, PM and PSM provide means to describe the functional and architectural properties of application subsystems, and the architecture of platforms conforming to the DREAMS architectural style (cf. Sec. 0).Section 1). They are complemented with the meta-models summarized below that provide means to describe application (sub-)systems' temporal behavior, as well as a number of additional view-points.

- The *Timing Requirements Meta-Model* (Sec. 9.5) is used for models describing the intended temporal behavior of applications to be implemented on a DREAMS platform, as well as the latency properties of concrete deployed DREAMS systems. Together with the architectural model of the application and the platform instance, it serves as input to the offline scheduling and timing analysis methods and tools (see Sec. 4, and Sec. 10).
- Based on the *Reliability / Safety Meta-Model* (Sec. 9.6), models describing the corresponding requirements of applications as well as platform properties (e.g., safety integrity level) can be established.
- The *Energy / Power Analysis Model* (Sec. 9.7) is an analysis model describing the system-level average static and dynamic power consumption of the NoC. The *Energy / Power Analysis Requirements Meta-Model* can be used to describe the admissible energy/power consumption bounds of applications to be deployed to the DREAMS platform. In combination with timing requirements models and reliability requirements models (see above), it is used as goal specification for the design-space exploration (see Sec. 4).
- Using the *Security Meta-Model* (Sec. 9.8), security requirements and corresponding security mechanisms can be described.
- The subtopic *Variability Meta-Model* (Sec. 9.9) contains requirements on meta-models for the specification of explicit variability models. They are used as the basis for product-line-based design methods (e.g., architectural exploration...).

9.1 Overall Organization of Meta-Models

ID	R 9.1.1	
Topic	Modeling	
Subtopic	Overall Organization of Meta-Models	
Name	Separation of concerns	
Responsibility	WP	1
	Lead partner	FORTISS
	Participating partners	FORTISS, SINTEF, RTaW, ALSTOM, IKL, TRT, TUKL
Description	The meta-model shall be organized in such a way that different aspects are covered by sub-meta-models (e.g., architecture meta-models for application and platforms, requirements meta-models (e.g., real-time, energy...)).	
Rationale	A separation of different aspects into dedicate meta-models is the prerequisite for a scalable and maintainable overall meta-model.	

	It corresponds to the concept of multi-view modeling where different aspects of the system under design are covered by different views (represented by dedicated meta-models).
Significance	High
Means for validation/verification	The requirement will be validated by the providers of the different modules of the meta-model (FORTISS, RTaW, IKL, USIEGEN, SINTEF) in T1.8 by analyzing the models created for the demonstrator use cases. Feedback from the demonstrator lead partners (TRT (T6.3), ALSTOM (T7.3), STM (T8.3)) will be incorporated.
Source	DoW (T1.4, T1.6), ISO/IEC 42010
Additional Information	

ID	R 9.1.2	
Topic	Modeling	
Subtopic	Overall Organization of Meta-Models	
Name	Adequate degree of abstraction (Measure of success)	
Responsibility	WP	1
	Lead partner	FORTISS
	Participating partners	FORTISS, SINTEF, RTaW, ALSTOM, IKL, TRT, TUKL
Description	<p>On the one hand, the proposed meta-models shall abstract as many details as possible in order to hide details that are not relevant in the corresponding step of the development process. In particular, the meta-model shall be (pragmatically) minimized.</p> <p>On the other hand, the meta-model shall provide all the information that is required during the development process by capturing the relevant aspects of the application under design and the corresponding instance of the DREAMS architecture.</p>	
Rationale	<p>Abstract models that raise the level of abstraction are the prerequisite for frontloading activities into early phases of the development process (e.g., exploration of the design space).</p> <p>The sufficient level of abstraction and the pragmatic minimization ensure the clarity and therefore the acceptance of the meta-model by its users.</p> <p>On the other hand, models are required to contain sufficient information in order to allow for the automation of steps in the development process (i.e., tool support).</p>	

Significance	High
Means for validation/verification	<p>The requirement will be validated by the providers of the different modules of the meta-model (FORTISS, RTaW, IKL, USIEGEN, SINTEF) in T1.8 as follows</p> <ul style="list-style-type: none"> • Analysis of the models created for the demonstrator use cases. Feedback from the demonstrator lead partners (TRT (T6.3), ALSTOM (T7.3), STM (T8.3)) will be incorporated. • Analysis of use of meta-model by tool-chain. Feedback from the tool providers (IKL, RTAW, TTT, FORTISS, SINTEF, TUKL, ONERA, UPV; all T4.4) will be incorporated.
Source	DoW (T1.4, T1.6)
Additional Information	

ID	R 9.1.3	
Topic	Modeling	
Subtopic	Overall Organization of Meta-Models	
Name	Domain-independance (measure of success).	
Responsibility	WP	1
	Lead partner	FORTISS
	Participating partners	FORTISS, SINTEF, RTaW, ALSTOM, IKL, TRT, USIEGEN, TUKL
Description	The proposed meta-models shall be domain-independent. This means on the one hand that they shall not employ domain-specific constructs. On the other hand, this means that they shall be generic enough in order to be applicable for all DREAMS application domains.	
Rationale	The DREAMS project provides a cross-domain architecture. Hence, domain-independent meta-models are required that provide the basis for a cross-domain development process.	
Significance	High	
Means for validation/verification	The requirement will be validated by the providers of the different modules of the meta-model (FORTISS, RTaW, IKL, USIEGEN, SINTEF) in T1.8 by analyzing the models created for the demonstrator use cases. Feedback from the demonstrator lead partners (TRT (T6.3), ALSTOM (T7.3), STM (T8.3)) will be incorporated.	
Source	DoW (T1.4, T1.6)	

Additional Information	
------------------------	--

9.2 Platform-independent Application Meta-Model

ID	R 9.2.1	
Topic	Modeling	
Subtopic	Platform-independent Application Meta-Model	
Name	Application architecture	
Responsibility	WP	1
	Lead partner	FORTISS
	Participating partners	FORTISS, SINTEF, IKL
Description	The application meta-model shall capture the structure of applications in terms of their component architecture. In particular, this meta-model should be independent of the execution platform (i.e., it should not contain any platform-specific constructs), and therefore be applicable to different platforms.	
Rationale	<p>On the one hand, the platform-independence of the application model is a consequence of the goals to separate different concerns into dedicated meta-models (see R9.1.1) and to raise the level of abstraction (see R9.1.2).</p> <p>On the other hand, it also fosters the re-usability and exploitability of results from the DREAMS project.</p>	
Significance	High	
Means for validation/verification	<ul style="list-style-type: none"> • Module test: FORTISS will validate the suitability of the application architecture meta-model in T1.4 by creating representative prototype models inspired by the demonstrator use case specifications available at that time. • Integration test: Based on the models from the module test, FORTISS will analytically validate the integration of the application architecture meta-model into the overall meta-model (intermediate integration: T1.5 / final integration: T1.7) and its use by the tool-chain. Feedback from the tool providers (IKL, RTAW, TTT, FORTISS, SINTEF, TUKL, ONERA, UPV; all: intermediate integration T4.{1,2,3} / final integration T4.4) will be incorporated. • Acceptance test: The requirement will be validated by FORTISS (T1.8) by analyzing the models created for the demonstrator use cases. Feedback from the demonstrator 	

	lead partners (TRT (T 6.3), ALSTOM (T 7.3), STM (T 8.3)) will be incorporated. •
Source	DoW (T1.4)
Additional Information	

ID	R 9.2.2	
Topic	Modeling	
Subtopic	Platform-independent Application Meta-Model	
Name	Precise execution semantics	
Responsibility	WP	1
	Lead partner	FORTISS
	Participating partners	FORTISS, SINTEF, IKL, TRT
Description	For the application meta-model, precise (platform-independent) execution semantics should be defined.	
Rationale	On the one hand, executable platform-independent application models are the prerequisite to front-load activities into early stages of the development process, e.g. using functional simulation and formal verification. On the other hand, they are also required for the automatic generation of source code from models.	
Significance	Medium	
Means for validation/verification	See R 9.2.1.	
Source	DoW (T1.4)	
Additional Information	Note: The use of code generation is optional, i.e. applications can also be coded manually. The application meta-model may also be used to model the application architecture, and to apply the methods provided with the DREAMS development process that do not rely on a detailed specification of the on the functional behavior (e.g., mapping and scheduling).	

ID	R 9.2.3	
Topic	Modeling	
Subtopic	Platform-independent Application Meta-Model	
Name	Support for modeling memory requirements	

Responsibility	WP	1, 2, 4
	Lead partner	FORTISS
	Participating partners	FORTISS, ONERA, RTAW , FENTISS, TUKL
Description	The DREAMS architecture should provide means for defining the memory needs of the different application services and the platform services.	
Rationale	A specification of the memory needs of application and platform services is required to analyze platform and deployment choices (in combination with the platform model)	
Significance	Medium	
Means for validation/verification	<p>Module test: The providers of the relevant modules of the meta-model will validate the suitability of the selected approach in T1.4 by creating representative prototype models inspired by the demonstrator use case specifications available at that time.</p> <ul style="list-style-type: none"> • Integration test: Based on the models from the module test, the providers of the relevant modules of the meta-model will analytically validate the integration of the selected approach into the overall meta-model (intermediate integration: T1.5 / final integration: T1.7) and its use by the tool-chain. Feedback from the tool providers (IKL, RTAW, TTT, FORTISS, SINTEF, TUKL, ONERA, UPV; all: intermediate integration T4.{1,2,3} / final integration T4.4) will be incorporated. • Acceptance test: The requirement will be validated by the providers of the relevant modules of the meta-model in T1.8 by analyzing the models created for the demonstrator use cases. Feedback from the demonstrator lead partners (TRT (T 6.3), ALSTOM (T 7.3), STM (T 8.3)) will be incorporated. 	
Source	WG Avionics, Standards	
Additional Information	<p>Due to the nature of critical systems, memory is either statically allocated. If present at all, the extent of dynamic memory allocation is controlled and can be bounded.</p> <p>The timing aspect of memory requirements modeling support can be described using the meta-models provided to satisfy R9.5.*.</p>	

9.3 Platform Meta-Model

ID	R 9.3.1	
Topic	Modeling	
Subtopic	Platform Meta-Model	
Name	Platform architecture	
Responsibility	WP	1
	Lead partner	FORTISS
	Participating partners	USIEGEN, FORTISS, SINTEF, IKL
Description	The platform meta-model shall capture the topology and the hierarchic structure of instances of the DREAMS architecture.	
Rationale	<p>The topology of instances of the DREAMS architecture is the prerequisite for mapping applications to the platform and to derive the corresponding platform-specific models (see R9.4.1).</p> <p>The platform meta-model shall be hierarchic in order to capture the different levels of the DREAMS architecture (e.g., cluster level, chip level, etc.)</p>	
Significance	High	
Means for validation/verification	<ul style="list-style-type: none"> • Module test: FORTISS will validate the suitability of the platform architecture meta-model in T1.4 by creating representative prototype models inspired by the demonstrator use case specifications available at that time. • Integration test: Based on the models from the module test, FORTISS will analytically validate the integration of the platform architecture meta-model into the overall meta-model (intermediate integration: T1.5 / final integration: T1.7) and its use by the tool-chain. Feedback from the tool providers (IKL, RTAW, TTT, FORTISS, SINTEF, TUKL, ONERA, UPV; all intermediate integration T4.{1,2,3} / final integration T4.4) will be incorporated. • Acceptance test: The requirement will be validated by FORTISS (T1.8) by analyzing the models created for the demonstrator use cases. Feedback from the demonstrator lead partners (TRT (T6.3), ALSTOM (T7.3), STM (T8.3)) will be incorporated. • 	
Source	DoW (T1.4)	
Additional Information		

ID	R 9.3.2	
Topic	Modeling	
Subtopic	Platform Meta-Model	
Name	Platform services	
Responsibility	WP	1
	Lead partner	FORTISS
	Participating partners	USIEGEN, FORTISS, IKL, SINTEF
Description	The platform-meta model shall distinguish different types of building blocks contained in instances of the DREAMS architecture (e.g., ECUs, processors, on-/off chip network, memories, I/Os, OS or hypervisor partitions).	
Rationale	For the deployment of an application model to a model of the DREAMS platform, a rich component model of the platform is required.	
Significance	High	
Means for validation/verification	See R 9.3.1	
Source	DoW (T1.4)	
Additional Information		

9.4 Platform-specific Meta-Model

ID	R 9.4.1	
Topic	Modeling	
Subtopic	Platform-specific Meta-Model	
Name	Representation of Deployed Applications	
Responsibility	WP	1
	Lead partner	FORTISS
	Participating partners	FORTISS, TTT, ONERA, TUKL, IKL, UPV, SINTEF, RTaW
Description	<p>The platform-specific meta-model shall provide means to describe applications that are deployed to instances of the DREAMS architecture. Examples:</p> <ul style="list-style-type: none"> Mapping of application components to execution units of the platform (e.g. core, partition, ...), 	

	<ul style="list-style-type: none"> • WCETs • Mapping of application messages to platform channels • Message lengths • Task and message schedules
Rationale	The platform-specific meta-model defines the output of the mixed-criticality aware mapping and scheduling of applications to a DREAMS platform. A common meta-model ensures the compatibility of the results obtained by different methods and tools.
Significance	High
Means for validation/verification	<ul style="list-style-type: none"> • Module test: FORTISS will validate the suitability of the platform-specific meta-model by the end of T1.6 by creating representative prototype models inspired by the demonstrator use case specifications available at that time. • Integration test: Based on the models from the module test, FORTISS will analytically validate the integration of the specific-platform architecture meta-model into the overall meta-model (T1.7) and its use by the tool-chain. Feedback from the tool providers (IKL, RTAW, TTT, FORTISS, SINTEF, TUKL, ONERA, UPV; all in T4.4) will be incorporated • Acceptance test: The requirement will be validated by FORTISS (T1.8) by analyzing the models created for the demonstrator use cases. Feedback from the demonstrator lead partners (TRT (T 6.3), ALSTOM (T 7.3), STM (T 8.3)) will be incorporated.
Source	DoW (T1.6.1, T4.1)
Additional Information	How to obtain estimations on WCET is not a DREAMS topic. Existing techniques are supposed to be used.

9.5 Timing Requirements Meta-Model

ID	R 9.5.1	
Topic	Modeling	
Subtopic	Timing Requirements Meta-Model	
Name	Latency constraints	
Responsibility	WP	1
	Lead partner	RTaW

	Participating partners	RTaW, TUKL
Description	<p>The Timing Requirements Meta-Model shall allow the specification of latency constraints (local or end-to-end). These consist in an upper bound and an optional lower bound, in order to allow constraining the jitter of the response times.</p>	
Rationale	<p>Control algorithms are not able to effectively perform the control of a system if the delay between the acquisition of sensor data and the corresponding application of the command by the actuator is not bounded. Thus a mean must exist to describe these constraints so that their validation/verification can be assessed and traced.</p>	
Significance	High	
Means for validation/verification	<ul style="list-style-type: none"> • Module test: RTaW will validate the suitability of the proposed description of latency constraints in T1.4 by incorporating them to representative prototype models inspired by the demonstrator use case specifications available at that time. • Integration test: Based on the models from the module test, RTaW will analytically validate the integration of the selected description approach into the overall meta-model (intermediate integration: T1.5 / final integration: T1.7) and its use by the tool-chain. Feedback from the tool providers (intermediate integration T4.{1,2,3} / final integration T4.4) will be incorporated. • Acceptance test: The requirement will be validated by RTaW in T1.8 by analyzing the models created for the demonstrator use cases. Feedback from the demonstrator lead partners (TRT (T 6.3), ALSTOM (T 7.3), STM (T 8.3)) will be incorporated. 	
Source	DoW (T1.4), TIMMO-2-USE	
Additional Information	<p>Latency constraints will be used in the platform-independent model to describe timing needs of the applications.</p> <p>These are required inputs of the allocation and scheduling algorithms developed in T4.1.</p> <p>They can optionally be used in the platform-specific model in order to describe how the end-to-end latency budget is decomposed among off-chip and on-chip network delays and the processing delays.</p>	

ID	R 9.5.2
Topic	Modeling
Subtopic	Timing Requirements Meta-Model
Name	Repetition constraints

Responsibility	WP	1
	Lead partner	RTaW
	Participating partners	RTaW, TUKL
Description	<p>The Timing Requirements Meta-Model shall allow the specification of repetition constraints.</p> <p>An example is periodic repetition with optional jitter. This constraint can for example be applied to the execution of functions or tasks or the transmissions of messages.</p>	
Rationale	Control algorithms generally require minimal sampling rates in order to guarantee an effective control. A minimal sampling rate can be enforced by a periodic repetition constraint on the execution of the sensor driver.	
Significance	High	
Means for validation/verification	<ul style="list-style-type: none"> • Module test: RTaW will validate the suitability of the proposed description of repetition constraints in T1.4 by incorporating them to representative prototype models inspired by the demonstrator use case specifications available at that time. • Integration test: Based on the models from the module test, RTaW will analytically validate the integration of the selected description approach into the overall meta-model (intermediate integration: T1.5 / final integration: T1.7) and its use by the tool-chain. Feedback from the tool providers (intermediate integration T4.{1,2,3} / final integration T4.4) will be incorporated. • Acceptance test: The requirement will be validated by RTaW in T1.8 by analyzing the models created for the demonstrator use cases. Feedback from the demonstrator lead partners (TRT (T 6.3), ALSTOM (T 7.3), STM (T 8.3)) will be incorporated 	
Source	TIMMO-2-USE, T1.4	
Additional Information	<p>Repetition constraints will be used in the platform-independent model to describe timing needs of the applications.</p> <p>These are required inputs of the allocation and scheduling algorithms developed in T4.1.</p>	

ID	R 9.5.3	
Topic	Modeling	
Subtopic	Timing Requirements Meta-Model	
Name	Synchronization constraints	
Responsibility	WP	1

	Lead partner	RTaW
	Participating partners	RTaW, TUKL
Description	The Timing Requirements Meta-Model shall allow the specification of synchronization constraints (based on events).	
Rationale	Control algorithms may require data from several sensors. In order to guarantee an effective control in such a case, it is generally necessary that all sensor data has been sampled at the same moment in time, with a bounded temporal distance of the sampling times.	
Significance	High	
Means for validation/verification	<ul style="list-style-type: none"> • Module test: RTaW will validate the suitability of the proposed description of synchronization constraints in T1.4 by incorporating them to representative prototype models inspired by the demonstrator use case specifications available at that time. • Integration test: Based on the models from the module test, RTaW will analytically validate the integration of the selected description approach into the overall meta-model (intermediate integration: T1.5 / final integration: T1.7) and its use by the tool-chain. Feedback from the tool providers (intermediate integration T4.{1,2,3} / final integration T4.4) will be incorporated. • Acceptance test: The requirement will be validated by RTaW in T1.8 by analyzing the models created for the demonstrator use cases. Feedback from the demonstrator lead partners (TRT (T 6.3), ALSTOM (T 7.3), STM (T 8.3)) will be incorporated. 	
Source	TIMMO-2-USE, T1.4	
Additional Information	Synchronization constraints will be used in the platform-independent model to describe timing needs of the applications. These are required inputs of the allocation and scheduling algorithms developed in T4.1.	

ID	R 9.5.4	
Topic	Modeling	
Subtopic	Timing Requirements Meta-Model	
Name	Coverage of tools and demonstrators	
Responsibility	WP	1, 2, 3, 4
	Lead partner	RTaW

	Participating partners	RTaW, TUKL
Description	<p>The timing requirements meta-model should provide the information that is required to describe the desired timing behaviour of the DREAMS demonstrator applications.</p> <p>Additionally, the timing requirements meta-model should express exactly the timing constraints information handled by the timing analysis and scheduling tools developed in the course of the project. I.e., all information that is needed should be provided, and all information that is provided is actually used in at least one tool.</p>	
Rationale	<p>Depending on the required input of the aforementioned tools, and the needs of the demonstrators, additional timing constraints might be needed (e.g., precedence constraints, offset constraints...).</p>	
Significance	medium	
Means for validation/verification	<p>The requirement will be validated by RTaW in T.4.4 together with the tool provides by analyzing the capacities of the tools of the DREAMS tool chain and in T1.8 by analyzing the models created for the demonstrator use cases. Feedback from the demonstrator lead partners (TRT (T6.3), ALSTOM (T7.3), STM (T8.3)) will be incorporated.</p>	
Source	DoW	
Additional Information	<p>See also R9.1.2 (Adequate degree of abstraction), esp. the remark on (pragmatic) minimization of meta-models.</p>	

9.6 Reliability / Safety Meta-Model

ID	R 9.6.1	
Topic	Modeling	
Subtopic	Reliability / Safety Meta-Model	
Name	Policies according to IEC-61508	
Responsibility	WP	1,5
	Lead partner	IKL
	Participating partners	FORTISS, IKL, TUV, VOSYS
Description	<p>The Reliability / Safety Meta-Model should allow to specify policies according to IEC-61508 realization phase (system architecture definition).</p>	

Rationale	<p>When modeling a mixed-criticality system, it may be built from scratch or re-using safety compliant items, as for example software systems/subsystems, partitions, hypervisors and hardware platforms. Each safety compliant item can specify a failure probability by means, for example, of an assigned SIL level. So it is required that during modeling process safety policies (defined in the form of safety consistency rules) are checked, as for example: "SIL claimed cannot be higher than the maximum allowable SIL".</p> <p>These rules will enable to check the consistency of the specification and to eventually generate some artefacts (e.g., partitioning specification, scheduling, evidences, etc.)</p> <p>Models will ease system deployment taking into account non-functional properties through policies and consistency rules of compliant items (in the form of safety manuals) such as partitions, platforms, processors, etc.</p>
Significance	Medium
Means for validation/verification	IKL will validate these meta-models in T 1.4 and T 1.6 and support their use in T 1.8 in the Wind-Power demonstrator, thereby enabling their validation in the demonstrators. Feedback from the demonstrator lead partner ALSTOM will be incorporated.
Source	DoW (T1.6, T1.4)
Additional Information	R 9.6.1 is related to B5.1.3 MultiPARTES Methodology & B5.1.4 certification Strategy.

ID	R 9.6.2	
Topic	Modeling	
Subtopic	Reliability / Safety Meta-Model	
Name	Criticality levels	
Responsibility	WP	1,5
	Lead partner	IKL
	Participating partners	FORTISS, IKL, TUV, VOSYS
Description	The Reliability / Safety Meta-Model shall allow specifying criticality-levels according to IEC-61508.	
Rationale	Criticality level should be specified depending on their nature: Safety with SIL levels, reliability, etc.	
Significance	High	

Means for validation/verification	IKL will validate these meta-models in T 1.4 and T 1.6 and support their use in T 1.8 in the Wind-Power demonstrator, thereby enabling their validation in the demonstrators. Feedback from the demonstrator lead partner ALSTOM will be incorporated.
Source	DoW (T1.4)
Additional Information	R 9.6.2 is related to B5.1.3 MultiPARTES Methodology & B5.1.4 certification Strategy.

ID	R 9.6.3	
Topic	Modeling	
Subtopic	Reliability / Safety Meta-Model	
Name	Traceability	
Responsibility	WP	1,5
	Lead partner	FORTISS
	Participating partners	FORTISS, SINTEF, RTaW, IKL
Description	The meta-models should support the traceability between the artefacts used in the different steps of the development process.	
Rationale	Prerequisite for certifying designs built using the DREAMS approach.	
Significance	Medium	
Means for validation/verification	The requirement will be validated by the providers of the different modules of the meta-model (FORTISS, RTaW, IKL, USIEGEN, SINTEF) and TUV in T1.8 by analyzing which steps of the development process can be traced in the models created for the demonstrator use cases.	
Source	DoW (WP5)	
Additional Information		

9.7 Energy / Power Analysis Model and Meta-Model

ID	R 9.7.1	
Topic	Modeling	
Subtopic	Energy / Power Analysis Model and Meta-Model	
Name	System-level NoC static/dynamic power consumption analysis model	
Responsibility	WP	1, 2

	Lead partner	ST
	Participating partners	ST, TEI, FORTISS
Description	A high level analytical model covering average static and dynamic power consumption at the system-level shall be provided, e.g., a mathematical function.	
Rationale	A NoC static/dynamic power consumption meta-model is fundamental to enable system-level power characterization. It can be invoked a design-space exploration in order to evaluate the power consumption of the NoC for a given configuration.	
Significance	High	
Means for validation/verification	ST will validate the model against the RTL simulation models and synthesis.	
Source	DoW (T 1.4)	
Additional Information		

ID	R 9.7.2	
Topic	Modeling	
Subtopic	Energy / Power Analysis Model and Meta-Model	
Name	System-level energy / Power requirements meta-model	
Responsibility	WP	1, 4
	Lead partner	FORTISS
	Participating partners	FORTISS , ST,TEI
Description	The Energy / Power requirements meta-model should be suitable to define requirements on the energy / power consumption of a DREAMS system at the system-level.	
Rationale	Models defined using the energy / power requirements meta-model can be used to define goals for the design-space exploration. Examples include bounds on the energy consumption for a given application, or bounds on the peak power consumption of system components.	
Significance	Medium	
Means for validation/verification	<ul style="list-style-type: none"> Module test: FORTISS will validate the suitability of the system-level energy / power requirements meta-model in T1.4 by creating representative prototype models inspired by the demonstrator use case specifications available at that time. 	

	<ul style="list-style-type: none"> • Integration test: Based on the models from the module test, FORTISS will analytically validate the integration of the application architecture meta-model into the overall meta-model (intermediate integration: T1.5 / final integration: T1.7) and its use by the DSE tools (T4.4). • Acceptance test: The requirement will be validated by FORTISS (T1.8) by analyzing the models created for the demonstrator use cases. Feedback from the demonstrator lead partners (TRT (T 6.3), ALSTOM (T 7.3), STM (T 8.3)) will be incorporated.
Source	DoW (T1.4)
Additional Information	

9.8 Security Meta-Model

ID	R 9.8.1	
Topic	Modeling	
Subtopic	Security Meta-Model	
Name	Security Meta-Model for Data Confidentiality	
Responsibility	WP	1
	Lead partner	USIEGEN
	Participating partners	FORTISS, USIEGEN, TTT, ST, TEI, VOSYS
Description	<p>The Security Meta-Model for Data Confidentiality shall allow modelling the varying needs of confidentiality. This includes the confidentiality of the data in memory as well as the data transferred over the network. In case of confidentiality of data exchanged over the network, the meta model should allow the choice of end-to-end security or link level security.</p> <p>Different applications have different needs regarding confidentiality, e.g., real-time applications with sporadic data bursts might need stream ciphers for efficiency, whereas non-real-time applications might need block ciphers. The cryptographic strength is also a choice that depends on the application type and shall be definable in the model.</p>	
Rationale	<p>For some applications, privacy is of utmost importance, e.g., in patient health monitoring systems. One way to ensure privacy is through data confidentiality.</p>	

Significance	Medium
Means for validation/verification	<ul style="list-style-type: none"> • Module test: USIEGEN will validate the suitability of the Security Meta-Model for Data Confidentiality in T1.4 (Development of Methods for Application, Platform and Variability Modelling) by creating representative prototype models inspired by the demonstrator use case specifications available at that time. • Integration test: Based on the models from the module test, USIEGEN will validate the integration of the Security Meta-Model for Data Confidentiality into the overall meta-model in T1.5 (intermediate integration) and T1.7 (final integration). • Acceptance test: The requirement will be validated by the demonstrator partners with the help of USIEGEN in T1.8 by analysing the models created for the demonstrator use cases.
Source	WP1 / T1.4 and T1.6
Additional Information	

ID	R 9.8.2	
Topic	Modeling	
Subtopic	Security Meta-Model	
Name	Security Meta-Model for Data Integrity	
Responsibility	WP	1
	Lead partner	USIEGEN
	Participating partners	FORTISS, USIEGEN, TTT, ST, TEI, VOSYS
Description	<p>The Security Meta-Model for Data Integrity shall allow modeling the varying needs of data integrity. Data integrity in the context of security is provided through message authentication codes (MAC). MAC ensures that tampering of data/message is detectable (by the receiver).</p> <p>MAC length is a parameter that will define the strength of data integrity verification. Choosing larger MAC length might enhance the level of trust on integrity verification but increase the overhead. Choosing small MAC length will decrease the level of trust by increasing the collisions and thus chances for forgery attacks but might decrease the overhead. The tradeoff will depend on the type of application and the criticality of messages exchanged.</p>	
Rationale	Data in transit (or in storage) can be modified by an adversary. If data tampering cannot be avoided, it should at least be possible to	

	detect it. Such tampering is detectable using message authentication codes.
Significance	Medium
Means for validation/verification	<ul style="list-style-type: none"> • Module test: USIEGEN will validate the suitability of the Security Meta-Model for Data Integrity in T1.4 by creating representative prototype models inspired by the demonstrator use case specifications available at that time. • Integration test: Based on the models from the module test, USIEGEN will validate the integration of the Security Meta-Model for Data Confidentiality into the overall meta-model in T1.5 (intermediate integration) and T1.7 (final integration). • Acceptance test: The requirement will be validated by the demonstrator partners with the help of USIEGEN in T1.8 by analysing the models created for the demonstrator use cases.
Source	WP1 / T1.4 and T1.6
Additional Information	

ID	R 9.8.3	
Topic	Modeling	
Subtopic	Security Meta-Model	
Name	Security Meta-Model for Authentication	
Responsibility	WP	1
	Lead partner	USIEGEN
	Participating partners	FORTISS, USIEGEN, TTT, ST, TEI, VOSYS
Description	The Security Meta-Model for Authentication shall allow modeling the needs for establishing the authenticity of communication partner and the authentication of data origin. An entity involved in communication may have the need to ensure that the other end of the communication is trustworthy for communications involving exchange of private data and that the actual origin of the data is the same as the claimed origin.	
Rationale	For trustworthy communications, the communicating entities shall be able to authenticate each other and the origin of the data.	
Significance	Medium	
Means for validation/verification	<ul style="list-style-type: none"> • Module test: USIEGEN will validate the suitability of the Security Meta-Model for Authentication in T1.4 by creating representative prototype models inspired by the demonstrator use case specifications available at that time. 	

	<ul style="list-style-type: none"> • Integration test: Based on the models from the module test, USIEGEN will validate the integration of the Security Meta-Model for Authentication into the overall meta-model in T1.5 (intermediate integration) and T1.7 (final integration). • Acceptance test: The requirement will be validated by the demonstrator partners with the help of USIEGEN in T1.8 by analysing the models created for the demonstrator use cases.
Source	WP1 / T1.4 and T1.6
Additional Information	

9.9 Variability Meta-Model

ID	R 9.9.1	
Topic	Modeling	
Subtopic	Variability Meta-Model	
Name	Separate variability description	
Responsibility	WP	1, 4, 5
	Lead partner	SINTEF
	Participating partners	SINTEF; IKL, FORTISS
Description	The variability meta-model shall allow specifying variations of base models in order to define product lines.	
Rationale	<p>DREAMS systems need to be automatically adaptive, and this requirement will help automating the production of a configuration.</p> <p>The description of variability should be kept separate from the base models, but still referring to the base models. The separate approach makes it possible to associate several different variability models with one (set of) base model(s). The visualization of the variability may still very well be superimposed on the base models.</p>	
Significance	High	
Means for validation/verification	SINTEF will validate this requirement by applying the variability meta-model to examples developed in T1.4, T4.1, T4.3 and T5.5. Furthermore, variability meta-model will be applied to examples from appropriate pilot cases.	

	D1.4.1 will define the first version of the variation meta-model and by the feedback from validation activities improved and defined in D1.7.1.
Source	DoW (T1.4)
Additional Information	Even though we are making requirements on variability meta-model does not mean that DREAMS need to create such variability meta-model totally from scratch.

ID	R 9.9.2	
Topic	Modeling	
Subtopic	Variability Meta-Model	
Name	Layered and modularized variability description	
Responsibility	WP	1, 4, 5
	Lead partner	SINTEF
	Participating partners	SINTEF; IKL, FORTISS
Description	The variability meta-model shall allow to describe different feature sets of applications.	
Rationale	<p>DREAMS systems need to be automatically adaptive, and this requirement will help automating the production of a configuration in particular in making a pragmatic adaptation to enhancing the variability possible.</p> <p>Since similarities will appear in clusters, it is important that the variability description allows describing modules of variability that may represent sub-product-lines or just clusters of variation. This structuring may be applied when having to find adequate sub-optimal resolutions (see other requirement in this cluster).</p>	
Significance	High	
Means for validation/verification	<p>SINTEF will validate this requirement by applying the variability meta-model to examples developed in T1.4, T4.1, T4.3 and T5.5. Furthermore variability meta-model will be applied to examples from appropriate pilot cases.</p> <p>D1.4.1 will define the first version of the variation meta-model and by the feedback from validation activities improved and defined in D1.7.1.</p>	
Source	DoW (T1.4)	
Additional Information	Even though we are making requirements on variability meta-model does not mean that DREAMS need to create such variability meta-model totally from scratch.	

ID	R 9.9.3	
Topic	Modeling	
Subtopic	Variability Meta-Model	
Name	Flexible variability resolution	
Responsibility	WP	1, 4, 5
	Lead partner	SINTEF
	Participating partners	SINTEF, IKL, FORTISS
Description	The variability meta-model shall allow to describe different implementation alternatives of applications.	
Rationale	<p>DREAMS systems need to be automatically adaptive, and this requirement will help automating the production of a configuration in particular in adapting to different platform technologies (e.g., hardware).</p> <p>The concrete implementation of the resolution should be left to a separate variability realization description which is flexible itself to different environments and technologies.</p>	
Significance	High	
Means for validation/verification	<p>SINTEF will validate this requirement by applying the variability meta-model to examples developed in T1.4, T4.1, T4.3 and T5.5. Furthermore variability meta-model will be applied to examples from appropriate pilot cases.</p> <p>D1.6.1 will define the first version of the platform specific variation meta-model and by the feedback from validation activities improved and defined in D1.7.1.</p>	
Source	DoW (T1.4, T1.6)	
Additional Information	Even though we are making requirements on variability meta-model does not mean that DREAMS need to create such variability meta-model totally from scratch.	

ID	R 9.9.4	
Topic	Modeling	
Subtopic	Variability Meta-Model	
Name	Flexible variability implementation platform	
Responsibility	WP	1, 4, 5
	Lead partner	SINTEF

	Participating partners	SINTEF; IKL, FORTISS
Description		The variability meta-model shall allow to describe instances of the DREAMS architecture (w.r.t. a given base instance, e.g., core with or without OS).
Rationale		DREAMS systems need to be automatically adaptive, and this requirement will help automating the production of a configuration in particular in adapting to different platform technologies (e.g. hardware). The concrete implementation of the resolution should be left to a separate variability realization description which is flexible itself to different platforms
Significance		High
Means for validation/verification		SINTEF will validate this requirement by applying the variability meta-model to examples developed in T1.4, T4.1, T4.3 and T5.5. Furthermore variability meta-model will be applied to examples from appropriate pilot cases. D1.6.1 will define the first version of the platform specific variation meta-model and by the feedback from validation activities improved and defined in D1.7.1.
Source		DoW (T1.4, T1.6)
Additional Information		This requirement is strongly associated with R9.9.3. Even though we are making requirements on variability meta-model does not mean that DREAMS need to create such variability meta-model totally from scratch.

9.10 Complexity Management

ID	R 9.10.1	
Topic	Development Process	
Subtopic	Complexity Management	
Name	Definition of Model-to-model transformations	
Responsibility	WP	1
	Lead partner	FORTISS
	Participating partners	FORTISS, RTAW, IKL, UPV, SINTEF, USIEGEN

Description	The development process should define the model-to-model transformations required to implement application subsystems on top of the DREAMS platform.
Rationale	The development of a DREAMS-based systems involves several related models (e.g., PIM, PM, PSM). Model-to-model transformations define the steps needed to derive a given model from its source models. The definition of model-to-model transformations is therefore a prerequisite to provide tool support for a development process.
Significance	Medium
Means for validation/verification	The requirement will be validated by the tool providers whose tools implement the relevant transformations / produce the relevant artefacts (IKL, RTAW, TTT, FORTISS, SINTEF, TUKL, ONERA, UPV) and TUV by analyzing the models created for the demonstrator use cases. Feedback from the demonstrator lead partners (TRT (T6.3), ALSTOM (T7.3), STM (T8.3)) will be incorporated.
Source	DoW (T 1.3)
Additional Information	Not all model-to-model transformations need necessarily be automated.

ID	R 9.10.2	
Topic	Development Process	
Subtopic	Complexity Management	
Name	Definition of implementation artefacts	
Responsibility	WP	1
	Lead partner	FORTISS
	Participating partners	FORTISS, RTAW, IKL, UPV, SINTEF, USIEGEN
Description	The development methodology should allow the definition of the implementation artifacts, i.e. its end products that have to be produced for a DREAMS-based system.	
Rationale	The set of end-products defines the steps required to produce the required implementation artifacts.	
Significance	Medium	
Means for validation/verification	See R 9.10.1	

Source	DoW (T 1.3)
Additional Information	Generation of implementation artifacts can be both manual and automatic.

9.11 Resource Allocation and Design-Space Exploration

ID	R 9.11.1	
Topic	Development Process	
Subtopic	Resource Allocation and Design-Space Exploration	
Name	Offline mapping and real-time scheduling methods for MC systems	
Responsibility	WP	2, 3, 4
	Lead partner	TUKL
	Participating partners	RTaW, TUKL, FORTISS, IKL, UPV, ONERA, TTT
Description	<p>The development process shall support offline real-time scheduling methods for mixed-criticality systems that provide the following services</p> <ul style="list-style-type: none"> • Allocation of functional parts to partitions • Time triggered schedules • Static virtual circuit arbitration and port scheduling 	
Rationale	<p>Static allocation and scheduling is a prerequisite for the efficient certification of mixed-criticality systems.</p> <p>Static virtual circuit arbitration and port scheduling schemes at the NoC and network interface layer will help extend current on-chip interconnects towards supporting low-level hierarchical real-time communication schemes</p>	
Significance	High	
Means for validation/verification	<p>TUKL and IKL will participate in the validation of the offline scheduling and allocation methods at the module and integration level within WP4, specifically in task 4.1.</p> <p>The offline scheduling algorithms shall be implemented in the Avionics Demonstrator, so validation based on the demonstrator will also take place.</p>	
Source	DoW (T2.2, T4.1)	
Additional Information	In order to provide the adaptivity required to foster the efficiency of in mixed-criticality systems, the design-time mapping and	

	scheduling methods need to consider the guarantees provided by dynamic resource management. This is covered by R10.2.1.
--	---

ID	R 9.11.2	
Topic	Development Process	
Subtopic	Resource Allocation and Design-Space Exploration	
Name	Consideration of online adaptation and resource management strategies	
Responsibility	WP	2, 3
	Lead partner	TUKL
	Participating partners	TUKL,ST, TEI, TRT, UPV, ONERA, TTT, VOSYS, ALSTOM
Description	The development methodology shall support online resource allocation and management strategies, and their relationship to the static methods described in R10.2.1. The online resource managers (GRM, LRM) allow for online changes of the allocation plans including allocation and schedules, in order to permit adaptation to foreseen (and possible unforeseen) changes in the availability of the resources (which must complete within in a predictable time span).	
Rationale	Online resource allocation is necessary for adaptation, a pre-requisite for the efficient use of resources. Its effects (e.g., guarantees, overhead) and its relationship to static resource management methods (see R10.2.1) need to be considered in the development process.	
Significance	High	
Means for validation/verification	The global resource management services shall be implemented in the Avionics Demonstrator, and TUKL shall participate in the experimental validation of this requirement, together with the demonstrator lead partner (TRT).	
Source	DoW (T2.2, T 3.2, 4.1)	
Additional Information		

ID	R 9.11.3	
Topic	Development Process	
Subtopic	Resource Allocation and Design-Space Exploration	
Name	Design-space exploration	
Responsibility	WP	4

	Lead partner	FORTISS
	Participating partners	FORTISS, SINTEF
Description	<p>The development process should support the design space exploration (DSE) method that provides means for semi-automatic architectural exploration. Based on an initial system model and design objective specifications (timing constraint specifications, energy requirements models (R9.7.2), the DSE can be used by a designer to explore the set of variants of the initial design that are pareto-optimal w.r.t. the design objectives.</p> <p>Architectural exploration supports the designer in obtaining an optimized system configuration which can be derived from the original system specification (e.g., using a model-to-model transformation).</p> <p>The DSE uses external analyses and models in order rate the fitness of a particular solution</p> <ul style="list-style-type: none"> • Reliability analysis (R9.13.6) • Offline mapping and real-time scheduling (R9.11.1) • NoC energy analysis model (R9.7.1) 	
Rationale	The explorations of the design space and configuration optimization are required to optimize DREAMS-based systems.	
Significance	Medium	
Means for validation/verification	The requirement will be validated by FORTISS and SINTEF in T4.4 by comparing the candidate solutions for the demonstrator use cases obtained using the DSE methods with the selected demonstrator implementations. Feedback from the demonstrator lead partners will be incorporated (TRT (T6.3), ALSTOM (T7.3), STM (T8.3)).	
Source	DoW (T4.1)	
Additional Information	See also: R5.1.6	

ID	R 9.11.4	
Topic	Development Process	
Subtopic	Resource Allocation and Design-Space Exploration	
Name	Timing analyses	
Responsibility	WP	4
	Lead partner	RTaW

	Participating partners	RTaW, TUKL, FORTISS, ST, TEI, TRT, UPV, ONERA, TTT
Description		The development process should support timing analyses, such as the analysis of end-to-end response times induced by a given allocation and scheduling configuration.
Rationale		The outcomes of the response time analysis algorithm are needed to verify the end-to-end latency constraints.
Significance		Medium
Means for validation/verification		The requirement will be validated by RTaW in T1.4, T4.1, T 5.4, T4.4, by analyzing the proposed development process and tool support.
Source		DoW (T4.1)
Additional Information		

9.12 Overall Development Approach

ID	R 9.12.1	
Topic	Development Process	
Subtopic	Overall Development Approach	
Name	Meet-in-the-middle development methodology	
Responsibility	WP	1, 5
	Lead partner	IKL
	Participating partners	FORTISS, SINTEF, RTaW, IKL, USIEGEN
Description		The development process should support "top-down" and "bottom-up" development of DREAMS-based applications ("meet-in-the-middle methodology"), if possible aligned with existing practices and workflows
Rationale		<p>A top-down approach, i.e. continuously refining requirements into an implementation is a pre-requisite for certification.</p> <p>A bottom-up approach, i.e. the construction of a system based the reuse components reduces effort and cost. Product-lines are a methodology to systematize this for well-defined platforms. However, the re-use of existing components requires modular certification approaches that allow the re-use of certification arguments.</p>
Significance		High

Means for validation/verification	IKL will support the use of the development approach in the Wind-Power demonstrator in T 1.8, thereby enabling its validation in the demonstrators. Feedback from the demonstrator lead partner ALSTOM will be incorporated.
Source	DoW (T5.1, T5.3, T5.5)
Additional Information	R 9.12.1 is related to B5.1.3 MultiPARTES Methodology & B5.1.4 certification Strategy.

ID	R 9.12.2	
Topic	Development Process	
Subtopic	Overall Development Approach	
Name	Consideration of Certification, Validation and Verification	
Responsibility	WP	1,5
	Lead partner	IKL
	Participating partners	USIEGEN, IKL, TUV
Description	The development process shall harmonize with the methods provided by WP5 Certification, Validation and Verification.	
Rationale	Development processes (e.g., model-driven development processes) must be thought to be used in functional safety projects and may have an impact on TUV's inspection of the Functional Safety Management.	
Significance	High	
Means for validation/verification	TUV will validate the requirement in T 1.3 (D1.3.1 Description of development process with model transformations) by inspecting and assessing the development approach.	
Source	DoW (T1.3 Definition of Development Process)	
Additional Information	R 9.12.2 is related to B5.1.3 MultiPARTES Methodology & B5.1.4 certification Strategy.	

ID	R 9.12.3	
Topic	Development Process	
Subtopic	Overall Development Approach	
Name	Time needs considered from design	
Responsibility	WP	WP1, WP4
	Lead partner	RTAW

	Participating partners	ONERA, RTAW, TTT, FENTISS, TUKL, USIEGEN
Description	The development (design, verification, validation) process shall foresee the definition of application timing requirements.	
Rationale	Need to validate system properties at the early stages of development.	
Significance	High	
Means for validation/verification	<ul style="list-style-type: none"> • Module test: RTaW will validate the suitability of the proposed description of application timing requirements in T1.4 by incorporating them to representative prototype models inspired by the demonstrator use case specifications available at that time. • Integration test: Based on the models from the module test, RTaW will analytically validate the integration of the selected approach into the overall meta-model (intermediate integration: T1.5 / final integration: T1.7) and its use by the tool-chain. Feedback from the tool providers (intermediate integration T4.{1,2,3} / final integration T4.4) will be incorporated. • Acceptance test: The requirement will be validated by RTaW in T1.8 by analyzing the models created for the demonstrator use cases. Feedback from the demonstrator lead partners (TRT (T 6.3), ALSTOM (T 7.3), STM (T 8.3)) will be incorporated. 	
Source	WG Avionics, TRT	
Additional Information		

9.13 Design, Development and Validation of MC Systems

ID	R 9.13.1	
Topic	Development Process	
Subtopic	Design, Development and Validation of MC Systems	
Name	Development of safety related parts	
Responsibility	WP	1, 5
	Lead partner	IKL
	Participating partners	IKL, TUV
Description	Safety related parts shall be developed according to IEC 61508.	

Rationale	In order to allow reuse of safety related parts the development of such a component needs to be done according to IEC 61508.
Significance	
Means for validation/verification	Validation partner: TUV Validation activity: inspection, assessment Task of Validation: T 1.3 (D1.3.1 Description of development process with model transformations)
Source	TUV
Additional Information	Within the DREAMS project it is not intended to develop a product that will be certified. On the other hand certification needs to be kept in mind in order to allow a reuse of the results of the dreams project in industrial projects. R 9.13.1 is related to B5.1.3 MultiPARTES Methodology & B5.1.5 certification Strategy based on COTS.

ID	R 9.13.2	
Topic	Development Process	
Subtopic	Design, Development and Validation of MC Systems	
Name	V-shape development process	
Responsibility	WP	1,5
	Lead partner	IKL
	Participating partners	FORTISS, IKL, TUV, SINTEF, FENTISS, UPV

Description	<p>The development process shall follow a V-shape development lifecycle model, e.g., as the one depicted in the picture below:</p>
Rationale	Taking into account that the DREAMS development process is not intended to develop hardware, but to build software systems to be deployed to instances of the DREAMS architecture, the design lifecycle for safety projects has the V-shape lifecycle according to IEC-61508-3 for software.
Significance	High
Means for validation/verification	<p>Validation partner: TUV</p> <p>Validation activity: inspection, assessment</p> <p>Task of Validation: T 1.3 (D1.3.1 Description of development process with model transformations)</p>
Source	DoW (T1.3 Definition of Development Process)
Additional Information	R 9.13.2 does not need any BB or GAP since it is a following the standard (IEC 61508) requirement.

ID	R 9.13.3	
Topic	Development Process	
Subtopic	Design, Development and Validation of MC Systems	
Name	Requirement traceability support	
Responsibility	WP	1,5
	Lead partner	IKL
	Participating partners	IKL, TUV, SINTEF

Description	The development process shall support the traceability for requirements regarding: safety, security, etc. Traceability will help in the avoidance of systematic faults in both HW (IEC 61508-2 Table B.6) and SW (IEC 61508-3 Tables A.1 to A.10) development.
Rationale	As stated in IEC 61508-7 C.2.11: In order to ensure that the software that results from lifecycle activities meets the requirements for correct operation of the safety related system, it is essential to ensure consistency between the lifecycle stages, traceability between activities.
Significance	High
Means for validation/verification	Validation partner: TUV Validation activity: inspection, assessment Task of Validation: T 1.3
Source	DoW (1.1.2.4 Objective 4: Development Methodology and Tools based on Model-Driven Engineering)
Additional Information	R 9.13.3 does not need any BB or GAP since it is a following the standard (IEC 61508) requirement.

ID	R 9.13.4	
Topic	Development Process	
Subtopic	Design, Development and Validation of MC Systems	
Name	Consideration of time and space partitioning mechanisms	
Responsibility	WP	1,5
	Lead partner	IKL
	Participating partners	USIEGEN, FORTISS, IKL, TUV, SINTEF, TTT, FENTISS, UPV
Description	The development process should take into account time and space partitioning mechanisms.	
Rationale	Partitioning services for resources (e.g., networks, processing cores, inputs/ outputs, memory) must ensure timeliness, data and energy integrity based on a priori knowledge of the permitted component behavior.	
Significance	Medium	
Means for validation/verification	Validation partner: TUV Validation activity: inspection, assessment Task of Validation: T 1.3	

Source	DoW (1.2.2 Virtualization Technologies, Intelligent on-chip and off-chip communication systems with TSP and T 1.2 Definition of Cross-Domain Architectural Style for Mixed-Criticality Systems)
Additional Information	R 9.13.4 is related to B 5.1.2 and B 3.3.1 in its relation to the temporal and spatial partitioning.

ID	R 9.13.5	
Topic	Development Process	
Subtopic	Design, Development and Validation of MC Systems	
Name	Integration of an additional application subsystems into an existing system	
Responsibility	WP	1,5
	Lead partner	IKL
	Participating partners	TUV, SINTEF, TTT, FENTISS, UPV
Description	The development process should support the integration of an additional application subsystem into an existing system.	
Rationale	The development process is a key part while certifying a safety critical system and is also applicable to mixed criticality system.	
Significance	Medium	
Means for validation/verification	Inspection of D1.3.1 Description of the development process with model transformations	
Source	DoW (T 1.3 Definition of the Development Process) T1.3 definition of the Development Process / 610640 DREAMS - Work plan table - Page 14 of 79	
Additional Information	R 9.13.5 is related to B 5.1.1, B 5.1.3.	

ID	R 9.13.6	
Topic	Development Process	
Subtopic	Design, Development and Validation of MC Systems	
Name	Reliability analysis / methods for active redundancy	
Responsibility	WP	4, 5
	Lead partner	FORTISS
	Participating partners	FORTISS, USIEGEN, TUV, IKL
Description	The development process should support a reliability analysis, and methods for the introduction of active redundancy that derive a	

	<p>fault-tolerant design from the original design in order to meet reliability requirements. The fault-tolerant design should apply active redundancy and voting at the system level.</p> <p>The analysis should consider the results obtained by fault-injection (for example using a system level or component level FMEA (Failure Mode Effect Analysis)). The transformation should consider architecture requirements (e.g., “ON-Chip Redundancy” acc. IEC61508-2, Annex E).</p>
Rationale	<ul style="list-style-type: none"> • Safety related applications should be analysed with typical methods like performing a FMEA (e.g., on block level (using a reliability block diagram), or on component level (later in the development process)). • Implementing active redundancy at the system-level is one method of increasing fault-tolerance and reliability of a design. It allows to separate application development from fault-tolerance and ensures a more efficient development process (c.f. R1.3.3, R1.3.4) • Fault injection is a method to verify the predictions done within the FMEA (c.f. R5.7.8 – R5.7.11) • The use of on-chip redundancy requires the consideration of the requirements imposed by IEC61508-2, Annex E.
Significance	Medium
Means for validation/verification	<p>FORTISS will validate the reliability analysis and the model-to-model transformations used to obtain fault-tolerant design-variants at the module and integration level in T4.1</p> <p>In T4.4, FORTISS will support the use of these methods in the demonstrators, thereby enabling their validation the demonstrators. Feedback from the demonstrator lead partners (TRT (T 6.3), ALSTOM (T 7.3), STM (T 8.3)) will be incorporated.</p>
Source	DoW (T 4.1, T 5.2)
Additional Information	Note that the “consideration of IEC61508-2, Annex E” for on-chip redundancy in this context refers to the use of appropriate models and analysis/transformation algorithms. The construction of a platform-architecture that conforms to these requirements is not in the scope of this requirement onto the development process.

ID	R 9.13.7
Topic	Development Process
Subtopic	Design, Development and Validation of MC Systems

Name	DREAMS solutions integration in current development processes	
Responsibility	WP	WP1, WP4
	Lead partner	RTAW
	Participating partners	TRT, ONERA, RTAW, TTT, FENTISS, TUKL
Description	The development methodology provided by the DREAMS architecture should be able to be integrated in a development process as described in the relevant certification standards. That is, the development should be traceable, provide means for verification and validation of requirements, provide means for generating reports/records/documentation and provide means for controlled configuration of the final solution among others.	
Rationale	Without the compatibility with existing standards, DREAMS results would not be applicable to real world systems.	
Significance	Medium	
Means for validation/verification	The requirement will be validated by RTaW in T1.4, T 5.4, T4.4, by analyzing the proposed development process and tool support and in T1.8, T5.6, T6.3, T7.3, T8.3 by analyzing the actual usage by the demonstrators. Feedback from tool providers and demonstrators leader will be taken into account.	
Source	WG Avionics, DoW, Industrial practices, ...	
Additional Information	This requirement doesn't mean that the demonstrators and the used DREAMS framework will be developed using these standards, but that the development of the framework considers them so that the solution can be applied to real world critical systems in the future.	

9.14 Variability Binding

ID	R 9.14.1	
Topic	Development Process	
Subtopic	Variability Binding	
Name	Process for variability resolution	
Responsibility	WP	1, 4, 5
	Lead partner	SINTEF
	Participating partners	SINTEF, IKL, FORTISS

Description	The development process shall define how variability is bound.
Rationale	<p>Fulfilling this will be necessary to achieve high efficiency on adaptivity and to be able to keep adaptivity under adequate control. This is important in order to achieve safety and to achieve certifiability.</p> <p>We should expect that variability in DREAMS will be defined in several different ways (as is normally the case for complex adaptive systems). To get a unified approach to the handling of the adaptivity and the variability the development process should clearly define how variability is defined and how it is resolved. This should include both means for definition/resolution and the binding times.</p>
Significance	High
Means for validation/verification	<p>SINTEF will validate the variability binding process at the module and integration level in T4.3.</p> <p>In T4.4, SINTEF will support the use of these methods in the demonstrators, thereby enabling their validation the demonstrators.</p> <p>Feedback from the demonstrator lead partners (TRT (T 6.3), ALSTOM (T 7.3), STM (T 8.3)) will be incorporated.</p>
Source	DoW (T1.3, T1.4, T4.3, T5.5)
Additional Information	The definition of such a process will involve a set of tools. One such tool would be a feature modeling tool.

10 Requirements for Resource Management

In this section, the requirements on the DREAMS approach for global resource management are presented. The following discussion includes the organization of these requirements into subtopics and their relation to other requirement (sub-) topics in this document.

The goal of the integrated resource management services of DREAMS is the reconfiguration of a mixed-criticality system upon foreseen and unforeseen changes in its operational and environmental conditions. Autonomous detection mechanisms will be devised for operational and environmental changes (e.g., monitoring deadlines, overload detection). In addition, the DREAMS architecture will offer adaptability mechanisms for securely reconfiguring the system, without interrupting or interfering with execution. Important use cases of the integrated resource management will be the establishment of energy integrity in a mixed-criticality system and adaptive fault tolerance mechanisms based on multiple processor cores. DREAMS will provide services for system-wide adaptivity of mixed-criticality applications consuming several resources via global integrated resource management. The approach will be based on the separation of system-wide decisions to meet global constraints from the local execution on individual resources:

- Resources will be monitored individually (requirements provided in section 10.2)
- Resources will be scheduled individually (requirements provided in section 10.3)
- Should significant changes demand adaptation, global resource management (GRM) will take decisions on a system-wide level, based on offline computed configurations (requirements provided in section 10.1)
- These orders, representing bandwidth assignment, or scheduling parameters will be on an abstract level to separate system wide decisions from implementation details (requirements provided in section 10.6)
- The orders are translated into resource specific parameters and controlled by local resource management (LRM).

Thus, system-wide constraints, such as end-to-end timing, reliability, of energy integrity, can be addressed without incurring the complexity and overhead of individual negotiations among resources directly, providing scalability (requirements provided in section 10.7), reliability and safety (requirements provided in section 10.5), and timely adaptation (requirements provided in section 10.4).

10.1 Global Resource Management in Networked Multi-Core Chips

ID	R 10.1.1	
Topic	Resource Management	
Subtopic	Global Resource Management in Networked Multi-Core Chips	
Name	Global Resource Management	
Responsibility	WP	
	Lead partner	TUKL
	Participating partners	RTAW, UPV, FENTISS,

Description	<p>The Global Resource Manager (GRM) shall perform global decisions with information from resource monitors (see R 10.2.1). It provides new configurations for Local Resource Managers (see R 10.3.1) to virtualize resources (e.g., partition scheduling tables, resource budgets).</p> <p>System-wide constraints shall be supported such as end-to-end timing, reliability, energy integrity without increasing the complexity and overhead of individual negotiations among resources directly. The system-wide resource management decisions and their local execution will be separated.</p>
Rationale	Basic principle of global resource management.
Significance	High
Means for validation/verification	TUKL shall participate in the validation of the Global Resource Management services (D3.2.3) implemented in the Avionics Demonstrator (T3.4 Integration of Network Services and Support for use in Demonstrators).
Source	DoW
Additional Information	

ID	R 10.1.2	
Topic	Resource Management	
Subtopic	Global Resource Management in Networked Multi-Core Chips	
Name	Behaviour in the presence of faults of the GRM	
Responsibility	WP	
	Lead partner	TUKL
	Participating partners	RTAW, ONERA, UPV, FENTISS, VOSYS, TTT
Description	In case of failure of GRM, it has to be ensured that the safety of the system is not compromised. This encompasses hardware faults and component failures.	
Rationale	Failure in the GRM should not impair safety of an otherwise operational system. This can be achieved by e.g. fail silent behaviour of the GRM, so that a GRM failure will not impeded continuity of system operation, or making GRM part of the safety kernel. Optionally, redundant GRMs could be considered, which will fit into the general operation of resource management in DREAMS.	

Significance	Medium
Means for validation/verification	Experimental evaluation with the use of test cases that cover a faulty GRM in the Avionics Demonstrator.
Source	DoW
Additional Information	

ID	R 10.1.3	
Topic	Resource Management	
Subtopic	Global Resource Management in Networked Multi-Core Chips	
Name	Reconfiguration based on external inputs	
Responsibility	WP	3
	Lead partner	TUKL
	Participating partners	TTT, RTAW, ONERA, UPV, FENTISS, VOSYS
Description	A trigger for reconfiguration of LRMs can be initiated directly from the GRM based on external (e.g. user) input.	
Rationale	Resource management can be performed proactively on-demand.	
Significance	High	
Means for validation/verification	Experimental evaluation based on demonstrators (Avionics).	
Source	DoW	
Additional Information		

10.2 Resource Monitoring

ID	R 10.2.1	
Topic	Resource Management	
Subtopic	Resource Monitoring	
Name	Local Resource Monitoring	
Responsibility	WP	1,2,3
	Lead partner	TUKL
	Participating partners	TUKL, USIEGEN, RTAW, ST, TEI, ONERA, UPV, FENTISS, TTT, TRT, VOSYS

Description	<p>Resource Monitors (MON) shall monitor the resource availability. Significant changes will be reported to the GRM, who in turn can provide a different configuration at system-level.</p> <p>Local resource monitors shall be associated with the different virtualization building blocks (e.g., hypervisor and network interfaces in WP2, network switches in WP3).</p> <p>Standard interfaces (API) between the different configurations must be defined, providing information/control up to the application (from R10.2.2)</p>
Rationale	The information provided by the local resource monitors is a required input by the GRM.
Significance	High
Means for validation/verification	Experimental validation. Module test of the local resource monitors and the runtime services associated (low-level and high-level monitors). Integration test in WP2 (Chip-level building blocks).
Source	DoW
Additional Information	

ID	R 10.2.2	
Topic	Resource Management	
Subtopic	Resource Monitoring	
Name	Mixed-Criticality Monitoring	
Responsibility	WP	1, 2
	Lead partner	TRT
	Participating partners	TRT, ONERA, RTAW, TTT, UPV, FENTISS, TUKL, VOSYS
Description	Monitoring of attributes that are relevant for non-safety critical services (e.g., performance monitoring of non safety-critical services) shall not interfere with safety-relevant monitoring activities (e.g., health monitoring for safety)	
Rationale	Effect containment.	
Significance	High	
Means for validation/verification	Architecture(WP1), implementation (WP2)	
Source	DoW, TRT	
Additional Information		

ID	R 10.2.3	
Topic	Resource Management	
Subtopic	Resource Monitoring	
Name	Technology-independent monitoring	
Responsibility	WP	1, 2
	Lead partner	TRT
	Participating partners	TRT, ONERA, RTAW, TTT, UPV, FENTISS, TUKL, VOSYS, ST
Description	DREAMS approach to monitoring and reconfiguration should be applicable to different hardware platforms.	
Rationale	Related to Technology Independence, several solutions can be envisioned, e.g., virtualization	
Significance	High	
Means for validation/verification	Architecture definition (WP1)	
Source	WG Avionics, DoW, TRT	
Additional Information		

10.3 Local Resource Management and Local Resource Scheduling

ID	R 10.3.1	
Topic	Resource Management	
Subtopic	Local Resource Management and Local Resource Scheduling	
Name	Local Resource Managers (LRM)	
Responsibility	WP	2
	Lead partner	TUKL
	Participating partners	TUKL, IKL, USIEGEN, ONERA, UPV, FENTISS, VOSYS, ST, RTAW
Description	Algorithms shall be provided to dynamically configure virtualized resources using local resource managers (LRM) and local resource schedulers (LRS) to implement the decisions from the global resource management (GRM) such as resource-specific	

	<p>configurations, as well as monitoring their behaviour with feedback to the GRM.</p> <p>The LRMs shall adopt the configuration from the GRM at particular resources (e.g., processor core, memory, I/O). The LRM is responsible for mapping global decisions to the local scheduling policy of the LRS.</p>
Rationale	The separation of system wide decisions of global resource management and their local execution depends on LRS and LRM.
Significance	High
Means for validation/verification	TUKL participates in the experimental evaluation of the LRMs in the Avionics demonstrator.
Source	DoW
Additional Information	

ID	R 10.3.2	
Topic	Resource Management	
Subtopic	Local Resource Management and Local Resource Scheduling	
Name	Local Resource Scheduling (LRS)	
Responsibility	WP	2
	Lead partner	TUKL
	Participating partners	TUKL, IKL, ONERA, USIEGEN, TTT, UPV, FENTISS, VOSYS, ST, RTAW
Description	<p><i>Local Resource Schedulers (LRS)</i> shall perform the runtime scheduling of resource requests (e.g., execution of tasks on processor, processing of queued memory and I/O requests). The LRS in DREAMS will support different scheduling policies (e.g., dispatching of time-triggered actions, priority-based scheduling).</p> <p>The configuration of the LRS shall be controlled by the GRM via the LRM.</p>	
Rationale	The separation of system wide decisions of global resource management and their local execution depends on LRS and LRM.	
Significance	High	
Means for validation/verification	TUKL participates in the experimental evaluation of the LRSs in the Avionics demonstrator.	
Source	DoW	

Additional Information	
------------------------	--

10.4 Timely Adaptation (Measure of Success)

ID	R 10.4.1	
Topic	Resource Management	
Subtopic	Timely Adaptation (Measure of Success)	
Name	Bounded Reconfiguration Time	
Responsibility	WP	2, 4
	Lead partner	TUKL
	Participating partners	RTAW, ONERA, UPV, FENTISS, TTT, TRT, VOSYS
Description	<p>The system can be reconfigured upon foreseen and unforeseen changes in its operational and environmental conditions within in a predictable time span.</p> <p>Reconfiguration (changing the scheduling) time should be measurable and considered in the global scheduling.</p>	
Rationale	<p>The resource allocation strategies should allow for online changes of the allocation plans in order to permit adaptation to foreseen (and possible unforeseen) changes in the availability of the resources. Such changes must complete within in a predictable time span.</p>	
Significance	High	
Means for validation/verification	Implementation (WP2, WP3), Tools to compute it (WP4), assessment (WP6)	
Source	WG Avionics, DoW, TRT	
Additional Information		

ID	R 10.4.2	
Topic	Resource Management	
Subtopic	Timely Adaptation (Measure of Success)	
Name	Predictable Reconfiguration Results	
Responsibility	WP	4
	Lead partner	RTAW
	Participating partners	TUKL, ONERA, UPV, FENTISS, TRT, TTT,

Description	Ressource allocation plans which are activated through reconfiguration must guarantee predictable performances. This should be achieved through a set of precomputed (formally proven offline) ressource allocation plans, between which the resource manager switches deterministically at runtime. For the case of unforeseen changes, a fall back ressource allocation plan should be foreseen that allows bringing the system into a safe state.
Rationale	The complexity of determining system wide scheduling decisions for all resource is far too high to be handled dynamically or at runtime. Furthermore such algorithms are difficult to certify.
Significance	High
Means for validation/verification	D4.1.1, D4.1.2, D4.1.3, Assessment of demonstrators
Source	WG Avionics, DoW, TRT
Additional Information	

ID	R 10.4.3	
Topic	Resource Management	
Subtopic	Timely Adaptation (Measure of Success)	
Name	Continuity of Service during Reconfiguration	
Responsibility	WP	4
	Lead partner	TUKL
	Participating partners	RTAW, ONERA, UPV, FENTISS, VOSYS, TTT, TRT
Description	The system has to continue to provide required results during normal operation as well as during reconfiguration.	
Rationale	Necessary for timeliness in system for different criticalities.	
Significance	High	
Means for validation/verification	Experimental validation in the Avionic demonstrator.	
Source	DoW	
Additional Information		

ID	R 10.4.4	
Topic	Resource Management	

Subtopic	Timely Adaptation (Measure of Success)	
Name	Timely communication between GRM and LRM	
Responsibility	WP	3
	Lead partner	TTT
	Participating partners	TUKL, RTAW, UPV, FENTISS, VOSYS , TRT
Description	Information about the state of individual resources, as provided by the local resource managers, has to be transmitted to the GRM in a timely manner. Similarly, the decisions about resources stated taken by the GRM have be communicated to the LRMs in a timely fashion.	
Rationale	Needed for timely operation of resource management. Variations or large delay may lead to incorrect state information or no longer valid resource management decisions.	
Significance	high	
Means for validation/verification	Assurance is achieved through implementation (T3.2) and assessment (T6.3, T7.3, T8.3)	
Source	Dow	
Additional Information		

10.5 Reliability and Safety

ID	R 10.5.1	
Topic	Resource Management	
Subtopic	Reliability and Safety	
Name	Definition of Real-time faults detection and recovery strategies	
Responsibility	WP	4
	Lead partner	ONERA
	Participating partners	TRT, TUKL
Description	Definition of fault model such as overuse of shared resources, deadline exceeding, rules violation need to be provided. For each fault, adequate detection and recovery strategies. E.g. switching between off-line scheduling tables at runtime need to be determined	
Rationale	Increase of performance, management of mixed-criticality	

Significance	High
Means for validation/verification	The requirement will be verified with formal methods (T4.1). The requirement will be experimentally evaluated by ONERA, TRT and TUKL through the use of the fault injection in the avionic demonstrator (T6.3).
Source	DOW, ONERA
Additional Information	

ID	R 10.5.2	
Topic	Resource Management	
Subtopic	Reliability and Safety	
Name	Safe reconfiguration of resources	
Responsibility	WP	3, 4
	Lead partner	TUKL
	Participating partners	RTAW, UPV, FENTISS, VOSYS, TTT
Description	Reconfiguration of resources (e.g. activation of a new network schedule, switching between task schedules) must be done synchronously in a coordinated safe manner for all resource types.	
Rationale	Requirement for consistency	
Significance	High	
Means for validation/verification	Experimental assessment of synchrony of the reconfiguration in the demonstrators.	
Source	DoW	
Additional Information		

10.6 Abstract Service Levels of Resource States

ID	R 10.6.1	
Topic	Resource Management	
Subtopic	Abstract Service Levels of Resource States	
Name	Separation of local and global resource management	
Responsibility	WP	
	Lead partner	ONERA

	Participating partners	TUKL, RTAW, UPV, FENTISS, VOSYS, ST, TRT
Description	Resource management in DREAMS is based on the separation of system-wide decisions to meet global constraints from the local execution on individual resources: resources will be monitored individually with abstract information provided to global resource management (GRM). GRM will take decisions on a system-wide level, with orders, such as bandwidth assignment, or scheduling parameters for all resources, which are controlled by local resource management (LRM). Local algorithms monitor and dynamically configure virtualized resources as local resource monitors and local resource schedulers to implement decisions from global resource management into resource specific configurations, as well as monitoring their behaviour with feedback to global resource management.	
Rationale	Principle for global resource management to achieve efficiency and adaptability.	
Significance	high	
Means for validation/verification	<p>The requirement will be partially verified with formal methods by TUKL and ONERA(T4.1).</p> <p>The requirement will be experimentally evaluated by ONERA, TUKL, FENTISS, UPV and TRT through the use of the fault injection in the avionic demonstrator (T6.3).</p> <p>The requirement will be experimentally evaluated by VOSYS and ST in the health care demonstrator (T8.3).</p>	
Source	DOW	
Additional Information		

10.7 Scalability

ID	R 10.7.1	
Topic	Resource Management	
Subtopic	Scalability	
Name	Computational complexity of resource management algorithms	
Responsibility	WP	3
	Lead partner	TUKL
	Participating partners	RTAW, ONERA

Description	Due to the separation of system wide decisions taken by the global resource manager and their local execution on resources, the actual resource management decisions shall be based on small, abstract view of the system state with low complexity
Rationale	Basic principle of global resource management.
Significance	medium
Means for validation/verification	TUKL performs the analytical evaluation of the GRM design in D3.2.1 in WP3 (Task 3.2 Global Resource Management). ONERA will perform some formal analysis in T4.1.
Source	DOW
Additional Information	

11 Requirements for Security

In this section, the requirements for security aspects in DREAMS are presented. The following discussion explains the subtopics in this section and their relationship to other topics in this deliverable.

To ensure security in a system, security services, such as confidentiality, integrity, availability and authenticity need to be provided for the different sub-systems. Confidentiality ensures the privacy of information, integrity ensures that data cannot be modified unnoticeably, availability ensures that the system or a sub system is available when needed and authenticity ensures that the data is genuine and that the actual origin of the data is the same as the claimed origin. These security services are provided through security mechanisms, such as encryption to provide confidentiality or digital signatures to provide authenticity in communications.

Security is mostly an invisible attribute of a system. This implies that the end users use security features implicitly and therefore tend to forget about the importance of security. In the wake of the recent information leakages, information security has become an integral part of modern systems. Security has many dimensions and every component of a large system might need security of a certain dimension. For some systems, privacy has higher importance, whereas for others authenticity of data and its originator is more important. Security should be considered at different levels for distributed systems such as DREAMS. In the context of DREAMS, security issues need to be addressed at different levels. This includes security of data in the memory, on the chip, off the chip, transmission of data from the chip to the off-chip network (cluster), in the hypervisors and in the software design of the applications running on the DREAMS architecture.

In this section, the security requirements for different components of the DREAMS architecture are identified. Security in DREAMS is a horizontal activity which appears in the different work packages and tasks of the DREAMS project. The security requirements are grouped into the following categories,

- **Top-level security architecture (Sec. 11.1):** Identifies the core security concept for on-chip and off-chip security in the DREAMS architecture. This includes a threat model and global properties concerning security.
- **Chip-level security (Sec. 11.2):** Covers the requirements for on-chip security in the virtualization aware context. On-chip security services shall provide protection from logical attacks against the virtual machine and the resource management (LRM, LRS, MON), especially through the network interfaces.
- **Cluster-level security (Sec 11.3):** Covers the requirements for security services, such as confidentiality, integrity, availability and authenticity for communication between different chips to prevent attacks, such as man-in-the-middle attacks, replay attacks, spoofing attacks, denial of service attacks and data masquerading attacks etc. The requirements in this section also concern secure communication between the GRM, LRMs, LRSs and MONs. Time distribution for synchronization is also a network level activity which needs to be done securely and is covered by the requirement of secure time distribution for global time base. A global time base is essential for predictable virtualization of resources and TSP in DREAMS. The time distribution and synchronization shall be protected against active attacks. Last but not the least; cryptographic key management is essential for the provision of these security services.
- **Security properties and validation (Sec. 11.4):** Covers the requirements for the validation of security in the DREAMS architecture. This concerns not only security aspects themselves but also the Quality of Service (QoS) aspects such as bounded effects on timing and extra-functional requirements, e.g., reliability and energy.

- **Application-level security (Sec. 11.5):** Covers the specific security requirements for the application-level which are not covered by the other requirements.
- **Security in the development process (Sec. 11.6):** Identifies the requirement that the development process for applications shall address security issues in the design phase. This can obviate security risks in a later development phase and reduces the need for applying security patches later on.

In addition to the security requirements listed in this section, there are requirements for security in other topics, e.g., requirements for healthcare demonstrator in Sec. 8 and requirements for modelling and development process in Sec. 9. The security subtopic in the healthcare demonstrator covers the specific security requirements for this demonstrator, e.g., the communication security between the specific participating components as well as privacy of data exchanged. The security requirements for modelling and development process in Sec. 9 covers the modelling part and includes the meta-models for data confidentiality, data integrity and authenticity the of communication partner.

The majority of the security services and mechanisms will be integrated transparently into the DREAMS architecture. All demonstrators will use them implicitly without much additional work, e.g., by using the network level communications without knowing that the communications take place with integrated security services. However, the healthcare demonstrator has explicit requirements for security and privacy and therefore it uses the security services and mechanisms explicitly.

11.1 Top-level Security Architecture (off-chip and on-chip)

ID	R 11.1.1	
Topic	Security	
Subtopic	Top-level Security Architecture (off-chip and on-chip)	
Name	Domain Independent core security services	
Responsibility	WP	1
	Lead partner	USIEGEN
	Participating partners	USIEGEN, ST, TEI, TTT, VOSYS
Description	The core security services shall include secure communications, secure time distribution and secure execution environment.	
Rationale	The DREAMS architecture shall be based on a minimal set of essential security services. These services shall provide a foundation for other services to be built on top.	
Significance	High	
Means for validation/verification	The requirement will be evaluated analytically by USIEGEN and by ALSTOM and ST through the wind power and healthcare demonstrators in T7.3 and T8.3.	
Source	DOW (WP1 / T1.2)	

Additional Information	
------------------------	--

ID	R 11.1.2	
Topic	Security	
Subtopic	Top-level Security Architecture (off-chip and on-chip)	
Name	Identification of security services, security policies and threat models for mixed criticality systems	
Responsibility	WP	1
	Lead partner	USIEGEN
	Participating partners	USIEGEN, TTT, VOSYS
Description	Identification of core and optional security services, security policies and threat models in the DREAMS architecture and provision of security mechanisms to address those identified services.	
Rationale	Security aspects shall be identified and considered in the architectural style of DREAMS. Global properties concerning security shall be identified and listed. Security shall be embedded into the development process e.g., via security patterns.	
Significance	Medium	
Means for validation/verification	The requirement will be analytically verified by USIEGEN and validated by ALSTOM and ST through the wind power and healthcare demonstrators in T7.3 and T8.3.	
Source	DOW (WP1 / T1.2)	
Additional Information	The architecture of DREAMS will have its specialized requirements of security, which may not have been addressed in the past.	

11.2 Chip-level security (logical and physical security)

ID	R 11.2.1	
Topic	Security	
Subtopic	Chip-level security (logical and physical security)	
Name	Security of the monitoring components	

Responsibility	WP	2
	Lead partner	USIEGEN, VOSYS
	Participating partners	USIEGEN, VOSYS , ST, TEI
Description	Security mechanisms shall be provided to verify the authenticity and integrity of the MON, LRM and LRS.	
Rationale	Authenticity and integrity of monitoring and resource management components shall be provided to ensure trust worthy communications and executions.	
Significance	High	
Means for validation/verification	Experimental validation of the authenticity and integrity of the monitoring and scheduling components in T2.1, T2.2 and T2.3 will be performed by USIEGEN, VOSYS, ST and TEI and through the demonstrators in T7.3 and T8.3.	
Source	DoW (WP2/ T2.1, T2.2 and T2.3)	
Additional Information	<p>We consider protection against physical adversaries attempting to compromise the monitoring components and their external communication channel (gateway and/or memory).</p> <p>These attacks could probe the external memory bus or tamper integrity of the system's communication facility. Man-in-the-middle attacks and internal disgruntled worker attacks fall within this scope, including spoofing, splicing and replay.</p>	

ID	R 11.2.2	
Topic	Security	
Subtopic	Chip-level security (logical and physical security)	
Name	Security services at the network interface layer	
Responsibility	WP	2
	Lead partner	TEI, VOSYS
	Participating partners	TEI, VOSYS, ST, USIEGEN
Description	Security services shall provide protection from logical attacks, in the virtualization-aware context, through the network interfaces (NI).	
Rationale	Enhanced chip-level virtualization security at the Spidergon STNoC network interfaces. The details of this requirement are based on the threat analysis done in R 11.1.2. This will include protections against security attacks such as eavesdropping.	

Significance	High
Means for validation/verification	Validation will be done through simulations as well as through the wind power and healthcare demonstrators in T7.3 and T8.3 by ST, TEI, VOSYS and USIEGEN.
Source	DoW (WP2 / T2.1, T2.3 (drivers))
Additional Information	

ID	R 11.2.3	
Topic	Security	
Subtopic	Chip-level security (logical and physical security)	
Name	Security through the VM	
Responsibility	WP	2
	Lead partner	ST
	Participating partners	ST, TEI, VOSYS, USIEGEN
Description	Mechanisms for enhancing the on chip and off chip security through the VM shall be provided. This includes authenticity of the components of the applications, VMs and the hypervisor.	
Rationale	Enhanced chip-level virtualization security shall be provided by strong guest isolation support and integration of the mandatory access control (MAC) through the sVirt for RT-KVM.	
Significance	High	
Means for validation/verification	<p>Consistent RTL integration, drivers and software security manager development, and KVM extensions with FPGA platform prototyping by TEI and ST.</p> <p>Validation will also be done through the wind power and healthcare demonstrators in T7.3 and T8.3 by USIEGEN, ST and ALSTOM.</p>	
Source	DoW (WP2 / T2.1, T2.3)	
Additional Information	<p>This requirement focuses on logical threats inside the chip, e.g., VMs attacking other VMs or the hypervisor.</p> <p>In addition, the Denial-of-Service attacks shall be subverted, where malicious code injected by attackers prevents intended users from using a system service, e.g. by saturating the NoC through massive unauthorized accesses.</p> <p>We partly consider robustness against certain software vulnerabilities, e.g. exploiting bounded buffer overflows and we target enhanced chip level virtualization security.</p>	

ID	R 11.2.4	
Topic	Security	
Subtopic	Chip-level security (logical and physical security)	
Name	Protection from physical attacks shall be considered	
Responsibility	WP	2
	Lead partner	USIEGEN
	Participating partners	USIEGEN, ST, TEI
Description	Mechanisms for protection against physical attacks, such as side channel attacks, shall be evaluated and provided if found adequate, e.g., if they do not affect the QoS requirements.	
Rationale	Security of the on chip communication can be compromised by a physical access to the chip or access to the vicinity of the chip. Security attacks, such as side channel attacks, can deduce the encrypted texts and cryptographic keys without actually seeing the communication but by merely observing it, e.g., by monitoring the electromagnetic radiations emitted from the chip.	
Significance	Medium	
Means for validation/verification	Analysis and experimentation will be done by USIEGEN, ST and TEI in T2.1. This requirement will also be validated in T8.3 through the healthcare demonstrator by USIEGEN and ST.	
Source	DOW (WP2 / T2.1)	
Additional Information		

ID	R 11.2.5	
Topic	Security	
Subtopic	Chip-level security (logical and physical security)	
Name	System software for security services	
Responsibility	WP	2
	Lead partner	ONERA
	Participating partners	ONERA, USIEGEN, ST, TEI, VOSYS
Description	Drivers shall be provided to access the generic security services of the underlying platform in the execution environments.	

Rationale	The drivers will integrate the security mechanisms that support trustworthy communications between the system components, e.g., between the MON, LRM and LRS.
Significance	High
Means for validation/verification	This requirement will be validated in the wind power and healthcare demonstrators in T7.3 and T8.3.
Source	DoW (WP2 / T2.3)
Additional Information	

11.3 Cluster-level Security

ID	R 11.3.1	
Topic	Security	
Subtopic	Cluster-level Security	
Name	Security services for Ethernet-related protocols (e.g., TTEthernet)	
Responsibility	WP	3
	Lead partner	USIEGEN
	Participating partners	USIEGEN, TTT
Description	Security services, such as confidentiality, integrity, availability, authenticity of origin etc., shall be provided for the TTEthernet based on the MACsec (IEEE 802.1AE).	
Rationale	Ethernet related protocols, e.g., TTEthernet, will form the backbone of the cluster level communication in the DREAMS architecture. All the communications involved shall be protected using security services, such as protection from message forgery, eavesdropping, impersonation, replay, modification etc.	
Significance	High	
Means for validation/verification	<p>Reasonable network level security attacks on the protocols.</p> <p>Experimental validation of cluster level security will be performed in T3.3 by USIEGEN and TTT.</p> <p>The requirement will also be verified by USIEGEN, ST and ALSTOM in T7.3 and T8.3.</p>	

Source	DOW (WP3 / T3.3)
Additional Information	Protection mechanisms against adversaries for the cluster level communications.

ID	R 11.3.2	
Topic	Security	
Subtopic	Cluster-level Security	
Name	Secure time distribution for global time base	
Responsibility	WP	3
	Lead partner	USIEGEN
	Participating partners	USIEGEN, TTT
Description	Secure time distribution and synchronization shall be provided in the DREAMS architecture.	
Rationale	Time synchronization for establishing a global time base is a key mechanism for the predictable virtualization of resources and TSP in DREAMS. Time synchronization shall be protected against active attacks.	
Significance	High	
Means for validation/verification	Experimental validation of secure time distribution and synchronization in T3.3 will be performed by USIEGEN and TTT. The requirement will be used and validated in the healthcare, wind power and avionics demonstrators in T6.3, T7.3 and T8.3.	
Source	DOW (WP3 / T3.3)	
Additional Information	This requirement will be addressed by integration of the TTEthernet and MACsec protocols.	

ID	R 11.3.3	
Topic	Security	
Subtopic	Cluster-level Security	
Name	Provision of cryptographic primitives and cipher suites	
Responsibility	WP	Responsibility
	Lead partner	USIEGEN
	Participating partners	

Description	A choice of cipher suites shall be provided. A cipher suite includes cryptographic algorithms and their parameters, e.g., key sizes etc.
Rationale	Privacy of the communications is extremely important for certain applications, such as the patient monitoring system. The privacy of cluster level communication needs to be given special considerations because the communication might take place over public networks.
Significance	High
Means for validation/verification	Experimental validation and testing by USIEGEN. The requirement will be validated in the healthcare and wind power demonstrators by ALSTOM and ST in T7.3 and T8.3.
Source	DOW (WP3 / T3.3)
Additional Information	

ID	R 11.3.4	
Topic	Security	
Subtopic	Cluster-level Security	
Name	Core security services at the cluster level.	
Responsibility	WP	3
	Lead partner	TTT
	Participating partners	USIEGEN, TTT
Description	Core security services on the cluster level shall be identified and provided. This includes services such as end-to-end security (e.g., privacy and authentication).	
Rationale	Analysis of the security functions on the cluster level in order to establish a core set of security services shall be done. This will reach from end-to-end encapsulation to specific safety features that cover needs in the security domain (e.g., security of time, cryptographic mechanisms).	
Significance	High	
Means for validation/verification	This requirement will be validated by USIEGEN and TTT through experimental analysis as well as through the demonstrators in T7.3 and T8.3.	
Source	DOW (WP3 / T3.3)	
Additional Information		

ID	R 11.3.5	
Topic	Security	
Subtopic	Cluster-level Security	
Name	Secure communications between GRM, LRM and MON	
Responsibility	WP	2, 3
	Lead partner	USIEGEN
	Participating partners	USIEGEN, TTT, TEI, ST
Description	Mechanisms for secure end-to-end communications between the monitoring and resource management components shall be provided.	
Rationale	Specification of security concept and implementation of necessary security mechanisms to provide trustworthy communications between the GRM, LRMs and MON. Trust level depends on the applications and measuring “trust level” is an active research topic.	
Significance	High	
Means for validation/verification	<p>Experimental validation of security concepts and mechanisms in T3.3 will be performed by USIEGEN and TTT.</p> <p>The requirement will also be validated through the demonstrators by ALSTOM and ST in T7.3 and T8.3.</p>	
Source	DOW (WP3 / T3.3)	
Additional Information		

ID	R 11.3.6	
Topic	Security	
Subtopic	Cluster-level Security	
Name	Security services in the gateways	
Responsibility	WP	3
	Lead partner	TTT
	Participating partners	TTT, USIEGEN

Description	Provision of standard communications security services, for confidentiality, integrity, non-repudiation and availability in the gateways at the network interfaces.
Rationale	<p>The gateways are specified and implemented in order to allow the abstraction of communication between multiple chips in a transparent way. Gateways are a key point in the security of the overall communications infrastructure. A few services of the gateway that are prone to security attacks are,</p> <ul style="list-style-type: none"> • Clock synchronization • Selective redirection • Resolving of property mismatches • Routing and mapping of namespaces <p>Protective measures need to be taken, e.g., by providing secure clock synchronization, secure routing etc., to avoid disruption of the above services by the attacker.</p>
Significance	High
Means for validation/verification	The requirement will be validated through implementation of security attacks and through the demonstrators in T7.3 and T8.3.
Source	DOW (WP3 / T3.1)
Additional Information	

ID	R 11.3.7	
Topic	Security	
Subtopic	Cluster-level Security	
Name	Key Management	
Responsibility	WP	3
	Lead partner	USIEGEN
	Participating partners	USIEGEN, TTT
Description	Key management for secure communication between the entities on a cluster shall be provided (Mechanisms for key generation, key distribution/exchange, key destruction etc.).	
Rationale	The key management service includes key generation, secure key distribution/exchange and key destruction when a key is no longer needed. Other key management services include key storage and retrieval.	
Significance	High	

Means for validation/verification	The requirement will be tested by USIEGEN in T3.3 and validated through the demonstrators by ALSTOM and ST in T7.3 and T8.3.
Source	DOW (WP3 / T3.3)
Additional Information	Key will be generated and exchanged based on standard algorithms, such as DH Key Exchange Algorithms etc.

ID	R 11.3.8	
Topic	Security	
Subtopic	Cluster-level Security	
Name	Replay Protection	
Responsibility	WP	3
	Lead partner	USIEGEN
	Participating partners	USIEGEN, TTT
Description	Protection mechanisms for protection against replay attacks shall be provided.	
Rationale	Replay attacks can have disastrous effects on the system. Examples include disruption of the patient health status through the patient monitoring system and disturbing the clock synchronization by replaying old messages.	
Significance	High	
Means for validation/verification	The requirement will be validated experimentally by USIEGEN and by the demonstrators by ALSTOM and ST in T7.3 and T8.3.	
Source	DOW (WP3 / T3.3)	
Additional Information		

ID	R 11.3.9	
Topic	Security	
Subtopic	Cluster-level Security	
Name	Protection against man-in-the-middle attacks	
Responsibility	WP	3
	Lead partner	USIEGEN
	Participating partners	USIEGEN, TTT

Description	Communication shall be protected against man-in-the-middle attacks
Rationale	Man in the middle attacks might result in denial of service, access to confidential information, modification of confidential data etc. and shall be avoided.
Significance	High
Means for validation/verification	The requirement will be validated through testing by USIEGEN and TTT and also through the demonstrators in T7.3 and T8.3.
Source	DOW (WP3 / T3.3)
Additional Information	

ID	R 11.3.10	
Topic	Security	
Subtopic	Cluster-level Security	
Name	Protection against traffic analysis	
Responsibility	WP	3
	Lead partner	USIEGEN
	Participating partners	USIEGEN, TTT
Description	Cluster level communication, e.g., over TTEthernet, shall be protected against traffic analysis attacks.	
Rationale	Traffic analysis is a passive attack that helps in the deduction of information from patterns in communications. It can be performed even in the presence of encryption and other means shall be employed for protection against traffic analysis.	
Significance	High	
Means for validation/verification	The requirement will be validated through experimental analysis of the system by USIEGEN and TTT.	
Source	DOW (WP3 / T3.3)	
Additional Information		

ID	R 11.3.11	
Topic	Security	
Subtopic	Cluster-level Security	
Name	Protection against denial of service attacks	

Responsibility	WP	3
	Lead partner	USIEGEN
	Participating partners	USIEGEN, TTT
Description	Critical components in DREAMS shall be analysed and mechanisms shall be proposed to minimize the denial of service attacks.	
Rationale	Denial of service attacks might be launched against the critical components of dreams clusters, resulting in denial of service. This can be very critical for certain applications, e.g., healthcare. Such attacks should be minimized.	
Significance	High	
Means for validation/verification	The requirement will be validated through experimental analysis of the system by USIEGEN and TTT and by the healthcare demonstrator in T8.3.	
Source	DOW (WP3 / T3.3)	
Additional Information		

11.4 Security Properties and Validation

ID	R 11.4.1	
Topic	Security	
Subtopic	Security Properties and Validation	
Name	Integrity, authenticity and availability based on the attacker and security model of DREAMS	
Responsibility	WP	1,2,3
	Lead partner	USIEGEN
	Participating partners	USIEGEN, TTT, , VOSYS
Description	Integrity, authenticity and availability shall be ensured for communications and communications partners, in the presence of security threats, such as message sniffing, insertion, modification and denial of service.	
Rationale	Confidentiality, Integrity and Availability (CIA) are the pillars of security in any modern architecture. These pillars help in protection against the different perceived threats and ensure that the system or components of a system are available at all time and trustable. The CIA triad also ensures that the integrity of the	

	message exchange between different components is not violated. The authenticity of communicating entities is another prime aspect of security and must be provided by the DREAMS architecture.
Significance	High
Means for validation/verification	Reasonable attack scenarios by USIEGEN and TTT. The requirement will be validated in the wind power and healthcare demonstrators by ALSTOM and ST in T7.3 and T8.3.
Source	DOW / Measures for success of project objectives (Obj. 1)
Additional Information	

ID	R 11.4.2	
Topic	Security	
Subtopic	Security Properties and Validation	
Name	Validation of security services	
Responsibility	WP	1
	Lead partner	USIEGEN
	Participating partners	USIEGEN, TTT, VOSYS
Description	Security services shall be validated using reasonable attack scenarios and related penetration tests. Attack scenarios in the context of the DREAMS architecture need to be envisaged and implemented to validate the strength of the security services of DREAMS.	
Rationale	The strength of security services can be validated only if as many as possible attack-scenarios are envisaged and reasonable attacks are implemented. This includes penetration tests, denial of service attacks, message sniffing and spoofing attacks etc. This will validate the strength of perceived security of the DREAMS architecture and contributes towards the measure of success.	
Significance	High	
Means for validation/verification	Experimental validation of security services based on reasonable attacks will be performed by USIEGEN, TTT and VOSYS. The requirement will also be validated through the demonstrators by ALSTOM and ST in T7.3 and T8.3.	
Source	DOW / Measures for success of project objectives (Obj. 1)	
Additional Information		

ID	R 11.4.3	
Topic	Security	
Subtopic	Security Properties and Validation	
Name	Bounded effects of security mechanisms on timing and EFR, e.g., reliability and energy	
Responsibility	WP	2
	Lead partner	VOSYS
	Participating partners	VOSYS,ST
Description	Security mechanisms (at chip and cluster level) shall have a bounded effect on timing and EFP such as reliability and energy efficiency	
Rationale	Efficient HW/SW virtualization-aware protection mechanisms.	
Significance	Medium	
Means for validation/verification	<p>System-level and RTL models will be used</p> <p>Drivers and software security manager will also need to be examined experimentally.</p> <p>The requirement will also be validated through the demonstrators by ALSTOM and ST in T7.3 and T8.3</p>	
Source	DOW / Measures for success of project objectives (Obj. 1)	
Additional Information		

11.5 Application-level Security

ID	R 11.5.1	
Topic	Security	
Subtopic	Application-level security	
Name	Data origin authentication	
Responsibility	WP	2, 3, 8
	Lead partner	USIEGEN
	Participating partners	USIEGEN, TTT
Description	The communication messages between the resources shall be trustworthy. It shall be possible to verify the origin of messages.	
Rationale	In order to ensure trustworthy communication between different resources, like media home gateway (MHG) and remote	

	monitoring system, the actual origin of the messages should be the same as the claimed origin.
Significance	High
Means for validation/verification	The requirement will be used in the demonstrators by the demonstrator partners, e.g., by ST in T8.3.
Source	DOW (WP2 / T2.2, WP3 / T3.3)
Additional Information	

11.6 Security in the Development Process

ID	R 11.6.1	
Topic	Security	
Subtopic	Security in the Development Process	
Name	Security in the development process	
Responsibility	WP	1
	Lead partner	USIEGEN
	Participating partners	USIEGEN, TTT, VOSYS
Description	Integration of security in the development process, i.e., security by design, e.g., using security patterns.	
Rationale	The development process shall address security issues in the design phase. Known and proven techniques shall be employed for secure software development. This can be accomplished e.g., via security patterns.	
Significance	High	
Means for validation/verification	Will be validated through the demonstrators in T7.3 and T8.3.	
Source	DOW (WP1 / T1.3)	
Additional Information		

12 Building Blocks

The following sections list already available building blocks to address different aspects needed for the DREAMS concept. A building block may provide a technique, a dedicated algorithm or an overall

methodology as a solution to a general purpose or specific problem. They are often available as results from previous projects or they exist as intellectual property.

12.1 Building Blocks for Architecture

12.1.1 GENESYS

ID	B 1.1.1	
Name	Cross-domain architectural style	
Source	GENESYS	
Owner	TTT	
Description	<p>The GENESYS cross-domain architectural style is characterized by architectural principles and structuring rules for dependable distributed embedded systems. A principle is an accepted statement about some fundamental insight in a domain of discourse. Principles form the basis for the formulation of operational rules. In GENESYS these principles are operationalized in the reference architecture template of the architectural service specification, which is covered in the following chapters. The cross-domain architectural style is the result of extensive</p> <p>The cross-domain architectural style of GENESYS is the result of extensive discussions among the members of the GENESYS consortium, which included experts from the diverse application domains, ranging from safety-critical embedded systems to dynamic multimedia systems, such as mobile phones.</p>	
Building block type	Document.	
Services provided	The architectural principles include strict component orientation, separation of computation from communication, availability of a common time, hierarchical system structure, adherence to message passing, state awareness, fault isolation and integrated resource management.	
Dependencies / applicability		
Relationship to requirements	Related to all architecture requirements.	

ID	B 1.1.2	
Name	Reference architecture template	
Source	GENESYS	
Owner	TTT	
Description	<p>The reference architecture template is a template for building a concrete instantiation of the GENESYS architecture. The reference architecture template provides specifications for a comprehensive set of platform services. The core services of GENESYS (i.e., basic configuration, execution control, basic time, basic communication) are implemented by a trusted subsystem, which consists of the Trusted Resource Manager (TRM), the NoC and the communication interfaces to the components.</p>	
Building block type	Document.	
Services provided	Domain-independent optional services are open set of services realized on the core services:	

	<pre> graph LR Root[Domain-Independent Optional Services] --> Robustness[Robustness Services] Root --> Memory[External Memory Management Services] Root --> Security[Security Services] Root --> Resource[Resource Management Services] Root --> Gateway[Gateway Services] Root --> Mobility[Mobility Services] Root --> Higher[Higher Communication Services] Robustness --> SE[State Externalization] Robustness --> MS[Membership Service] Robustness --> ADI[Analysis of Diagnostic Information] Robustness --> CRS[Component Restart Service] Robustness --> VS[Voting Service] Memory --> ACM[Access Control of Memory Partitions] Memory --> SS[Stable Storage] Memory --> SecS[Secure Storage] Security --> SKM[Security Key Management] Security --> ED[Encryption and Decryption] Security --> RNG[Random Number Generation] Security --> SA[Service Authentication] Security --> SBS[Secure Boot Service] Security --> SAC[Service Access Control] Resource --> LRM[Local Resource Management] Resource --> GRM[Global Resource Management] Resource --> DLRM[Device Level Resource Management] Resource --> CR[Configuration and Reconfiguration] Gateway --> WC[Wireless connection] Gateway --> IC[Internet Connection] Gateway --> LI[Legacy Integration] Gateway --> FCS[Fault tolerant Clock Synchronization] Gateway --> PIO[Process Input Output] Mobility --> CSD[Component/Service Detection] Mobility --> CM[Connectivity Management] Mobility --> MDCM[Mobile Device Controlled Mobility] Mobility --> ICM[Infrastructure Controlled Mobility] Higher --> HLP[High-Level Protocol Implementation] Higher --> RCS[Receiver Controlled Streaming] </pre>
Dependencies / applicability	
Relationship to requirements	Related to all architecture requirements.

12.2 Building Blocks for Multicore Virtualization Technology

12.2.1 TRESCCA

ID	B 2.1.1
Name	Virtualization extensions
Source	TRESCCA
Owner	ST, TEI, VOSYS
Description	Virtualization-aware protection mechanisms
Building block type	SystemC, RTL (ongoing)
Services provided	ST and TEI currently extend the Spidergon STNoC network interface with novel virtualization-aware hardware security modules, that act as NoC Firewalls. In order to manage efficiently and securely different types of connected resources corresponding system drivers, software configuration mechanisms and specialized VMs are being developed by VOSYS.
Dependencies / applicability	Virtualization-aware technology must be adapted and extended (in the presence of mixed criticality and safety constraints. This is not limited to security, but includes address translation, device tables, TLB, page walk, synchronization, monitoring and decision support.
Relationship to requirements	Related to chip level and security requirements.

12.2.2 vIrtical

ID	B 2.2.1
Name	I/O virtualization solutions
Source	vIrtical
Owner	ST, TEI, VOSYS
Description	I/O virtualization (customized and ARM-v7 compatible solution)
Building block type	SystemC, RTL (ongoing)
Services provided	ST, TEI and VOSYS are working on hardware/software extensions related to supporting full system virtualization by pursuing the design of an integrated IOMMU component to perform virtual-to-physical address translation and provide task isolation and monitoring functionalities.
Dependencies / applicability	Centralized IOMMU technology must be adapted to develop an efficient distributed solution that work at the network interface layer in the presence of mixed criticality and safety constraints.
Relationship to requirements	R 1.1.5

12.3 Building Blocks for Mixed-criticality Network

12.3.1 ACROSS

ID	B 3.1.1	
Name	TTNOC to TTNOC gateway	
Source	ACROSS	
Owner	TTT	
Description	Within the ACROSS project, TTT has implemented a prototypical gateway between two TTNOC networks using a TTEthernet network, thus allowing cores to communicate transparently across a TTEthernet network.	
Building block type	Software	
Services provided	The “transparent gateway service” provides a (software-based) gateway between two TTNOC networks. Parts of the concept of this gateway service are reused in DREAMS.	
Dependencies / applicability	WP3	
Relationship to requirements	R 12.3.1	

12.3.2 SCARLETT

ID	B 3.2.1	
Name	TTEthernet Reconfiguration	
Source	SCARLETT	
Owner	TTT	
Description	Within the SCARLETT project, reconfiguration capabilities were provided in switches to allow location specific modification of switch functionality to support reallocation of processing resources following various failure conditions.	
Building block type	Switch functionality	
Services provided	Switch reconfiguration functionality to implement mode changes	
Dependencies / applicability	WP3	
Relationship to requirements	R 12.3.2	

12.3.3 Internal Projects: TTT

ID	B 3.3.1	
----	---------	--

Name	TTEthernet Protocol	
Source	TTT	
Owner	TTT	
Description	The TTEthernet protocol SAE6802 developed by TTT provides the core functionality for mixed-criticality networking.	
Building block type	Chip IP, Software	
Services provided	<ul style="list-style-type: none"> • System-wide safe clock synchronization • Time- and space partitioning at the cluster level • Bounded WCTT for cluster-level communication 	
Dependencies / applicability	WP3	
Relationship to requirements	R 1.1.3, R 12.3.3, R 12.3.4, R 12.3.5, R 12.3.6, R 12.3.7, R 12.3.8	

12.4 Building Blocks for Architecture Tooling, Scheduling and Analysis

12.4.1 Internal Projects: ONERA

ID	B 4.1.1	
Name	SchedMCore (schedlabiltiy, execution layer, dispatcher, PRELUDE)	
Source	Internal Projects:ONERA	
Owner	Onera	
Description	<p>Prelude/SchedMcore is an end-to-end framework that goes from the formal description of a multi-periodic assembly of sub-tasks down to the execution on real-target.</p> <p>Prelude is a formal synchronous data-flow language designed for the specification of the software architecture of a critical embedded control system. Prelude compiler generates a set of dependent periodic tasks that preserves the semantics of the original program, based on the use of a communication protocol.</p> <p>SchedMcore offers three tools:</p> <ul style="list-style-type: none"> • schedulability analyser based on an exhaustive analysis, using model checking or constraint solving techniques. It includes the generation of static off-line schedule, • simulator for Prelude programs to validate the execution • several hard real-time executive layers for multi/many-core (POSIX schedulers , or directly bare-metal libraries) 	

Building block type	Tool, SW (bare-metal libraries)
Services provided	Prelude: <ol style="list-style-type: none"> 1. formal high level specification SchedMcore <ol style="list-style-type: none"> 1. schedulability analyser 2. simulation of Prelude programs 3. executive services for multi/many-core
Dependencies / applicability	WP2, WP4, WP6
Relationship to requirements	

12.4.2 PEGASE

ID	B 4.2.1	
Name	Algorithms for bounding delays in Ethernet networks	
Source	PEGASE	
Owner	ONERA	
Description	Algorithms to compute bounds on delay (latency and jitter) and memory usage in a Store-And-Forward network, with sporadic traffics (minimal duration between two emissions and maximal frame size), without cyclic dependencies, with FIFO and Static priority policies. AFDX and a TTEthernet with only RC class are such networks. Formal proofs of algorithms exist.	
Building block type	Algorithms	
Services provided	<ul style="list-style-type: none"> • Bound on network jitter • Bound on network delay • Bound on per-switch memory usage 	
Dependencies / applicability		
Relationship to requirements	R 3.1.3 to R 3.1.5, R 4.1.3	

ID	B 4.2.2	
Name	RTaW-PEGASE tool	
Source	PEGASE	
Owner	RTAW	

Description	Tool that implements the “Algorithms for bounding delays in Ethernet networks”.
Building block type	Software tool
Services provided	Actual computation <ul style="list-style-type: none"> • Bound on network delay • Bound on per-switch memory usage
Dependencies / applicability	
Relationship to requirements	R 3.1.3 to R 3.1.5, R 4.1.3

12.4.3 SCARLETT

ID	B 4.3.1
Name	Reconfigurable IMA platform
Source	SCARLETT
Owner	ONERA
Description	Airbus and ONERA have defined a reconfigurable IMA platform based on specific architecture (including a reconfiguration supervisor). Reconfigurable IMA should be able to change the configuration of the platform by moving applications hosted on a faulty computing module to spare computing modules. The main objective of such an extension is to reduce the cost of unscheduled maintenance and to improve the operational reliability of the aircraft while preserving current safety levels. ONERA and Airbus assessed the safety of the reconfigurable platform: Functional Hazard Analysis (FHA) and Preliminary System Safety Assessment (PSSA) of the Scarlett demonstrator.
Building block type	Architecture, model
Services provided	Concepts for reconfiguration in a spatial and temporal domains
Dependencies / applicability	WP2, WP4, WP6
Relationship to requirements	Related to Resource management requirements

12.4.4 Internal Projects: TTT

ID	B 4.4.1
Name	TTEthernet tool chain
Source	Other
Owner	TTT

Description	<p>The TTEthernet tool chain allows the user to perform the following activities:</p> <ul style="list-style-type: none"> • Model TTEthernet network traffic (messages) to be transmitted between the end-systems, provided in a “Network Description XML” • Calculate a schedule for these messages and generate these into device specifications, provided in “Device Specification XML” • Translate the device specification into device dependent and implementation-specific configurations in human-readable form summarized in “Device Configuration XMLs” • Translate the human-readable device configuration into binary files that can be loaded into the switches and nodes in a TTEthernet network, i.e. “Device Configuration HEX-files”
Building block type	Set of Tools
Services provided	<ol style="list-style-type: none"> 1. TTEPlan: The input to the TTEPlan tool is a network description XML file. This file specifies high-level communication requirements for the system. Examples for these communication requirements are message flows, the network topology, message timing requirements, Virtual Links (VLs), including their IDs, timing and frame sizes. From this input, the TTEPlan tool then automatically creates the network schedules/configuration records 2. TTEBuild: allows converting XML-based device configuration database files into binary configuration images required by the TTE Switches and the TTE End Systems. 3. TTeloLoad: is a Windows application suitable to configure a TTEthernet Switches in a network. It connects to the Management Interface of the switch, and performs a safe unlocking procedure before reprogramming the static configuration memory of the switches.
Dependencies / applicability	WP3, WP4
Relationship to requirements	<ul style="list-style-type: none"> - R 4.4.1 (Tool chain) - R4.4.2 (Continuous data flow through the tool chain) - R 4.5.1 (Configuration file generators)

12.4.5 Internal Projects: TUKL

ID	B 4.5.1
----	---------

Name	Offline scheduler	
Source	Internal Projects:TUKL	
Owner	TUKL	
Description	TUKL has developed a number of offline scheduling algorithms, which are candidates as basis for the ones developed in DREAMS. In particular, this includes a scheduling algorithms constructing scheduling tables for single resource nodes in a distributed systems connected via broadcast medium, and one creating a number of scheduling tables for different modes and transitions among them.	
Building block type	Algorithms	
Services provided	offline scheduling	
Dependencies / applicability	WP2, WP4, WP6	
Relationship to requirements	R4.1.1, R4.1.2	

12.5 Building Blocks for Mixed-Criticality Certification

12.5.1 MultiPARTES

ID	B 5.1.1	
Name	Safety concept for Wind Power based on multi-core and hypervisor	
Source	MultiPARTES	
Owner	ALSTOM, IKL	
Description	The safety concept for a Wind Power use case based on MultiPARTES technology demonstrates the certifiability (with respect to IEC-61508 and ISO-13849) of a platform which is composed of heterogeneous multi-core and managed by an hypervisor	
Building block type	Reference architecture	
Services provided	Safety, Certifiability	
Dependencies / applicability	Wind Power	
Relationship to requirements	R 5.4.1, R 5.4.2, R 5.4.3, R 5.4.4, R 5.4.5, R 5.4.6 , R 5.5.3, R 7.3.1	

ID	B 5.1.2	
Name	XtratuM	
Source	MultiPARTES	
Owner	FENTISS	
Description	XtratuM is a hypervisor designed for real-time safety-critical embedded systems. It provides the essential services of a partitioning kernel, according to the IMA concept, mainly consisting in spatial and temporal isolation for partitions. The isolation is accomplished by means of para-virtualisation technologies.	
Building block type	SW, partitioning kernel	
Services provided	<ul style="list-style-type: none"> • temporal and spatial isolation • virtualisation of • partition management • inter-partition communication • fault management mechanisms • tracing 	
Dependencies / applicability	(none)	
Relationship to requirements	R 1.13.4, R5.3.4, R7.2.4 , R5.3.1, R5.3.2, R5.3.3, R5.4.4	

ID	B 5.1.3	
Name	MultiPARTES methodology	
Source	MultiPARTES	
Owner	IKL	
Description	<ul style="list-style-type: none"> • Safety certification strategy for FSM based on multicore partitioning • Model-driven applied to safety constraints for multicore partitioning • Time Triggered heterogeneous multicore platform • Safety assessment of the hypervisor <p>The MultiPARTES methodology provides the application model, platform model, attributes (criticality, timeliness) modeling tool, hypervisor configuration generation tool and Functional Safety</p>	

	Management (FSM) process. It is described in deliverables D5.1, D5.2 and D5.3.
Building block type	Methodology, Tools
Services provided	Modelling approach
Dependencies / applicability	Cross-domain applicability in mixed-criticality developments
Relationship to requirements	R 5.X.X reusable in all requirements of WG5 R 3.2.1 to R 3.2.9 and R 3.1.2 to R 3.1.6

ID	B 5.1.4	
Name	Certification strategy	
Source	MultiPARTES	
Owner	IKL	
Description	<ul style="list-style-type: none"> • Application model • platform model • attributes (criticality, timeliness) • Modelling tool • configuration generation tool • Functional Safety Management (FSM) process <p>The certification strategy is based on a model-driven approach and consists on defining safety constraints for multi-core partitioning. The safety assessment of the hypervisor is provided so that the hypervisor is seen as a compliant item.</p>	
Building block type	Methodology	
Services provided	Certification, Safety	
Dependencies / applicability	Cross-domain applicability in mixed-criticality developments	
Relationship to requirements	R5.4.2, R 5.5.1, R 5.5.2	

ID	B 5.1.5	
Name	IEC-61508 Certification strategy based on COTS multicore partitioning for safety critical embedded systems	
Source	MultiPARTES	
Owner	IKL, ALSTOM	
Description	<p>A certification strategy has been defined for IEC-61508 compliant safety critical embedded systems, based on COTS and multicore partitioning.</p> <p>This certification strategy has been used for the definition of a Wind Power Safety concept</p>	
Building block type	Reference architecture / Method	
Services provided	Safety, Certifiability	
Dependencies / applicability	Wind Power	
Relationship to requirements	R 5.3.1, R 5.4.5	

12.5.2 Internal Projects:FENTISS

ID	B 5.2.1	
Name	Xoncrete (configuration and scheduling for mono/multi core)	
Source	Internal Projects:FENTISS	
Owner	FENTISS	
Description	<p>Xoncrete is an integrated editor and analysis tool that performs the schedulability analysis of a partitioned system. Rather than a general purpose scheduling tool, Xoncrete has been designed to meet the ARINC 653 system model as the executive platform environment in general, and in particular the XtratuM framework.</p>	
Building block type	SW, modelling	
Services provided	<ul style="list-style-type: none"> • Rich temporal modelling (MARTE-UML based) • Cyclic schedule generation • XM Configuration File (XMCF) generation • Incremental development of the temporal model 	
Dependencies / applicability	XtratuM (B1.19)	
Relationship to requirements	R 1.13.4, R5.3.4, R7.2.4	

12.5.3 TERESA

ID	B 5.3.1	
Name	Dependability and security patterns	
Source	TERESA	
Owner	IKL	
Description	Safety design/implementation patterns that provide dependability and/or security properties (e.g. watchdog, safety communication layer, majority voter, data agreement, clock synchronization, triple modular redundancy, etc.)	
Building block type	Pattern, System, SW, Modelling	
Services provided	Dependability, Security	
Dependencies / applicability	Cross-domain applicability in developments with dependability and/or security requirements	
Relationship to requirements	R5.1.1	

ID	B 5.3.2	
Name	Model-driven pattern based methodology	
Source	TERESA	
Owner	IKL	
Description	The methodology defined in TERESA is a model-driven approach for easy integration of dependability and security patterns along the different development phases for safety railway signaling systems	
Building block type	Methodology, Tools, Modelling	
Services provided	Modelling approach, Dependability, Security	
Dependencies / applicability	Cross-domain applicability in developments with dependability and/or security requirements	
Relationship to requirements	R 5.2.X (all requirements in 5.2 section)	

12.6 Building Blocks for Avionics Demonstrator

12.6.1 Internal Projects:TRT

ID	B 6.1.1	
Name	Interference measurement	
Source	Internal Projects:TRT	
Owner	TRT	
Description	<p>Methodology for the evaluation of interferences on a multi-core platform.</p> <p>The methodology provides a set of stressing benchmarks used to compute the resources required by tasks and estimate degradation of WCET against the task running in isolation.</p> <p>The approach is black box, i.e., doesn't require the source of the tasks, only the binaries.</p>	
Building block type	Methodology, SW (bare-metal libraries)	
Services provided	<p>Timing computation</p> <p>Resources</p>	
Dependencies / applicability	WP2, WP6	
Relationship to requirements	R 4.1.3	

12.7 Building Blocks for Wind-power Demonstrator

12.7.1 Internal Projects: ALSTOM

ID	B 7.1.1	
Name	Use case specifications	
Source	Internal Projects: ALSTOM	
Owner	ALSTOM	
Description	GALILEO real-time platform based on industrial PC	
Building block type	Architectural, HW	
Services provided	Resources	
Dependencies / applicability		
Relationship to requirements	R 7.2.2, R 7.2.3	

12.7.2 MultiPARTES

ID	B 7.2.1
----	---------

Name	Use case Methodology
Source	MultiPARTES
Owner	ALSTOM/IKL
Description	Safety Analysis of a Wind Power Use-Case with Multicore & Virtualization
Building block type	Safety, Architectural
Services provided	Virtualization, Mixed-criticality approach
Dependencies / applicability	
Relationship to requirements	R 7.2.1

12.8 Building Blocks for Healthcare Demonstrator

12.8.1 TRESCCA

ID	B 8.1.1
Name	STNoC firewall
Source	TRESCCA
Owner	ST
Description	The underlying NoC communication infrastructure enforces strong isolation of VM by checking the underlying transactions. What this means is that a potentially compromised Guest OS in a Virtual Machine cannot access data that is tagged by another VM.
Building block type	RTL implementation
Services provided	Enable SoC security
Dependencies / applicability	
Relationship to requirements	This block enables to strong isolation among different VMs

ID	B 8.1.2
Name	Secure Hypervisor
Source	TRESCCA
Owner	VOSYS
Description	A software module or architecture that enables the invocation of secure services from the non-secure world.
Building block type	SW

Services provided	Enable the system security at chip level
Dependencies / applicability	WP2, WP8
Relationship to requirements	R 11.1.1, R 11.2.1, R 11.2.2, R 11.2.3

12.8.2 vIrtical

ID	B 8.2.1	
Name	KVM on ARM	
Source	vIrtical	
Owner	VOSYS	
Description	Open source hypervisor supporting the ARM architectures	
Building block type	SW	
Services provided	Hardware assisted virtualization on ARM	
Dependencies / applicability	WP2, WP8	
Relationship to requirements	R 3.5.1, R 3.5.2, R 3.5.4	

12.9 Building Blocks for Modelling and Development Process

12.9.1 TIMMO-2-USE

ID	B 9.1.1
Name	TADL (Timing Augmented Description Language)
Source	TIMMO-2-USE
Owner	RTAW
Description	<p>The TIMMO-2-USE project has defined a “timing augmented description language” that allows to precisely specify timing constraints, such as latency constraints, recurrence constraints, synchronization constraints and precedence constraints, with the help of so called “timing events” and “timing chains”.</p> <p>Examples of “timing events” are “data d1 has been produced by function fct1”, “frame frm1 has been filled with data and queued for transmission” or “frame frm1 is available in the input port of a gateway”.</p> <p>These events can be linked into (hierarchical) “timing chains” in order to specify cause and effect relationships between these events. This way, a “timing chain” allows to exactly specifying information transformation paths through a system (e.g. sensor data that is transformed into a corresponding command at some actuator).</p>

	<p>The hierarchical structure allows refining a timing chain segment defined at platform independent level, depending on the allocation choices made at platform specific level. For example, depending on the allocation, a timing segment that spans between the production of a data d1 by some function f1 to the consumption of this data by some function f2, could be refined into</p> <ul style="list-style-type: none"> • an “EtherCAT communication sub chain” • an “TTEthernet communication sub chain” • a “local memory” sub chain in the same partition • a “inter-partition data transmission sub chain” <p>It will have to be checked which timing events are relevant in the context of DREAMS how far timing chains need to be decomposed hierarchically. Furthermore, the actually useful degree of decomposition may vary from demonstrator to demonstrator.</p>
Building block type	Meta-Model
Services provided	<ul style="list-style-type: none"> • Definition of event and timing chains • Specification of timing constraints based on timing event and chains <ul style="list-style-type: none"> ○ Latency constraints ○ Recurrence constraints ○ Synchronization constraints ○ Precedence constraints ○ Offset constraints
Dependencies / applicability	
Relationship to requirements	R9.5.1, R9.5.2, R9.5.3, R 4.1.3

ID	B 9.1.2
Name	TIMMO-2-USE Generic Method Pattern
Source	TIMMO-2-USE
Owner	RTAW
Description	<p>The TIMMO-2-USE project has developed a generic method pattern that consists of a generic sequence of design tasks related to timing constraints and this for the automotive domain. It can be instantiated at every phase/level of the development process and for every timing aspect, such as latency constraints, synchronization constraints, etc.</p> <p>The TIMMO-2-USE Generic Method Pattern consists of the six tasks:</p>

	<ul style="list-style-type: none"> • “Create Solution” describes the definition of the architecture without any timing information. • “Attach Timing Requirements to Solution” describes the formulation of timing requirements in terms of the current architecture. • “Create Timing Model” describes the definition of a formalized model for the calculation of specific timing characteristics based on properties of the current architecture. • “Analyze Timing Model” describes the actual execution and evaluation of all necessary “calculations” according to the timing model. • “Verify Solution against Timing Requirements” describes the comparison of the obtained analysis results with the specified timing requirements. • “Specify and Validate Timing Requirements” describes the identification of mandatory timing characteristics and their promotion to timing requirements for the next development phase. <p>The generic method pattern has been applied in TIMMO-2-USE to several to timing aspect that are relevant in the automotive domains. The following are very likely to be also relevant in the context of DREAMS:</p> <ul style="list-style-type: none"> • “Specify timing budgets” • “Specify synchronization timing constraints”
Building block type	Development process (elements)
Services provided	<ul style="list-style-type: none"> • Generic Method Pattern (GMP) • Instantiations of the GMP for several timing aspects relevant in the automotive domain
Dependencies / applicability	R 12.9.1

Relationship to requirements	Contributes to the achievement of the requirements R 9.14.*, with respect to timing.
------------------------------	--

12.9.2 ACROSS

ID	B 9.2.1	
Name	ACROSS tool-chain	
Source	ACROSS	
Owner	FORTISS	
Description	Eclipse-based tool-chain used to design & implement applications for the ACROSS MPSoC	
Building block type	Meta-models, Tooling	
Services provided	<ul style="list-style-type: none"> • Meta-models: <ul style="list-style-type: none"> ○ PIM (generic: KPN, domain-specific: IEC 61131-3 FBD & SFC) ○ PM of ACROSS platform ○ Adapters to PSMs (TTNoC, PikeOS) ○ Extra-functional requirements meta-models (timing, reliability) • Design-space exploration: Mapping, scheduling, instantiation of fault-tolerance mechanisms based on reliability analysis • Code / configuration generator 	
Dependencies / applicability	<ul style="list-style-type: none"> • Dependencies <ul style="list-style-type: none"> ○ Eclipse ○ TTT scheduler and configuration tool to configure the ACROSS MPSoC ○ Sysgo CODEO (to configure PikeOS) • Applicability: Platform-independent modules and algorithms can be re-used directly, others need adaptation 	
Relationship to requirements	Requirements on modeling, development process, and tooling	

12.9.3 RECOMP / ARAMIS

ID	B 9.3.1
Name	AutoFocus 3

Source	RECOMP / ARAMIS
Owner	FORTISS
Description	Eclipse-based tool-chain providing “seamless” model-based development (use of models in all development phases)
Building block type	Meta-model, Tooling
Services provided	<ul style="list-style-type: none"> • Meta-Models <ul style="list-style-type: none"> ○ Requirements ○ PIM: components, state automata ○ PM: Extensible, e.g. generic distributed system, shared-memory multicore platform • Modeling and Simulation • Code generation • Scheduling Synthesis • Formal Verification • Testing
Dependencies / applicability	<p>Dependencies</p> <ul style="list-style-type: none"> • Eclipse • External verification tools (NuSMV, Yices) <p>Applicability:</p> <ul style="list-style-type: none"> • Open Source (http://af3.fortiss.org/) • Platform-independent modules and algorithms can be re-used directly, others need adaptation
Relationship to requirements	Requirements on modeling, development process, and tooling

12.9.4 CESAR

ID	B 9.4.1
Name	Variability on CESAR platform
Source	CESAR
Owner	SINTEF
Description	Tooling for variability modeling. Principles for including tests in the variability models
Building block type	SW (Tool), Principles
Services provided	Combining testing with modelling in one integrated model.
Dependencies / applicability	Depends on building blocks MoSiS-CVL Tool and ICPL Tool. In CESAR it was also integrated partially with the CESAR platform, but this is not a necessary dependence

Relationship to requirements	Relates to requirements on Variability
------------------------------	--

12.9.5 MOSIS

ID	B 9.5.1	
Name	MoSiS CVL Tool	
Source	MOSIS	
Owner	SINTEF	
Description	Prototype tooling for MoSiS CVL. Product Line language.	
Building block type	SW Tool	
Services provided	Definition of variability, production of product model. Some associated techniques for variability analysis	
Dependencies / applicability	<p>Depends on Eclipse.</p> <p>The CVL Tool of MoSiS is open source, but is not maintained any more. It will eventually be replaced by VARIES Tool.</p> <p>See also http://variabilitymodeling.org</p>	
Relationship to requirements	Related to requirements on variability	

12.9.6 VARIES

ID	B 9.6.1	
Name	VARIES Prototype Tool	
Source	VARIES	
Owner	SINTEF	
Description	Variability mangement tool. Tools for testing and analysis of product lines.	
Building block type	SW Tool	
Services provided	Feature model editing. Resolution model. Realization model for production of products.	
Dependencies / applicability	Runs on Eclipse. Work in progress. Will become open source early 2014	
Relationship to requirements	<p>Relates closely to requirements on variability</p> <p>R5.5.1</p>	

12.9.7 VERDE

ID	B 9.7.1	
Name	ICPL	

Source	VERDE
Owner	SINTEF
Description	Prototype tool for testing product lines. Selection of optimal products.
Building block type	SW Tool
Services provided	Based on feature model it selects the most optimal set of resolutions (configurations) to test.
Dependencies / applicability	Eclipse. Works with but are not absolutely dependent on the VARIES Prototype Tool.
Relationship to requirements	Related to requirements on variability and on architectural exploration. R 5.5.1

12.10 Building Blocks for Resource Management

12.10.1 ACTORS

ID	B 10.2.1
Name	Resource management
Source	ACTORS
Owner	TUKL
Description	The resource manager of the ACTORS project provides adaptivity within a single device, based on abstract service levels of CPU availability expressed as serves and application demands and adaptivity as expressed in the data-flow language CAL. The main application area was adaptive MPEG video streaming.
Building block type	algorithms
Services provided	Concepts for resource management on single device for not safety critical, adaptive applications.
Dependencies / applicability	
Relationship to requirements	R10.*

12.11 Building Blocks for Security

12.11.1 OVERSEE

ID	B 11.1.1	
Name	Security services for vehicles based on virtualization and secure communication;	
Source	OVERSEE	
Owner	USIEGEN, UPV	
Description	Security service for open vehicular platform using virtualization and secure communication	
Building block type	HW, SW, Modelling, Hypervisor (XtratuM)	
Services provided	Security services for vehicles such as secure communication, access control for I/O devices, isolating secure I/O partition from other user partitions	
Dependencies / applicability	Security services and virtualization	
Relationship to requirements	R 11.2.1, R11.2.3	

12.11.2 TERESA

ID	B 11.2.1	
Name	Model-driven pattern based approach for dependability and security	
Source	TERESA	
Owner	USIEGEN	
Description	Model-driven pattern based approach for secure software development for resource constraint systems, such as smart meter gateways.	
Building block type	HW, SW, Modelling	
Services provided	Security services such as secure software development, confidentiality, authenticity, integrity of communication. Random number generation. Random number testing.	
Dependencies / applicability	Security services and cryptographic mechanisms. Key Management.	
Relationship to requirements	R 11.6.1, R11.3.7, R 11.3.8	

12.11.3 TRESCCA

ID	B 11.3.1	
Name	Input related to network interface security solutions	
Source	TRESCCA	

Owner	ST, TEI, VOSYS
Description	Protection security blocks and testbenches
Building block type	SystemC, RTL (ongoing)
Services provided	ST and TEI currently extend the Spidergon STNoC network interface with novel hardware security modules, that operate as NoC Firewalls. In order to manage efficiently and securely different types of connected resources corresponding system drivers, software configuration mechanisms and specialized VMs are necessary and are being developed by VOSYS.
Dependencies / applicability	The security technology must be adapted to develop an efficient virtual machine isolation solution in the presence of mixed criticality and safety constraints.
Relationship to requirements	R 11.1.1, R 11.2.2, R 11.2.3, R 11.2.4

ID	B 11.3.2
Name	Secure Hypervisor
Source	TRESCCA
Owner	VOSYS
Description	Strong isolation between VM
Building block type	Software component
Services provided	The realtime part will be protected against attach coming from the non secure and no realtime par of the systems
Dependencies / applicability	The security technology must be adapted to develop an efficient virtual machine isolation solution in the presence of mixed criticality and safety constraints.
Relationship to requirements	R 11.2.1, R 11.2.2, R 11.2.3, R 11.2.4

12.11.4 vIrtical

ID	B 11.4.1
Name	Security input related to I/O virtualization solutions
Source	vIrtical
Owner	ST, TEI, VOSYS
Description	Secure I/O virtualization (customized and ARM-v7 compatible solution)
Building block type	SystemC, RTL (ongoing)
Services provided	ST, TEI and VOSYS have pursued the design of an integrated IOMMU component. This component performs virtual-to-physical

	address translation and related task isolation capabilities by examining the validity of access to specific memory pages.
Dependencies / applicability	IOMMU technology must be adapted to develop a new, more efficient distributed solution that work at the network interface layer in the presence of mixed criticality and safety constraints.
Relationship to requirements	R 11.2.1, R 11.2.2, R 11.2.3, R 11.2.4

12.11.5 MACsec

ID	B 11.5.1	
Name	Data confidentiality, integrity and source authentication for MAC access independent protocols	
Source	IEEE 802.1AE	
Owner	IEEE	
Description	MACsec standard defines a set of protocols for the security of connectionless data transmission, including confidentiality, integrity and source authentication for Ethernet based protocols	
Building block type	IEEE standard	
Services provided	Data confidentiality, integrity and source authentication for connectionless data transmission	
Dependencies / applicability	TTEthernet shall provide support for confidential message exchange, data integrity and source authentication for connectionless data transmission at MAC which can be achieved via an implementation and integration of the MACsec standard	
Relationship to requirements	R 11.3.1, R 11.3.2, R 11.3.3, R 11.3.4, R 11.3.5, R 11.3.6, R 11.3.7, R 11.3.8	

12.11.6 Internal Projects: TTT

ID	B 11.6.1	
Name	TTEthernet Safety and Availability Features	
Source	Other: TTT	
Owner	TTT	
Description	A subset of security features required are potentially covered by the safety concept and related availability measures currently implemented in TTT solutions. Part of the DREAMS project will be the evaluation and assessment of this safety/availability-reuse for security.	

Building block type	Concepts
Services provided	For example, existing features that target availability of the network covers potential denial of service attacks.
Dependencies / applicability	WP3
Relationship to requirements	R 12.11.1, R 12.11.2, R 12.11.3, R 12.11.4

12.11.7 D-MILS

ID	B 11.7.1	
Name	Prototype MACsec implementation	
Source	D-MILS Project	
Owner	TTT	
Description	Within the D-MILS project, TTT has implemented an early prototype of parts of the IEEE 802.1AE set of security protocols in FPGA for analysis purposes.	
Building block type	VHDL code	
Services provided	IEEE 802.1AE (MACsec) specifies the provision of connectionless user data confidentiality, frame data integrity, and data origin authenticity by media access independent protocols. It shall provide the basis for secure communication through networks such as TTEthernet.	
Dependencies / applicability	WP3	
Relationship to requirements	R 12.11.5, R 12.11.6, R 12.11.3, R 12.11.4, R 12.11.5, R 12.11.6, R 12.11.7, R 12.11.8	

12.11.8 IEEE 802.1X

ID	B 11.8.1	
Name	Key management for MACsec	
Source	IEEE 802.1X-2010	
Owner	IEEE	
Description	For establishing and managing secure associations in MACsec, key management needs to be performed.	
Building block type	IEEE standard	
Services provided	Key management Secure connectivity associations Secure associations	

Dependencies / applicability	TTEthernet shall provide support for secure communications using MACsec. MACsec by itself does not provide the key management and building secure associations between the communicating ends. This can be done via IEEE 802.1X-2010
Relationship to requirements	R 11.3.1, R 11.3.2, R 11.3.3, R 12.11.4, R 11.3.5, R 11.3.6, R 11.3.7, R 11.3.8

13 Gaps

Gaps define solutions needed to satisfy the different requirements stated in sections 1 – 11. They are available for all requirement subsections and provide detailed information on the missing services and the relationship to requirements.

13.1 Architecture

ID	G 1.1	
Topic	Architecture	
Name	Time and Space Partitioning for Mixed-Criticality Systems	
Source	USIEGEN	
Description	Methods for time and space partitioning with the boundary conditions of mixed-criticality systems is missing. For example time and space partitioning must support heterogeneous models of computation, different timing models and different underlying technologies (e.g., networks, operating systems).	
Missing Services	<ul style="list-style-type: none"> • Time and space partitioning for computational resources • Time and space partitioning for communication resources 	
Gap Type	HW/SW/Model	
Relationship to requirements	R 1.1.1, R 1.5.1, R 1.10.1 R 1.10.3 R 1.1.3, R 1.13.1	
Relationship to building blocks	B1.1, B1.2	
Responsibility	WP	1
	Deliverable	D1.2.1
	Lead Partner	USIEGEN
	Participating partners	USIEGEN, TTT, ST, TEI, TUKL, UPV, FENTISS, VOSYS

ID	G 1.2	
Topic	Architecture	
Name	Architecture for networked multi-core chips	
Source	USIEGEN	
Description	A mixed-criticality architecture for networked multi-core chips with real-time support, fault isolation, security is missing in the state-of-the art. In addition, the integration of on-chip and off-chip networks with different	

	protocols into a coherent embedded architecture for networked multi-core chips is not available.	
Missing Services	<ul style="list-style-type: none"> • Gateway services between on-chip and off-chip networks with selective redirection of information, fault isolation, name space mapping, reconfiguration support • Gateway services between different off-chip networks • Access to remote virtualized resources and seamless virtualization of on-chip and off-chip resources 	
Gap Type	HW/SW/Model	
Relationship to requirements	R 1.1.3, R 1.1.6, R 1.1.7, R 1.2.1 , R 1.2.2, R 1.2.3, R 1.2.4, R 1.2.6, R 1.3.1, R 1.4.1, R 1.4.2, R 1.4.3, R 1.5.1 ,R 1.6.1, R 1.9.1, R 1.9.2, R 1.10.2 , R 1.13.1	
Relationship to building blocks	B1.1, B1.2	
Responsibility	WP	1
	Deliverable	D1.2.1
	Lead Partner	USIEGEN
	Participating partners	USIEGEN, TTT, ST, TEI, TUKL

ID	G 1.3	
Topic	Architecture	
Name	Resource virtualization	
Source	USIEGEN	
Description	Existing architectures do not address the efficient virtualization of all relevant resources (i.e., communication network, processor cores, memory, I/O) of mixed-criticality applications on networked multi-core chips.	
Missing Services	<ul style="list-style-type: none"> • On-chip communication • Off-chip communication • IO Services • Memory Services • Processor cores 	
Gap Type	SW/HW/Model	
Relationship to requirements	R 1.1.3, R 1.3.1 , R 1.5.1, R 1.11.1 ,R 1.13.4	
Relationship to building blocks	B1.1, B1.2	
Responsibility	WP	1

	Deliverable	D1.2.1
	Lead Partner	USIEGEN
	Participating partners	ST, TTT, VOSYS, UPV

ID	G 1.4	
Topic	Architecture	
Name	Fault-tolerance for mixed-criticality systems	
Source	USIEGEN	
Description	Architectural support for fault-tolerance based on the boundary conditions of mixed-criticality systems (e.g., heterogeneous models of computation, non-deterministic subsystems, and different timing models) is missing.	
Missing Services	<ul style="list-style-type: none"> • Active redundancy for safety-relevant components • Fault recovery strategies based on dynamic reconfiguration 	
Gap Type	HW/SW/Model	
Relationship to requirements	R 1.1.6, R 1.1.7, R 1.4.1 ,R 1.9.3 ,R 1.10.1	
Relationship to building blocks	B1.1, B1.2	
Responsibility	WP	1
	Deliverable	D1.2.1
	Lead Partner	USIEGEN
	Participating partners	USIEGEN, TTT, TUKL, ONERA, TRT

ID	G 1.5	
Topic	Architecture	
Name	Timing guarantees in mixed-criticality systems	
Source	USIEGEN	
Description	Existing architectures do not support different types of timing guarantees (e.g., minimal jitter, bounded delay, best effort) for communication activities, computational activities and resource management for subsystems with different criticality, different timing models and different models of computation.	
Missing Services	<ul style="list-style-type: none"> • Time predictable communication (e.g., deadlines, jitter, throughput) 	

	<ul style="list-style-type: none"> Time predictable execution (e.g., deadlines, jitter, temporal order) Time predictable reconfiguration and resource management (e.g., bounded reconfiguration time) 	
Gap Type	HW/SW/Model	
Relationship to requirements	R 1.2.1, R 1.2.3, R 1.2.6 ,R 1.5.1,R 1.9.3, R 1.10.1, R 1.10.3	
Relationship to building blocks	B1.1, B1.2	
Responsibility	WP	
	Deliverable	D1.2.1
	Lead Partner	
	Participating partners	

ID	G 1.6	
Topic	Architecture	
Name	Fault detection and fault information	
Source	USIEGEN	
Description	Support for fault detection and fault information in networked multi-core chips with mixed-criticality applications is missing.	
Missing Services	<ul style="list-style-type: none"> Fault detection based on generic fault detectors in the architecture and information about faults provided by the application Health monitoring with consistent fault information for the application Provision of fault information for the resource management 	
Gap Type	HW/SW/Model	
Relationship to requirements	R 1.3.1, R 1.3.2, R 1.3.3, R 1.4.3	
Relationship to building blocks	B1.1, B1.2	
Responsibility	WP	1
	Deliverable	D1.2.1
	Lead Partner	USIEGEN
	Participating partners	USIEGEN, TTT, ONERA

ID	G 1.7
----	-------

Topic	Architecture	
Name	Fault recovery strategies	
Source	USIEGEN	
Description	Fault tolerance through recovery strategies and dynamic reconfiguration for networked multicore chips with mixed-criticality applications is missing.	
Missing Services	<ul style="list-style-type: none"> • Reconfiguration service • Fault detection • Real-time fault recovery service 	
Gap Type	HW/SW/Model	
Relationship to requirements	R 1.4.3, R 1.13.3	
Relationship to building blocks	B1.1, B1.2	
Responsibility	WP	1
	Deliverable	D1.2.1
	Lead Partner	USIEGEN
	Participating partners	USIEGEN, ONERA, TRT, TUKL

ID	G 1.8	
Topic	Architecture	
Name	Domain-independent core architectural services	
Source	USIEGEN	
Description	A domain-independent definition of core architectural services for networked multi-core chips and mixed-criticality systems is missing, where core architectural services abstract from the underlying technology (e.g., different protocols for communication) and support the modular refinement through higher architectural services.	
Missing Services	<ul style="list-style-type: none"> • Global time base service • Communication service • Execution service • Resource management service 	
Gap Type	HW/SW/Model	
Relationship to requirements	R 1.5.1, R 1.5.2, R 1.7.1, R 1.10.1, R 1.10.2, R 1.10.2, R 1.13.1, R 1.13.3, R 1.13.4	
Relationship to building blocks	B1.1, B1.2	

Responsibility	WP	1
	Deliverable	D1.2.1
	Lead Partner	USIEGEN
	Participating partners	USIEGEN,TUKL, TTT, ST, TEI

ID	G 1.9	
Topic	Architecture	
Name	Energy and power efficiency	
Source	USIEGEN	
Description	Today's architectures do not support energy and power efficiency in mixed-criticality applications.	
Missing Services	Resource monitors for energy	
Gap Type	HW/SW/Model	
Relationship to requirements	R 1.12.1	
Relationship to building blocks	B1.1, B1.2	
Responsibility	WP	1
	Deliverable	D1.2.1
	Lead Partner	ST
	Participating partners	ST, FORTISS

13.2 Multicore Virtualization Technology

ID	G 2.1
Topic	Virtualization-Chip
Name	On-chip monitoring and dynamic configuration of virtualized resources
Source	TRT
Description	The on-chip network shall ensure performance isolation via time and/or space partitioning based on a priori knowledge of the permitted behavior of cores.
Missing Services	On-chip network supporting time triggered message delivery has to be developed within the project

Gap Type	HW/ Model	
Relationship to requirements	R 1.12.1 R2.7.1 R 2.8.1 R 2.8.3	
Relationship to building blocks		
Responsibility	WP	WP2, WP3
	Deliverable	
	Lead Partner	USIEGEN
	Participating partners	VOSYS, UPV, TEI, ST

ID	G 2.2	
Topic	Virtualization-Chip	
Name	Memory subsystem	
Source	ST	
Description	The architecture shall support memory interleaving	
Missing Services	Memory interleaving algorithm integrated to on-chip network to balance automatically the Network load over several DDR sub-systems	
Gap Type	HW/ Model	
Relationship to requirements	R 1.12.1	
Relationship to building blocks		
Responsibility	WP	WP2
	Deliverable	
	Lead Partner	ST
	Participating partners	TEI, ST

ID	G 2.3	
Topic	Virtualization-Chip	
Name	Cache Memory Resource Efficiency	
Source	T2.1 (DOW)	
Description	Partitioning mechanisms for a shared L2 Cache shall be provided	
Missing Services	No HW service allows today at platform level to manage L2 caches. L2 management is performed in SW resulting in performance drop.	

Gap Type	HW/ Model	
Relationship to requirements	R 2.3.2	
Relationship to building blocks		
Responsibility	WP	WP2
	Deliverable	
	Lead Partner	ST
	Participating partners	TEI, ST, VOSYS

ID	G 2.4	
Topic	Virtualization-Chip	
Name	On-chip monitoring and dynamic configuration of virtualized resources	
Source	TRT	
Description	MON, LRS and LRM shall be implemented at chip level	
Missing Services	MON, LRS and LRM chip-level implementation is not fully included in any of the BBs. Part of it is covered by ACTORS (single-device monitoring), but this is mostly a gap to be addressed in DREAMS	
Gap Type	HW/ Model	
Relationship to requirements	R 2.4.1, R 10.2.1 (Local Resource Monitoring), R 10.3.1 (Local Resource Managers (LRM)), R10.3.2 (Local Resource Scheduling (LRS))	
Relationship to building blocks	B 10.3.1 (ACTORS)	
Responsibility	WP	WP2
	Deliverable	
	Lead Partner	TRT
	Participating partners	TUKL, ONERA, UPV, FENTISS, VOSYS, ST(supporting role)

ID	G 2.5	
Topic	Virtualization Chip	
Subtopic	Multicore Virtualization	
Name	Support for STNoC network interface virtualization extensions at hypervisor level	
Source	VOSYS	

Description	The hypervisor will need to exploit the novel virtualization extensions supported by NoC the interface layer. In case of KVM, this gap will fill the implementation differences between a standard network interface and the STNoC specific features and provide a standardised interface to guest operating systems.	
Missing Services	Hypervisor implementation for using virtualization and memory interleaving extensions developed at Network Interface (NI) levels, in order to expose virtual network interfaces for the guest systems. This gap will be validated using the most appropriate hardware platform.	
Gap Type	SW	
Relationship to requirements	R 2.5.1	
Relationship to building blocks	B 3.25	
Responsibility	WP	WP2
	Lead Partner	ST
	Participating partners	ST, TEI, VOSYS

ID	G 2.6	
Topic	Virtualization Chip	
Subtopic	Multicore Virtualization	
Name	Distributed IOMMU Virtualization	
Source	VOSYS	
Description	A software and/or hardware mechanism will be needed to bridge the gap between multiple and distributed IOMMU instances and present a unified view to the guest systems.	
Missing Services	An IOMMU provides address translation for I/O devices. In case of distributed IOMMUs, this address translation support has been extended for the whole set of systems. This gap will be validated using the most appropriate hardware platform.	
Gap Type	SW, HW	
Relationship to requirements	R 2.5.5	
Relationship to building blocks	B 3.25	
Responsibility	WP	WP2
	Lead Partner	VOSYS/TEI
	Participating partners	ST

ID	G 2.7	
Topic	Virtualization-Chip	
Name	On chip time and space partitioning	
Source	USIEGEN	
Description	The on-chip network shall ensure performance isolation via time and/or space partitioning based on a priori knowledge of the permitted behavior of cores.	
Missing Services	On-chip network supporting time triggered message delivery has to be developed within the project	
Gap Type	HW/ Model	
Relationship to requirements	R 1.12.1	
Relationship to building blocks		
Responsibility	WP	WP2, WP3
	Deliverable	
	Lead Partner	USIEGEN
	Participating partners	VOSYS, UPV, TEI, ST

13.3 Mixed-criticality Network

ID	G3.1	
Topic	Gateways	
Name	STNOC OnChip – Off-chip Gateway	
Source	TTT	
Description	In order to provide a deterministic bridge between two on-chip networks, a gateway must be implemented that transfers messages from one NOC to another NOC.	
Missing Services	Transparent Gateway Service to bridge communication from one chip to another chip transparently through a TTEthernet network	
Gap Type	SW and/or HW implementation	
Relationship to requirements	R 13.3.1	
Relationship to building blocks	B 3.1.3	
Responsibility	WP	3

	Deliverable	D3.1.1, D3.1.2, D3.1.3
	Lead partner	TTT
	Participating partners	ST / TEI

ID	G3.2	
Topic	Gateways	
Name	Deterministic Wireless Gateway	
Source	TTT	
Description	In order to connect wireless devices deterministically to the network, a gateway must be implemented. This requires a set of features currently present in TTEthernet to be translated to a wireless channel in order to provide transparent or at least compatible operation.	
Missing Services	Wireless Gateway Service: <ul style="list-style-type: none"> - Wireless clock synchronisation with TTEthernet clock - Deterministic Wireless RX and TX - End system implementation - Switch / Access Point implementation 	
Gap Type	HW + SW implementation	
Relationship to requirements		
Relationship to building blocks		
Responsibility	WP	3
	Deliverable	D3.1.1, D3.1.2, D3.1.3
	Lead partner	TTT
	Participating partners	

ID	G3.3	
Topic	Networking	
Name	Safety and Fault Handling	
Source	IKL	
Description	Mechanisms for Safety Assurance according to IEC 61508 for mixed criticality networks	
Missing Services	<ul style="list-style-type: none"> • Definition of safety certification of mixed-criticality network 	

	<ul style="list-style-type: none"> Implementation of the SCL 	
Gap Type	Safety, Methodology	
Relationship to requirements	R 3.2.1 to R 3.2.9 and R 3.1.2 to R 3.1.6	
Relationship to building blocks	B 5.1.3	
Responsibility	WP	3, 5
	Deliverable	D3.3.1, D3.3.2, D3.3.3
	Lead partner	IKL
	Participating partners	TTT

13.4 Tooling, Scheduling and Analysis

ID	G 4.1	
Topic	Tools & Scheduling	
Name	Design-space exploration	
Source	FORTISS	
Description	<p>The design space exploration (DSE) shall allow for semi-automatic architectural exploration of DREAMS-based systems.</p> <p>The DSE used in the ACROSS tool-chain uses genetic algorithms to optimize ACROSS-based designs w.r.t. to temporal and reliability objectives based on the rating of the design candidates by different analyses. The DSE provided by AutoFocus3 combines a tree-based search procedure with an SMT-based offline scheduler.</p> <p>The SINTEF product line testing engine shall be enhanced to find optimal or almost-optimal configurations based on the explicit variability descriptions. We foresee up-front processing and analysis combined with direct adaptation at runtime among preselected configurations. The SINTEF product line testing engine is a fast engine for advanced covering array analysis.</p>	
Missing Services	<p>In DREAMS, the methods need to be extended to consider the following additional concerns:</p> <ul style="list-style-type: none"> DREAMS architecture Integration of Energy consumption analysis (see R 9.7.1) Reliability analysis considering physical separation-constraints such as the ones implied by IEC61508-2, Annex E ("ON-Chip Redundancy"). 	

	<ul style="list-style-type: none"> • Code / configuration generator for tool-supported implementation of the fault-tolerant deployments (special case of G 4.3) • Optimal configuration analysis (off-line and on-line) based on product line analysis 	
Gap Type	Tooling	
Relationship to requirements	R 4.1.4, R 9.11.3	
Relationship to building blocks	B 9.2, B 9.3	
Responsibility	WP	4
	Deliverable	D4.1.2, D4.1.3
	Lead partner	FORTISS
	Participating partners	FORTISS, SINTEF

ID	G 4.2	
Topic	Tools & Scheduling	
Name	Continuous data flow through the tool chain	
Source	RTaW	
Description	The exchange of data between consecutive tools in the DREAMS development process should be automated so that it can be performed without “manual” recopying or reworking of the data. Here, a public meta-model API such as the one provided by the ACROSS tool-chain or AutoFocus3 is the prerequisite for the implementation of automatic model-to-model-transformations.	
Missing Services	For each tool of the tool-chain, the provider of the tool needs to implement a model-to-model transformation, for importing the needed input data from a DREAMS model into the tool and for exporting the resulting output data back to a DREAMS model. Some tools may only need unidirectional data exchange.	
Gap Type	Tooling	
Relationship to requirements	R 4.4.2	
Relationship to building blocks	B 9.2, B 9.3	
Responsibility	WP	1, 4
	Deliverable	D1.5.1, D1.7.1, D4.4.2
	Lead partner	RTaW
	Participating partners	RTAW, FORTISS, IKL, TTT, SINTEF, ONERA, UPV

ID	G 4.3	
Topic	Tools & Scheduling	
Name	Configuration file generators	
Source	RTaW	
Description	The code / configuration generator (model-to-text-transformations based on the EMF technology) used in the ACROSS-tool-chain can be used as the basis for the generation of configuration files for DREAMS platform services.	
Missing Services	<p>For each configurable service of the DREAMS platform, the actual implementation of a configuration file generator is required. This comprises in particular:</p> <ul style="list-style-type: none"> • configuration file for the execution model based on XtratuM • configuration files of TTEthernet based DREAMS clusters 	
Gap Type	Tooling	
Relationship to requirements	R 4.5.1	
Relationship to building blocks	B 9.2	
Responsibility	WP	4
	Deliverable	D4.2.1, D4.2.2
	Lead partner	RTaW
	Participating partners	RTaW, FORTISS, IKL, TTT, SINTEF, ONERA, UPV

ID	G 4. 4	
Topic	Tools & Scheduling	
Name	Offline Schedulers	
Source	TUKL	
Description	Offline schedulers for mixed criticality systems	
Missing Services	<ul style="list-style-type: none"> • Extension of existing schedulers to deal with mixed criticality • Decomposition of global constraints into local constraints • Allocation of functional parts to partition • Generation of static real-time scheduling plans for partition management • Analysis of end-to-end latency constraints (from sensor to actuator). 	

Gap Type	SW	
Relationship to requirements	R4.1.1, R4.1.2, R4.1.3, R4.1.5	
Relationship to building blocks	B 4.5.1, B4.2.1, B4.2.2	
Responsibility	WP	4
	Deliverable	D4.1.2, D4.1.3
	Lead partner	RTaW
	Participating partners	RTaW, FORTISS, IKL, TTT, SINTEF, ONERA, UPV, TUKL

ID	G 4.5	
Topic	Tools & Scheduling	
Name	Explicit Variability Management in DREAMS platform	
Source	SINTEF	
Description	<p>The explicit variability well integrated with the implicit variability given in other ways than the variability met model</p> <p>This relates also to other gaps G4.* and G9.*</p>	
Missing Services	<ul style="list-style-type: none"> Integration of the generic variability management tooling and the product line analysis tooling with the DREAMS tooling platform Extension of the generic transformation defining the CVL execution. 	
Gap Type	Tooling	
Relationship to requirements	R 4.2.1, R4.2.2, R5.1.1, R9.9.x, R9.15.1	
Relationship to building blocks	B 9.6.1, B9.7.1	
Responsibility	WP	4,5
	Deliverable	D4.*.*, D5.5.*
	Lead partner	SINTEF
	Participating partners	SINTEF, IKL, ++

ID	G 4.6	
Topic	Tools & Scheduling	
Name	Platform service configuration file formats	

Source	RTaW	
Description	Based on the DREMAS meta model, the model-to-text generation framework from the ACROSS-tool-chain will be used to generate configuration files of platform services for a specific instance of a DREAMS architecture.	
Missing Services	For each configurable service of the DREAMS platform, the specification of the configuration file format needs to be defined.	
Gap Type	Tooling	
Relationship to requirements	R 4.5.1	
Relationship to building blocks	B 9.2	
Responsibility	WP	4
	Deliverable	D4.2.1, D4.2.2
	Lead partner	RTaW
	Participating partners	RTaW, FORTISS, IKL, TTT, SINTEF, ONERA, UPV, USIEGEN

13.5 Mixed-Criticality Certification

ID	G 5.1	
Topic	Certification V&V	
Name	Building Blocks for the Simulation of the DREAMS architecture	
Source	USIEGEN	
Description	A simulation framework of multicore chips based on mixed criticality levels and the core architectural services of DREAMS are missing.	
Missing Services	<ul style="list-style-type: none"> Gateway simulation building block for on-chip off-chip communication. Simulation building blocks for network interfaces of processor cores to the on-chip network Application component simulation building block that uses the architectural services of DREAMS (e.g., communication) Execution platform environment simulation building block (e.g., time triggered dispatcher) Sink simulation building block 	

	<ul style="list-style-type: none"> Multi-source simulation building block that supports traffic from multiple application subsystems and with multiple traffic types to be sent at one communication interface 	
Gap Type	SW/Model	
Relationship to requirements	R 5.6.1, R 5.6.2, R 5.6.3, R 5.6.4, R 5.6.5, R 5.6.6 and R 5.6.11	
Relationship to building blocks	B5.X	
Responsibility	WP	5,2
	Deliverable	D2.1.1, D5.2.1, D5.2.2
	Lead Partner	USIEGEN
	partner	IKL,RTAW,ST,TEI

ID	G 5. 2	
Topic	Certification V&V	
Name	Simulation environment as part of the tool chain	
Source	USIEGEN	
Description	<p>There is no simulation framework that is integrated with a model-driven development process and corresponding tools for mixed-criticality systems.</p> <p>The testbed should be part of the tool chain and the configuration information should be generated by the tools from WP4.</p>	
Missing Services	<ul style="list-style-type: none"> Specification of input/output formats of simulation environment Tool for generation of input models for simulation 	
Gap Type	SW/Model	
Relationship to requirements	R 5.6.6, R5.6.10	
Relationship to building blocks		
Responsibility	WP	5
	Deliverable	D5.2.1, D5.2.2
	Lead Partner	RTAW
	partner	TTT,IKL, USIEGEN

ID	G 5.3	
Topic	Certification V&V	
Name	Simulation building Block for Fault injection	
Source	USIEGEN	
Description	In simulation environments for networked multi-core chips with the DREAMS core architectural services and mixed-criticality applications, fault injection mechanisms are missing.	
Missing Services	<ul style="list-style-type: none"> • The fault model of IEC61508-2 is relevant for the fault injection, fault containment and the sinks for the observation of fault effects. • Configuration interface of fault injection mechanisms • fault injection at simulated on-chip network interface • fault injection in a simulated on-chip partition • fault injection in execution platform environment (e.g., hypervisor) • fault injection within a complete simulated node of the cluster • fault injection at communication networks (bus, links and Switches, etc..) 	
Gap Type	SW/Model	
Relationship to requirements	R 5.6.7, R 5.6.8 and R 5.6.9	
Relationship to building blocks		
Responsibility	WP	5
	Deliverable	D5.2.3
	Lead Partner	USIEGEN
	partner	RTAW,FENTISS,ST,TT,IKL

ID	G 5.4
Topic	Mixed-Criticality Certification
Name	Modeling variability and product line features to support the certification of mixed-criticality product lines
Source	SINTEF

Description	Model the variability and product line features as defined in the certification strategy for mixed-criticality product lines (e.g. parametrizable compliant items) SINTEF provides the product line technology foundation for mixed criticality systems.	
Missing Services	<ul style="list-style-type: none"> - Manage variability and product line features in order to support and simplify the effort required to certify new configurations and possibilities. - TUV will review the approach to assess certification. - Shall be analyzed the implications to certify the Wind Power Demonstrator. 	
Gap Type	Method / Guidelines	
Relationship to requirements	R 4.2.1, R4.2.2, R5.1.1, R9.9.x, R9.15.1	
Relationship to building blocks	B 9.6.1, B9.7.1	
Responsibility	WP	5
	Deliverable	D4.*.*, D5.5.*
	Lead partner	SINTEF
	Participating partners	TUV, IKL, ALSTOM, FENTISS

ID	G 5.5	
Topic	Mixed-Criticality Certification	
Name	Modular Safety Case	
Source	IKL	
Description	The already existent Safety Concept of the Wind Power demonstrator (from FP7 MULTIPARTES) will serve as starting input for the wind power demonstrator, generic heterogeneous multicore and hypervisor.	
Missing Services	Adaptation to DREAMS architecture	
Gap Type	Reference architecture	
Relationship to requirements	R 7.3.1, R 5.4.1, R 5.4.2, R 5.4.3, R 5.4.4, R 5.4.5, R 5.4.6	
Relationship to building blocks	B 5.1.1, B 5.1.5, B 5.1.3, B 5.1.4	
Responsibility	WP	5
	Deliverable	D5.1.X
	Lead partner	IKL
	Participating partners	IKL, ALSTOM

ID	G 5.6	
Topic	Cross-Domain Mixed-Criticality Patterns	
Name	Definition of V&V strategy	
Source	IKL	
Description	<p>A set of cross domain mixed-criticality patterns shall be collected and described, these patterns will encompass common occurring design solutions.</p> <p>Then a Verification and Validation strategy will have to be defined to be applied on these patterns, the way that once the pattern is validated all the implementations arising from it are pre-validated already.</p>	
Missing Services	Adaptation of patterns to DREAMS architecture.	
Gap Type	Dependability patterns/ Deliverable	
Relationship to requirements	R 5.1.1, R 5.3.4	
Relationship to building blocks	B 5.3.1	
Responsibility	WP	5
	Deliverable	D 5.3.1
	Lead partner	IKL
	Participating partners	TÜV, ALSTOM, FENTISS, UPV

ID	G 5.7	
Topic	Modeling	
Name	Reliability/Safety Meta-Model	
Source	IKL	
Description	<p>Model-driven pattern based Methodology that aims to define reusable common safety solutions to common safety problems that can be or could be found in the development of mixed-criticality systems. For example, an I/O server partition could be used by a single safety partition to manage all shared I/Os of a multicore processor. The Modeling of such a service can be high level or low level as a technical solution. The selection of design-patterns will consider proposals from TÜV Rheinland and IK4-Ikerlan.</p>	
Missing Services	Not applicable, the representation of design-pattern provides a modeled representation of a reusable safety solution.	
Gap Type	Design of dependability Patterns	
Relationship to requirements	R 9.6.1, R 9.6.2	

Relationship to building blocks	B 5.3.2	
Responsibility	WP	5
	Deliverable	D 1.4.1, D 1.6.1
	Lead partner	IKL
	Participating partners	FORTISS, TÜV, VOSYS

ID	G 5.8	
Topic	Mixed-Criticality Certification	
Name	Modular safety-case for Hypervisor certification	
Source	IKL	
Description	Requirements from WP1, WP2, and WP3 should build the modular safety case.	
Missing Services	Definition of safety cases based on the requirements defined in WP1 (T 1.1), WP2 (T 2.3), and WP3 (T 3.1 to T 3.3).	
Gap Type	Safety-case / Deliverable	
Relationship to requirements	R5.4.1, R 5.4.2, R 5.4.4, R 5.5.1	
Relationship to building blocks	B 5.1.1, B 5.1.2, B 5.1.3, B 5.1.5	
Responsibility	WP	5
	Deliverable	D 5.1.1
	Lead partner	IKL
	Participating partners	TUV, FENTISS, ALSTOM

ID	G 5.9	
Topic	Mixed-Criticality Certification	
Name	Modular safety-case for selected COTS multicore processor	
Source	IKL	
Description	Requirements from WP1, WP2 and WP3 should build the modular safety case.	
Missing Services	Definition of safety cases based on the requirements defined in WP1 (T 1.1), WP2 (T 2.3), and WP3 (T 3.1 to T 3.3).	
Gap Type	Safety-case / Deliverable	
Relationship to requirements	R 5.4.1, R 5.4.2, R 5.4.3, R5.4.5, R 5.5.1	
Relationship to building blocks	B 5.1.1, B 5.1.3, B 5.1.5	
Responsibility	WP	5

	Deliverable	D 5.1.2
	Lead partner	IKL
	Participating partners	TUV, FENTISS, ALSTOM
ID	G 5.10	
Topic	Mixed-Criticality Certification	
Name	Modular safety-case for selected mixed-criticality network	
Source	IKL	
Description	Requirements from WP1, WP2 and WP3 should build the modular safety case.	
Missing Services	Definition of safety cases based on the requirements defined in WP1 (T 1.1), WP2 (T 2.3), and WP3 (T 3.1 to T 3.3).	
Gap Type	Safety-case / Deliverable	
Relationship to requirements	R5.4.1, R 5.4.2, R 5.4.3, R 5.4.6, R 5.4.7, R 5.4.8, R 5.5.1	
Relationship to building blocks	B 5.1.1, B 5.1.2, B 5.1.3, B 5.1.5	
Responsibility	WP	5
	Deliverable	D 5.1.3
	Lead partner	IKL
	Participating partners	TUV, FENTISS, ALSTOM

ID	G 5.11	
Topic	Certification V&V	
Name	Tool integration in industrial (safety) engineering process	
Source	RTaW	
Description	<p>In order to foster the usage of tools in an industrial safety engineering process, the following aspects need to be defined for each tool:</p> <ul style="list-style-type: none"> • Phase where the tool is used (system architecture, software architecture, etc.) • Actions to be performed before and after each usage of a tool. • Contribution to the overall safety engineering process • Inputs/Outputs with respect to the safety engineering process • Actor(s) to use it (architect engineer, design engineer, test engineer, certification authority, etc.). 	

	<ul style="list-style-type: none"> Relationship with other available COTS tools (compatible, replacement, complement, extension, etc.) <p>The IKL SIL3 certified Functional Safety Management (FSM) will be used as reference safety engineering process.</p>	
Missing Services	Application to the tools of the DREAMS tool chain.	
Gap Type	Method / Guidelines	
Relationship to requirements	R 5.7.1	
Relationship to building blocks		
Responsibility	WP	5
	Deliverable	D5.4.X
	Lead partner	RTaW
	Participating partners	IKL, SINTEF, TUV

ID	G 5.12	
Topic	Certification V&V	
Name	Transfer test results from simulation to physical system	
Source	TTT	
Description	DREAMS shall develop a process describing how tests executed in simulation can be transferred and tested on a real physical target.	
Missing Services	Establish a framework for the transfer of test case results from the simulation environment to physical environment. Thereby, TTT will improve the test reusability to utilize this framework on the components of WP3.	
Gap Type	Method / Guidelines	
Relationship to requirements	R 5.6.12	
Relationship to building blocks	B 5.1.3	
Responsibility	WP	3,5
	Deliverable	D5.2.2
	Lead partner	TTT
	Participating partners	USIEGEN, RTAW

ID	G 5.13	
Topic	Certification V&V	
Name	Support for formal verification	

Source	ST	
Description	The functional verification shall be implemented through a coverage-driven approach based on dynamic and static methodologies. In dynamic context automatic e-coded checkers shall be used to test functionalities while both code and functional coverage are used to verify the random stimuli generation. This methodology shall be applied to black-box functionalities. In static (or formal) context, white-box assertions shall be used in order to target sub-modules functionalities that cannot be simply addressed at top level. Same assertions should then be activated during dynamic simulation.	
Missing Services	Development a formal verification framework via a process algebra language for modeling and a model checking tool for verification of temporal logic properties of the DREAMS architecture. Will be validated in the STNoC technology.	
Gap Type	Method / Guidelines	
Relationship to requirements	R 5.6.13	
Relationship to building blocks	B 5.1.3	
Responsibility	WP	5
	Deliverable	D5.2.2
	Lead partner	ST
	Participating partners	TEI

13.6 Modeling and Development Process

ID	G 9.1	
Topic	Modeling	
Name	Meta-Model Architecture and Technology	
Source	FORTISS	
Description	<p>The overall architecture of the DREAMS meta-model and the implementation technology used to provide appropriate tool-support should satisfy the following properties:</p> <ul style="list-style-type: none"> • Separation of concerns • Extensibility • Adequate degree of abstraction • Domain-independence <p>The meta-models provided by both building blocks both directly satisfy the requirements “separation of concerns”, “extensibility”, and “domain-independence”. For both meta-models,</p>	

	implementations based on the Eclipse Modeling Framework (EMF) are available.	
Missing Services	<p>The requirement “adequate degree of abstraction” can be addressed best by a combination of both approaches:</p> <ul style="list-style-type: none"> • AutoFocus3 provides meta-models to specify requirements, meta-models for the logical and technical architecture (corresponding to the PIM and the PM of the system under design), the behavioral specification of systems, as well as an extensible annotation meta-model. Its main focus is the platform-independent specification of applications. • In addition to domain-specific and generic application meta-models, the ACROSS tool-chain provides fine-grained meta-models to specific the execution platform (PM), as well as platform-specific representation of applications (PSMs). 	
Gap Type	Meta-Model	
Relationship to requirements	R 9.1.1, R 9.1.2, R 9.1.3	
Relationship to building blocks	B 9.2, B 9.3	
Responsibility	WP	1
	Deliverable	D 1.4.1, D 1.6.1, D 1.8.1, D 1.8.2
	Lead partner	FORTISS
	Participating partners	FORTISS, SINTEF, RTaW, ALSTOM, IKL, TRT, TUKL

ID	G 9.2	
Topic	Modeling	
Name	DREAMS cross-domain application meta-model	
Source	FORTISS	
Description	<p>The DREAMS application meta-model should provide means for the cross-domain, platform-independent specification of applications. Hence, it should support the description of the application architecture, provide a strict execution semantics as well as means to describe the dependency of an application onto the resources of the execution platform (in particular: memory requirements).</p> <p>The meta-models provided by both building blocks provide component models with well-defined execution semantics,</p>	

	covering the first two sub-requirements. Here, AutoFocus3 follows a more formal approach, providing behavioural models of different degrees of abstraction, such as message sequence charts, or automata.	
Missing Services	The mechanisms to describe the dependency of applications onto the resources of the platform used in the meta-model of the ACROSS tool-chain need to be integrated into the AutoFocus3 meta-model.	
Gap Type	Meta-Model	
Relationship to requirements	R 9.2.1, R 9.2.2, R 9.2.3,	
Relationship to building blocks	B 9.2, B 9.3	
Responsibility	WP	1
	Deliverable	D1.4.1
	Lead partner	FORTISS
	Participating partners	FORTISS, SINTEF, TRT, IKL

ID	G 9.3	
Topic	Modeling	
Name	DREAMS platform meta-model	
Source	FORTISS	
Description	<p>The platform meta-model should provide means to capture the different component types of the DREAMS platform, as well as the topology and the hierarchic structure of its instances.</p> <p>The technical architecture AutoFocus3 is a component-based, typically flat platform meta-model that provides a coarse categorization of resources. The platform meta-model of the ACROSS platform provides a more fine-grained approach that covering the components of the ACROSS architecture.</p>	
Missing Services	<ul style="list-style-type: none"> In DREAMS, the AutoFocus3 technical architecture needs to be extended to a more fine-grained hierarchical platform-meta model (using concepts and parts of the platform-meta model used in the ACROSS tool-chain). Support for components specific to the DREAMS platform need to be added (e.g., off-chip network). 	
Gap Type	Meta-Model	
Relationship to requirements	R 9.3.1, R 9.3.2	
Relationship to building blocks	B 9.2, B 9.3	

Responsibility	WP	1
	Deliverable	D1.4.1
	Lead partner	FORTISS
	Participating partners	USIEGEN, FORTISS, SINTEF, IKL

ID	G 9.4	
Topic	Modeling	
Name	DREAMS platform specific meta-model	
Source	FORTISS	
Description	<p>The platform-specific meta-model should provide means to describe applications that have been deployed to instances of the DREAMS platform.</p> <p>Both building blocks provide deployment meta-models that provide means to specify the mapping of tasks to execution units, and platform-independent channels to platform-level messages (both focus on time-triggered systems).</p>	
Missing Services	For DREAMS, the platform-specific meta-model will be generalized, in order to take into account the particular properties of the DREAMS platform, such as HW and SW separation mechanisms, security mechanism, etc.	
Gap Type	Meta-Model	
Relationship to requirements	R 9.4.1	
Relationship to building blocks	B 9.2, B 9.3	
Responsibility	WP	1
	Deliverable	D1.6.1
	Lead partner	FORTISS
	Participating partners	FORTISS, TTT, ONERA, TUKL, IKL, UPV, SINTEF, RTaW

ID	G 9.5	
Topic	Modeling	
Name	Timing constraints	
Source	RTaW	
Description	The DREAMS meta-model should provide cross-domain applicable means for describing timing constraints.	

Missing Services	In order to achieve “cross-domain applicability”, existing (possibly domain specific) descriptions of timing constraints need to be adapted or extended.	
Gap Type	Meta-Model	
Relationship to requirements	R 9.12.3, R 9.5.1, R 9.5.2, R 9.5.3, R 9.5.4	
Relationship to building blocks	B 9.1.1	
Responsibility	WP	1
	Deliverable	D 1.4.1, D1.6.1
	Lead partner	RTaW
	Participating partners	RTaW, TUKL

ID	G 9.6	
Topic	Modeling	
Name	Traceability	
Source	FORTISS	
Description	<p>The meta-models should support the traceability between the artefacts used in the different steps of the development process.</p> <p>AutoFocus3 is based on a “seamless” integrated model-based development approach, providing meta-models for artefacts of different steps in the development process, and a bidirectional traceability at all levels. Requirements specification can be traced to elements of the logical architecture, which in turn are mapped to the technical architecture during the deployment phase.</p>	
Missing Services	In DREAMS, the tracing mechanism will be extended to the hierarchical platform meta-model.	
Gap Type	Meta-Model	
Relationship to requirements	R 9.6.3	
Relationship to building blocks	B 9.3	
Responsibility	WP	1
	Deliverable	D1.4.1, D1.6.1
	Lead partner	FORTISS
	Participating partners	FORTISS, SINTEF, RTaW, IKL

ID	G 9.7
----	-------

Topic	Modeling	
Name	System-level energy / Power requirements meta-model	
Source	FORTISS	
Description	The Energy / Power requirements meta-model should be suitable to define requirements on the energy / power consumption of DREAMS system at the system-level.	
Missing Services	The AutoFocus3 annotation meta-model will be used to provide an energy / power requirements meta-model. It will be based on the DSE objective meta-model used in the ACROSS platform.	
Gap Type	Meta-Model	
Relationship to requirements	R 13.69.7.2	
Relationship to building blocks	B 9.2, B 9.3	
Responsibility	WP	1
	Deliverable	D1.4.1
	Lead partner	FORTISS
	Participating partners	FORTISS, ST,TEI

ID	G 9.8	
Topic	Development Process	
Name	Description of model-to-model / model-to-text transformations	
Source	FORTISS	
Description	<p>One part of the definition of the development process is the description of the required transformation steps and implementation artifacts.</p> <p>This includes both model-to-model transformations (e.g., the ones required to obtain of fault-tolerant design variants based on DSE-results) and model-to-text transformations (e.g., code, configuration).</p> <p>Here, the relevant part of the development process description for the ACROSS platform will be used as a starting point.</p>	
Missing Services	A description of the transformations required for DREAMS-based applications needs be provided.	
Gap Type	Development Process	
Relationship to requirements	R 9.1.1, R 9.1.2, R 9.1.3, R 9.10.1, R 9.10.2, R 9.11.3	
Relationship to building blocks	B 9.2	
Responsibility	WP	1, 4

	Deliverable	D 1.3.1
	Lead partner	FORTISS
	Participating partners	FORTISS, RTAW, IKL, UPV, SINTEF

ID	G 9.9	
Topic	Development Process	
Name	Timing constraints related activities of the development process	
Source	RTaW	
Description	The DREAMS framework should provide a model transformation based development process that can be applied to different application domains. One of the important aspects is the description, the traceability and the verification of timing constraints.	
Missing Services	cross-domain applicable development activities related to the description, the tracing and the verification of timing constraints	
Gap Type	Development process (elements)	
Relationship to requirements	R 9.5.1, R 9.5.2, R 9.5.3, R 13.6.1	
Relationship to building blocks	B 9.1.2	
Responsibility	WP	1
	Deliverable	D 1.3.1
	Lead partner	RTaW
	Participating partners	RTaW, FORTISS,

ID	G 9.10	
Topic	Modeling	
Name	Security Meta-Models	
Source	USIEGEN	
Description	The Security Meta-Models shall allow modelling the varying needs of security.	
Missing Services	Models for <ul style="list-style-type: none"> • data confidentiality • data integrity • authentication 	
Gap Type	Meta-Model	

Relationship to requirements	R 9.8.1, R 9.8.2, R 9.8.3	
Relationship to building blocks	B 9.3.1	
Responsibility	WP	1
	Deliverable	D 1.4.1
	Lead partner	USIEGEN
	Participating partners	USIEGEN, FORTISS, TTT, ST, TEI, VOSYS

ID	G 9.11	
Topic	Modeling	
Name	NoC static/dynamic power consumption model	
Source	ST	
Description	The NoC static/dynamic power consumption model can be used to obtain approximate average static and dynamic power consumption at the system-level.	
Missing Services	A modeling methodology to design static cost analysis of NoC components We thus propose a fully automated modeling flow which can be applied directly to any architecture and technology. The output of the flow is a NoC component cost predictor able to estimate a metric of interest for any configuration in the design space in few seconds.	
Gap Type	Model	
Relationship to requirements	R 9.7.1	
Relationship to building blocks	Existing Kriging theory	
Responsibility	WP	1
	Deliverable	D1.4.1
	Lead partner	ST
	Participating partners	ST , TEI, FORTISS

13.7 Resource Management

ID	G 10.1	
Topic	Resource Management	
Name	TTEthernet Network Reconfiguration	
Source	TTT	

Description	In order to reconfigure the network at runtime, a reconfiguration service must be implemented to switch between various schedules.	
Missing Services	<ul style="list-style-type: none"> Concept and implementation to safely switch between schedules in the network 	
Gap Type	SW	
Relationship to requirements	R 3.4.1, R3.2.2, R3.2.3, R3.2.4, R3.2.10	
Relationship to building blocks	Extension of B3.1.3 TTEthernet Reconfiguration	
Responsibility	WP	3
	Deliverable	D3.2.1, D3.2.2, D3.2.3
	Lead partner	TTT
	Participating partners	TUKL, IKL, ONERA

ID	G 10.2	
Topic	Resource Management	
Name	Distributed Resource Management	
Source	TUKL	
Description	<p>Distributed resource management services in DREAMS shall be implemented by: Global Resource Manager (GRM) and the local components, which are the Local Resource Managers (LRMs), Local Resource Schedulers (LRS) and Local Monitors (MON).</p> <p>The resource management in DREAMS shall provide the base services for fault detection and recovery strategies. Furthermore, reconfiguration of the resources shall produce on-time predictable results.</p>	
Missing Services	Concepts and implementation to extend concepts for single resources and devices to distributed resource management	
Gap Type	SW	
Relationship to requirements	R 2.4.1, R10.*	
Relationship to building blocks	B 10.3.1 ACTORS (single-divide resource management)	
Responsibility	WP	2,3,4
	Deliverable	D2.2.2, D3.2.1, D3.2.2, D3.2.3
	Lead partner	TUKL
	Participating partners	ONERA, RTAW, UPV, FENTISS, VOSYS, ST, TRT

ID	G 10.3	
Topic	Resource Management	
Name	Global Resource Manager	
Source	TUKL	
Description	The Global Resource Manager (GRM) shall integrate a system wide view on an abstract level (provided by local monitors). Based on the system state, it shall make global reconfiguration decisions to fulfil high-level constraints. Such decisions shall be distributed to the Local Resource Managers (LRM), and subsequently to the Local Resource Schedulers (LRS) of each resource involved.	
Missing Services	GRM implementation.	
Gap Type	SW	
Relationship to requirements	R10.*	
Relationship to building blocks	B 10.3.1 ACTORS (single-divide resource management)	
Responsibility	WP	3
	Deliverable	D3.2.1, D3.2.2, D3.2.3
	Lead partner	TUKL
	Participating partners	ONERA, RTAW, UPV, FENTISS, VOSYS, ST, TRT

13.11 Security

ID	G 11.1	
Topic	Security	
Name	Security in the development process	
Source	USIEGEN	
Description	Secure software development for security by design. Security shall be included and embedded into the development process.	
Missing Services	<ul style="list-style-type: none"> Security software design, e.g., via security patterns 	
Gap Type	SW, Model	
Relationship to requirements	R 11.6.1	
Relationship to building blocks	B 11.2.1	
Responsibility	WP	1

	Deliverable	D 1.3.1
	Lead Partner	USIEGEN
	Participating partners	USIEGEN, FORTISS

ID	G 11.2	
Topic	Security	
Name	Secure communications on the cluster level using Ethernet related protocols, such as TTEthernet.	
Source	USIEGEN	
Description	Security services for secure communication on the cluster level for privacy, authenticity, integrity etc.	
Missing Services	<ul style="list-style-type: none"> Implementation of confidentiality, integrity and authenticity services, based on MACsec, and their integration with TTEthernet 	
Gap Type	SW	
Relationship to requirements	R 11.3.1, R 11.3.2, R 11.1.2, R 11.3.3, R 11.3.4, R 11.3.5, R 11.3.6, R 11.3.7, R 11.3.8	
Relationship to building blocks	B 11.5.1	
Responsibility	WP	3
	Deliverable	D 3.3.1, D 3.3.2, D 3.3.3
	Lead Partner	USIEGEN
	Participating partners	USIEGEN, TTT

ID	G 11.3	
Topic	Security	
Name	Protection against man in the middle attacks on TTEthernet.	
Source	USIEGEN	
Description	Communications shall be protected from man in the middle attacks. One way to limit the man in the middle attacks is through the integration of MACsec protocol with TTEthernet.	
Missing Services	<ul style="list-style-type: none"> Implementation and integration of MACsec with TTEthernet Additional mechanisms to prevent man in the middle attacks 	
Gap Type	SW	

Relationship to requirements	R 11.3.9	
Relationship to building blocks	B 11.5.1	
Responsibility	WP	3
	Deliverable	D 3.3.1, D 3.3.2, D 3.3.3
	Lead Partner	USIEGEN
	Participating partners	USIEGEN, TTT

ID	G 11.4	
Topic	Security	
Name	Key management	
Source	USIEGEN	
Description	Key management for security services should be provided. This includes key generation, key distribution or exchange and key destruction etc.	
Missing Services	<ul style="list-style-type: none"> • Key management for cluster level security services needs to be implemented and integrated with MACsec using IEEE802.1X • Key management for security services at the memory controller and for GRM, LRM, MON etc. needs to be implemented 	
Gap Type	SW	
Relationship to requirements	R 11.3.7, R11.2.2	
Relationship to building blocks	B 11.6.1	
Responsibility	WP	3
	Deliverable	D 3.3.1, D 3.3.2, D 3.3.3, D2.2.2
	Lead Partner	USIEGEN
	Participating partners	USIEGEN, TTT

ID	G 11.5	
Topic	Security	
Name	On-chip communications security	
Source	USIEGEN	
Description	Security services to protect on-chip communications from logical attacks are missing	

Missing Services	<ul style="list-style-type: none"> Protection of on-chip communication from security attacks, e.g., side channel attacks 	
Gap Type	SW, HW	
Relationship to requirements	R 11.2.2, R 11.2.3	
Relationship to building blocks	B 11.3.1, B 11.3.2, B 11.4.1	
Responsibility	WP	2
	Deliverable	D 2.1.1, D 2.1.2, D 2.1.3
	Lead Partner	TEI, VOSYS
	Participating partners	TEI, VOSYS, ST

ID	G 11.6	
Topic	Security	
Name	Network safety and security trade-off and threat analysis	
Source	TTT	
Description	In order to specify an adequate security concept and implementation of necessary security mechanisms, a threat analysis on the network level shall be performed. Also, potential trade-offs and benefits between safety and security functionality must be analysed.	
Missing Services	Security services to target the existing needs and not influencing safety functionality in a negative way. In the case of negative effects, solutions to these shall be described.	
Gap Type	Analysis	
Relationship to requirements	R 13.11.2	
Relationship to building blocks	B 11.7.1	
Responsibility	WP	3
	Deliverable	D3.3.1
	Lead partner	USIEGEN
	Participating partners	TTT

ID	G 11.7
Topic	Network Security

Name	TTEthernet MACsec Security	
Source	TTT	
Description	TTEthernet security infrastructure on the cluster level based on the MACsec (IEEE 802.1AE) protocol set focusing on data origin authentication, data integrity, confidentiality and replay protection.	
Missing Services	Implementation of the MACsec security protocol in the TTEthernet Switch IP	
Gap Type	VHDL implementation	
Relationship to requirements	R 13.113.1, R 13.11.2, R 13.11.3, R 13.11.4, R 13.11.5, R 13.11.6, R 13.11.8, R 13.11.9	
Relationship to building blocks	B 11.7.1	
Responsibility	WP	3
	Deliverable	D3.3.1, D3.3.2, D3.3.3
	Lead partner	TTT
	Participating partners	

ID	G 11.8	
Topic	Security	
Name	Protection against replay attacks.	
Source	USIEGEN	
Description	Communications shall be protected from replay attacks. Replay attacks are very critical in certain applications, such as healthcare. Measures such as those that come with the integration of MACsec shall be adopted.	
Missing Services	<ul style="list-style-type: none"> Implementation and integration of MACsec with TTEthernet 	
Gap Type	SW	
Relationship to requirements	R 11.3.8	
Relationship to building blocks	B 11.5.1	
Responsibility	WP	3
	Deliverable	D 3.3.1, D 3.3.2, D 3.3.3
	Lead Partner	USIEGEN
	Participating partners	USIEGEN, TTT

ID	G 11.9	
Topic	Security	
Name	Protection against traffic analysis.	
Source	USIEGEN	
Description	Protection against traffic analysis attacks is not covered by any building block.	
Missing Services	<ul style="list-style-type: none"> Implementation of methods for protection against traffic analysis attacks Integration in cluster level security mechanisms 	
Gap Type	SW	
Relationship to requirements	R 11.3.10	
Relationship to building blocks		
Responsibility	WP	3
	Deliverable	D 3.3.1, D 3.3.2, D 3.3.3
	Lead Partner	USIEGEN
	Participating partners	USIEGEN, TTT

ID	G 11.10	
Topic	Network Security	
Name	Secure time distribution for global time base	
Source	USIEGEN	
Description	<p>Secure time distribution and synchronization shall be provided in DREAMS architecture as a core service.</p> <p>Time synchronization for establishing a global time base is a key mechanism for the predictable virtualization of resources and TSP in DREAMS.</p>	
Missing Services	<ul style="list-style-type: none"> Secure time distribution and synchronization 	
Gap Type	SW	
Relationship to requirements	R 11.3.2	
Relationship to building blocks	B 11.5.1	
Responsibility	WP	3
	Deliverable	D3.3.1, D3.3.2, D3.3.3
	Lead partner	USIEGEN

	Participating partners	USIEGEN, TTT
--	------------------------	--------------

ID	G 11.11	
Topic	Chip-level security (logical and physical security)	
Name	Security of the on-chip resource, scheduling and monitoring components	
Source	USIEGEN	
Description	Authenticity and integrity of monitoring and resource management components shall be provided to ensure trust worthy communications and executions.	
Missing Services	<ul style="list-style-type: none"> Security mechanisms for the protection and authentication of resource managements and scheduling components 	
Gap Type	SW	
Relationship to requirements	R 11.2.1	
Relationship to building blocks	B 11.3.2, B 11.4.1	
Responsibility	WP	2
	Deliverable	D2.2.2
	Lead partner	USIEGEN, VOSYS
	Participating partners	USIEGEN, VOSYS, ST, TEI

ID	G 11.12	
Topic	Security	
Name	Security mechanisms for trustworthy communications between the GRM, LRMs and MONs	
Source	USIEGEN	
Description	Specification of security concept and implementation of necessary security mechanisms to provide trustworthy communications between the GRM, LRMs and MONs.	
Missing Services	<ul style="list-style-type: none"> Security mechanisms to achieve trustworthy communications between the GRM and the LRMs/MONs etc. 	
Gap Type	SW	
Relationship to requirements	R 11.3.5	
Relationship to building blocks	B 11.3.2, B 11.4.1	
Responsibility	WP	2, 3

	Deliverable	D2.2.2, D3.3.3
	Lead partner	USIEGEN
	Participating partners	USIEGEN, TTT

Part C

Terminology

Terminology

This part describes a common terminology for the DREAMS architecture, which was converged from the different application domains (i.e., avionics, wind power, healthcare) and the technological areas (i.e., multi-core chips, networks, operating systems, development methods).

Relationships between the different terms are visualized with the help of diagrams. Figure 11 serves as overview for the DREAMS terminology. It sets the umbrella terms *Development Methodology*, *Architectural Style*, *Platform*, *Fault Tolerance and Segregation*, *Mixed-Criticality System* and *Architecture* in relation to the *Mixed-Criticality Architecture* term itself.

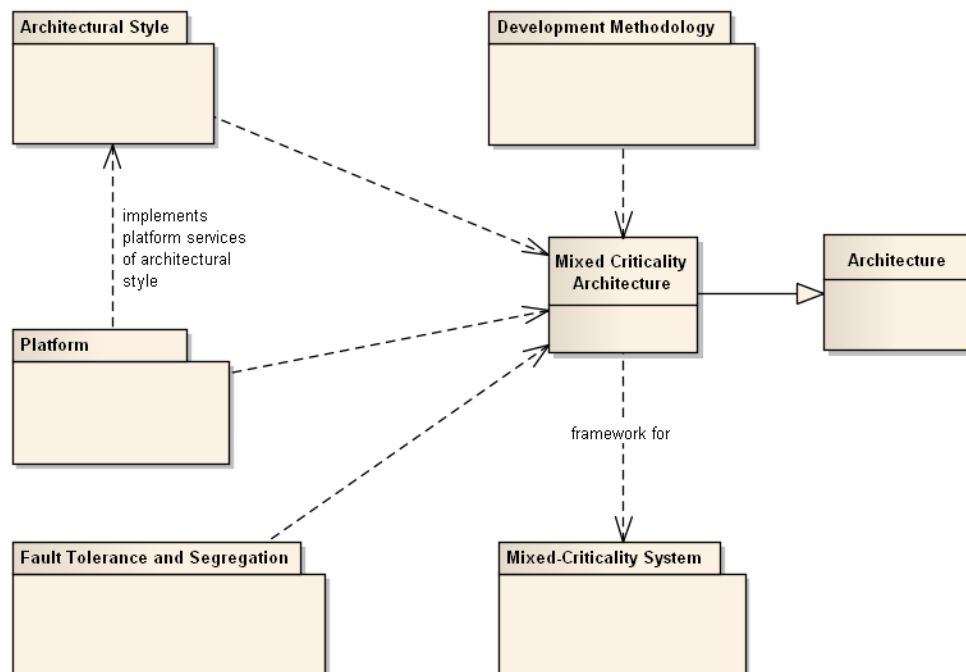


Figure 11: Overview of the DREAMS terminology

Relations between *Development Methodology*, *Meta-Models* and the *Development Process* are provided by Figure 12. The *Architectural Style* including the associations with *System Properties*, *Integration Levels* and the *Platform Services* is shown in Figure 13.

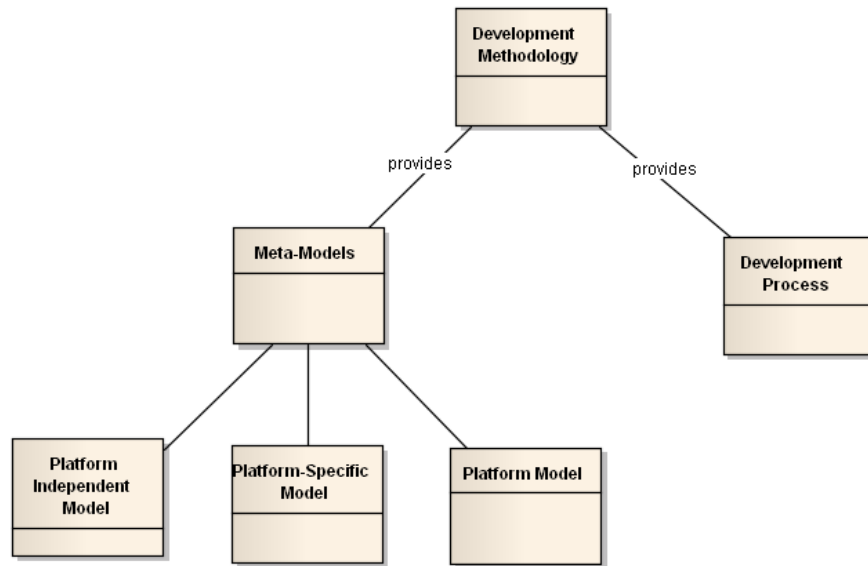


Figure 12: Development Methodology

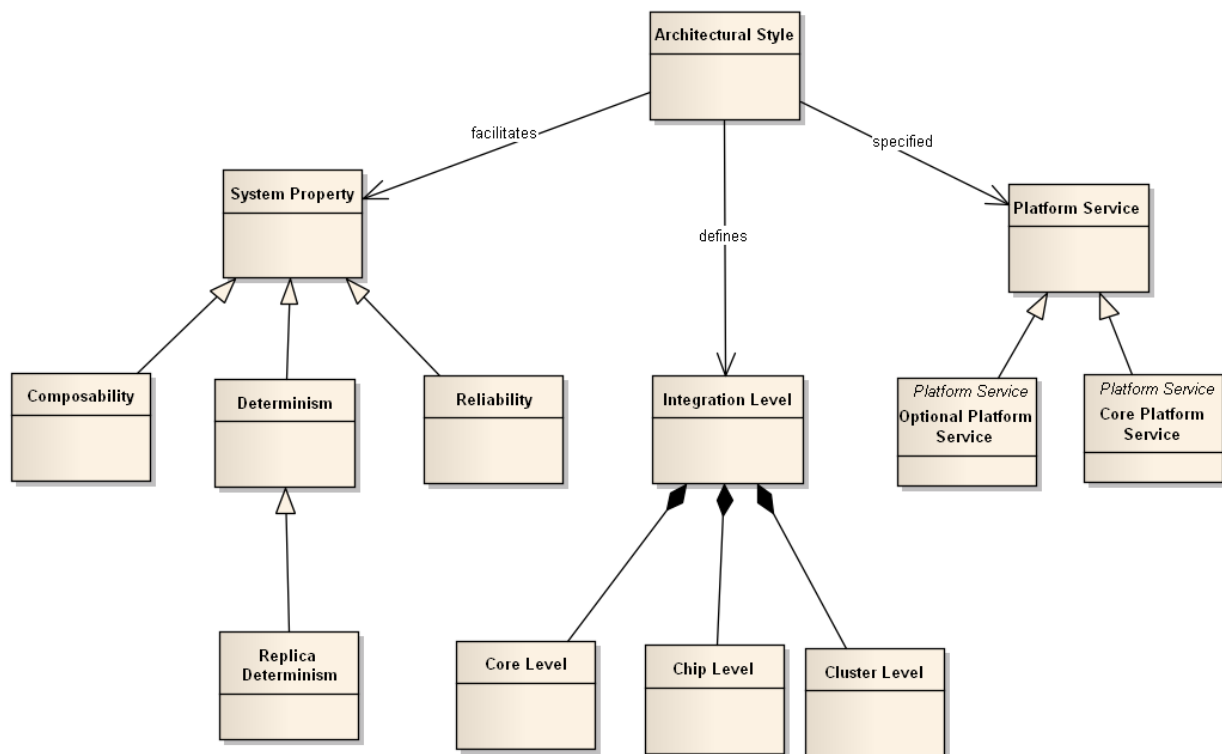


Figure 13: Architectural style

Information on the Core platform Services and their relation to Partitions and Channels is given in Figure 14 while Figure 15 illustrates the associations between terms of the fault tolerance and segregation group.

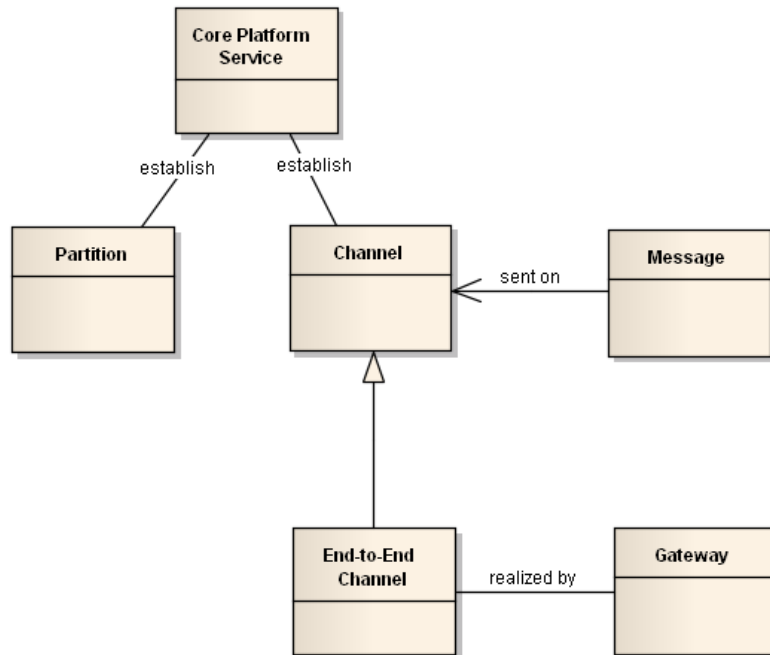


Figure 14: Platform

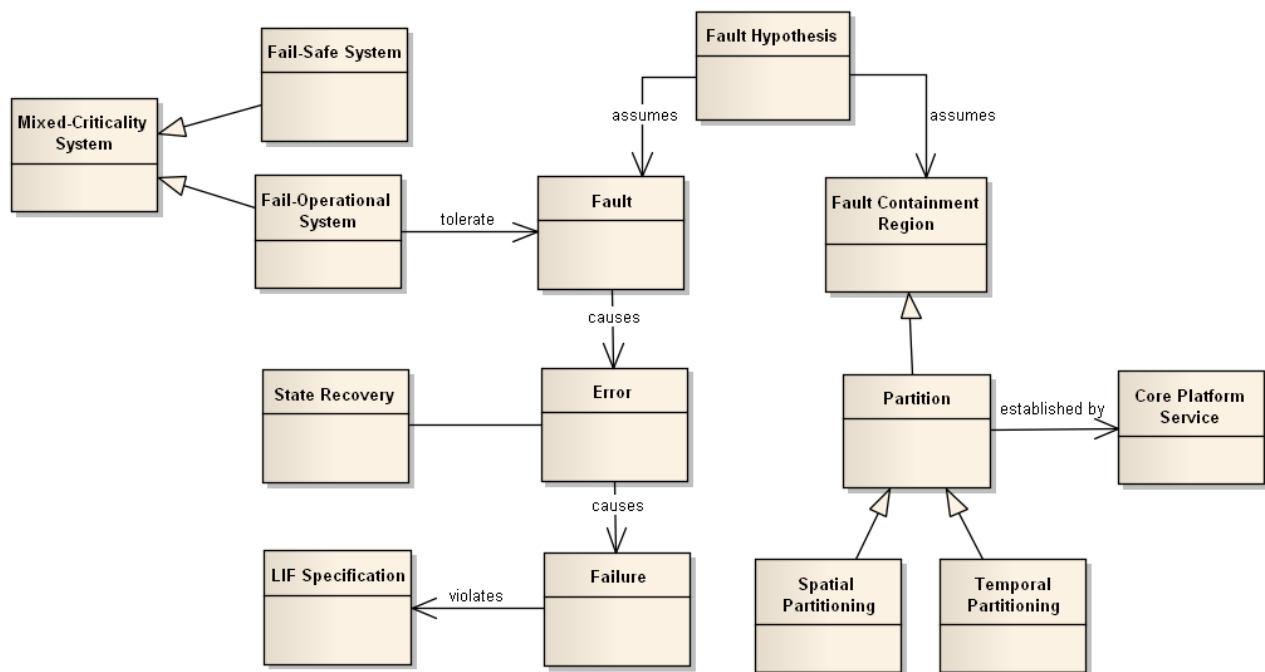


Figure 15: Fault tolerance and segregation in a mixed-criticality system

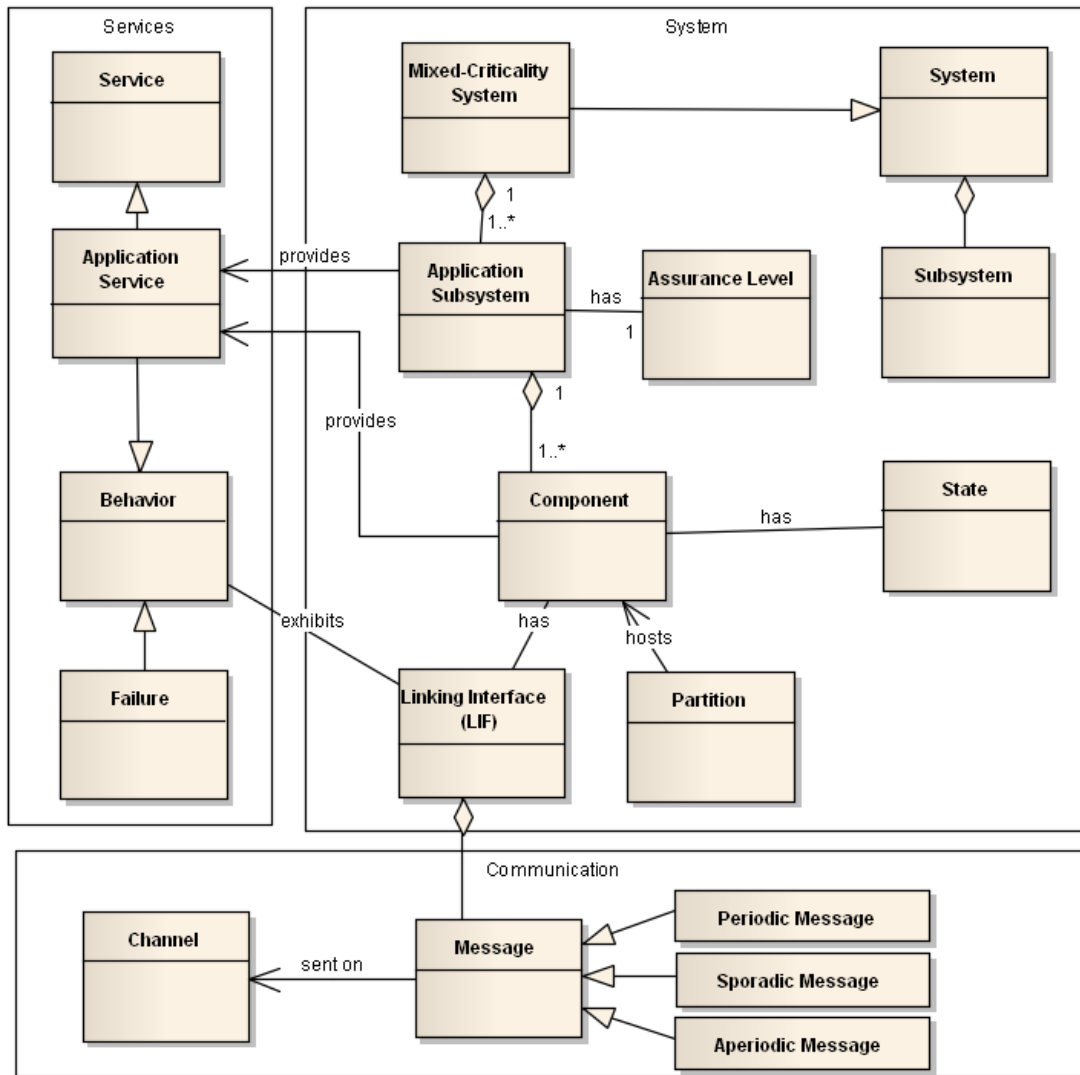


Figure 16: Mixed criticality system

Figure 16 defines the general terms for a mixed-criticality system, such as for instance *System*, *Subsystem*, *Component*, *State* and *Message*. Figure 17 concludes by providing detailed information on association of security related terms. All terms necessary to establish a common understanding are described in the following subsections.

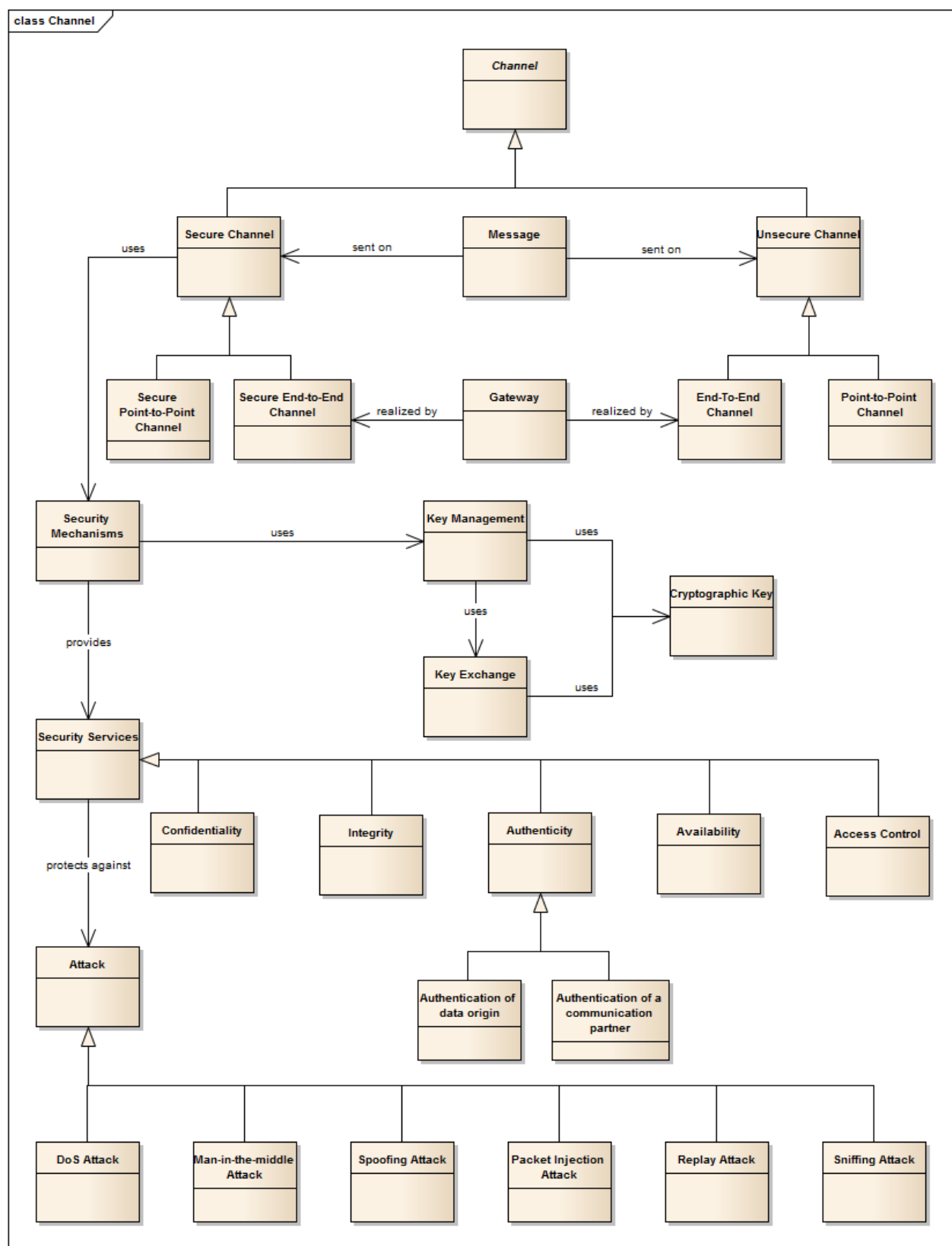


Figure 17: Security terminology

1 Aperiodic Message

An aperiodic message is a [message](#) without timing constraints on the arrival of the messages. Hence, only a best-effort transmission of aperiodic messages is possible. There is no guarantee if and when these messages will be transmitted, what delays may occur, or if aperiodic messages will be delivered at the recipient location. Best-effort messages typically use the remaining bandwidth of the network and have lower priority than [periodic messages](#) and [sporadic messages](#).

2 Application Service

The application service is the intended sequence of [messages](#) that is produced by a [component](#) via output ports at the [LIF](#) and the controlled object interface in response to the progression of time, inputs (via input ports at the LIF and the controlled object interface), and state.

3 Application Subsystem

An application subsystem is a nearly independent distributed subsystem of a large distributed real-time system that provides a well-specified [application service](#) with a corresponding [assurance level](#).

Examples of subsystems in a present day avionic application are the cabin pressurization system, the fly-by-wire system, and the in-flight entertainment system. Application subsystems are often developed by different organizational entities (e.g., by different vendors) and maintained by different specialists.

Since application subsystems may be of different criticality (e.g., safety system vs. multimedia system), the probability of fault propagation across subsystem boundaries must be sufficiently low to meet the dependability requirements. A subsystem is further decomposed into smaller units called [components](#).

4 Architectural Style

The architectural style provides a common notion of basic terms and concepts as well as rules and guidelines for the structuring of a [mixed-criticality system](#) into subsystems and for the design of the interactions among the subsystems. The architectural style also introduces architectural building blocks (e.g., network interfaces, global resource manager, local resource manager, gateways) and provides a high-level specification of the [platform services](#) provided by these building blocks.

The rules, guidelines and specifications of platform services constrain the implementation of a mixed-criticality system in such a way that the desired properties are obtained (e.g., security, safety, real-time performance, fault containment, data and system integrity, understandability) and it can be built cost-effectively.

5 Architecture

A technical system architecture (or architecture for short) is a framework for the construction of a [system](#). It imposes an [architectural style](#) for constraining an implementation in such a way that the desired properties of the system are obtained (cf. architectural style). The architecture also provides a

[development methodology](#) with a corresponding development process, modeling techniques, algorithms, tools, validation and certification techniques.

6 Assurance Level

The assurance level is determined from the safety assessment process and hazard analysis by examining the effects of a failure condition in the system.

For example, DO-178B distinguishes five assurance levels in avionics: Level A (Catastrophic) refers to systems where a failure that may cause a crash, Level B (Hazardous) implies a large negative impact on safety), Level B (Major) involves a significant, but lesser impact than a hazardous failure, Level C (Minor) refers to an even lesser impact on safety. The failure of a Level E system has no safety effect.

7 Behavior

The behaviour of a subsystem is the sequence of [message](#) (i.e., intended and unintended) that is produced by the subsystem at its [LIF](#).

8 Channel

A channel serves for the exchange of [messages](#) between ports. A channel is associated with a communication topology, a data-direction (e.g., unidirectional or bidirectional), temporal properties and dependability properties.

9 End-to-End Channel

An end-to-end channel is a [channel](#) that can include on-chip and off-chip communication links over hierarchical, heterogeneous and mixed-criticality networks. [Gateways](#) enable the horizontal integration at the cluster-level across different off-chip communication networks with different protocols (e.g., TTEthernet, EtherCAT, etc.), different reliabilities (e.g., fault-tolerant networks with media redundancy and active star couplers, low-cost fieldbus networks). [Gateways](#) between NoCs and off-chip networks enable the vertical integration through the seamless communication in hierarchical networks respecting mixed-criticality safety and security requirements.

10 Error

An error is that part of the system state which is liable to lead to a subsequent failure. A [failure](#) occurs when the error reaches the service interface.

11 Fail-operational System

A fail-operational system is able to tolerate one or several faults. Fail-operational systems send correct messages despite the failure of their subsystems.

12 Fail-safe System

If a fail-safe system one or more safe states can be reached in case of a system failure. Fail-safeness is a characteristic of the controlled object, not the computer system. In fail-safe systems the computer system must have a high error-detection coverage.

13 Fault

A fault is the adjudged or hypothesized cause of an [error](#). Faults can be internal or external of a system. Examples of types: An external fault (e.g. a malicious attack) causes an error, and possible a subsequent [failure](#). An internal fault (i.e. vulnerability) allows an external fault to harm the system and has to pre-exist in the system.

14 Fault-Containment Region

A Fault Containment Region (FCR) is a [subsystem](#) that operates correctly regardless of any arbitrary logical or electrical [fault](#) outside the region.

15 Fault Hypothesis

The fault hypothesis is the specification of the [faults](#) that must be tolerated without any impact on the essential system services. The fault hypothesis states the assumptions about units of failure (see [Fault Containment Region](#)), failure modes, failure frequencies, failure detection, and state recovery.

16 Failure

A failure occurs when the delivered service deviates from fulfilling its specification.

17 Cluster

A cluster is a physically distributed computer system that consists of a set of nodes interconnected by a physical network. Each node can be a multi-core chip with multiple IP cores interconnected by a network-on-a-chip. A cluster can be connected to another cluster using a [gateway](#).

18 Component

A component is a constituting element of an [application subsystem](#) and forms the basic unit of work. It interacts with other components through the exchange of [messages](#) across [LIFs](#) in order to work towards a common goal and provide the [application services](#).

A component is regarded as a self-contained building block that can be used in the design of a larger system. The component can have a complex internal structure that is neither visible, nor of concern, to

the user of the component. In the context of embedded real-time systems, it is essential that the component [behavior](#) can be specified in the domains of value and time.

19 Composability

Composability is a concept that relates to the ease of building [systems](#) out of [subsystems](#). A system, i.e., a composition of subsystems, is considered composable with respect to a certain property (functional or non-functional) if this property, given that it has been established at the subsystem level, is not invalidated by the integration. Examples of such properties are timeliness or certification.

For example, some embedded systems closely interact with their environment and they have to produce intended results at intended points of time. Temporal composability is a prerequisite for the feasible construction of such temporally predictable systems of high complexity. In [architectural styles](#) that support temporal composability, determining the emergent temporal [behavior](#) of the resulting system is eased by the fact that the individual subsystems retain their temporal properties after integration.

20 Core Platform Service

Core platform services (or core services for short) are mandatory in every instantiation of the [architecture style](#). The core platform services provide the foundation for higher-level, [optional platform services](#). For instance, a message-based communication service is a core service. At any given [integration level](#), the core services form a waist that can be realized using a multitude of implementation choices. Also, they form the starting point for the domain-customization using optional services. Exemplary categories of core services are communication services, execution services, time services and resource management services.

21 Determinism

A model behaves deterministically if and only if, given a full set of initial conditions (the initial [state](#)) at time t_0 , and a sequence of future timed inputs, the outputs at any future instant t are entailed.

22 Development Methodology

The development methodology is a framework consisting of a development process, a set of methods, techniques and tools for [mixed-criticality systems](#) based on networked multi-core chips.

23 Integration Level

The integration level denotes the layer in a system-of-systems at which it is composed out of its [components](#). Different integration levels can be distinguished in embedded systems including the [chip-level](#), the [cluster-level](#) and the [core-level](#).

24 Integration Level: Chip-Level

The chip-level is an [integration level](#) where IP cores are integrated using an on-chip network.

25 Integration Level: Cluster-Level

The cluster-level is an [integration level](#) where multiple chips are interconnected to a cluster using one or more off-chip communication networks (e.g., TTEthernet, EtherCAT). Thereby, applications can be supported that need more resources than are available on a single SoC. In addition, a distributed system with multiple SoCs is a prerequisite for implementing safety-critical [application subsystems](#), because today's semiconductor technology does not support the manufacturing of chips with a reliability that is suitable for ultra-dependability.

26 Integration Level: Core-Level

The core-level is an [integration level](#) where components are integrated using a hypervisor.

27 Gateway

A gateway is an IP core with two communication interfaces for connecting either two off-chip networks or an off-chip network and an on-chip network.

A gateway enables the horizontal integration at the [cluster-level](#) across different off-chip communication networks with different protocols (e.g., TTEthernet, EtherCAT, etc.), different reliabilities (e.g., fault-tolerant networks with media redundancy and active star couplers, low-cost fieldbus networks).

A gateway between a network-on-a-chip and an off-chip network enables vertical integration between [cluster-level](#) and [chip-level](#) through the seamless communication in hierarchical networks respecting mixed-criticality safety and security requirements.

28 Linking Interface

A [component](#) provides its real-time services, and accesses the real-time services of other components by the exchange of [messages](#) across its Linking Interface (LIF). These messages have to be fully specified in a [LIF specification](#) which consists of an operational specification and a LIF service model specification.

29 Linking Interface Specification

The linking interface specification is the mediating middle between a service supplier and the service user. On the one hand, the LIF service specification should be complete in the sense that it contains all information required to understand and use the [services](#) of the [component](#) that are offered at the particular LIF. On the other hand, the LIF specification should be minimal in the sense that it contains only information that is required by the user of the services.

The LIF service specification comprises a syntactic specification, a temporal specification, and a LIF service model specification. We subsume under the term operational specification of an interface the syntactic

specification and the temporal specification. The syntactic specification forms out of the sequence of bits in a [message](#) larger (information) chunks (such as a number, a string, or a method call, a structure consisting of a combination thereof, or a complex data object, such as a picture) and assigns a name to each chunk.

The temporal specifications of the messages defines their send and receive instants, e.g., at what instants the messages are sent and arrive, how the messages are ordered, and the rate of message arrival. This information can be formalized if an appropriate model of real-time is available, as for example Timing Events and Constraints. In non-safety critical applications the temporal specification can be expressed in probabilistic terms.

The LIF service model specification provides a conceptual interface model that relates the names of the chunks to the user's conceptual world and thus assigns a deeper meaning to the chunks generated by the syntactic specification. It follows that the LIF service model must be expressed in concepts that are familiar to the user of the interface services.

30 Message

A message is any data structure that is formed for the purpose of inter-[component](#) communication. Message passing is a universal model. Different interaction patterns, such as a shared memory, can be realized on top of message passing. When a message is send over a network, then it is contained in the payload portion of a Frame, possibly together with other messages.

The timing of message exchanges can be explicitly defined and the need for separate synchronization is eliminated. Different types of timing models of messages can be distinguished such as [periodic messages](#), [sporadic messages](#) and [aperiodic messages](#).

An unidirectional message has one sender component and one or more receiver components. The knowledge of the sender identity is essential for fault-containment and diagnosis (e.g., masquerading failures), whereas knowing the timing of a message (e.g., period) enables the containment of temporal faults.

31 Message Sent (Event)

A Message Sent is a Timing Event that describes the fact that a Component has sent out a message, i.e. has handed over the message to the communication service of the platform.

32 Message Arrived (Event)

A Message Arrived is a Timing Event that describes the fact that a message has arrived at a destination Component, i.e. the message is available for being read by the destination Component.

33 Mixed-Criticality System

Mixed-criticality is the concept of allowing [application subsystems](#) that must meet different [assurance levels](#) (e.g., ranging from DAL A to DAL E in RTCA DO-178B, SIL1 to SIL4 in EN ISO/IEC 61508) to seamlessly interact and co-exist on the same networked distributed computational [platform](#).

34 Mixed-Criticality Architecture

A mixed-criticality architecture is an architecture that provides [platform services](#) and a [development methodology](#) supporting [mixed-criticality](#) (e.g., temporal and spatial partitioning, modular certification methods).

35 Optional Platform Services

The optional platform services which are built upon the [core platform services](#) can be generic in the sense that they can be used in multiple application domains or specific for a focused domain. These services are optional in the sense that they are not required in every instantiation of the [architecture](#). If needed, developers can pick them out of the [architectural style](#), which includes a set of existing, validated component libraries for the different [integration levels](#). For instance an encryption service could be a generic optional service.

36 Partition

A partition is the execution environment for a [component](#) with corresponding resources (e.g., processor, memory, communication, input/output). The resources for a partition are protected by [temporal partitioning](#) and [spatial partitioning](#) in order to avoid unintended feature interaction and fault propagation between components.

37 Periodic Message

Periodic messages are specified by a period and phase, which can be expressed with respect to a system-wide synchronized global time base.

Periodic messages can be exchanged using *time-triggered communication*, where the instants of periodic message transmissions are specified by an a priori planned conflict-free communication schedule. For time-triggered communication, the communication infrastructure is deterministic and guarantees temporal properties such as latency, latency jitter, bandwidth, and message order.

38 Platform

A platform is the hardware/software foundation for the execution of applications. The platform instantiates the [architectural style](#) and implements generic services for the development of applications, which are denoted as [platform services](#) (see [core platform services](#) and [optional platform services](#)).

39 Platform Services

Platform services facilitate the development of [applications subsystems](#) and separate the application functionality from the underlying platform technology to reduce design complexity and to enable design reuse. We differentiate between two different types of platform services: [core platform services](#) and [optional platform services](#).

40 Platform-Independent Model

A Platform Independent Model (PIM) is a model of a system that is independent of the specific technological [platform](#) used to implement it.

41 Platform-Specific Model

A Platform Specific Model (PSM) is a model of a system that is linked to a specific technological [platform](#) used in implementation.

42 Reliability

Reliability is the ability of an [application subsystem](#) to perform its required functions under stated conditions for a specified period of time.

43 Replica Determinism

Replica determinism is a desired property between replicated [components](#). A set of replicated components is replica determinate if all components in this set produce exactly the same output [messages](#) that are at most an interval of d time units apart, as seen by an omniscient outside observer. In a time-triggered system, components are considered to be replica-deterministic, if they produce the same output messages at the same global ticks of their local clock.

44 Service

The service delivered by a [system](#) (in its role as a provider) is its intended [behavior](#) as it is perceived by its users. The behavior is the sequence of observable outputs of a system.

45 Spatial Partitioning

Spatial partitioning ensures that the service in one [partition](#) cannot alter the code or private data of another partition. Spatial partitioning shall also prevent a partition from interfering with control of external devices (e.g., actuators) of other partitions.

46 Sporadic Message

Sporadic messages establish rate-constrained data-flows with maximum bandwidth use, which helps to guarantee bounded latencies. Successive transfers of sporadic messages belonging to the same rate-constrained dataflow are guaranteed to be offset by a minimum duration (also called minimum inter-arrival time of sporadic messages).

The temporal behavior of sporadic messages can further be specified by sporadic repetition constraints.

47 State

The state enables the determination of a future output solely on the basis of the future input and the state the system is in. In other word, the state enables a "decoupling" of the past from the present and future. The state embodies all past history of the given system. Apparently, for this role to be meaningful, the notion of the past and future must be relevant for the system considered.

48 State Recovery

State recovery is the action of re-establishing a valid state in a subsystem after a [failure](#) of that subsystem.

49 Subsystem

A subsystem is a part of a [system](#) that represents a closure with respect to a given property.

50 System

A system is a set of [subsystems](#).

51 Temporal Partitioning

Temporal partitioning ensures that a [partition](#) cannot affect the ability of other partitions access shared resources, such as the network or a shared CPU. This includes the temporal behavior of the services provided by resources (latency, jitter, duration of availability during a scheduled access).

52 Design Pattern

A Design Pattern is a general reusable solution to a commonly occurring problem within a given context. It is a description or template for how to solve a problem that can be used in many different situations. Patterns are formalized best practices.

53 Dependability Patterns

Design patterns that focus on finding common links on dependability as a measure of a system's availability, reliability, and its maintainability.

54 Compliant Item

A compliant item is any item (e.g. an element) on which a claim is being made with respect the clauses of IEC 61508 series.

55 Safety manual for compliant items

Safety manual for compliant items is a document that provides all the information relating to the functional safety of an element, in respect of specified element safety functions, that is required to ensure that the system meets the requirements of IEC 61508 series.

56 Event

“An event denotes a distinct form of state change in a running system, taking place at distinct points in time called occurrences of the event. That is, a running system can be observed by identifying certain forms of state changes to watch for, and for each such observation point, noting the times when changes occur. This notion of observation also applies to a hypothetical predicted run of a system or a system model — from a timing perspective, the only information that needs to be in the output of such a prediction is a sequence of times for each observation point, indicating the times that each event is predicted to occur.” – TIMMO-2-USE

57 Timing Event

Timing Events are identifiable state changes that are possible to constrain with respect to timing. Examples of timing events are: Message Sent, Message Arrived, Task Activation, Task Execution End, Frame Instantiation, Frame Transmission Start, Frame Transmission End.

The most common timing constraints are Latency constraint, Repetition Constraint, Synchronization Constraint.

58 Task Activation (Event)

A Task Activation is a Timing Event that describes the fact that a recurring task has entered the scheduling queue, i.e. will be considered by the scheduler for allocation of the processing unit.

Task Activations may occur for example periodically, with a certain jitter (see also Repetition Constraint).

59 Task Execution End (Event)

A Task Execution End is a Timing Event that describes the fact that a recurring task has executed all its instructions and is therefore removed from the scheduling queue.

60 Frame Instantiation (Event)

A Frame Instantiation is a Timing Event that describes the fact that a recurring frame has been filled with data and is ready for being transmitted, as soon as the protocol allows this.

61 Frame Transmission Start (Event)

A Frame Transmission Start is a Timing Event that describes the fact that the first bit of a frame is about to be transmitted.

62 Frame Transmission End (Event)

A Frame Transmission End is an event that describes the fact that the last bit of a frame has been transmitted and the frame is ready for being decoded at the receiver side.

63 Timing (Event) Chain

A Timing Chain specifies that a certain “response” event is causally related to a certain “stimulus” event. In other words, the “stimulus” event is supposed to induce the “response” event.

Prominent examples of stimulus and response events are changes of sensor values and corresponding changes of actuator values. However, a Timing Chain may be defined for any pair of causally related Timing Events.

In particular, a Timing Chain allows putting into a temporal / causal order all Timing Events related to the communication of a Message through an (End-to-End) Channel and to formally impose timing constraints on the communication.

A Timing Chain may be decomposed hierarchically into sub-chains. This allows to describe more precisely how the “stimulus” relates to the “response” and also how a global time budget, given by a latency constraint, may be decomposed into sub-budgets.

64 Timing Constraint

A Timing Constraint is a constraint on the occurrence times of one or more Timing Events.

65 Latency Constraint

A latency constraint describes how occurrences of a “target” event are placed relative to each occurrence of a “source” event. Source and target events are specified by a timing event chain.

Every instance of the source event must be matched by an instance of the target event, within a time window starting at lower and ending at upper time units relative to the source occurrence.

66 Repetition Constraint

A Repetition constraint describes the distribution of the occurrences of a single event. Typical examples of these events are Task Activation, Frame Instantiation, Task Execution End, Frame Transmission End.

Prominent examples of repetition constraints are periodic repetition with jitter and sporadic repetition with minimal inter-occurrence time.

67 Synchronization Constraint

A Synchronization constraint describes how tightly the occurrences of a group of events follow each other. This is typically expressed by a temporal window, i.e. an upper bound on the temporal distance between the occurrences of the events of the group.

An example is the reading of input data from different sensors, which must occur in a small time window to ensure a temporally consistent view of the environment.

68 Worst Case Execution Time (WCET)

The Worst Case Execution Time is the maximal delay needed to execute all instructions of a task, excluding interruption or preemption delays.

69 Worst Case Response Time (WCRT)

The Worst Case Response Time is the worst delay between the occurrence time of the Task Activation and the occurrence time of the Task Execution End. With respect to the WCET, it includes interruption/preemption or initial blocking delays (non-preemptive scheduling).

70 Worst Case Traversal Time (WCTT)

The Worst Case Traversal Time is the worst delay between the occurrence time of the Frame Instantiation and the occurrence time of the Frame Transmission End.

71 Secure End-to-End Channel

Using a secure end-to-end channel means that the communication is uninterruptedly protected between two communicating parties, e.g., PGP (e-mail), ZRTP (VoIP), etc.

72 Secure Point-to-Point Channel

Using a secure point-to-point Channel means that the communication is uninterruptedly protected between two points/nodes in a network, e.g., VPN, MACsec, IPsec etc.

73 Security Mechanisms

Security mechanisms are used to provide security services, e.g., encryption is used to ensure confidentiality.

74 Security Services

Security services define different classes to protect a system against attacks. Security services include authentication, access control, confidentiality, integrity and non-repudiation.

75 Confidentiality

Confidentiality ensures the privacy of information. Only authorized users can read the data. This includes the data stored in memory as well as the data transferred over a network.

76 Integrity

Data integrity means that the data cannot be modified unnoticeably. Every intended and unintended modification of the data should be detectable.

77 Authenticity

Authenticity ensures that data is genuine and that the actual origin of the data is the same as the claimed origin.

78 Authentication of data origin

Authentication of data origin ensures that the actual origin of the data is the same as the claimed origin.

79 Authentication of a communication partner

Authentication of a communication partner ensures that the actual communication partner is the same as claimed.

80 Availability

If an Information or access to a service is needed, it must be available. Additionally, it must also function correctly.

81 Access control

Access control includes authorization, identification and authentication (I&A), access approval, and audit. Authorization specifies what a subject can do, e.g., read, write or execute a file. Access approval grants or rejects access to the requested resource. Audit records the access to a resource. For Identification and authentication please refer to the topic on authentication.

82 DoS attack

A denial-of-service (DoS) attack tries to make a system unavailable to legitimate users of a service or a system. The user might still be able to access the system but might not be able to use it in the way he wants to, e.g., the connection becomes too slow or parts of the system are inaccessible.

83 Man-in-the-middle attacks

Man-in-the-middle attacks aim that an attacker gets control of a communication between two other parties relaying messages by inserting itself between them. The two parties believe that they are talking to each other directly but the attacker can eavesdrop, suppress or modify the exchanged messages and can create new messages pretending to be originating from one of the communicating partners.

84 Spoofing attack

In spoofing attacks, an attacker tries to masquerade as another user or program to get an advantage, e.g., in e-mail spoofing, the attacker manipulates the "From" field of an e-mail.

85 Packet injection attack

In packet injection attacks, the attacker inserts new packets or messages into a communication stream. The attacker creates the new packets in a way that he gets an advantage. If the new packet is copy of an old one, then the attack is called a replay attack.

86 Replay attack

An attacker makes a copy of a valid data packet/message and sends it later once again to achieve certain objectives, such as repeat.

87 Sniffing attack

A sniffing attack captures the transmitted packets/message. The captured packets can be analyzed later.

88 Key Management

Key Management includes key generation, key exchange, key destruction, etc.

89 Cryptographic key

A cryptographic key is a parameter that influences the output of a cryptographic algorithm and is shared between the communicating parties. Using an algorithm with the same input, but with a different key, the output of the algorithm will be quite different.

90 Key Exchange

Key exchange defines the way how a cryptographic key and relevant parameters are shared between the communicating partners so that no one else can obtain the information.

Bibliography

- [1] "664P1-1 Aircraft Data Network, Part 1, Systems Concepts and Overview", ARINC, 06-2006
- [2] DO-297, "Integrated Modular Avionics (IMA) Development Guidance and Certification Considerations", RTCA, 2005
- [3] "PikeOS™ SIL 4 certification on multi-core platform", Sysgo, <http://www.sysgo.com/news-events/press/press/details/article/pikeosTM-sil-4-certification-on-multi-core-platform/>, October 2013
- [4] "664P7-1 Aircraft Data Network, Part 7, Avionics Full-Duplex Switched Ethernet Network", ARINC, 09-2009
- [5] "AS6802: Time-Triggered Ethernet", SAE International, 2011-11
- [6] RTCA, „DO-254, "Design Assurance Guidance For Airborne Electronic Hardware", 2000.
- [7] RTCA, „DO-178, "Software Considerations in Airborne Systems and Equipment Certification", 2012.
- [8] EC, "Directive 2006/42/EC of the European Parliament and of the Council of 17 May 2006 on machinery, and amending Directive 95/16/EC (recast)".
- [9] IEC, "IEC 61400: Wind Turbines generator systems," ed, 2005.
- [10] ISO, "ISO 13849: Safety of Machinery," ed, 2006.
- [11] IEC, "IEC 62061: Safety of machinery: Functional safety of electrical, electronic and programmable electronic control systems," ed, 2005.
- [12] IEC, "IEC 61508: Functional safety of electrical/electronic/programmable electronic safety-related systems," ed, 2010.
- [13] GL, "Guidelines of Germanischer Lloyd Industrial Services (GL)," ed, 2010.
- [14] Y.B Choi, J.S Krause , H. Seo, K.E. Captain, K. Chung , "Telemedicine in the USA: standardization, through information management and technical applications",
- [15] Pellizzoni, A. Schranzhofery, J. Cheny, M. Caccamo, and L. Thiele, "Worst case delay analysis for memory interference in multicore systems," in Design, Automation & Test in Europe Conference & Exhibition (DATE), 2010. IEEE, 2010,pp. 741–746
- [16] Heechul Yun "Improving Real-Time Performance on Multicore Platforms Using MemGuard"
- [17] L. Tang et al., "The Impact of Memory Subsystem Resource Sharing on Datacenter Applications", ISCA'11, IEEE/ACM international conference