A PRACTICAL APPROACH FOR AIRCRAFT SYSTEMS REQUIREMENTS VALIDATION IN COMPLIANCE WITH ARP4754

Luigi Turco, Valerio Zanetti, Nicole Cardili, Maurizio Parolini
Leonardo Helicopters, Cascina Costa di Samarate, Italy

Abstract

This paper illustrates the process applied to the validation of AW609 Flight Control System (FCS) requirements to comply with ARP4754 guidelines. The approach takes into accountthe application of the guidelines to a project that was already at an advanced stage of maturity, therefore requiring a deep reorganization of the requirement review to comply with ARP guidelines, with particular emphasis to the validation aspects. In addition, the difficulty of the task was amplified by the fact that system in object is a complex, highly integrated system within the type certification process of a powered lift, a new special class of aircraft.
This paper presents the LHD / AWPC approach employed to successfully achieve the objectives of the system validation review process that have been approved by the Certification Authority.

1. NOTATION

The following acronyms will be used throughout the paper:

ARP	Aerospace Recommended Practice
AWPC	Agusta Westland Philadelphia Corp.
CLAW	Control Law
DAL	Design Assurance Level
DOORS	Dynamic Object-Oriented Requirements System
DSR	Derived Safety Requirements
ERF	European Rotorcraft Forum
ECR	Engineering Change Request
FAA	Federal Aviation Administration
FBW	Fly by Wire
FCS	Flight Control System
FHA	Functional Hazard Assessment
FuSE	Function-Based Systems Engineering
IBM	International Business Machines
IRS	Interface Requirement Specification
LHD	Leonardo Helicopters Division
PLM	Product Lifecycle Management
PSSA	Preliminary System Safety Assessment
RMT	Requirements Management Tool
SAE	Society of Automotive Engineers
SAP	System Applications and Products in Data Processing
SSA	System Safety Analysis
SSS	System / Subsystem Specification

2. INTRODUCTION

Since the SAE ARP4754 (Ref. 1) guideline has been recognized by aeronautics certification authorities as ‘an acceptable method for establishing a development assurance process’ (Ref. 2), aircraft manufacturers have worked to define and review their own development processes to comply with it.
In order to define a suitable and compliant process, several challenges have to be faced. Firstly, the guideline requires starting the system validation and safety assessment activities prior to the system design stage, by documenting and motivating design choices and assumptions. Moreover, for projects that were already in an advanced development stage when compliance to ARP4754 became mandatory, the task was even more difficult to be accomplished due to the need of a rearrangement of the certification documentation, including validation evidence.
Defining such a process is much more difficult when the system of interest is a complex system made of several items and deeply interconnected with other aircraft systems and the operating environment, be-

ing the validation of requirements allocated to the system heavily dependent upon activities carried out by multiple company departments.
These aspects require technical activities and increased paperwork, impacting project cost and schedule.
In the following paragraphs an effective approach is presented, to address the validation process in a holistic manner by function and sub-function, to maximize the similarities among the sub-functions and especially among the requirements which define the sub-functions in object.
This paper proposes a method to apply SAE ARP4754 guidelines for system requirements validation, applicable to any aircraft complex system, either to establish a new process or tailor a pre-existing one.The system validation process described has been developed and applied for the civil certification of the Fly-By-Wire (FBW) Flight Control System (FCS) of the AW609 tiltrotor by Leonardo Helicopters and reviewed by the Federal Aviation Administration (FAA). FBW FCS for tiltrotor aircraft is a fitting example of complex system, since it is required to provide safety-critical functions continuously throughout the mission, and it is deeply interconnected with most of the aircraft systems, e.g., hydraulics, drive system, powerplant and conversion systems, avionics, etc.
According to the systems engineering standard ISO 15288 (Ref. 3), the validation process can be divided in three phases: prepare for validation, perform validation, and manage validation results. The paper offers methodology solutions and guidance for each phase, as listed in Table 1, to address many validation facets.

Table 1: Process solutions for each validation phase.

Prepare for validation	Perform Validation	Management Validation Results
Process definition with FuSE approach	Validation of Functions	Validation summary Structure
Digital development environment setup	Aircraft systems integration requirements validation	Validation knowledge management
Validation matrix structure	Concurrence of validation and safety assessment activities	
	Validation evidence selection from requirement types	

3. PREPARE FOR VALIDATION

3.1. Process definition with FuSE approach

Function-Based Systems Engineering (FuSE) is an approach proposed in (Ref. 4), it is based upon the definition and usage of a functional architecture, to perform product planning, conceptual and detailed design. In the example AW609 FCS validation case, the FuSE approach was applied for validation activities. This means carrying out validation function-by-function, instead of requirement-by-requirement. According to Ackoff (Ref. 5), a function is the ability of the System to produce an outcome that contributes to the achievement of the System goals and objectives. Thus, the effects and efficiency of a system function can be gauged by observing the outputs produced by a system in response to certain inputs, in a black-box representation.

Approaching the validation per functions has many advantages.

1. The FuSE approach offers an overall vision of the System functionality to the reviewer and allows to easily identify any missing, wrong, unclear or duplicated requirement. Validating a function equal validating its allocated requirements.
2. Simulation and testing scenarios trigger the functions behaviors at once. These validation evidences can be traced directly to the function, to validate multiple functional requirements at once.
3. Validation by functions makes concurrence of safety assessment and validation activities easier since functions are the starting point for the safety process, as per ARP4761 (Ref.

6). Additionally, the system requirements validation rigor, i.e. the Development Assurance Level (DAL), depends on the severity of the failures associated to the parent function, hence the functional requirements have homogeneous DAL.

Definition of end-to-end functions is necessary to assess the effects of their functional failures on the entire aircraft and its passengers, to establish failure severity and DAL. However, they encompass large sets of functional requirements that are difficultly validated all at once. To manage the complexity of validation activities, a functional decomposition from top-level functions to sub-functions is necessary, as in Figure 1. The functional architecture diagrams should be captured in the System architecture and design description document requested by ARP4754 objective 4.4.4.

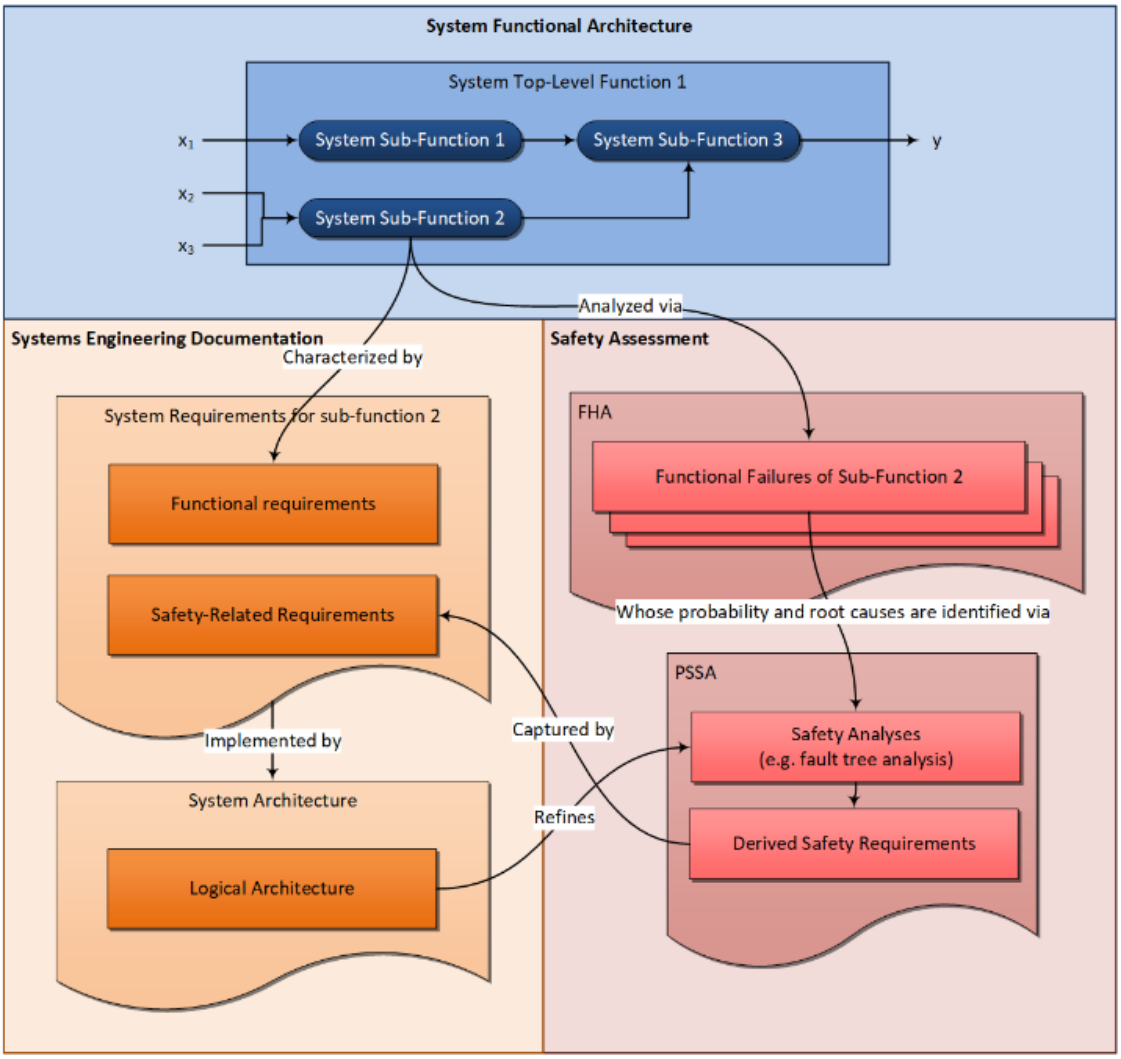

Figure 1: FuSE approach to the validation process.

To nurture the benefits of the FuSE approach, the following rules for sub-function identification have been adopted in the AW609 FCS validation activities:

1. Sub-functions perimeter have been defined so that no requirement pertains to more than one sub-function. Preexisting requirements whose meaning covered more than one sub-function have been rephrased and split.
2. Even though sub-functions are not end-to-end, they have been defined so that their contribution on the System outputs is identifiable in simulations and testing.

3. Decomposition of system top-level, end-to-end functions into sub-functions should be captured verbally in dedicated requirements statements. This helps communicating and agreeing the functional decomposition with external stakeholders since the early phases of the project, before the systems design is mature enough to issue the system architecture and design description document. Validating such requirements translates into providing a justification for the functional decomposition.

The requirements capture, validation, and safety assessment processes start at sub-function level, as portrayed in Figure 1.
In the FuSE framework, requirements describe the properties of the parent function, in terms of performance, availability, trigger conditions, functional inputs required, and functional outputs produced. Thus, achieving the requirements correctness and completeness objectives due for ARP4754 equals demonstrating that each function and function property is necessary and adequate to support aircraft-level functions and certification requirements. In the example case, AW609 FCS functions were classified to distinguish the kind of properties requirement types should have. To ensure the requirements validation review would be carried out thoroughly and homogeneously by various reviewers, a checklist template for each function validation was prepared and agreed with the Certification Authority.

3.2. Digital development environment setup

In the system requirements validation context, the digital development environment is the set of tools, databases, and their customizations, including embedded workflows, used to perform validation process activities. As a minimum, the environment should provide the capabilities to store and edit requirements and validation evidence, support change control and signature processes involving the Certification Authority or its delegates and allow data sharing with the various technical specialists that will receive or refine the requirements. To address the above, the environment needs to include the following kinds of tools:

1. A Requirements Management Tool (RMT), with requirements modules and traceability relationships among requirements, functions, and other design data.
2. A versioning database to collect supporting evidence in various formats: test procedures, analysis documents, review meeting minutes, etc.
3. A tool to issue official documentation and support the signature process.

For the AW609 FCS case, the three tools adopted were respectively IBM DOORS, Serena Dimensions, and SAP (Figure 2). The second and third kinds can be merged in a unique PLM tool properly configured, but this option was not adopted for the AW609 FCS example case due to company policies.

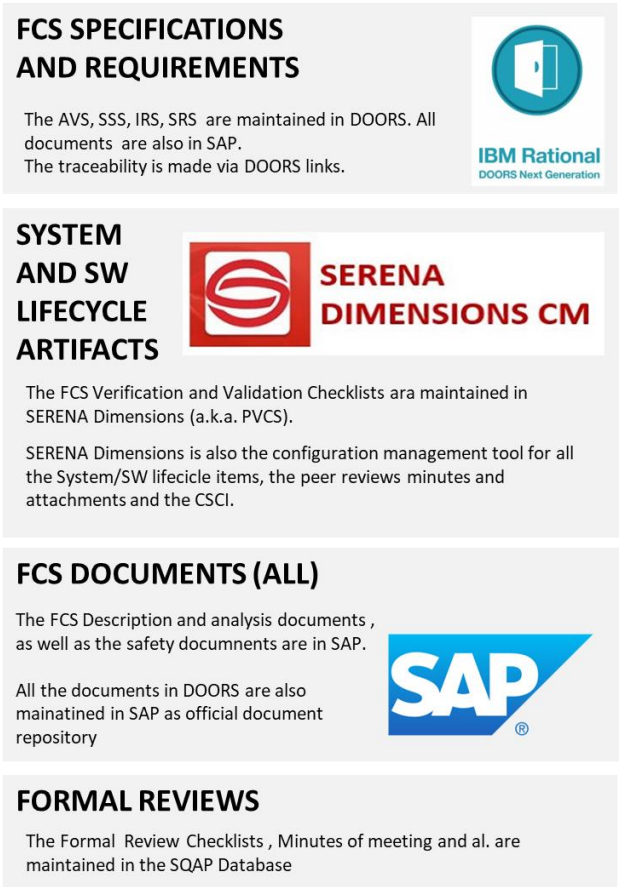

Figure 2: Development Environment Elements

AW609 FCS requirements validation activities had additional needs for the environment due to the program peculiarities. A tiltrotor FBW FCS is a highly interconnected system; hence requirements validation activities heavily depend on information exchanges with other technical areas and subject matter experts. Various integration requirements are levied upon

FCS system, e.g., to enforce command limitations due to flight envelope limits or drive system ratings. On the other hand, FCS requires energy supply, input data, and actuating capabilities that should be provided by other systems. This translates in the necessity for the digital environment to support cross-team collaboration and enable users with variegated levels of skills in the tools' usage and in ARP4754 to work effectively in it. The solution adopted was to finely configure the environment, especially the requirements management tool, to embed ARP4754 process constraints in it. This was achieved by the following means:

1. Mapping of all data required to RMT structure elements and other tools, to define a module and inter-module links structure compliant to the ARP guideline, as shown in Figure 3.
2. Definition of views in each tool and RMT module in line with the deliverables required by ARP4754 and agreed with the Authority. In particular, the validation data represented in Figure 3 was allocated to the deliverables as per Table 2: ARP4754 Design Data to Deliverable Documents Map. Only selected attributes of each data kind is added to each deliverable document.
3. Definition of a change control workflow, with each step mapped to a tool and to a user profile.
4. Definition of tools user profiles with detailed permissions, aligned to the workflow roles.

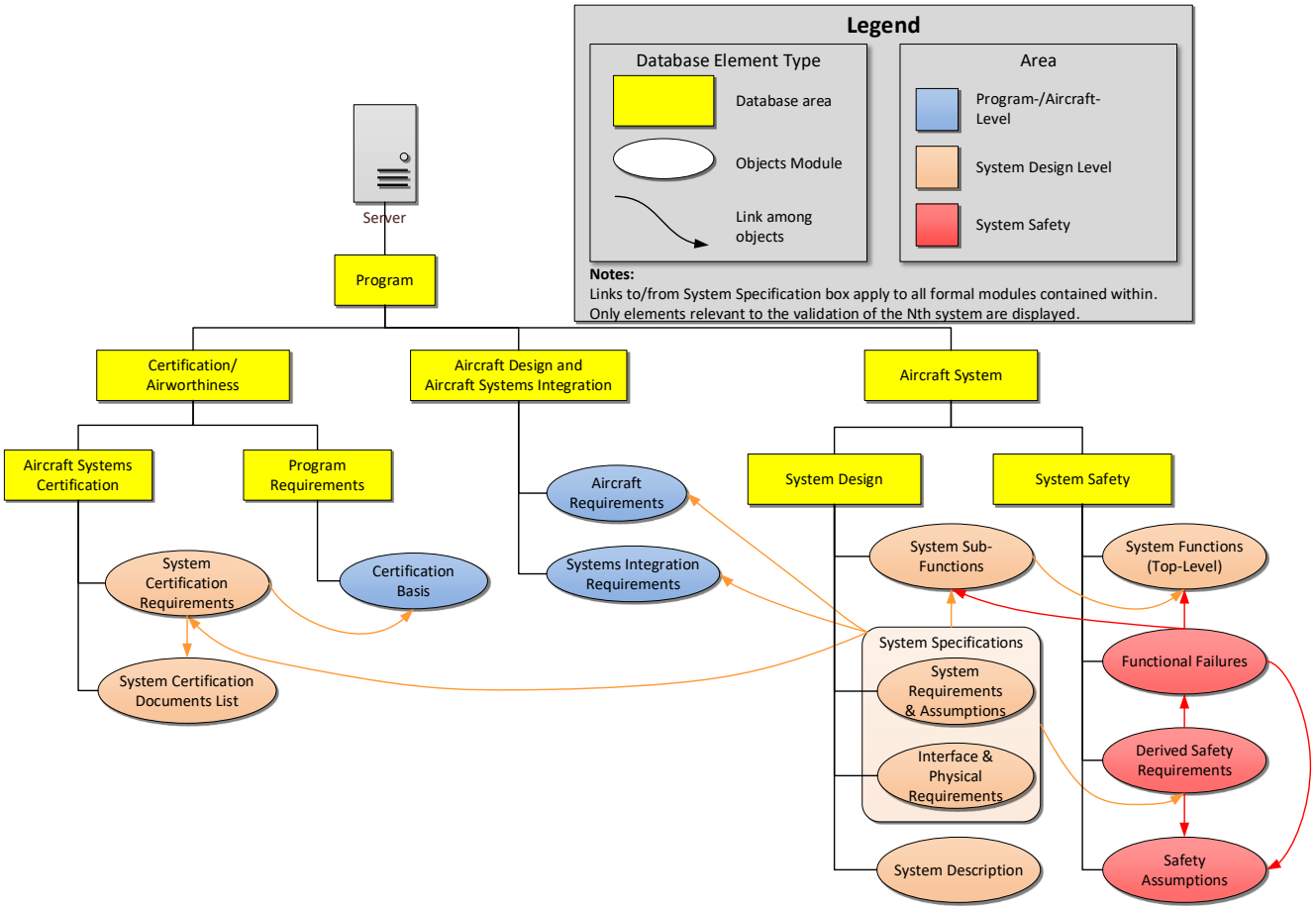

Figure 3: Requirements Management Tool Modules and Links Architecture

Table 2: ARP4754 Design Data to Deliverable Documents Map

ARP4754 Data in RMT	Data Description	Certification Plan	Configuration Index	Architecture and design description	FHA	PSSA	Requirements Specification	Validation Summary
System Certification requirements	*Certification basis requirements allocated to the system and their*	X						
System certification documents list	*as title*	X	X	X				
Certification Basis	*Laws and certification prescriptions levied by the Authority on the aircraft program*	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Aircraft requirement	*requirements on platform design: performance, capabilities*	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Systems Integration Requirements	*Requirements for functions allocation to onboard systems and architectural provisions*	N/A	N/A	N/A	N/A	N/A	N/A	N/A
System Functions (Top-Level)	*end-to-end system functions*			X	X		X	X
System sub-functions	*decomposition of top-level functions*			X	X	X	X	
Functional Failures	*Including attributes required to produce the FHA and reference FTA elements*				X	X		
Safety Assumptions	*Considerations to support functional failure severity classification and preliminary fault tree analysis*				X	X	X	
Derived Safety Requirements	*Constraints on system design emerging from PSSA analyses*					X		
System Requirements & Assumptions	*Statements and attributes*						X	X
Interface & Physical Requirements	*Statements and attributes*						X	X
System Description	*Description of the system design and architecture by both functional and physical points of view*	X		X			X	X

3.3. Validation matrix structure

During a validation process, there are two main streams that need to be kept under control: progress status and requirement categorization.

First, the status of the sub-function. Given that the sub-function is a cluster of requirements sharing the same functionality, the status of the sub-function depends on the status of each requirement. For each requirement it must kept track if the requirements have been validated, hence is valid, if is not valid or if the validation process is under work. Is also important to keep record of the status of the requirement itself, meaning if the requirement is completed, or has been

reviewed. This allows to have an immediate overall picture of the validation status.
On the other hand, is also fundamental to categorize each requirement and to track how it has been validated. Using both information, the validation process can be sped up taking or using similar validation evidence for requirement in the same category.
Per each requirement, it should be indicated how it has been validated, meaning the validation methods used: if traceability to an upper-level requirement is provided, if an analysis has been done, or a validation through simulation or test is available. Along with the simulation or the analysis, a track of result of the simulation or the documentation produced for the analysis should be recorded.
It must be made clear if the requirement is derived, if it is an assumption, or if it is refined from another requirement (from aircraft-level, interfacing system, etc.). In the latter case, the traceability needs to be easily retrievable.
To be fully integrated also with the safety process, it is important to record the DAL of the Sub-function and each requirement that pertains to it, indicating also if that requirement comes from a safety request.
To keep track of all this information, a validation matrix has to be produced, with at least the following attributes:

- Unique ID requirement
- Validation status of the requirement
- Requirement categorization (including pertaining FCS function and in case of integration/interface requirement the other applicable area)
- Validation methods used (Traceability, Simulation, Test, Analysis, Safety Request, etc.)
- Validation Rationale
- Validation evidences collection (including all the documentation produced or used to validate the requirement)
- Indicate if Derived
- Indicate if Assumption
- DAL
- Indicate if related to Safety

4. PERFORM VALIDATION

4.1. Validation of Functions

As explained in §3.1, with the FuSE approach the sub-function are the elementary bricks of the whole system validation activity. Each sub-function is validated establishing objective criteria based on the requirements allocated to the sub-function itself and its intended behavior at system level.
The validation process and methods are captured in the Validation Plan as requested by the ARP4754. Is useful to highlight some core aspects of the applied methodology:

- grouping the sub-functions in two main categories, to take advantage of the discipline specialists composing the validation team. In the FCS case study, main categories identified were input/output Redundancy Management) and CLAWs (Control Laws).
- traceability from each sub-function upward to the end-to-end FCS functions assessed by safety department in the safety analysis (e.g. FHA and PSSA). This kind of traceability guarantees the effectiveness of the integral process that iterates, starting from the system architecture/system safety, passing through the system requirements and lastly reaching the item requirements and development.
- retrieval of artifacts inherited from past activities (before application of ARP4754 was mandatory). Lots of evidence such as simulations, analyses and tests were conducted in the past to substantiate the basis of the technical project, hence containing the core of the validation aspects and the rationale of important design choices.
- requirements set allocated to a sub-function is reviewed against the architectural diagram of the same sub-function to contribute assessing the completeness and correctness of the requirements set itself.

Being compliant with the ARP4754, the presented approach addresses most of the standard's objectives to establish completeness and correctness of a given sub-function by mean of a dedicated checklist. Once the validation activity is completed, one checklist review exists for each sub-function of the system.
Note that all the findings/comments from the system reviewer and system safety reviewer are stored in the checklist form. Moreover, in case those findings are driving changes at system requirements' level or at FHA/PSSA level, a proper Engineering Change Request (ECR) is created, to guarantee that impacts on the overall system are determined and addressed.

Important aspects of the presented approach to ensure efficiency and effectiveness of the validation effort are:

1. the rationale to sort out the functions to be validated
2. assigning the roles of “requirements originator” and “independent reviewer” taking into account the level of expertise of the team members

Properly sorting and clustering the sub-functions on which the team concentrates the effort in a defined time period is important to maximize the focus of the whole team on a homogenous subject before switching to the next one. This kind of approach is beneficial also for the independent review performed by the system safety specialist who takes advantage from reviewing sub-functions related to the same end-to-end function.
The second aspect is about taking advantage of the different levels of experience and expertise of the teams’ components. It is usually beneficial assigning the role of “requirement originator” to relatively less expert persons, especially when the project is in an advanced development stage. This allocation allows to reduce the impact of “previous knowledge bias” and to ease arising a higher number of questions about choices and rationale about the system intended behavior and design. On the other hand, assigning the “independent review” to more expert team members has positive effects on the validation process since those people can effectively challenge the work done on the system requirements being deeply aware of the intended system behavior and its architecture.

4.2. Aircraft systems integration requirements validation

System requirements of interface and physical types (as defined in ARP4754) for FCS system and system elements are collected in the FCS Interface Requirement Specification (IRS), all other system requirements for the FCS system and FCS items are captured in the System/Subsystem Specification (SSS). Due to the pivotal role of the FCS system to provide or support aircraft functions, its design needs to consider functional constraints related to other onboard systems, including those not physically interfacing with FCS. To distinguish these constraints, the requirement type ‘integration requirement’ was added to the types list defined in ARP4754. Integration requirements are identified and collected at system level in the IRS. The structure of the relationship between different systems is depicted ion the following figure.
The reference document is the FCS IRS, which together with the SSS represent the system requirements specification, as per Figure 4. The IRS in particular fulfills two important targets: it defines the interface and integration requirements established at FCS level to be passed to other systems and includes the requirements that other systems are allocating to the FCS, in order to perform the requested aircraft-level functions. Other systems requirements are typically defined in the IRS of other systems, but they may be in other documents as SSS or even detailed design ones, especially for simpler systems.

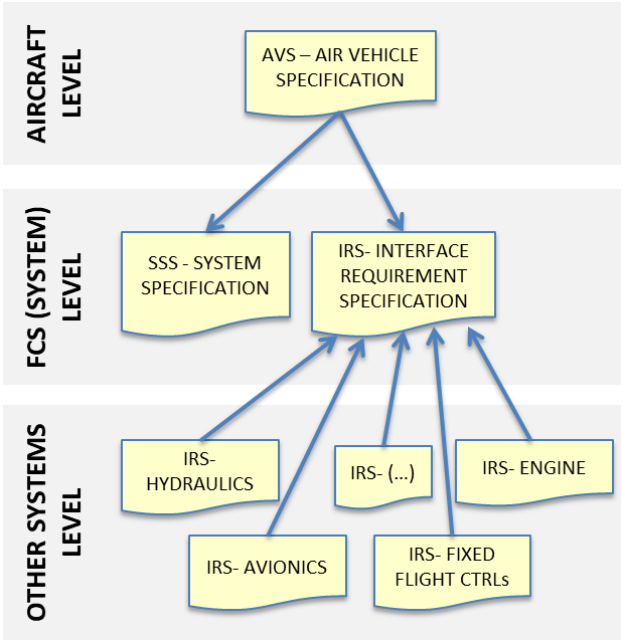

Figure 4: Interface Requirement Structure.

Validation of the system integration requirements is also addressed at function-level, structured per interfaced systems, as per the FuSE approach as applied to all other FCS requirements. Thus, even for these requirements, the validation is conducted by strictly following the checklist approach which is described in this paper.
The details of the process applied in the IRS case do not differ significantly by the checklist process applied to the functions of the system specified in the SSS, just its application is complicated by the need of the

reviewers to navigate through a wider range of documents of different sources.
The validation of the FCS IRS requirements in fact are mostly related to the traceability for those requirements which are dictated by characteristic (e.g. electrical or mechanical) typical of external equipment. In functional cases the requirements normally are not completely validated by traceability only, therefore a FCS level validation is needed (e.g. with closed-loop simulations).
In any case the requirements from other systems are captured by FCS IRS and from this point they are treated in the validation process in the same manner as for the other functional requirements, including all considerations related to DAL, derived requirements, assumptions.
Assignment of validation methods to each FCS SSS and IRS requirements is documented and tracked in electronic databases. The same database is used to document evidence of compliance, status of testing, analyses, etc. As the AW609 program progresses, validation results are collected against each requirement. The final validation compliance matrix then shows complete coverage for each FCS system requirement.
Management of the integration requirements needs a special attention due to its inherent complexity and multiple relationship to systems, subsystems or items. Therefore, a special attention is given to properly identify and trace all ramification of sources and contribution to the requirement itself and relevant validation artifacts. The FCS validation matrix is designed to clearly report for IRS requirements both the owning system and the allocation to aircraft systems and FCS subsystems. The owning system is the system that formulated the requirement in the first place, and levied it on other aircraft systems. In the system validation process scope, the owning system development team has the duty to justify the requirement existence, whereas development teams of any other system that has the requirement allocated need to trace the requirement source.

4.3. Concurrence of validation and safety assessment activities

The FuSE approach to validation is particularly efficient to carry out the validation process concurrently with the safety assessment process. In AW609 FCS development, validation process and safety process are carried out by two different figures: the system engineer and the safety engineer. Though both roles require a systems engineering and safety engineering background, including good knowledge of ARP4754 and ARP4761 processes, each one has broader expertise on its field. The two roles work side by side, carrying out in parallel the validation process and the safety process. To ease the collaboration between the two, safety requirements were split in two:

- Derived Safety Requirements (DSR), written in the Preliminary System Safety Assessment (PSSA) by the safety engineer, synthesize the safety prescriptions needed for the functional failures to meet the safety objectives resulting from PSSA analyses. Example prescriptions are redundancy, fault annunciation, etc.
- System safety-related requirements are system requirements captured in the SSS and IRS documents. They are refined by the system engineer from DSR, by including the information necessary to implement the safety prescriptions in the system design, such as monitoring tolerances, etc.

Thus, DSR are the justification of system safety-related requirements existence, hence the latter are validated by tracing to the former ones.
Figure 5 shows the DAL assignment mechanism to functions and requirements at various levels. Note that in the FuSE approach to validation, requirements pertaining to the same function inherit the same DAL. The system engineer fills out the attributes in the Validation matrix specific for DAL and if safety-related. During the safety review phase of the function, the safety engineer confirms per each requirement the DAL and if it is safety-related or not, and in case update the validation matrix

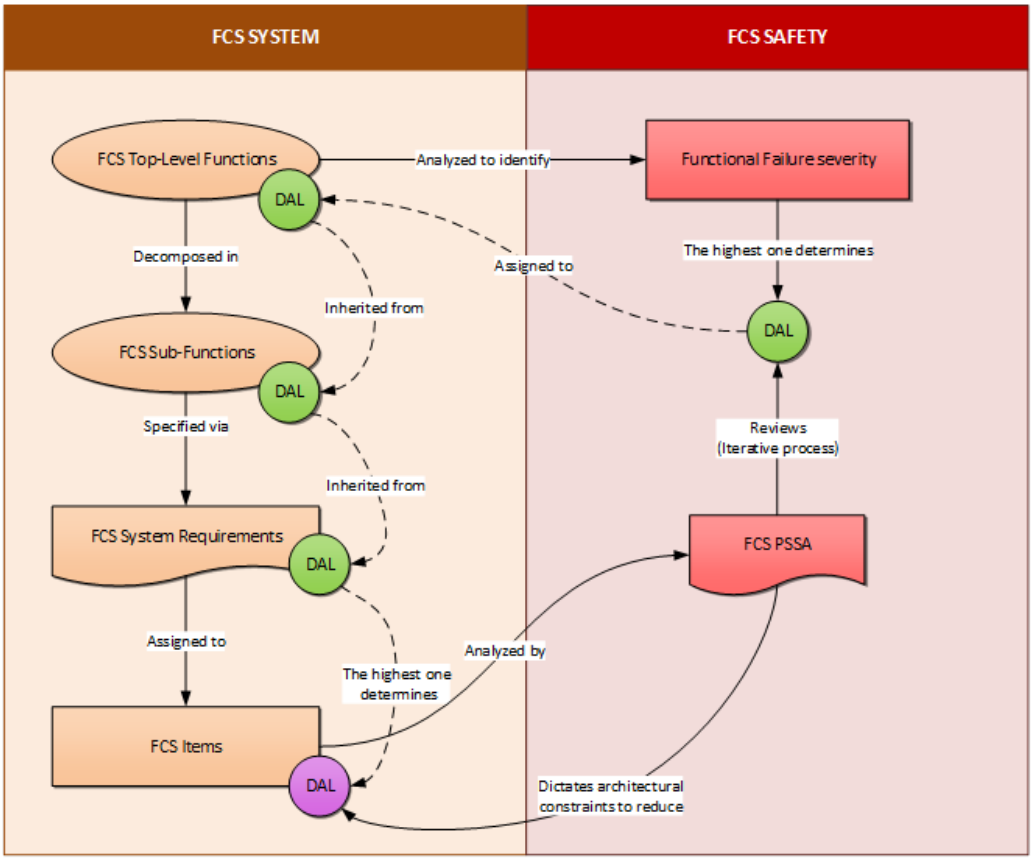

Figure 5: DAL assignment from Functional Failures.

4.4. Validation evidence selection from requirement types

As explained in §3.1, the requirements allocated to a specific sub-function have homogeneous DAL, driven by the severity of the parent function. For the "DAL A" sub-functions, a "third method" of requirements validation is requested in ARP4754 other than "traceability" (except for purely "Derived" requirements that must be always justified with a rationale) and "checklist review". The ARP4754 contains "Analysis, Modeling or Test" as generic options for the third method. Depending on the requirements' type, a "guideline/mapping" has been developed and reported in Table 3 for the most appropriate method to be applied and for the artifact that contains the validation evidence.

Table 3: Guideline for Validation Method selection

Requirement Type	Validation Method	Reference document/source of information
Customer Requirements	Inspection	Aircraft Requirements Document
Safety Requirement	Safety analysis	PSSA
Performance Requirements	Performance analysis OR Pilot in the loop lab test	Analysis Document Test report
Failure management *(sub-type of safety requirements)*	Tolerance analysis	Analysis Document ref
Physical requirements	Inspection OR Ground Test	Installation drawing to be inspected Test report reference
Interface and integration requirements	Inspection	Other systems' IRS Review Meeting Minutes

5. MANAGE VALIDATION RESULTS

5.1. Validation summary Structure

As per ARP4754 the validation summary is a report that mainly contains:

- Validation matrix
- Collection and description of any significant deviation from the validation plan

In the presented approach to system requirements validation, it has been agreed with the Certification Authority to capture this kind of report – as well as the verification summary – in the final System Safety Analysis (SSA). The background rationale behind this decision is that the whole validation effort is actually an integral and iterative process between system safety and system aimed to improve the global safety level of the aircraft.
Also, capturing the safety considerations/assumptions and the validation summary in the same document turns to be a great aid to assess the robustness and correctness of safety analysis of complex systems, such as the FCS of a FBW Tiltrotor.

5.2. Validation knowledge management

According to ISO 30401 (Ref. 7), knowledge management is any set of practices targeting the improvement of organizational performance obtained through knowledge creation and use In view of such definition, the overall requirements capture and validation process can be ascribed to the knowledge management discipline. More specifically, dedicated practices were employed in the validation of AW609 FCS requirements to capitalize the obtained knowledge to the benefit of the current and future programs. Among the many knowledge management strategies, the following ones were the most beneficial.

1. Preservation of all aircraft initial design objectives, including those not met by the final product. These objectives, stored together with their rationale and implementation status, provide valuable insights in terms of technical feasibility, lessons learned, and targets for the analysis of market competitiveness of future projects. To exploit the benefits, design objectives should be recorded together with their source, justification, achievement status, and any blocking points.
2. Definition of requirements templates for each category, based on source, functional properties covered, validation means of evidence, etc. Usage of templates eases validation review activities. Optimized templates are a value for any other programs.
3. Management of each requirement justification at multiple level of details, to cover specific needs. Requirements are typically captured as a result of design trade-off studies, past experiences, experimental testing, and other similarly complex activities. Requirements specification readers often do not have the time or the skills to look into the reports of said activities, that are referenced as validation evidences in the validation matrix. As a consequence, the requirement rationale could remain obscure or misinterpreted. To cope with this issue, for each AW609 FCS requirements, a brief justification paragraph was provided in the validation matrix, to summarize the more detailed validation evidences (that were referenced nonetheless). According to ARP4754, providing a rationale is mandatory only for derived requirements, but the team greatly benefitted from having it for all kinds of requirements.

6. CONCLUSIONS

The solutions presented in this paper were successfully applied for the certification process of AW609 onboard systems and approved by the Certification Authority. The main values nurtured by the proposed methodology are listed hereafter.

1. In a function-based approach, the validation subject is the function, i.e. a coherent subset of requirement to be reviewed as a whole, as opposed to isolated requirements review. Even though ARP4754 requires only to refer to top-level functions, decomposing them into sub-functions with adequate level of granularity is necessary.
2. Governance of validation activities, in terms of process definition, toolchain customization, requirements format and validation methods standardization is key to reduce effort and
3. Outputs of validation endeavor are used by different stakeholders, both internal and external, with variegated levels of understanding of the system.

Thus, it is important to store and share the information at the adequate level of detail via most appropriate the communication channels.

Despite the method was tailored for the first issue of ARP4754 and ARP4761, the authors believe that this work is a solid starting point to develop an approach compliant to the latest revisions. The function-based approach to validation is also prone to be complemented by model-based methods for systems and safety engineering.

Authors contacts:

Luigi Turco, luigi.turco@leonardo.com
Valerio Zanetti, valerio.zanetti@leonardo.com
Nicole Cardili, nicole.cardili@leonardo.com
Maurizio Parolini, maurizio.parolini@leonardo.com

7. REFERENCES

1. "Aerospace Recommended Practice (ARP) 4754 Certification Considerations for Highly-Integrated Or Complex Aircraft Systems" Society of Automotive Engineers (SAE), 1996-11-01
2. Hempe, D. W., "Development of Civil Aircraft and Systems," U.S. Department of Transportation Federal Aviation Administration, AC 20-174
3. "ISO15288 Systems and software engineering – system life cycle processes", International Organization for Standardization, First Edition, 2015-05-15.
4. Hutcheson, R. S., McAdams, D. A., Stone, R. B., and Tumer, I. Y., "Function-Based Systems Engineering (Fuse)," International Conference On Engineering Design, Paris, France, August, ICED'07/470.
5. Ackoff, R. L., "Towards a System of Systems Concept", Management Science, Vol. 17, (11), 1971.
6. "Aerospace Recommended Practice (ARP) 4761 Guidelines And Methods For Conducting The Safety Assessment Process On Civil Airborne Systems And Equipment" Society of Automotive Engineers (SAE), 1996-12
7. "ISO30401 Knowledge Management Systems – Requirements", International Organization for Standardization, First Edition, 2018-11-01.