INTERNATIONAL DESIGN CONFERENCE - DESIGN 2012
Dubrovnik - Croatia, May 21 - 24, 2012.

INTEGRATING SYSTEMS AND MECHANICAL/ ELECTRICAL ENGINEERING – HOW MODEL-BASED INTERFACE MANAGEMENT SUPPORTS MULTI-DOMAIN COLLABORATION

C. Tristl and A. Karcher

Keywords: model-based engineering, interface management, systems engineering, product lifecycle management, aircraft development

1. Introduction

In the next years Aerospace & Defence (A&D) industry will be particularly impacted by factors such as Climate Change, Energy price, Economics and Technology, further fueling the demand from customers to reduce development time and costs of products while new business requirements like network-centric interoperability lead to interdependent System of Systems (SoS) [Jamshidi 2009]. SoS deliver the required capabilities by combining multiple interacting systems, but at the cost of increasing complexity and uncertainty which directly reflects on their corresponding development processes [Browning 1998].

In order to design complex SoS like military aircraft in a tolerable time span, the different discipline-specific development processes have been parallelized, every stream managed rather independently. But this Concurrent Engineering (CE) paradigm conflicts with the iterative nature of interdisciplinary aircraft design requiring efficient cross domain information exchange. Therefore, these characteristics pose an important challenge to synchronized multi-domain collaboration, something traditional domain-separated engineering processes and heterogeneous tool environments cannot provide sufficiently [Broy et al. 2010] and hence, future integrated development processes have to focus on.

Daily business experience shows that especially during integration of domain-specific deliverables from Systems Engineering and Mechanical/Electrical Engineering the two processes under investigation, unforeseen problems can originate from unconsidered dependencies between the high number and types of system interfaces (e.g. logical, mechanical, electrical). Considering complexity as a structural issue, inconsistent system interface definitions fostered by interfaces that cross domain or company boundaries, can cause expensive rework and iterations in the development process [Browning 1998].

Consequently, by looking at a product and its development process within the context of distinct development phases, domains and interacting system elements Interface Management (IFM) on various dimensions is a key element for increasingly important multi-domain collaboration [Chen 2007]. Therefore, this paper presents an approach to develop a multi-dimensional model-based Interface Management Framework. It focuses on improving a holistic understanding of characterized interface issues (IFI) and provides a set of activities intended to help engineers and process managers to address these issues already during an early development stage in a consistent way on the dimensions Communication, Process-, Tool- and Product Data, using the emerging methods and tools of Model-based Engineering (MBE) and Product Lifecycle Management (PLM).

The paper is structured as follows: In Section 2 the background and motivation is presented by discussing future challenges in A&D. Further, the complexity of a concurrent and highly iterative aircraft development process will be described and the potential of Virtual Product Development analyzed as a unique reference for integrated product data. Section 3 points out the specific research questions addressed in this paper and explains the research methodology being used. Section 4 explores the interrelated key factors leading to interface issues and their root causes. Activities on multiple levels resolving the identified II will be subject of investigation in Section 5. Within this Section the multi-dimensional IFM approach will be explained and how only aligned concepts from the state of the art mapped onto a framework can substantially enhance collaboration between different engineering disciplines. Finally Section 6 draws the conclusions of this paper and describes further envisaged research.

2. Background

2.1 Challenges in aerospace & defence

One of the future challenges is the integration of new elements and technologies with varying Product Lifecycle (PLC) durations (e.g. avionics software) into existing systems, having an overall PLC of about 30 and more years in mind. High product costs are not always caused by expensive product components, but more often by the complexity of the SoS. System functions are often realized explicitly by the synergetic integration of work results from different domains. Elements of risk are especially missing or incomplete requirements, functionalities or interfaces leading to unforeseen interactions and thus unintended behavior of the system.

The OEM-supplier relationship within the A&D industry is evolving more towards a Value Network Chain. For efficient development of complex projects with many distributed partners an orchestrated and transparent collaboration (across-functions and across-divisions) is needed. Counterproductive to the goal of lead time reduction, the effort to coordinate the value-streams of separated development processes also increases due to domain specific local processes, methods and heterogeneous IT infrastructures which underlines the need for effective synchronization between participating domains [Arnold et al. 2011].

2.2 Interdisciplinary aircraft development process

How well a product performs in the end can be related directly to its development process. In general, development processes are ways of organizing and conducting product development projects, massively contributing to the product life cycle cost by defining the product characteristics during development. In accordance with Browning [1998], product development can be summarized in simplified terms as: The customer's requirements are *satisfied by* a product that is *developed through* a product development process *implemented by* a development organization.

The engineering of mechatronic systems like aircraft or cars requires integrated processes, methods and tools which can successfully address the product complexity. Systems Engineering (SE) is a fundamental discipline intended to embody these parts. As defined by the INCOSE Systems Engineering Handbook [2010], SE is an “interdisciplinary approach and means to enable the realization of successful systems”, focusing on coverage of customer needs and required functionality early in the development cycle, documenting requirements, and then proceeding with design synthesis and system validation while considering the complete problem. Therefore SE incorporates both technical and management processes. The boundaries and encapsulated elements of a *system* are defined by its context and level of perspective/interest. Decomposition is an important technique to address the complexity of a development project through breaking it down into smaller manageable fragments. According to “Systems thinking”, while aircraft systems engineering primarily focuses on *aircraft* as a system, other levels of interest may focus on aircraft *subsystems* (Hydraulic, Structure,…) as a system or on *Components* (Sensors, pumps, computers…).

For new aircraft development projects concurrent and iterative exploration of problem and solution space is typical. Rather than delivering “first time right” information - which is not available so early in the development life cycle for these complex products - initial assumptions about the aircraft design are incrementally refined on the way towards an global optimum within the interdependencies of involved engineering disciplines like Flight Control, Aerodynamics, etc. as described in Figure 1. The challenge is to manage the concurrent but correlated activities efficiently, avoiding waiting time [Autran et al. 2012]. These interdependencies are a major source for uncertainty and risk [Browning 1998] and can cause expensive iteration cycles through rework if errors are identified at a late development stage.

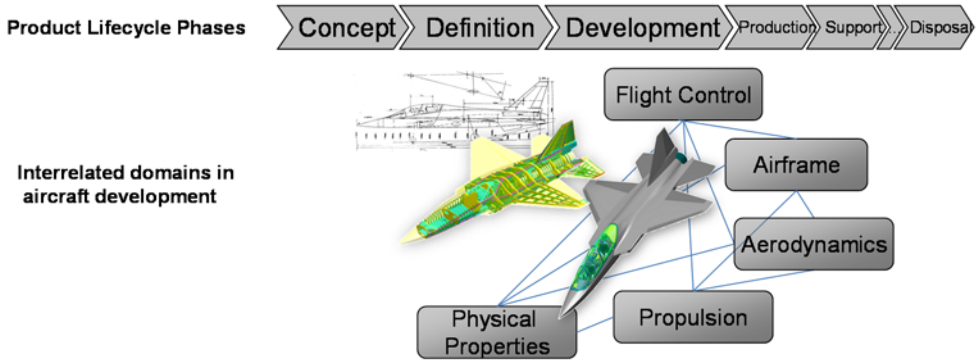

Figure 1. Product life cycle phases and interdependent domains during aircraft development

In the following, the two disciplines under investigation, Systems Engineering (SE) and Mechanical/Electrical Engineering (M/EE), shall be briefly characterized and the challenge of their integration pointed out.

SE follows a function and property oriented top-down approach dealing with the system requirements/functions, the system architecture and the decomposition into system elements and their interfaces, specification of required system properties, as well as the development of software-intensive electronic equipment. The output of SE is input to the subsequent M/EE process.

M/EE follows a geometry and assembly oriented bottom-up approach down to implementation level where the specified systems (e.g. engine control computer) are installed into the aircraft under certain constraints (e.g. electrical design standards, power supply, communication interfaces, mechanical structure/ available space, etc.)

Current domain-specific processes, design models and tools show some shortcomings when it comes to engineering information exchange between each other. Often information about components is managed in different domain-specific product structures or IT systems and SE work products like system block diagrams or specifications are structured differently than M/EE work products like wiring diagrams, MCAD or ECAD models. Thus, representations and attributes of common components between SE and M/EE cannot be easily linked and therefore collaboration between these domains still involves much manual work. The fact that most knowledge about how collaboration between these domains is working exists only implicitly and is relying heavily on personal experience, demands for a formalized framework integrating and optimizing the communication and data exchange between all participating disciplines [Arnold et al. 2011]. While local processes are already optimized to a certain extent, interfaces between different disciplines bear great improvement potential to reduce product development time and costs.

2.3 Virtual product development and model-based engineering

Virtual product development (VPD) promotes the vision of a seamless fully digital product development process elevating virtual product models to the core of the concept to reduce dependence on physical prototypes. A prerequisite for efficient VPD are harmonized processes, tools and an interdisciplinary product model to help understand system behavior and interdependencies throughout the lifecycle phases across many engineering domains. According to STEP- ISO 10303 a product model is a formal specification of any product related information through the product lifecycle.

One major challenge in system decomposition are logical interfaces, emerging from the iterative architecture definition (within SE) in concept phase and their gradual refinement down to physical interfaces (e.g. data buses) at lower levels (within M/EE).

Today's development is characterized by document driven and standalone models, revealing an important problem of information technology: Fragmentation and inconsistency of data through separated development phases and processes across disciplines and across organizational boundaries which leads to Islands of Knowledge & Information and finally to problems with traceability and reliability of engineering information over the PLC.

One way to confront these challenges in the future is to use a Model-based Engineering Environment with integrated processes and tools providing discipline oriented views on a common product model as shown in Figure 2. MBE is an approach to engineering that represents all relevant technical aspects of a system as a virtual model instead of a set of documents. A technical baseline is therefore established by a consistent set of models for Requirements, Functions and Architecture to be used for analysis, design, implementation, verification, simulation, certification, qualification and operation across the whole PLC.

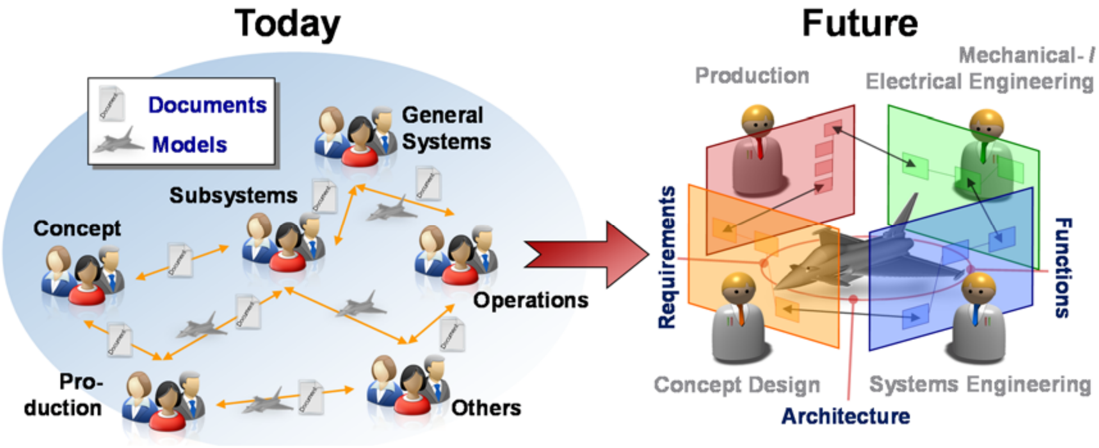

Figure 2. Paradigm shift from document-based towards integrated model-based engineering

Product Lifecycle Management (PLM) plays a crucial role in MBE by collecting all product data and serving as the single source for consistent information for all other product-related processes. It further intends to provide full traceability of dependencies between information of various domains from different local IT-systems [Autran et al. 2012]. Still Subject of current research is how to efficiently map MBE onto a PLM framework by elevating models to a central role in specification, design, integration and validation of systems. The interdisciplinarity of today’s and future products require appropriate new ways of development. MBE and PLM together could provide a digital model to the VPD, as a unique product data reference point for all product or service related disciplines.

MBE is subject of research since many years, having created a large body of concepts. However, the everyday use of sound model-based approaches in Aerospace or Automotive is still limited. The absence of seamlessly integrated tools and harmonized process chains which allow for the cyclic reuse and refinement of models, subsequently lead to problems like redundancy, inconsistency and lack of automation [Broy et al. 2010]. Therefore, one step to overcome the lack of common understanding between development processes like SE and M/EE is an integrated product- and process model, enabling the interdisciplinary and concurrent data processing supported by transparent processes.

3. Research questions and methodology

The challenges described above lead to the following main research questions which will be explored in the remainder of this paper:

1. How to enhance collaboration between separated Systems Eng. and Mech./ Elec.-Eng. processes in order to avoid integration issues discovered late at system integration?

2. How to improve interface definition and management in a Model-based Engineering Environment, based on consistent product data from a PLM System?

We intend to explore these questions as described in Figure 3. In order to define a theory, a case-study concerning an equipment modification and replacement on an aircraft was chosen to describe the challenging industry-related problem. Expert interviews and literature research verified that Interface Management has not been solved in an integrated manner satisfactorily in Academia yet and it is also relevant to none A&D companies, as the work of Chen [2007] show.

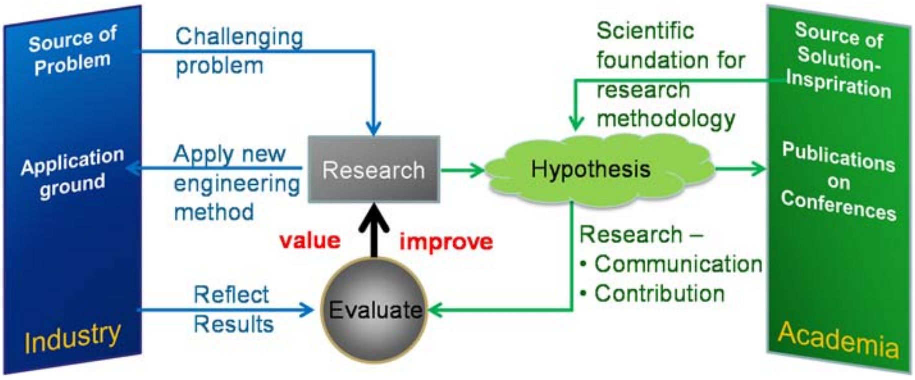

Figure 3. Applied research methodology (based on [Muller 2011])

While industry provides the possibility to investigate the challenges of interdisciplinary product development, Academia acts as the source of solution-inspiration through offering a scientific foundation for a problem-related research methodology. Besides contributing to Academia, this work adds practical value by focusing on successful transfer of new methods from Academia to applicable new practices in Industry.

4. Exploring key factors causing interface issues

According to Chen [2007] *Interfaces* (IF) can be defined as common boundary between two or more system elements (e.g. functions, physical elements, processes, organizations) enabling the exchange of e.g. information, material or energy. Basically, interfaces have two simple characteristics: a terminal at each end and a media of communication in between.

Applying a “comprehensive modeling theory” [Broy et al. 2010], virtual product models could help to address complexity in system architecture definition by consistently capturing all relevant product data from high level system requirements to definition of system elements interdependencies and their logical interfaces and allocation of functions to system elements. The complementary activity to decomposition is system integration and test of components and subsystems to verify compliance with the previously defined specifications [Muller 2007]. As illustrated in Figure 4, a major challenge in system decomposition are logical interfaces, emerging from the iterative architecture definition and their gradual refinement down to physical interfaces (e.g. data buses) at lower implementation level. Because crosscutting functionalities and qualities suffer from this decomposition, results can be a lack of responsibility and communication across organizational boundaries which are root causes for interface issues, finally the source for integration issues as stated by Muller [2007].

In order to understand which factors spark IFI, a comprehensive view is needed. As literature reviews showed, Interface issue related research mainly focuses on one specific area like physical, intercompany or process interfaces, thus missing the important relationships of cause factors. Therefore the IFI cause and effect diagram method for construction building from Chen [2007] was adopted to investigate main causes of IFI from the systems engineering perspective, extended by findings from Muller [2007], company expert interviews and further literature studies.

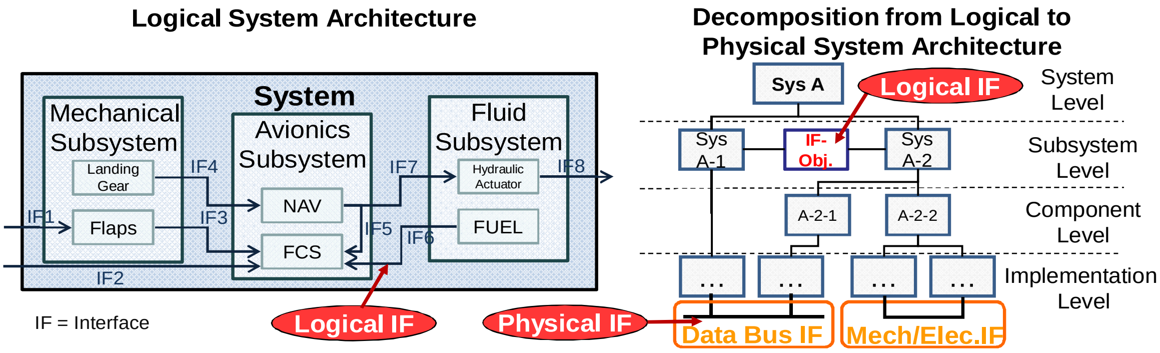

Figure 4. Gradual refinement from logical to physical system interfaces

Based on these findings, the collaboration between the two processes SE and M/EE was analyzed through conducting a case study about an equipment modification and replacement use case according to Form-Fit-Function with the goal of investigating interdisciplinary communication practices, interface classification, association of main causes leading to interface issues and identification of relationships. To identify relevant interfaces, system elements around the system of interest were visualized in a Digital Mock-Up (DMU) based on geometry data from a PLM system. Additionally, system block diagrams and product structures were compared. To make the implicitly known interdependencies between the requirements, functions, logical/physical architecture, work products, processes and resources explicit within the case study, an Multiple-Domain Matrix (MDM) [Biedermann 2008] was created according to the procedure shown in Figure 5, representing the As-is situation based on interviews and existing domain-specific partial process models.

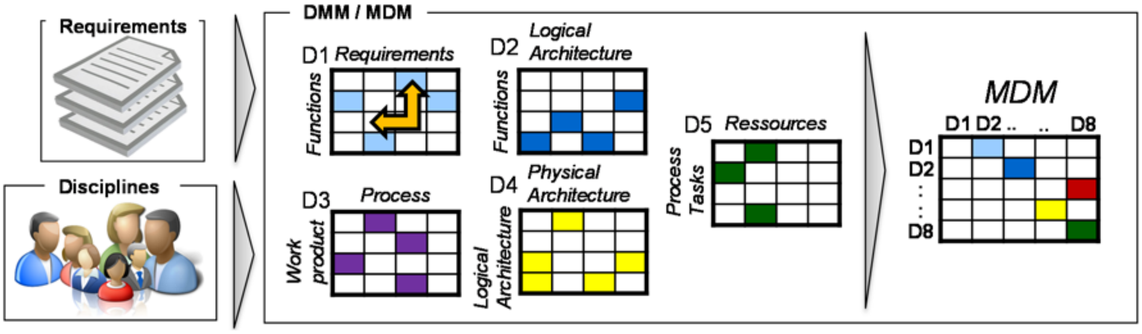

Figure 5. Sequence for making interdependencies explicit via a MDM

Aiming at the relationships, questions were raised like “Can a certain IF issue be traced back to shortcomings in collaboration between development process, their supporting tools or databases or data inconsistencies due to misinterpretations of system information spread across many documents?”. The findings and their relationships were captured in one Design Structure Matrix (DSM) [Browning 1998] per main cause. Due to space limitations Figure 6 shows the data briefly summarized in a cause and effect diagram. *Major causes* like domain-separated development process contribute to the *Main Cause* Processes, Practices & Methods and finally to the *Effect* or *Problem* Interface/Integration Issue. These factors represent the point of action for further investigations for counter measurs.

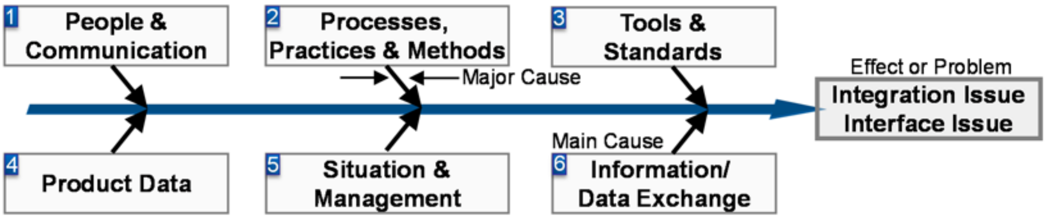

Figure 6. Key factors causing interface and integration issues

Consequently, the use case has underlined that synchronized development processes, based on common integrated process- and product models providing consistent interface data are critical for successful system development [Eversheim et al. 2005], [Chen 2007].

5. Approach to address model-based interface management on four dimensions

5.1 Successful combination of partial solutions

Section 4 revealed that Interface/Integration issues can have multiple interrelated main causes, thus the problem needs to be addressed on several dimensions in order to enhance the collaboration between different development processes. Literature studies showed that many concepts and technical optimized stand-alone solutions have been developed around VPD, focusing on one of the dimensions. But only a holistic approach can help to enable partial concepts from academia to be successfully applied to VPD in industry, where the complexity of a product can be another limiting factor.
By using the weighted objective method it was decided to focus on the four dimensions Communication, Processes, Tools and Product data, outlining the concept for a holistic Interface Management approach on multiple levels and serving as a framework for goal-oriented integration of existing partial solutions.
The presented approach in Figure 7 intends to describe a prototypical mapping of existing concepts for interface related problems onto the Framework, making their relative role and their individual but interdependent character more explicit. The main objectives are briefly summarized as follows:

- Synchronization of existing Systems and Mechanical/Electrical Engineering processes to support multi-domain collaboration in order to avoid system integration issues
- Exchange of consistent system interface data (power consumption, dimensions, mass, cooling, signals,…) across disciplines throughout the product lifecycle
- Embedding of partials solutions within methods of Model-based Engineering and Product Lifecycle Management

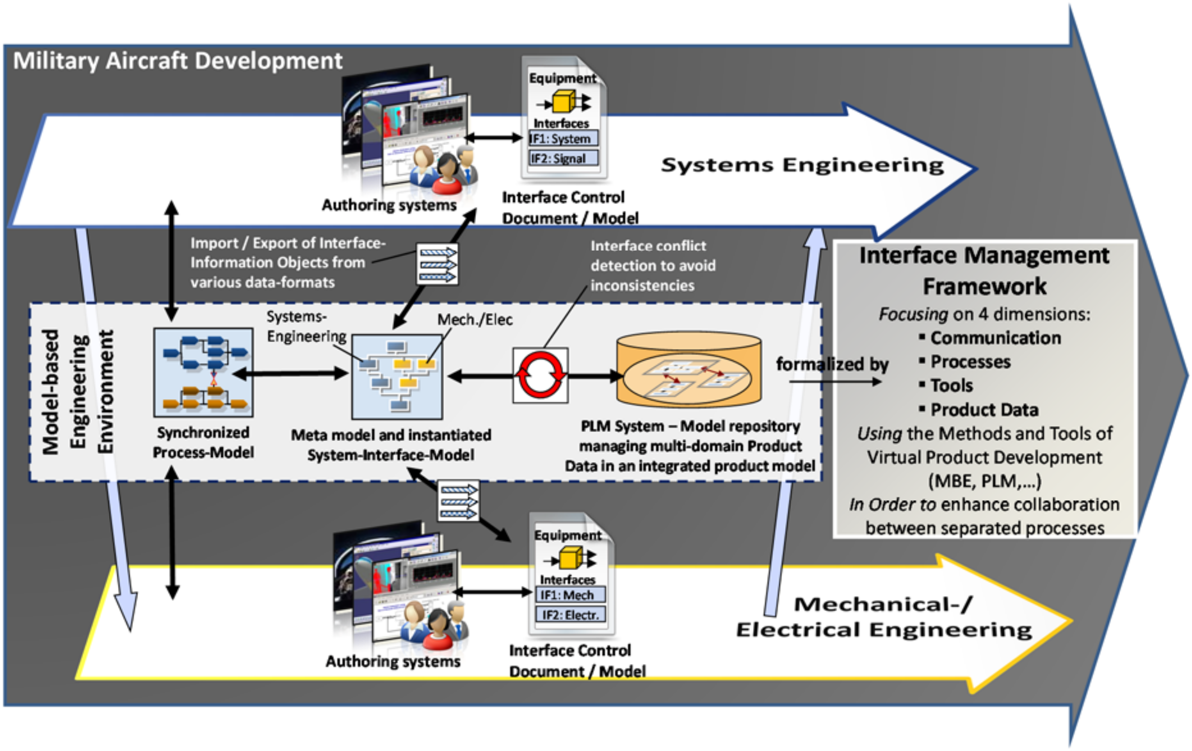
Military Aircraft Development
Equipment
Interfaces
IF1: System
IF2: Signal
Systems Engineering
Authoring systems
Interface Control Document / Model
Import / Export of Interface-Information Objects from various data-formats
Interface conflict detection to avoid inconsistencies
Interface Management Framework
Focusing on 4 dimensions:
▪ Communication
▪ Processes
▪ Tools
▪ Product Data
Using the Methods and Tools of Virtual Product Development (MBE, PLM,…)
In Order to enhance collaboration between separated processes
formalized by
Model-based Engineering Environment
Systems-Engineering
Mech./Elec
Synchronized Process-Model
Meta model and instantiated System-Interface-Model
PLM System – Model repository managing multi-domain Product Data in an integrated product model
Equipment
Interfaces
IF1: Mech
IF2: Electr.
Authoring systems
Interface Control Document / Model
Mechanical-/ Electrical Engineering

Figure 7. Interface management framework focusing on the four dimensions communication, processes, tools and product data

Just considering the technical dimension is not sufficient when dealing with large systems, developed by engineers from various disciplines. Efficient collaboration also depends on many influences and versatile components like communication and common goals. Therefore it must be remembered that people are in the center of collaboration, exchanging information and data via communication. Because *information objects* can contain a huge amount of data that has to be shared, the activity has to be supported by IT-tools with underlying databases.

This paper further focuses on the dimensions Processes, Tools and Product Data, due to space limitations, but not without underlining the importance of a sound concept of goal-oriented interrelating activities on *all* four dimensions.

5.2 Processes

Processes cover the product design by linking sub process, tools, data and people with their project specific roles reasonably together in order to structure the successful development of a product. In order to support the collaboration of parallelized but interacting domain-specific processes, a process-model will be defined which supports the systematic information exchange via synchronization points. Synchronization points can consist of interdisciplinary milestones, reviews or quality gates and are used to exchange information as early as possible in the development lifecycle to assure that the information provided satisfies the needs of the recipients. Thus, rework caused by missing information or not properly considered uncertainty is minimized.

In the following example the synchronization points shall support the regularly exchange of interface data between SE and M/EE development activities. As Arnold et al. [2011] state, especially quality gates support the splitting and realigning of disciplines without taking the necessary freedom of iterations. Further research will be attempted to evaluate the idea of Browning [1998] to use a design structure matrix (DSM) to analyze and modify the interfaces in a process, focusing on iterations and uncertainty.

According to Eversheim et al. [2005] the idea of an integrated process and product model is not new in academia but the efficient implementation has just started yet. Therefore, a visualization of interdisciplinary value streams linked with product data and development responsibilities support a common understanding through a transparent view. But considering the concept “understanding through visualization” in isolation, it does not support the actual “doing” of synchronization or simplifying access to information, if error prone manual data conversion is dominating the transfer of information between disciplines. Therefore it has to be mentioned that measures on the process level only are not sufficient without being supported by an appropriate data model and an integrated tool chain.

5.3 Tools & product data

Interface Management today involves a lot of manual work, coordination, mediation and searching for dependencies in documentation from different disciplines. Information is kept in several domain-specific tools and data formats, in modeling or special interface control tools. In general these “islands of automation” make it hard to trace certain information throughout the PLC over domain or organizational boundaries, due to a missing integration, local processes and incompatible tools. So, in a document-centric approach, interface data is cross-linked to all other relevant Interface Control Documents (ICD). In the future model-based Interface Management should be embedded in a harmonized Model-based Systems Engineering approach, where models are cross-linked to product data provided by a PLM system and downstream processes can access the information without barriers. But data consistency remains a hot topic, as consistency between interfaces data in models and/or documents has to be ensured.

In order to move from a document-centric towards a model-based development approach, a comprehensive Model-based Engineering (MBE) strategy is needed, intending to use models as a means for efficient investigation of design solution and means of communication between different disciplines. In order to develop a common culture of MBE, a Strategy was developed in order to align the different disciplines covering the three main areas *Common Methodology & Processes,* *Harmonized Tool-Environment* and a *Trained Workforce* to enhance a global understanding of Interface Management.

As stated in Section 2.3 VPD requires a model-based methodology for a seamless fully digital product development process. Therefore the MBE-Strategy is intended to be based on a PLM framework providing consistent product data from a single source, accessible by different disciplines. The intention in this context is to use the PLM system as a model repository managing and providing multi-domain product data within an integrated product model. Thus, the formal control of interface information is a means for a successful integration of complex systems. So a process-oriented integration of capable tools working on domain-specific views of models, based on MBE and PLM, are the foundation for seamless virtual product development.

In order to provide consistent interface data to engineers based on models or documents, the tools described as "Authoring systems" in Figure 7 have to be coupled in different ways to a consistent meta-model. While the tools could be coupled by e.g. converters transferring data between different data formats, or by exchanging information by using standardized data formats like ISO STEP ISO 10303 - AP233 for systems engineering or AP239 considering the PLC, documents could contain interface data embedded within XML objects.

The developed process model is also input for development of a Meta-model, describing the relations between interface information-objects from different disciplines. In this way interface data in different product artifacts can be managed. Thus it eases concurrent engineering by separation of concerns via multiple views on integrated but consistent models. In this use case, the instantiated meta-model is intended to capture interface data of a system having relations in the SE and M/EE processes, supporting traceability of data and thus, facilitating impact analysis of requirements and design changes. As the aircraft development was described before as highly iterative, starting with uncertain requirements, the meta-model needs to be able to account for the status of certain interface data. The general steps to develop a meta-model are outlined as follows:

- Identification of the domains to be involved in the model
- Identification of the viewpoints and views to be manipulated by the stakeholders
- Based on the processes used by the different disciplines to develop the product, identification of the data and information objects to be shared
- Identification of the common data and information objects and of their relationships
- Specification and implementation of the meta model

As described in Figure 7, the consistency between the instantiated system interface model and the consistent data model from the PLM system shall be maintained. Therefore an interface conflict detection and mediation tool shall be investigated where responsibles of a certain interface are notified about the conflict and offered a solution or at least help identify the responsible of the interface object (see Figure 4) or the terminal on the other side of the interface relying on deviating data. While the DMU clash analysis method is established in industry to detect geometrical interface issues, a sound approach for Systems Engineering (logical, electrical interfaces) has been missing yet. Therefore the four dimensional IFM Framework is intended to contribute towards consistent interface definition during interdisciplinary concurrent development projects.

6. Conclusion and further research

Today's product development is defined by the complexity in products, processes and a global collaboration via a distributed Value Network Chain. The ability to synchronize processes like Systems Engineering and Mechanical/Electrical Engineering running in parallel into an aligned value-stream is a key success factor for industry to address the concurrent, iterative and multidisciplinary challenges of today's aircraft programs. In practice, the efficient virtual product development suffers from a lack of harmonized processes and tools, fostering inconsistent system interfaces and finally expensive rework during system integration. Therefore, to improve collaboration of involved domains, this research focuses on establishing a framework combining state of the art solutions on different dimensions in order to address interface issues (IFI) and their root causes. For that purpose dependencies between the process and product-models were analyzed using the MDM method and an IFI cause and effects diagram was developed within this paper and applied to an equipment modification and replacement use case in order to verify the identified key factors. Associated domain expert interviews confirmed that the proposed IFM Framework concept is useful to improve synchronization of designs. To the current state, some parts of the Framework concept are not yet completed and will be tackled in further research work:

- Storage of interface data within a PLM system connected to domain-specific product data is not yet solved
- IF conflict detection program is still in conceptual state
- A central issue is how information from various domain-specific product models/ documents can be transferred from/to the System-Interface-Model

This paper presents an approach to develop a four-dimensional Interface Management Framework, aiming to contribute towards a holistic understanding of key factors for IFI and their relationships. As soon as IFI are characterized, a set of activities is provided helping engineers and process managers to address these issues already during an early development stage as a whole on Communication, Process, Tool and Product Data level. Future work will focus on seamless implementation of these partial IFM concepts, embedded in the emerging methods and tools of Model-based Engineering (MBE) mapped onto a PLM framework. In a subsequent step an evaluation phase must investigate the benefits and limitations of the presented approach in an industrial product development context.

References

Arnold, V., Dettmering, H., Engel, T., Karcher, A., "Product Lifecycle Management beherrschen: Ein Anwenderhandbuch für den Mittelstand", Springer, 2nd revised edition, Berlin Heidelberg, 2011

Autran, F., Scheithauer, D., Introducing Systems Engineering views in Product Lifecycle Management, 22nd Annual INCOSE International Symposium in Rome, July 9 – 12, 2012

Biedermann, W., Lindemann, U., Cycles in the Multiple-Domain Matrix Interpretation and Applications, In 10th International DSM Conference 2008, Stockholm, Sweden. München: Hanser Verlag 2008

Browning, T., Modeling and Analyzing Cost, Schedule, and Performance in Complex System Product Development, Massachusetts institute of technology, 1998

Broy, M., Feilkas, M., Herrmannsdoerfer, M., Merenda, S., Rati, D., Seamless Model-Based Development: From Isolated Tools to Integrated Model Engineering Environments, In Proceedings of the IEEE, pages 526 - 545, volume 98, number 4, IEEE, 2010

Chen, Q.,"An object model framework for Interface management in Building information models",Virginia Polytechnic Institute and State University, 2007

Eversheim, W., Schuh, G., Integrierte Produkt- und Prozessgestaltung, Springer-Verlag, Heidelberg, 2005

Jamshidi, M., System of Systems Engineering - Innovations for the 21st Century. John Wiley & Sons, Inc., 2009

Muller, G., Coping With System Integration Challenges in Large Complex Environments", Embedded Systems Institute, published by INCOSE, 2007.

Muller, G., Research Question and Hypothesis, The Gaudí project, Embedded Systems Institute, 2011.

Dipl.-Ing. Christian Tristl
PhD Student
Cassidian – An EADS company
Rechliner Strasse
85077 Manching, Germany
Email: Christian.Tristl@cassidian.com
URL: http://www.cassidian.com