AIRCRAFT CABIN ARCHITECTURES INCLUDING TOLERANCING USING A GRAPH-BASED DESIGN LANGUAGE IN UML

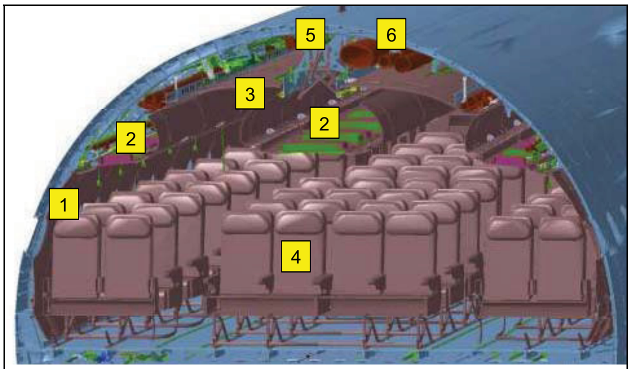
B. Landes, Airbus Operations GmbH, Kreetslag 10, 21129 Hamburg, Germany
S. Rudolph, Institute for Statics and Dynamics of Aerospace Structures,
University of Stuttgart, Pfaffenwaldring 27, 70569 Stuttgart, Germany

Summary

Current conceptual aircraft cabin architecture trade-off studies are traditionally done manually and are extensively based on experience and heuristics. This paper emphasizes that there is a need for formal methods to support the systematic analysis and evaluation of mechanical aircraft cabin architectures. Mechanical interfaces including their functional description act hereby as key integration data, and it is shown how tolerance management methods can be used as milestone for such an engineering analysis. Due to the multi-domain product data, which has to be considered – i.e. geometry data, physical data, functional data, tolerancing data, and assembly process data – the need for an abstract and formal model-based approach is described. Using these frame conditions, an implementation using a graph-based design language is proposed. A so-called ‘cabin design language’ in UML (Unified Modeling Language) is developed, offering multiple data visualizations in order to support design analysis and evaluation.

1. INTRODUCTION

Traditionally, the overall aircraft is divided into so-called ATA chapters [1], which represent a rule-set for technical aircraft documentation by breaking down all physical components and systems into sub-systems and by standardizing the functional allocation to design solutions. Each ATA chapter fulfills distinguished aircraft functions. For the aircraft cabin under consideration here, the major ATA chapters are the cabin modules (described in ATA 25), the supporting airframe substructure and brackets (mostly described in ATA 53) and lots of cabin systems like power supply (ATA 92), cabin intercommunication systems (ATA 44), or air supply (ATA 21), to name a few.



1: *Sidewall lining panels*
2: *Overhead stowage bins*
3: *Ceiling panels and air outlets*
4: *Passenger seats*
5: *Sub-structure for cabin module and system integration*
6. *Cabin systems (tubing, wiring etc.)*
(Cabin modules not shown: e.g. Galleys, Lavatories)

Figure 1: Typical Aircraft Cross Section with ATA 25 Cabin Modules, Systems and Airframe

The ATA 25 cabin modules, as shown in Figure 1, act as a visual interface between aircraft and passenger, as a ‘front-end’ for cabin systems or as functional devices for various purposes and, sometimes, as a pure cover for the airframe behind. Most cabin modules host components of further cabin systems from other ATA chapters and are attached to the fuselage using auxiliary substructures and brackets. The focus of this paper lies on these so-called mechanical interfaces between the cabin modules and the airframe and the corresponding brackets.

1.1. Paradigm Change for Cabin Engineering

In the global aircraft industry, many suppliers produced in the past single parts or small equipment, which had to be integrated into bigger cabin parts or even installed one by one into the aircraft directly. In this way, the aircraft manufacturing company was responsible for the definition and specification of almost every single part.

Nowadays, for the aircraft in general and for the cabin perimeter in particular, there is a strong tendency to profit from concurrent or as it is also often called ‘simultaneous engineering’, multi-ATA-chapter module development, requirement-based product specification and to employ a global policy of risk sharing partners and outsourcing [2].

However, recent production challenges in the aircraft industry tell an important lesson to learn: with a new way of work sharing it is of highest importance to understand the interdependencies between different work packages or modules and to manage the interfaces between the various involved ATA chapter responsibilities. In other words: with a changing design and production philosophy the focus of the aircraft manufacturers moves from the former detailed design towards responsible architecture definition and interface management.

1.2. Mechanical Cabin Architecture Trade-off Studies as Research Problem

For aircraft primary structure and in the field of cabin *systems* architecture – especially within single ATA chapters – you find good industrial experience in the field of modeling, simulation, and evaluation [3, 4] to anticipate the repercussions of architecture decisions.
The preparation and the execution of parameter and topology trade-off studies of conceptual *mechanical* cabin architecture scenarios are currently still done manually and are therefore relatively time-intensive. For the mechanical design part of cabin modules (ATA 25) it is state-of-the-art to create multi-model approaches manually: engineering iteration loops or trade-off studies are performed by the respective involved special engineering disciplines take several weeks and except for joint CAD models, individual data models are used or heuristic methods are followed.

The potential evolving shortcomings for the engineering work can be anticipated as follows: increasing communication efforts with an increased number of involved disciplines, the problem of data consistency, and the difficulty of model enrichment with further domain information, to name a few. On top of that, the required time to execute model-based multi-domain iteration loops or trade-off studies manually often leads to a decision based on time constraints. Linking the design data to cabin systems data models outside ATA 25 responsibility is even more complex.

Looking at the complex correlations between the involved engineering disciplines and facing the aforementioned paradigm change, it is obvious that there is a good opportunity for formal methods to support cabin architecture trade-off studies for decision finding during the concept phase in general and for the mechanical integration work in particular. Within this paper, the following three derived solution steps for such studies are investigated and proposed:

Solution Step 1:
Identifying the involved key engineering disciplines and defining the required product data domains.
It is shown in chapter 2 of this paper, that special focus has to be set on an appropriate mechanical interface definition, on the application of tolerance management methods for the interfaces between cabin and fuselage as well as on product data management

Solution Step 2:
Selecting a suitable modeling philosophy.
Chapter 3 in this paper will account for the need of a more holistic, formal and performant model-driven data model following a modern, abstract and generic modeling approach instead of using several dispersed and unconnected domain-specific data models.

Solution Step 3:
Selecting a pragmatic way of implementation.
Chapter 4 will present a graphical rule-based cabin design language as an implementation approach. It will show the benefits of semi-automated trade-off studies instead of using dispersed data models with manual interaction.

The paper closes with a discussion of open research questions in chapter 5 and with a summary in chapter 6.

2. TOLERANCE MANAGEMENT AND PRODUCT DATA MODELING IN THE FOCUS

2.1. Key Integration Engineering Disciplines

The cabin architecture complexity – and therefore the product data complexity – lies in the various spatial, physical, functional and procedural interactions between the cabin modules and the surrounding aircraft components. Three 'integration engineering disciplines' are identified (see solution step 1), as Figure 2 shows.

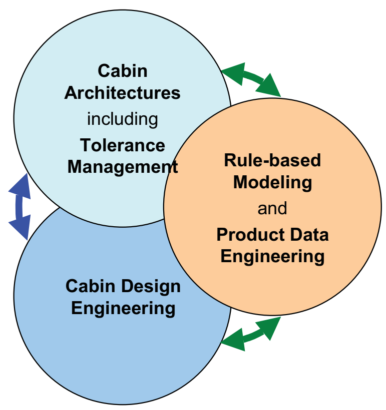


Figure 2 Interaction of three Cabin Integration Engineering Disciplines

Cabin architectures including *tolerance management* can thus be described with the following work focus:

- Definition of cabin module frame conditions as well as conceptual module geometry, gaps and split lines
- Definition of manufacturing and assembly philosophy for component suppliers and FAL
- Ensuring integration during development (top-down)
- Providing tolerance analysis (bottom-up).

Cabin design engineering primarily is responsible for:

- Conceptual design of cabin modules and systems
- Design of manufacturing and assembly processes
- Ensuring Integration during development (bottom-up).

To enable the modeling of mechanical cabin architecture and integration aspects, modeling by '*product data engineering*' is needed. In chapter 4 of this paper, an implementation approach using a graph-based design language with multi-domain model transformations will be described. In this context, the tasks for product data engineering are:

- Ensuring data consistency of multi-disciplinary product data
- Ensuring syntactic and semantic data checks
- Supporting cabin design engineers and cabin architects to identify and to formalize design rules
- Enabling fast trade-off studies with suitable methods and tools in order to support cabin architects and design engineers during decision finding
- Supporting data exchange with further data models, e.g. with cabin or aircraft systems.

2.2. Required Product Data Domains

ISO 10303 defines product data as '*a representation of information about a product in a formal manner suitable for communication, interpretation, or processing by human beings or by computers*' [5]. Scientifically, every model serves a purpose and this purpose determines the model content [6]. Looking at the goal to support cabin architecture decision finding, at least the following product data domains should form part of the targeted holistic product data model (solution step 2): *geometry data – physical data – mechanical interface data (tolerancing data) – manufacturing process data – cost data*.

2.2.1. Geometry Data Model

The geometry data model has to contain definitions of the ATA 25 cabin modules concerning size, shape as well as interface and attachment point location. Of course, parallel to the cabin module design data, it is mandatory to have an idea about the surrounding geometry as well. This at least asks for a conceptual representation of the supporting ATA 53 structure components and – if needed – for space allocation for systems and system routing. Figure 3 below gives an example for conceptual geometry data of **Cabin Modules** and surrounding **Structure Components**, which are the so-called **Physical Components**. If needed, **Sub-Components** can be added for CAD visualizations

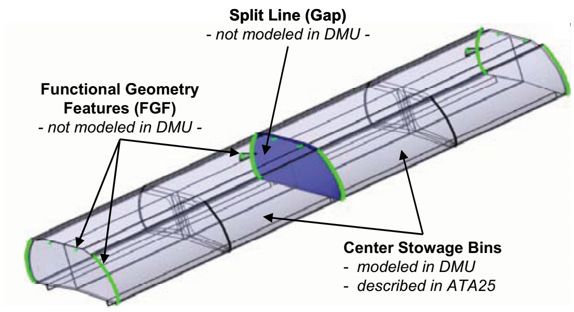


Figure 3: Conceptual Geometry Data of Stowage Bins including Split Line (Gap) and Functional Geometry Features (FGF)

However, although the module split lines or gaps play an important role in geometrical product specification and therefore in cabin architectural evaluation as well, they are not classified into ATA chapters. *Gaps* are special **Mechanical Interfaces** (see paragraph 2.2.3) between adjacent cabin modules and are therefore closely linked to tolerancing data models and installation process data models. The same applies to the so-called **Functional Geometry Features**, which constitute the mechanical interfaces to adjacent modules or to aircraft structure components. For a more holistic data model it is mandatory that both 'visible' geometrical product data (which is CAD data described in ATA 25) and 'non-visible' Gaps and Functional Geometry Features are modeled. It becomes clear, that already for modeling the geometry domain data, pure CAD modeling is not sufficient, but a more abstract modeling is necessary in order to represent Functional Geometry Features and Mechanical Interfaces in an appropriate manner.

2.2.2. Physical Data Model

Mass estimation plays an important role during decision finding as well. But due to outsourced production, a detailed modeling of the module materials is mostly not required. Parametric mass data with appropriate simplifications using key attributes and parameters e.g. from the geometry data model already provide the required detailing depth for mass estimations.
A load model e.g. by using finite element methods is not yet needed for conceptual cabin architecture decisions at this stage.

2.2.3. Mechanical Interface Data or Tolerance Management Data

SCARR talks of '*functional surfaces*' fulfilling at least one of five possible '*assembly functions*' [7]. MANTRIPRAGADA and WHITNEY propose the terminology '*joint*' between '*Key Characteristics*' of parts and distinguishes between joints transferring locational and dimensional constraints ('*mates*') and joints, which only support or fasten the parts [8]. MARGUET and MATHIEU use this concept as well [9]. These behavioral descriptions aim at modeling assembly processes.

The modeling purpose discussed in this paper aims at evaluating architectures of which assembly processes are a major contributor, but not the only one. So subsequently a more domain-independent and therefore more abstracted descriptive classification is proposed:

Mechanical Interfaces in a cabin architecture model have a spatial location and a functional purpose. Three functions are distinguished:

A mechanical Interface has a *locating function*, if it positions two components relative to each other geometrically. These Mechanical Interfaces are so-called **Kinematic Linkages** [10], which establish a static kinematic linkage system by blocking the kinematic degrees of freedom between two physical parts. Figure 4 shows examples for kinematic linkages as used in the computer-aided tolerancing (CAT) software MECAmaster.

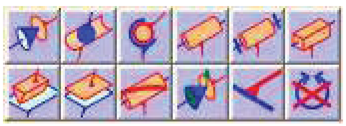


Figure 4: Kinematic Linkage types used in the 3D Tolerancing Software MECAmaster[1]

A mechanical Interface has a *fixing function*, if it links two components together in order to transmit a force or a moment. Such Mechanical Interfaces are named **Load Interfaces** and are needed for load models and therefore are of less importance at conceptual level. Often, they are directly coupled to the locating functions anyway, which would be a '*contact joint*' according to [8].

[1] MECAmaster by MECAmaster SARL, Ecully, France, see http://www.mecamaster.com.

A mechanical Interface serves a *tolerance function*, if it is a constraint for the static positioning of two components relative to each other e.g. at a gap or a split line. These Mechanical Interfaces are called **Tolerance Interface** because a tolerance as a geometric restriction applies.

Especially the locating functions and the tolerance functions play a key role for cabin architecture modeling and evaluation. In order to link Physical Components via any of these Mechanical Interfaces, **Functional Geometry Features** are introduced as geometry elements of a Physical Component. One or more attributes of the Functional Geometry Features can be Key Characteristics (KCs) according to EN 9100 [11], where a KC is defined as '*an attribute or feature whose variation has a significant effect on product form, fit, function, performance, service life or producibility, that requires specific actions for the purpose of controlling variation*'. Now a closer look at tolerance management [11] working steps is helpful:

1) Definition of Performance Key Characteristics including specification of target values
'Performance Key Characteristic' is a more detailed classification of the term KC, as shown e.g. in [9] and [12]. For the cabin, PKCs are Tolerance Interfaces which apply mainly on gaps and split lines between the cabin modules and are needed to ensure installability, functionality, or quality.

2) Identification of derived Key Characteristics and definition of locating functions
In order to validate and control the PKCs, further KCs (MKCs and AKCs, see below) are identified and are set in correlation by building a system of Kinematic Linkages between the involved components [10]. To do this in an explicit and non-ambiguous way, modern geometrical product specification (GPS[2]) methods are used [13]. In order to achieve the PKCs, top-down tolerance requirements for the derived KCs should be specified.

3) Capturing bottom-up values for AKCs and MKCs
For a robust bottom-up calculation of the PKC target values it is required to know about the manufacturing capability of the cabin modules for Manufacturing Key Characteristic (MKC) specification and about the assembly capabilities of the interfacing structure components for Assembly Key Characteristics (AKC) specification. For first calculations 'best engineering guesses' often are sufficient. Later in the design process, these tolerances have to be verified by proper formal means [4].

4) Calculation of PKC and iteration loops
Now it is possible to perform bottom-up PKC calculations [9], [14] followed by iteration loops of steps 1-4 in order to achieve converging top-down and bottom-up results.

To start the tolerancing process, only the geometrical definition of the considered Physical Components (Cabin Modules including the location of the Mechanical Interfaces, and – if helpful – also some Structure Components) as well as gap or split line definitions are required.

All other model aspects, like interface functions or the installation sequence are defined and modeled using the previous tolerancing work steps 1)-4). Consequently, the tolerancing iteration loops as mentioned in working step 4 are *multi-domain iteration loops*: if linkages or tolerances are changed, all contributing domains have to be considered and repercussions on all other model domains need to be checked in order to achieve a consistent trade-off study. By now it becomes clear that tolerancing data is the junction point between all considered model domains – and therefore tolerance management can be a key integration engineering discipline for mechanical cabin architecture definitions and evaluations.

2.2.4. Manufacturing Process Model

Concerning *Major Component Assembly (MCA)* processes (which is basically the airframe section assembly including bracket and substructure installation as well as equipment and systems installation), it is only necessary to address all relevant functional requirements in order to describe the assembly process constraints for bracket and supporting substructure: positional (and orientation) tolerances of Functional Geometry Features including defined datums and assembly sequence constraints, as well as limitations for in-flight deflections.
The same applies for *cabin component manufacturing*, as the current airplane manufacturer strategies mostly go towards outsourced production of cabin modules.

The focus of the cabin manufacturing process model is a detailed definition of the needed working processes at the *Final Assembly Line* (FAL), which is the joining of the fuselage sections, wings, empennage and engines and secondly cabin installation. This has to contain the cabin module installation sequence, the number of the overall needed working steps, and – if possible – an estimated working time. It can be assumed, that every Mechanical Interface also represents an **Installation Step** with an associated working time and cost. So the FAL process model has obviously overlapping product data with the mechanical interface data model and with the cost model.

2.2.5. Cost Model

Basically two major cost contributors can be identified for this modeling purpose: cost estimation per cabin module ship set, and estimation of FAL installation cost per aircraft respectively of MCA manufacturing cost. Links to a model for airframe and bracket cost estimation have to be foreseen. As for mass estimations, it is more important to cover the bandwidth instead of modeling all costs in detail. So the cost model at least has overlapping product data with the geometrical and physical data models and with the manufacturing process models.

3. A MODEL-BASED SOLUTION APPROACH AS CONSEQUENT MODELING PHILOSOPHY

Chapter 2 shows that almost any mentioned domain interacts with almost any other domain. In the center of

[2] See e.g. ISO 1101, ISO 5458, and ISO 5459 among others.

both the engineering challenge and the modeling challenge, tolerance management and the tolerancing data model can be identified. This leads to solution step 3 asking for an appropriate multi-domain modeling philosophy.

At first, the most promising candidate for multi-domain product data representation seems to be a CAD model. Also for cabin product data, geometry data forms a major part of the overall needed product data. But even though modern CAD software is extended to many additional engineering domains including CAT, it still does not leave the ‘geometry or 3D paradigm’ [15]: only geometry-based data structures can be modeled easily in a CAD system. Overloading CAD data with further design data does not permit domain-independent data structures and does not permit topology trade-offs independently from geometry.
To cure the gap to conceptual tolerancing, a method has been developed, which has the goal to embed geometrical Key Characteristics (KC) in the conceptual design processes based on interface functions, tolerance specification, and production frame conditions [16, 17]. An associative-parametric method for preliminary aircraft structure design has been proposed [18], and has been extended for preliminary aircraft weight estimations [3]. Other work focuses on the implementation of parameterized geometry handling for the interaction between CAD and CAE [19]. In [20, 21], so-called ‘High Level Primitives’ are proposed to model functional objects using KBE methods.

But even though these approaches either widen the view towards functional tolerancing or show the feasibility of multi-domain modeling in special use-cases, the afore named geometry paradigm is not really given up by any of the shown methods. So it seems to be difficult to extend them to a holistic support method for architectural cabin integration, which needs a more generic modeling approach. Since there is no known single-domain engineering data or software model that can represent all needed product data domain-independently, the search for a model-driven solution approach now is a logical consequence. In this context the implementation of a model-based approach (see solution step 3) has to stick to the following requirements:

– Ability to define adaptable flexible model structures.
– Ability to extend the model structure to any required, formally describable engineering domain.
– Ability to model and to represent abstracted data independently from the original domain to enable multi-domain and multi-model use.
– Ability to capture executable design rules in the product data model.
– Ensuring data consistency during model transformations.

4. GRAPH-BASED DESIGN LANGUAGES IN UML AS IMPLEMEMENTATION APPROACH

The implemented rule- and graph-based approach for multi-domain modeling using design languages [22] in the Unified Modeling Language[3] (UML) shows applications experiences in the aerospace and the automotive sector [15], [23], [24] and fulfills the above named requirements.

[3] See http://www.omg.org or http://www.uml.org

4.1. Rule-based Input of Tolerancing Data

Subsequently the implementation of a cabin architecture data model with the focus on geometry and tolerancing data modeling with a graph-based design language in UML is presented. Figure 5 below shows a simplified generic UML class diagram for a Cabin Design Language – a graph-based design language to model and evaluate mechanical cabin architectures. The domains, parameters and correlations as described in chapter 2 are represented. For specific use-cases, the class diagram can be extended by more detailed data.

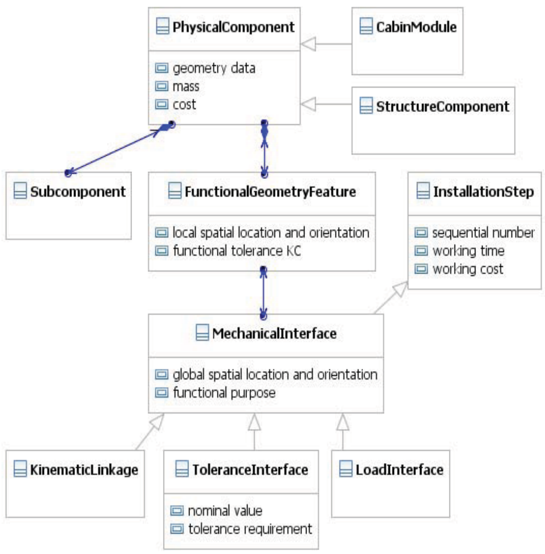


Figure 5: Class Diagram of a Cabin Design Language in UML

Figure 6 below shows a rule to insert a gap between two cabin modules (tolerancing step 1). The left hand side of the rule diagram shows the pre-modified status with the two unlinked cabin modules. The right-hand side shows these two modules now each with an associated Functional Geometry Feature and a gap in between as well as with two KCs describing the positional tolerance of the Functional Geometry Features. The rule is executed using parameters, so it can be used to model any gap or transition between any two parts.

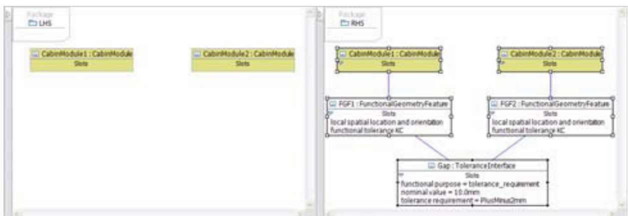


Figure 6: Design Rule to link two Cabin Modules with Key Characteristics and a Gap

The same principle can be used for rules to insert the kinematic linkages between cabin modules and the airframe substructure or the brackets (tolerancing step 2). In all cases, the datum of the tolerances has to be specified in subsequent design rules depending on the individual design.

Concerning tolerancing step 3, the tolerance value of a Functional Geometry Feature attribute (Key Characteristic) can be modeled using empirical formulas. In the example of Figure 5, the tolerances are defined as constant values.

4.2. Multi-Domain Output

In order to make evaluations of the modeled design studies, the modeling level has to be left and the domain-depending data models as shown in chapter 2 have to be extracted from the UML. Using the model transformation abilities of Design Compiler 43v2 within the current use-case it is possible to create various domain-specific output files, as shown in Figure 7. The model transformations can be made bidirectional, which enables to read back computed data from the linked models and methods (so-called 'Round Trip Engineering').

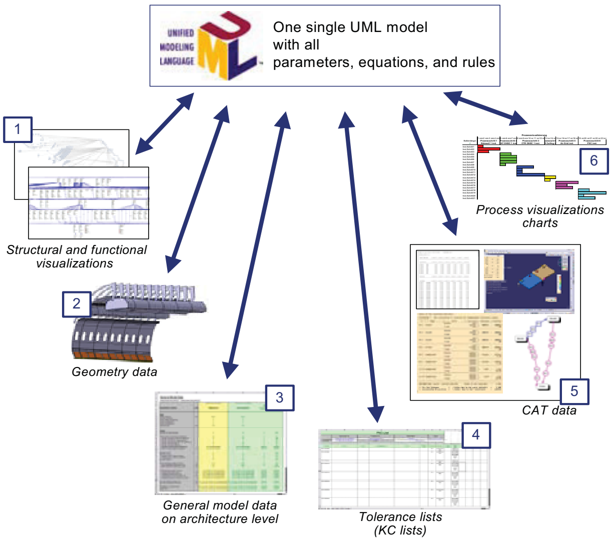


Figure 7: Multi-Domain Model Output from UML-based Cabin Design Languages using DC 43v2

Visualization of the CAD-based geometry data (see 2. in Figure 7), spreadsheet-based summary overviews about architecture evaluation criteria like mass, cost, interface and process data (see 3.), spreadsheet-based tolerance lists (KC lists, see 4.), FAL installation process overview charts (6.), and – as needed for tolerancing step 4 – input sheets for CAT-based tolerance analysis in conjunction with CAD data (see 5.). For the last one, the given implementation approach makes model transformations from the UML data model of the cabin architecture into MECAmaster UML-vocabulary and then into text-based MECAmaster input files, as Figure 8 shows. Aside these domain-specific model transformations, structural UML representations on model-level can be created for higher-level technical model structure evaluations or even for model validation purposes (see 1. in Figure 7).

Figure 8: Model Transformations from Cabin Design Languages to CAT input files

The targeted time reduction for the execution of concept trade-off studies as requested at the beginning of this paper can be realized using this multi-domain modeling approach. Not only does the presented structural approach using graph-based design languages increase speed for domain-specific modeling and data harmonization, but also helps the engineers to focus on the technical content which is needed to analyze and evaluate a proposed architecture scenario, instead of being held back by the fact to make these analyses manually.

5. OPEN RESEARCH QUESTIONS

Further research work will continue with the rule-based modeling of the complex correlation network between Physical Components, Mechanical Interfaces, and Process Steps. Especially aircraft structure tolerances and the associated manufacturing frame conditions are difficult to anticipate for all-new architecture scenarios, so here questions arise about how to capture tolerance synthesis rule-based and how to link the rules perhaps to knowledge data bases. The same applies to a potential future modeling of in-flight deflections between Cabin Modules.

From a technical point of view it would be useful to have harmonized architecture philosophies including tolerancing approaches for aircraft cabin and structure, which should be investigated further in detail e.g. by applying the method as proposed with this paper. Such a future task is to embed this new method into the daily work of cabin architects – e.g. as proposed by involving an engineering discipline for product data management.

Questioning the applicability of the current geometry paradigm (with its early assigning of functions to solutions) for the architecture phase is another possibility, as multi-ATA integration into cabin modules is complex and the required mechanical interface management is manifold.

Another research focus will be on the transformation of the cabin design languages in UML into the domain-specific CAT data models and into other models with relevancy for architecture evaluation. The door to a successful linking of the presented mechanical integration data model to further data models – especially for cabin systems architecture analysis and evaluation – is now wide open and can be subject of further investigations.

Due to the given design complexity, these transformations have huge topological effects on the models and therefore need smart topology search algorithms. It is easy to create generic topologies rule-based, but it is still difficult to enrich these structures with specific and very individual design parameters using automated rules.

6. SUMMARY AND OUTLOOK

With this paper it has been shown, that there is a need for a method to support mechanical cabin architecture evaluation beyond the classic geometry paradigm and its function allocation. This geometry-focused product data is not meant to handle mechanical interface data, which is key product data for architecture evaluation.

Tolerance management is a key to a more holistic integration method and to a multi-domain data model by providing functional links between all data domains of interest. It has been shown, that a model-driven approach is needed, as no single-domain data model can represent all required domains appropriately.

Finally, a graph-based cabin design language in UML as solution approach along with a multi-domain use-case are presented using rule-based model transformations of the product data and providing multi-domain data output to support cabin design analysis and evaluation.

REFERENCES

[1] ATA Specification 100 - Specification for Manufacturers' Technical Data, Revision No. 37 (1999). Air Transport Association of America.

[2] T.K. Cummings: Lessons Learned from 737 & 787 Jet Liner Programs, The Boeing Company, Space Exploration, Arlington, Virginia, AIAA SPACE 2007 Conference & Exposition, 18 - 20 September 2007, Long Beach, California (AIAA 2007-6125).

[3] K.W. Oltmann: Virtual engineering models for aircraft structure weight estimation. SAWE 66th Annual International Conference on Mass Properties Engineering, 2007.

[4] S. Marwedel, N. Fischer, H. Salzwedel: Improving the design quality of complex networked systems using a model-based approach. 3rd International Conference on Model–based Systems Engineering (IC–MBSE2010), Fairfax, VA, 2010.

[5] ISO 10303-1 (1994): Industrial automation systems and integration – Product data representation and exchange – Part 1: Overview and fundamental principles.

[6] H. Stachowiak: Allgemeine Modelltheorie. Springer-Verlag, 1973.

[7] A. Scarr, D. Jackson, R. McMaster: Product design for robotic and automated assembly. IEEE International Conference on Robotics and Automation, 1986.

[8] R. Mantripragada, D.E. Whitney: The Datum Flow Chain: A systematic approach to assembly design and modeling. Proceedings of the 1998 ASME Design Engineering Technical Conferences and Computers in Engineering Conference, ASME, New York, 1998.

[9] B. Marguet, L. Mathieu: Method for Geometric Variation Management from Key Characteristics to Specification. Geometric Product Specification and Verification: Integration of Functionality, 217-226. Kluwer Academic Publishers, 2003.

[10] P. Clozel: 3D Tolerances analysis, from preliminary study. Geometric Product Specification and Verification: Integration of Functionality, P. Bourdet and L. Matthieu eds., Kluwer Academic publishers, Dordrecht, p. 93-104, 2003.

[11] EN 9100 (2009): Quality Management Systems – Requirements for Aviation, Space and Defense Organizations.

[12] D.J. Lee, A.C. Thornton: The identification and use of key characteristics in the product development process. ASME Design Theory and Methodology Conference, 1996.

[13] B. Klein: Toleranzmanagement im Maschinen-und Fahrzeugbau. Oldenbourg Wissenschaftsverlag, 2005.

[14] F. Scholz: Tolerance Stack Analysis Methods, Boeing Information & Support Services, December 1995.

[15] D. Böhnke, A. Reichwein, S. Rudolph: Design Language for airplane geometries using the unified modeling language. ASME IDETC/CIE, San Diego, DETC2009-87368, 2009.

[16] B. Marguet, L. Mathieu: Tolerancing problems for aircraft industries. Proceedings of the 5th CIRP Seminar on Computer-Aided Tolerancing, Toronto, 1997.

[17] A. Ballu, H. Falgarone, N. Chevassus, L. Mathieu: A new Design Method based on Functions an Tolerance Specifications for Product Modeling. Annals of the CIRP, vol. 55/1, 2006.

[18] C. Ledermann, C. Hanske, J. Wenzel, P. Ermanni, R. Kelm: Associative parametric CAE methods in the aircraft pre-design. Aerospace Science and Technology, vol. 9, p. 641-651, 2005.

[19] P. Mawhinney, M. Price, R. Curran, E. Benard, A. Murphy, S. Raghunathan: Geometry-Based Approach to Analysis Integration for Aircraft Conceptual Design. 5th AIAA ATIO Forum, Arlington, Virginia, 2005.

[20] J. Schut: Conceptual Design Automation. Dissertation, Delft University of Technology, Faculty of Aerospace Engineering, 2010.

[21] G. La Rocca: Knowledge Based Engineering Techniques to Support Aircraft Design and Optimization. Dissertation, Delft University of Technology, Faculty of Aerospace Engineering, 2011

[22] S. Rudolph: Know-How Reuse in Conceptual Design Phase of Complex Engineering Products - Or: "Are you still constructing manually or do you already generate automatically?". Advances in Integrated Design and Manufacturing in Mechanical Engineering II. Berlin, Springer Science + Business Media Deutschland GmbH, 2007.

[23] J. Gross, A. Reichwein, S. Rudolph, D. Bock, R. Laufer: An Executable Unified Product Model Based on UML to Support Satellite Design. AIAA Space 2009 Conference & Exposition, Pasadena, California, USA, 2009.

[24] M. Haq, S. Rudolph: "EWS-CAR": Eine Entwurfssprache für den Fahrzeugentwurf. VDI-Berichte Nr. 1846, 2004.