FUTURE-PROOF INTERFACES: SYSTEMATIC IDENTIFICATION AND ANALYSIS

Wolfgang Bauer and Maik Maurer

Institute of Product Development, Technische Universität München, Germany

ABSTRACT

A system's functionality is often realized by interaction of many independent products. The interoperability can cause problems in management and design, because these products possess different life cycle durations. Improved interface definition can facilitate the interaction of products within the system even if products possess such differences in their life cycles. In this paper a matrix and graph based approach is presented, which enables developers to ensure robust interoperability of products during a system's life cycle. Relevant interfaces between interacting products are selected for more detailed consideration by systematically identifying critical and fast changing interfaces. This helps defining future-proof interfaces. The approach has been successfully applied in an industrial design project on safety relevant system development.

Keywords: Multiple-Domain Matrix, interface management, interface definition, life cycle, criticality

1   INTRODUCTION

Systems often consist of many single products or subsystems, which mutually interact for fulfilling the system's requirements (Jamshidi, 2009). This causes multiple complex relations and exchanges within such systems of systems (SoS) (Carlock and Fenton, 2001; Jamshidi, 2005). A lot of interfaces exist between the subsystems and ensure their interoperability. The interfaces possess the greatest leverage in systems architecting (Rechtin, 1991), but are not properly treated (Jackson, 1997).

Subsystems comprising the system often have different life cycles and technical innovations cycles. They also differ in the length of associated product phases (i.e. their development, production and utilization phases) (Hepperle et al., 2011). However, the interactions of these subsystems have to be ensured and managed regarding their unequal and fast changing life cycles; they still have to be able to interact with the SoS concerning their hardware, software characteristics, or nature (Jamshidi, 2009). These aspects are often not considered properly.

In this contribution an approach is introduced that supports designers to better handle the system of systems' interfaces by using a Multiple-Domain-Matrix (MDM) and force-directed graphs (Di Battista et al., 1999). Critical and fast changing interfaces are identified and assessed on the system level. Then relevant interfaces between selected subsystems are determined on a more detailed element level to define and design future-proof interfaces along the system's life cycle.

2   APPROACH ON FUTURE-PROOF INTERFACES

The proposed approach consists of four steps (see Figure 1), which are described in the following sections. The sequence of tasks is based on the general process of Structural Complexity Management (Lindemann et al., 2009).

Figure 1. Approach on future-proof interfaces

2.1 System definition and data acquisition

Step one of the approach comprises the system definition compliant with the first phase of the Structural Complexity Management (Lindemann et al., 2009). Relevant domains and implied elements at the right level of detail are collected in a MDM; as well, types of dependencies connecting domains (Domain Mapping Matrix, DMM) or within one domain (Design Structure Matrix, DSM) are defined. After the MDM is built the domains (e.g. different portfolio products) can be divided into system elements (e.g. physical components) thus the level of detail of the system model increases.

The dependencies within the MDM represent the type of interfaces between the domains and consequently the elements. The definition of these dependencies is very essential defining and analyzing the interfaces later on. Several types of relations between one or more domains can be outlined (Pimmler and Eppinger, 1994) addressing the modelling of different kinds of interfaces. The type and number of different relations depend on the project aims and the maintainable effort of the data acquisition. In this case, a "can-change" relation was defined between the component elements.

The procedure of data acquisition is performed as presented in the methodology of Structural Complexity Management (Lindemann et al., 2009).

2.2 Assessment of critical system elements

Step two of the approach is split into two parallel phases, which result in two system views for subsequent interface analyses. This step points out the relevant and fast-changing interfaces within the system. A time-consuming detailed analysis of all existing interfaces is avoided by rating system domains according to their criticality from a structural point of view and concerning their probability of change during the system life cycle.

2.2.1 Rating the probability of change

The subsystem's probability of change during the expected system life cycle is assessed in the following four categories. This allows identifying interfaces that have to be defined and designed future-proof:

> Core product (no changes expected nor allowed)
>
> Gentle product enhancements are possible
>
> Intended product exchange or enhancement planned
>
> Product innovations (huge or numerous changes)

The classification requires expert knowledge and experience for achieving high quality results. Subsystems declared as product innovations or products with intended exchanges or enhancements are extracted for further consideration within the interface analysis. In contrary, core products with no estimated modifications can be disregarded in subsequent analyses.

2.2.2 Criticality analysis

Based on the crosslinking of subsystems performed in the task of data acquisition, the resulting network is analysed regarding the criticality of subsystems. As stated by Daenzer and Huber (2002), the criticality is computed by multiplying the active sum (sum of outgoing relations) and the passive sum (sum of ingoing relations) of an element. The criticality is a characteristic for rating the degree of the integration of a subsystem or an implied element into the entire system.

Interfaces between critical elements can be extracted by colouring the subsystems according to their criticality and visualize them in a force-directed graph (see Figure 2a). From a structural point of view, these interfaces are to be defined as critical as they can possess huge impact to the system in case of adaptation.

2.3 Selection of relevant interfaces

Based on the ratings executed in criticality analysis, the most relevant interfaces for further analysis can be chosen. Transparency can be reached by visualizing all subsystems in the life cycle criticality portfolio according to their criticality (horizontal axis) and their probability of change (vertical axis), as shown in Figure 3. Thus, subsystems are classified concerning their importance in the prospect of modification and the structural criticality. The portfolio is divided into three sections (indicated by colourings in green, yellow, red; see Figure 3). Elements located in the green section are expected to only be objective to minor changes and their system interaction is of minor importance. Elements

located in the red area are expected to change often and fast during the life cycle and their interaction and consequently their change impact to the system is high. Interfaces of elements in this section have to be analyzed.

After identifying critical and fast changing subsystems (located in the red and yellow portfolio area), their related interfaces are considered using a force directed graph (introduced in section 2.2.2). Therefore, the locality (Lindemann et al., 2009) of the element in question is regarded. The linked subsystems are analyzed with respect to their location in the life cycle criticality portfolio. Other critical subsystems (yellow and red portfolio section) related to the initial one are identified and consequently their interfaces to be considered in a more detailed way, as presented in the following section.

2.4 Analysis of selected interfaces

Step 4 includes the analysis of relevant interfaces regarding their definition and design, because of their selection as being critical and fast-changing subsystems. This analysis is executed on the element level. For that purpose, elements of the selected subsystems and their cross-links are visualized by a graph indicating elements of subsystem 1 at one side and elements of subsystem 2 at the other side (see Figure 4a). The connections between elements of both subsystems represent the interface. Every interface has to be discussed for realizing a future-proof design. Experts have to be involved with the interfaces and the probable changes of the elements. Connections of already defined or designed interfaces can be faded out for executing this analysis. Consequently, a reduced network of elements and interfaces remains for future-proof interface design.

A classification is given depending on the assessments of the probability of change of the selected elements (compliant to the rating of the subsystems) to support the definition and design of the relevant interfaces:

> Core element ↔ core element: the interface is irrelevant for future-proof design. A definition of the interface design and responsibility has to be defined in an early development phase. A permanent communication of the interface definition among the involved engineers has to be ensured.
>
> Core element ↔ flexible element: the interface has to be designed future-proof and robust, mainly on the core-side element to minimize change impacts on the core product by changing the related flexible element.
>
> Flexible element ↔ flexible element: the interface has to be designed future-proof and sustainable. Both elements must be considered and flexible solutions for the interface must be developed (i.e. application of a consistent adapter to place a new element between the elements in question to provide a flexible and future-proof interface).

3   CASE STUDY

The case study covers an equipment system for fire-fighting and mining, consisting of 15 independent subsystems (SS). These are all offered as independent devices. The system contains one main device (SS5), which possesses an average usage cycle of 20 to 30 years. Associated subsystems only have life cycles of two to three years. During its life cycle time the whole system has to interact in a reliable way independent from technology changes of attached subsystems. For reasons of non-disclosure the data shown in the following are made anonymous.

The domain of physical systems was divided into 15 subsystems consisting of more than 180 physical component elements. The relations between components were defined as a possible change propagation. Next, the MDM was filled together with several system experts resulting in 1022 bidirectional relations.

In step 3 (Assessment of Critical System Elements) the network of subsystems resulting from the data acquisition was visualized with the elements coloured according to their criticality (see Figure 2a). Also the 15 subsystems were rated regarding their probability of change (see Figure 2b).
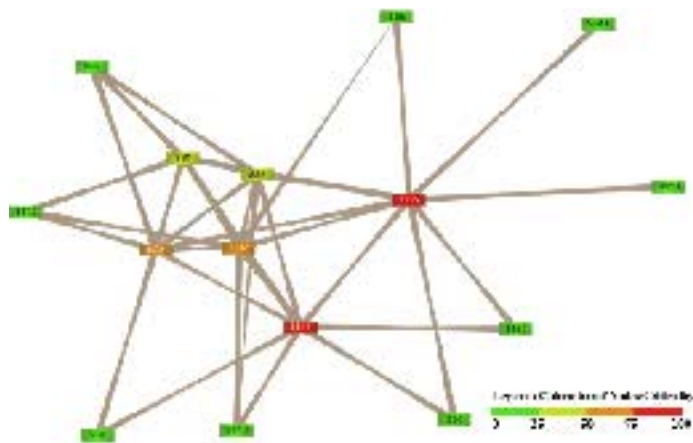
Figure 2a. Force-directed system graph

Figure 2b. Probability of change

The results from both assessments were transferred into the life cycle criticality portfolio to illustrate the rating information in a transparent way (see Figure 3). Based on this diagram, critical elements were picked out and their locality was analysed in the force-directed graph to identify interfaces to be designed future-proof.
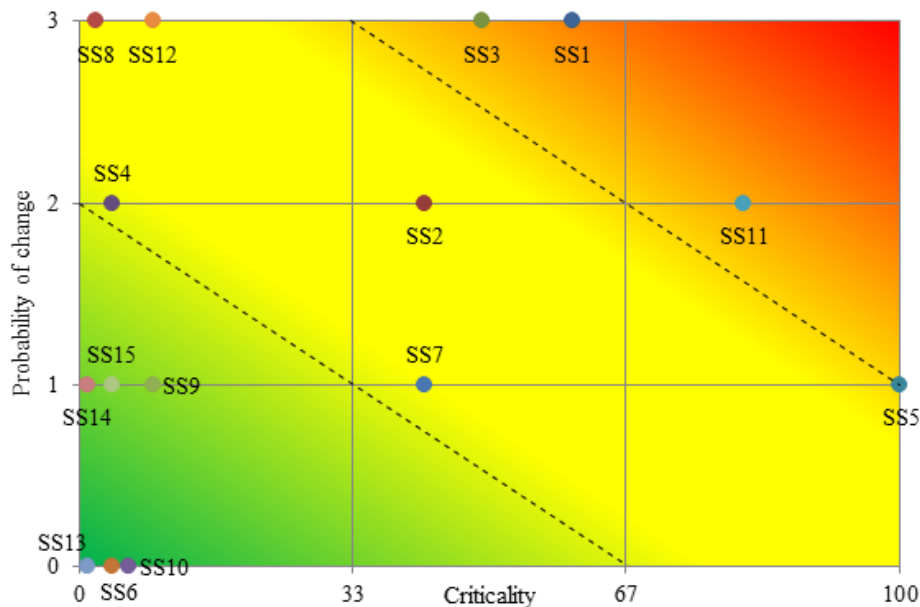

Figure 3. Life cycle criticality portfolio

The last step of the interface analysis is shown for the element level interfaces of SS1 and SS5. These are very important interfaces, not only because of the ranking in the portfolio but mainly because SS5 has the longest life cycle combined with possible gentle enhancements, and SS1has the shortest life cycle with a lot of expected technical innovations in the future. Moreover, the communication of these two systems is essential for the market success of the complete system.

Figure 4a shows the arrangement of the SS1 elements on the left (blue) and the SS5 elements on the right (red) in a graph. The green connections between the elements represent the interfaces to be discussed. Based on this visualization every interface was discussed by a team of eight experts. All elements and their interfaces were faded out, if they were already defined or not important for a future-proof design (because of not-expected element changes). The resulting graph is shown in Figure 4b. The remaining interfaces were discussed and future-proof solutions for the interface design were determined depending of the type of interaction.
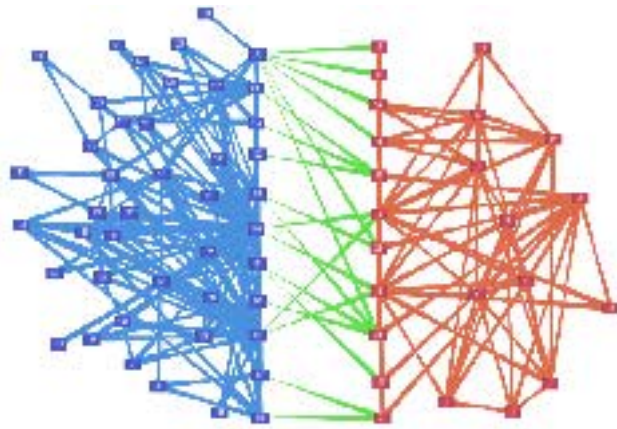
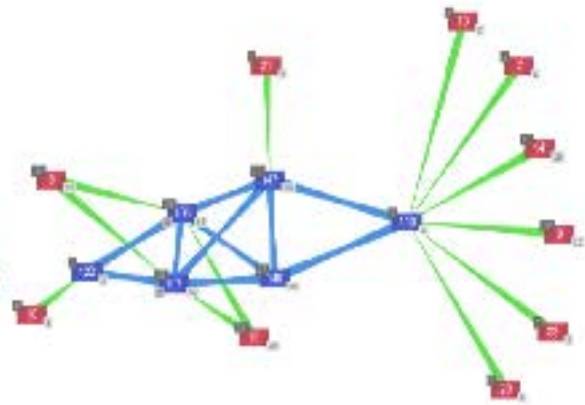*Figure 4a. Initial SS1 – SS5 graph*   *Figure 4b. Reduced SS1 – SS5 graph*

As shown in Figure 4b, the interfaces between these two systems could be reduced significantly to the relevant ones.

4   DISCUSSION

The presented approach can be applied for a systematical consideration of relevant interfaces with respect to their impact on the whole system as well as their possibility of change during the life cycle. The focus of an interface analysis can be channeled to critical interfaces in a system of systems from a structural point of view whereas the unnecessary effort on considering all interfaces is decreased. In the example shown above the relevant interfaces could be reduced significantly starting on a high system level and consequently breaking down the level of detail.

A long lasting interaction of the different related products along the systems' life cycle is ensured as aspects a different life cycles is integrated in the analysis. Therewith, the interoperability and interchangeablility of all subsystems building the SoS can be ensured with less change effort in product design as finally the systems' interfaces are defined in a future-proof and robust way.

A further benefit is brought by the applied approach by creating transparency in a SoS at different levels of detail regarding the existing interfaces. Adaptions raised from element change can be considered preventively by interface definition and design as the changes can be traced through the whole system, and the impacts of changes can be supervised and controlled. Due to the interface definition and design such change impacts can be better managed.

The presented approach can be applied to gain a more detailed representation of the system's interfaces by integrating more types of relations. In the presented case study, only one dependency type was applied. In case of considering different dependency types interfaces have to be optimized concerning different constraints.

Further views on the changeability and therefore for a comprehensive interface design can be introduced by integrating further domains in the MDM. Further views can be integrated in the system model by linking components to other domains. For example, elements can be tied to requirements, so the probability of requirement changes can be rated and this point of view can be integrated in a robust and future-proof interface design.

5   CONCLUSION

This paper presented an approach supporting the selection of relevant interfaces of a system of system for subsequent robust and future-proof definition. Characteristics such as the structural criticality of the elements as well as their possibility to change during the system's life cycle are applied. The interfaces can be reduced to those that require effort in interface definition and management.

REFERENCES

Carlock, P.G., Fenton, R.E. (2001). Systems of systems (SoS) enterprise systems for information-intensive environment. *Systems Engineering*, 4(4), 242-261.

Daenzer, W.F., Huber, F. (2002). *Systems Engineering – Methodik und Praxis*. Zurich: Industrielle Organisation.

Di Battista, G., Eades, P., Tamassia, R., Tollis, I.G. (1999). *Graph Drawing: Algorithms for the Visualization of Graphs*. Upper Saddle River, NJ: Prentice Hall.

Hepperle, C., Orawski, R., Langer, S., Mörtl, M., Lindemann, U. (2011). Temporal aspects in lifecycle-oriented planning of product-service systems. In A. Chakrabarti (Ed.), *Research into Design – Supporting Sustainable Product Development*. Research Publishing.

Jackson, S. (1997). *Systems Engineering for Commercial Aircraft.* Aldershot, UK: Ashgate Publishing.

Jamshidi, M. (2005). *System of Systems Engineering: Principles and Applications*. Boca Raton: Taylor & Francis Group.

Jamshidi, M. (2009). *System of Systems Engineering: Innovations for the 21st Century*. Hoboken: John Wiley & Sons.

Lindemann, U., Maurer, M; Braun, T. (2009). *Structural Complexity Management – An Approach for the Field of Product Design*. Berlin: Springer.

Maurer, M., Lindemann, U. (2007). Facing multi-domain complexity in product development. *Cidad Working Paper Series*, 3(1), 1-12.

Pimmler, T.U., Eppinger, S.D. (1994). Integration analysis of product decompositions. In *Proceedings of the 1994 ASME Design Theory and Methodology Conference*, Minneapolis, MN: ASME 1994.

Rechtin, E. (1991). *Systems Architecting: Creating and Building Complex Systems*. Englewood Cliffs, NJ: Prentice-Hall.

Contact: W. Bauer
Institute of Product Development
Technische Universität München
 Boltzmannstraße 15
85748 Garching
Germany
Phone: +49 (0)89 289.151.40
e-mail: bauer@pe.mw.tum.de
http://www.pe.mw.tum.de

95

Future-Proof Interface: Systematic Identification and Analysis

Wolfgang Bauer and Maik Maurer

Institute of Product Development, Technische Universität München, Germany

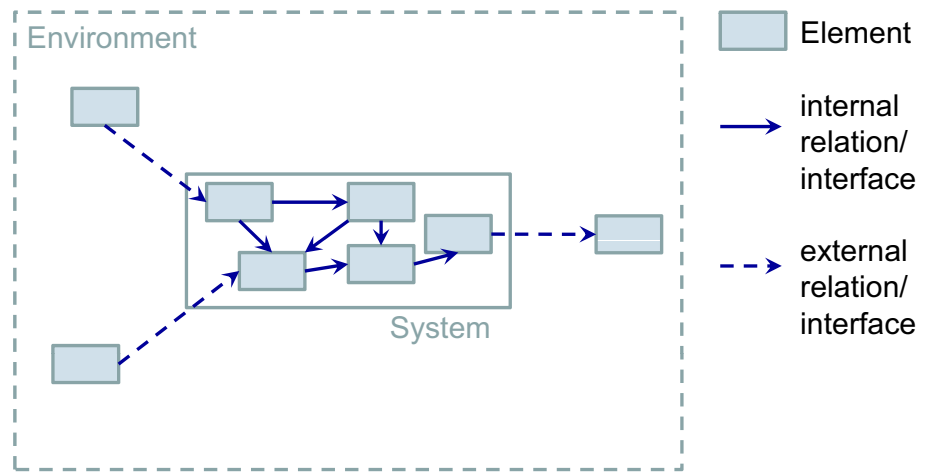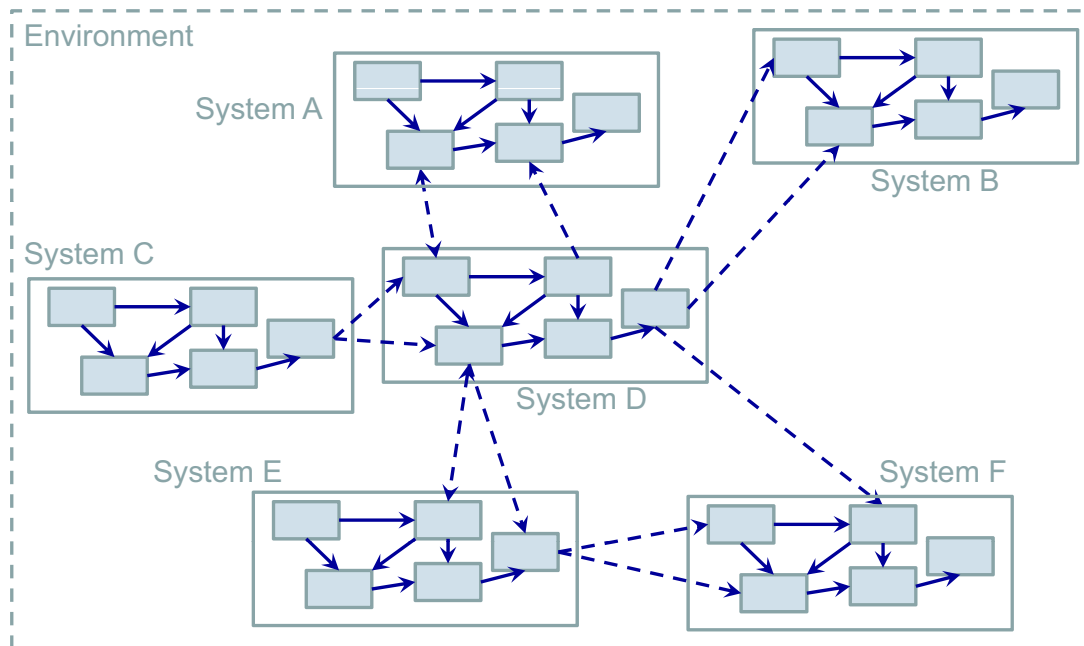Technische Universität München

---

Index

- Introduction of System of Systems' life cycle

- Overview of presented Approach
  - System Definition & Data Acquisition
  - Assessment of Critical System Elements
  - Selection of Relevant Interfaces
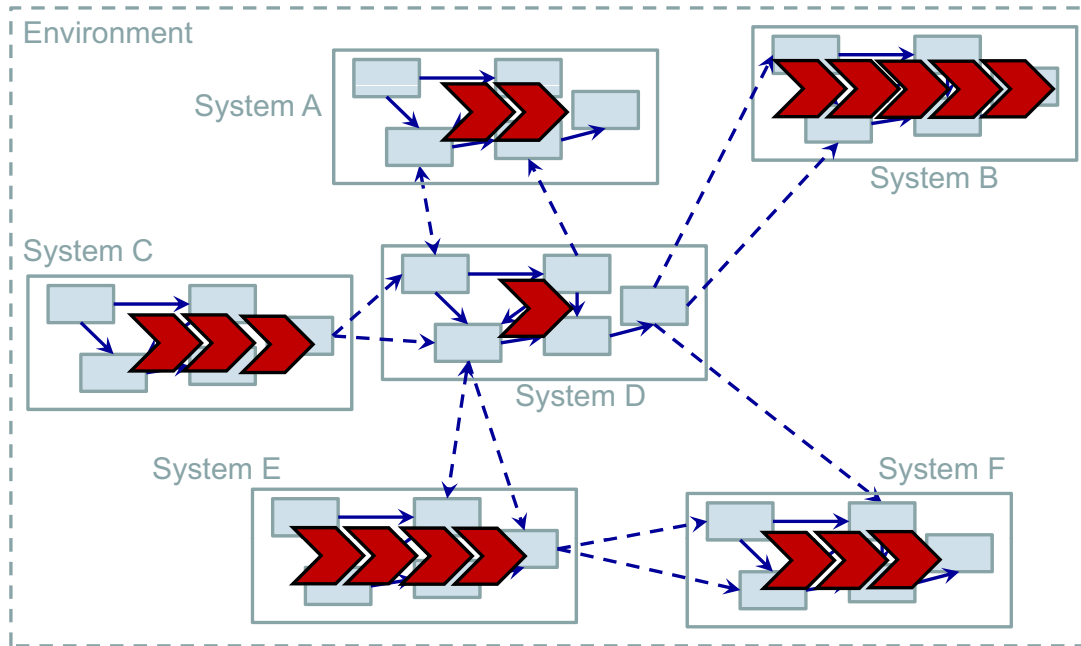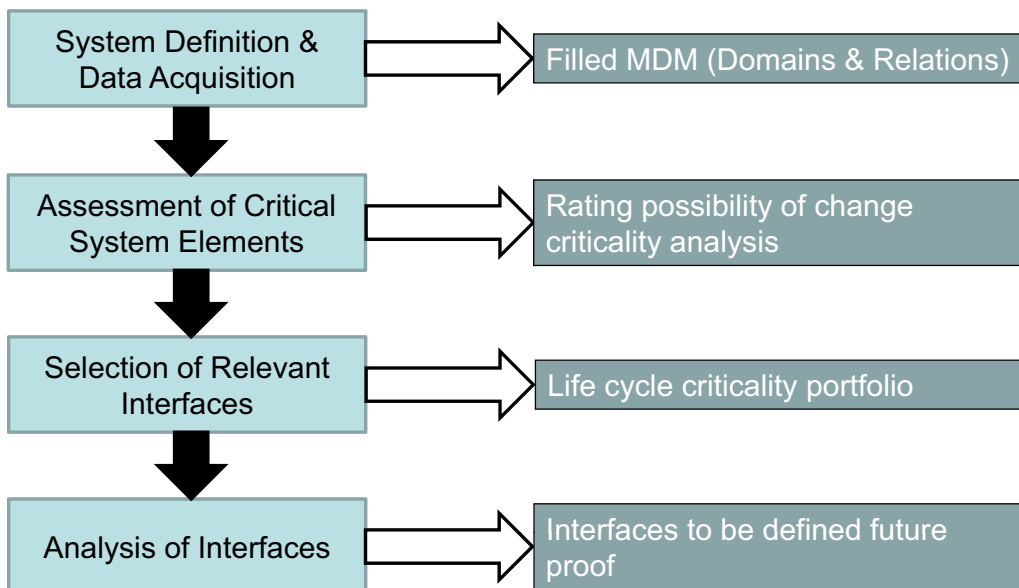 - Analysis of Selected Interfaces

- Benefits of the Approach

- Conclusion

Technische Universität München

System

System of Systems

Different Life Cycles in SoS

Approach on Future-Proof Interfaces

System Definition & Data Acquisition	Filled MDM (Domains & Relations)
Assessment of Critical System Elements	Rating possibility of change criticality analysis
Selection of Relevant Interfaces	Life cycle criticality portfolio
Analysis of Interfaces	Interfaces to be defined future proof

System Definition & Data Acquisition

- Build MDM
    - select domains
    - define relations
    - select elements

	Domain 1	Domain 2	Domain...	Domain n
Domain 1	Relation 1			
Domain 2		Relation 2	Relation 3	Relation 4
Domain...				
Domain n	Relation 5	Relation 6		Relation 7

- fill out matrices (e.g. DMM)

x: Relation 4

		Domain n						
Domain 2		1	2	3	4	5	6	7
	A		X		X		X	
	B	X		X		X		
	C		X		X		X	X

Assessment of Critical System Elements

- Two parallel steps

    - Rating the Probability of Change

    - Criticalitiy Analysis

Rating the Probability of Change

Rating the subsystems according their probability of change during the systems overall life cycle duration:

- Core product (no changes expected nor allowed)

- Gentle product enhancements are possible

- Intended product exchange or enhancement planned

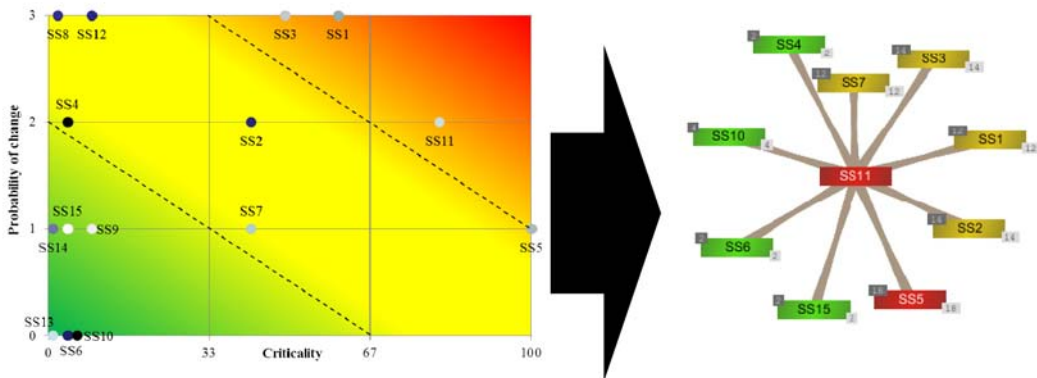- Product innovations (huge or numerous changes)

Criticality Analysis

Analysis of the structural criticality of a subsystem:

$$Criticality_n = Activ\ Sum_n * Passive\ Sum_n$$
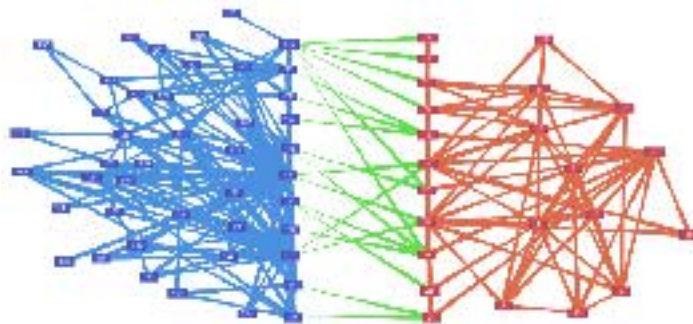
99

Selection of Relevant Interfaces

Identification of critical and fast changing subsystems in the life cycle criticality portfolio

Checking the locality of the subsystem in question and select critical interfaces as a result of the proximate element's criticality

Analysis of Selected Interfaces

Dicussion of interfaces to be designed future-proof (green) interfaces according to following rules:

- Core element ↔ core element: the interface is irrelevant for future-proof design → fade out
- Core element ↔ flexible element: the interface has to be designed future-proof and robust, mainly on the core-side element
- Flexible element ↔ flexible element: the interface has to be designed future-proof and sustainable.

Benefits of the Approach

- Focus of an interface analysis to critical interfaces in a system of systems from a structural point of view
  - → decrease unnecessary effort on considering all interfaces
- Ensure long lasting interaction of the different related products (SoS)
  - → reduce change effort because of new product releases or updates
- Create transparency in a SoS at different levels of detail regarding the existing interfaces
  - → preventively consider adaptions
  - → trace changes through the whole system
  - → supervise impacts of changes

Technische Universität München   TUM   MIT

Conclusion

- Presenting of an approach supporting the selection of relevant interfaces of a system of system for subsequent robust and future-proof definition

- Based on MDM-methodology

- Application of characteristics such as

  – structural criticality of the elements

  – possibility of change during the system's life cycle

- Selection and analysis of relevant interfaces regarding their future proof interaction

- Systematic reduction to interfaces that require effort in interface definition and management

Technische Universität München   TUM   MIT