

10 Use case: Automotive safety development

As depicted in Chapter 4, one can distinguish between handling problems and design problems, though these two types of problems can not always be clearly separated. In the use case presented here, the underlying problem can be described primarily as a handling problem resulting from the system's complexity.

The product – the safety system of an automobile – is already available. However, frequent requests for system adaptation (e.g., due to design changes executed to the car interior) generate problems of system controllability. Although a single adaptation request often does not appear to be too complex, system dependencies frequently result in the need for numerous subsequent adaptations. After- and side-effects are often unknown and cannot be anticipated; time and resource shortages are the consequences within the development process.

The objective aspired to in this use case was to provide guidance to product developers to minimize this handling problem of complexity. Product developers need a manual that provides relevant information in certain cases of system adaptation. They need, for example, knowledge about impact chains and the consequences of adaptations. If the handling of existing complexity is executed efficiently, the basic system design can be improved in a further step. That means that the system structure can be further developed in order to minimize the effort necessary for modifications in case of future system adaptations.

10.1 Problem Description

The fragmentation of the automobile market continues unabated, and automotive manufacturers continue to try to generate niches of demand (and occupy them with new products) between established automobile segments. This results in continually designing new carriage concepts. Today the challenge for all automotive OEMs consists of bringing more products of increasing quality to market within an increasingly shorter space of time, despite rising customer demands and continued pressure to keep costs low. Consequently, the complexity of development processes consistently increases. A successful strategy is to be able to control complexity rather than having to avoid the complexity that is essential to competition. This requires new solutions.

Complexity is continuously increasing, especially in the field of vehicle safety. The reasons for that include the fragmentation of vehicle variants offered and the

associated necessity for numerous adaptations of the safety systems. Most notably, however, the increasing number of influencing variables – such as crash cases to be met and dummy-types to be considered – and their interconnection represent a significant complexity. Figure 10-1 provides an illustration of the increasing set of safety requirements and functions.

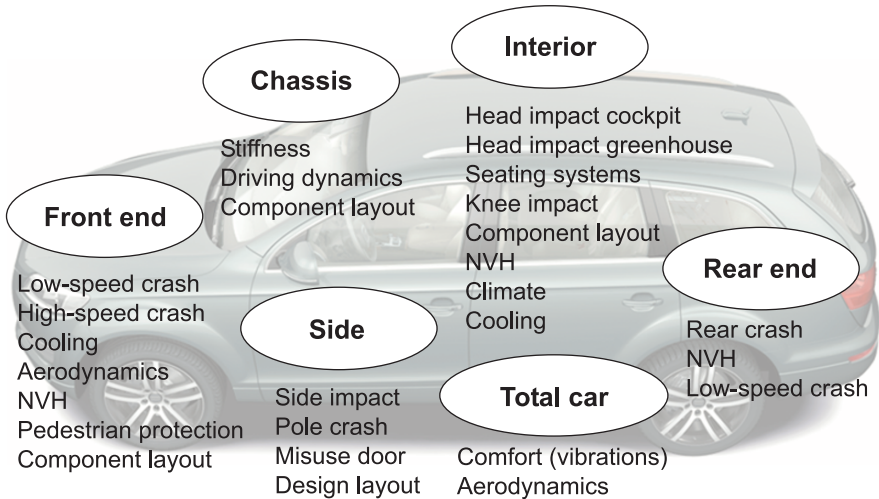


Fig. 10-1. Increasing set of safety requirements and functions

Influencing variables originate from several different domains. And linkages between parameters from different domains, e.g., physical components and functions, represent a further important source of complexity (see also Chapter 5).

In the present use case, the vehicle safety department of the automotive OEM faced the challenge of carrying out a structural analysis of different sub-systems of frontal and lateral safety to acquire the basis for future efficient complexity management.

One of the principal project goals was to establish transparency within the system comprising multiple domains. This is useful for reaching short response rates to emerging adaptation requirements when development times are tight. The following list contains the main objectives of the use case described derived from the problem description.

- Creation of system transparency and understanding
- Awareness of typical change impacts
- Comprehension of domain-spanning dependencies (components, features, test parameters, ...)
- Design of robust product structures concerning product adaptations
- Identification of opportunities and restrictions for product adaptation

10.2 System definition

The procedure of structural complexity management (see Chapter 4) was implemented for two reasons: to analyze the frontal protective system which focused on OOP (“out of position”: this means that the front-seat passenger is not in the optimal position during a collision); and to analyze a lateral protective system, which focused on understanding the overall system.

	C	CFS	CFD	TP	SPP	TC	DP	TCP	P
Components C	C has geometric dependency to C	Change in C can have influence on CFS	Change in C can have influence on CFD	--	--	--	--	--	--
Component features (static) CFS	Change in CFS can have influence on C	Change in CFS can have influence on CFS	Change in CFS can have influence on CFD	--	Change in CFS can have influence on SPP	--	--	Change in CFS can have influence on TCP	--
Component features (dynamic) CFD	Change in CFD can have influence on C	Change in CFD can have influence on CFS	Change in CFD can have influence on CFD	--	--	--	--	Change in CFD can have influence on TCP	--
Test parameters TP	--	--	Change in TP can have influence on CFD	Change in TP can have influence on TP	--	--	--	Change in TP can have influence on TCP	--
Seating procedure parameters SPP	--	--	Change in SPP can have influence on CFD	Change in SPP can have influence on TP	Change in SPP can have influence on SPP	--	--	Change in SPP can have influence on TCP	--
Test cases TC	--	--	Change in TC can have influence on CFD	--	--	--	--	TC determines TCP	--
Dummy parameters DP	--	--	Change in DP can have influence on CFD	Change in DP can have influence on TP	--	--	Change in DP can have influence on DP	Change in DP can have influence on TCP	--
Test case parameters TCP	--	--	--	--	--	--	--	Change in TCP can have influence on TCP	--
Persons P	P is responsible for C	--	--	--	--	--	--	--	--




	Acquired data (DSM)
	Acquired data (DMM)
	No data available

Fig. 10-2. MDM of the lateral protective system

In the system definition phase, the domains involved, the elements of the domains, and the types of dependencies within and between the domains, were defined. For this task the Multiple-Domain Matrix (MDM) was applied as storage for information about domains and dependency meanings. Figure 10-2 shows the matrix that resulted from the system definition for the case of the lateral protective system. For the compilation of the matrix system, experts were required, which

were interviewed in workshops. At first, domains and elements were gathered whose parameters exert an influence on the system. The collection was visually supported by creating a mind map. This enabled concepts and terms to be gathered within a short space of time and brought into a hierarchically-ordered structure. For example, in the case of the lateral protection, a total of eight domains could be identified with over a hundred elements in total. The domains were then carried over into the MDM. The dependency meanings connecting elements within and between these domains were then systematically defined for each subset of the MDM. Figure 10-2 shows the determined dependency meanings as entries in the subsets. It can clearly be seen that not all subsets of the MDM are occupied. Many dependency meanings between domains do not exist or were not relevant for this examination.

The methodical application of the MDM allowed a significant contribution to be made in order to create a system model as complete as possible. If all matrix subsets sequentially bring all domains in relation to each other systematically, no significant system connections are overlooked, which means that the MDM can serve as a check list for system definition.

10.3 Information acquisition

The system dependencies that are located in various DSM and DMM subsets in the MDM were collected in several workshops through interviews with the appropriate technical experts. Besides the experts for the entire safety system at the OEM, a number of additional experts from suppliers of the contained subsystems (e.g., an inflator) were called into the information acquisition workshops. The acquisition of the dependencies was first carried out for the DSMs along the diagonal of the MDM and subsequently for the DMMs.

The data collection and analysis processes were organized by complexity management experts and carried out using the complexity management software LOOME. Using special software that supports finding dependencies has numerous advantages: One is that the elements to be evaluated for probable dependencies can be displayed in a separate window with their attributes. This prevents the analyst from constantly twisting his head sideways to read vertical table entries, as is common in table worksheets. Also pictures can be inserted to illustrate the system elements actually discussed (see Figure 10-3).

With LOOME it is also possible to check for possible indirect dependencies during the acquisition phase. Plausibility analyses were carried out after each workshop. Discrepancies and conspicuous features in the structures were analyzed and brought forward at the beginning of the following acquisition workshop to be discussed in detail and incorporated into the system model.

Discussions often arise during the acquisition of dependencies, and the results must be documented as accurately as possible in the form of comments. It is by this means that an extremely valuable knowledge base is created.

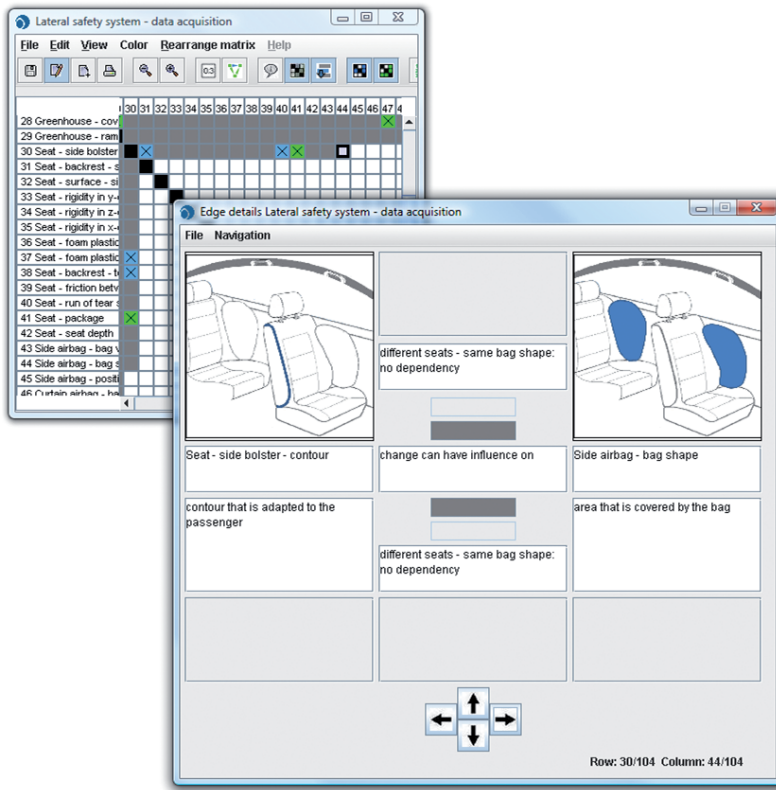


Fig. 10-3. Information acquisition with LOOME Software

10.4 Deduction of indirect dependencies

The most promising method of processing the handling problem of complex systems is in most cases to represent the acquired direct (native) dependencies in a suitable manner and to carry out the relevant evaluation for acute issues with appropriate methods. Here, the main focus was to establish transparency concerning the existing direct dependencies. But for certain issues it was necessary to deduce indirect dependencies as well. The network of indirectly linked people was deduced based on the dependencies within components and between components and people. This means that two people are linked, because both address (different) physical components, which depend on each other.

The computation of the indirect dependencies was carried out according to Case 4 described in Chapter 7 (see Figure 7-5 and Figure 7-9). For this computation, the DSM was derived from information stored in one DMM (connecting

elements from the domains of components and people) and one DSM (connecting elements within the domain of components). Resulting dependencies in the derived DSM (connecting elements within the domain of people) were condensed from relations between both domains as well as within the second domain.

In the example shown in Figure 10-4, both person A and person B are responsible for separate components, as can be understood from the information of the DMM; person A is responsible for component 1, which has a change dependency on component 2 (depicted in the DSM); this component is in the realm of responsibility of person B. Consequently, a dependency between person A and person B can be derived, because adaptations executed by person A affect person B by the indirect linking of the components.

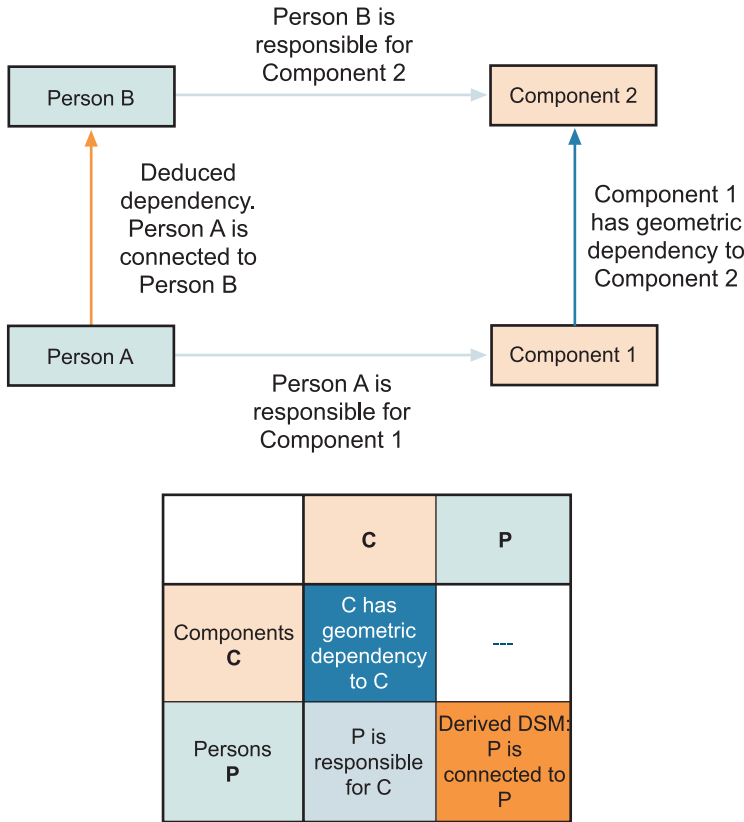


Fig. 10-4. Deduction of indirect dependencies between people

The DSM obtained of indirect dependencies between people was further investigated in the structural analysis and intensively drawn on in the phase of product design application in order to derive instructions to be acted on.

10.5 Structure analysis

In the present use case, methods for handling complexity, in particular, were applied. The structure analysis methods implemented were primarily those that produced the necessary transparency for carrying out interactions within the structure.

The initial point of every structure analysis is to obtain a comprehensive overview of the system. To this end various standard analyses are performed by applying methods described in Chapter 8 and in the Appendix. In this case, characteristics of the structure were identified, and the structure was checked for significant structural attributes in order to discover appropriate starting points for more detailed analyses. For example, an overall degree of cross-linking of 16% was identified for the frontal protection system and of 10% for the lateral protection system. The higher degree of cross-linking of the frontal protection system already indicates that in this system changes result in even more far-reaching consequences, and hence will be harder to control than in the lateral protection system. Furthermore, many hierarchies and closed circuits were identified within both safety systems. These structural attributes were identified as being responsible for the issues described at the beginning and are available for a follow-up evaluation in later product design application.

In the next step, the aim was to find those parts of the structure that had to be investigated in detailed analyses. If extensive structures are considered, it is often extremely difficult to detect the elements of particular significance in the system. Such a significance can mean that an element is responsible for a critical behavior of the entire system or that an element is extremely sensitive to system changes. The structure analysis offers various alternatives for detecting the relevant system elements: The representation of a network in force-directed graphs often provides initial transparency and enables the identification, for example, of elements that are centrally located. However, the larger a system becomes, the more difficult to identify system properties just by looking at force-directed graphs. Another possibility for obtaining key elements is to determine the active and passive sums of elements as well as the values for the elements' activity and criticality (see Chapter 8 and the Appendix) – as was executed in the present use case. The knowledge about such key parameters allowed better control of the vehicle safety system. These key parameters, together with parameters for which concrete problems had already been formulated, marked the base of operations for applying various advanced handling methods. An influence portfolio was created for the entire system as well as for the relevant subsystems (individual DSMs).

Influence Portfolio

According to the explications in Chapter 8 and in the Appendix, the active and passive sum of each element in the matrix has to be computed. The passive sum represents the amount of dependencies influencing the considered element (column total in a DSM). The active sum shows how much a considered element

affects the other system elements (row total in a DSM). This information forms the input for a two-dimensional diagram, the influence portfolio (Figure 10-5).

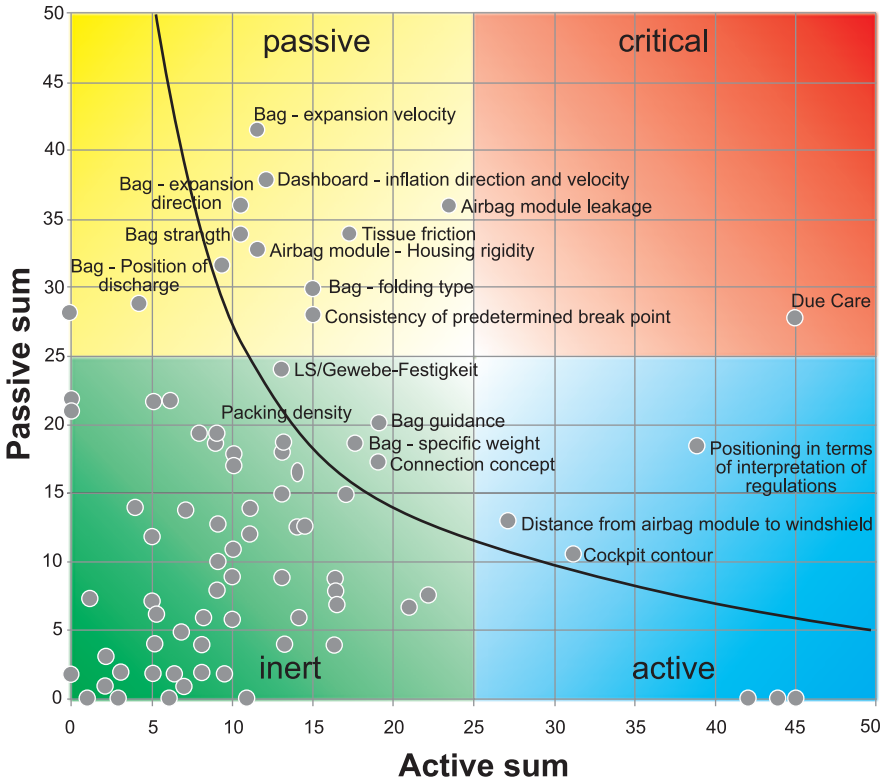


Fig. 10-5. Influence portfolio of elements of the safety system

For each element in the portfolio, the active sum (horizontal axis) is plotted against the passive sum (vertical axis). As shown in the diagram, it is helpful to divide the system into different sectors to get a classification of elements. First, an inert sector (green) is defined. Elements located in the inert sector are not of high significance regarding their impact on the network. That means that these elements do not largely affect other elements, and in addition they are not greatly affected by other elements. In contrast to inert elements, critical elements (red area) possess active links on many other elements, on the one hand, and are affected by many other elements, on the other hand. The blue sector contains active elements. The influence of these active elements on other elements in the network overshadows their passive dependencies. The yellow colored sector on the upper left side of the diagram shows passive elements. Passive elements notably are influenced to a greater degree by other elements than they influence system elements themselves.

To identify a set of critical parameters in the entire network, it is helpful to specify a line of equal criticality. Criticality is defined as the product of the active sum and the passive sum (see Chapter 8 and the Appendix). In Figure 10-5, a possible line of equal criticality is drawn. All elements above this line possess a higher product of their active and passive sum than indicated by the line of constant criticality. Here, it is not considered if, for example, the product value results from a high active sum multiplied with a low passive sum or vice versa. Of course, the criticality value for the line of constant criticality can be freely chosen and should result in the determination of a manageable quantity of system elements, which then depict prime candidates for further investigation.

10.6 Product design application

Answers to the initial problems were developed in the phase of product design application based on the findings from the structure analysis. The main focus was on problems dealing with the improvement of the handling of the existing complexity. That means that the objective was an improved system management. However, the results from applying the handling methods were also used in a subsequently improved system design.

10.6.1 Improved system management

In the product design application phase, different methods were applied to derive handling instructions for improved system management in product design.

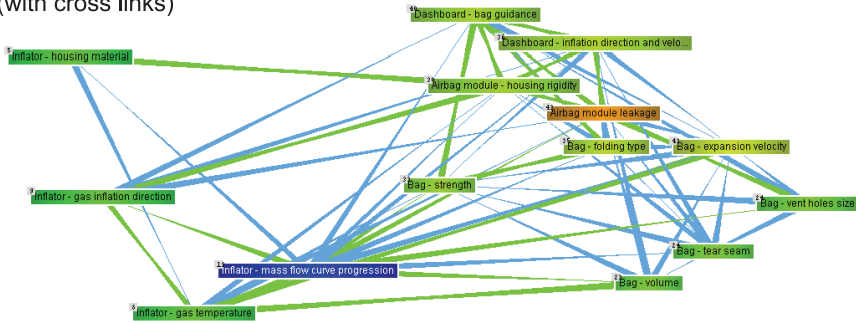
Impact check list

The transparency of consequences due to changes in the system was created for various elements by impact analyses. An impact analysis serves to represent the impact resulting from direct dependencies and dependency chains due to adaptations to the system. As already explained in Chapter 9, the impact check list represents a methodical analysis providing information about the nodes directly linked to one specific node in question. The provision of this information supports the systematic step-by-step evaluation of impact propagation resulting from the adaptation of a specific system element. If an adaptation has to be executed, product developers can sequentially evaluate the probability of change impacts to further elements. The systematic analysis of dependent elements guarantees that no possible impacts are neglected.

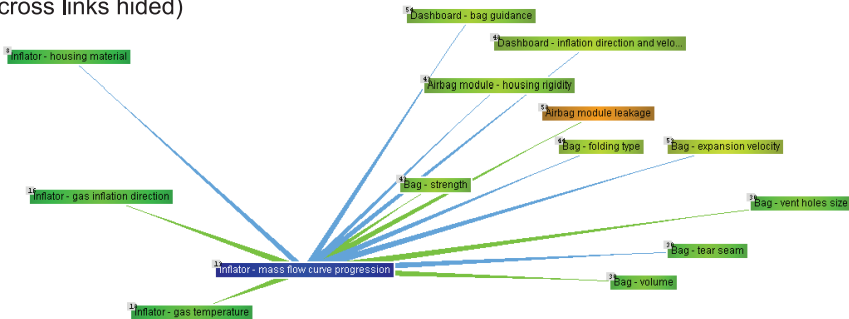
In the present use case, the impact check list comprises all nodes connected by outgoing dependencies only. In each case, the starting point is the element which has to be adapted. The structural environment is then modeled using this element

as a starting point. Outgoing linkages of the elements are tracked so that elements that are directly affected can be identified. After these affected elements have been evaluated, the next level of dependencies is focused on until the entire network is processed.

Active locality of “Mass flow curve progression” (with cross links)



Active locality of “Mass flow curve progression” (cross links hidden)



Edge coloring:

- Blue: dependency
- Green: reverse dependency exists

Node coloring:

- High (red) — low (green) criticality
(referring to the entire system)

Fig. 10-6. Impact check list for the system element “Mass flow curve progression”

One of several striking questions in the underlying use case of the front protective system was the question about the impact resulting to the system because of the weight reduction of the inflator. Such a weight reduction could be realized by a reduction of the wall thickness of the inflator – a bottle storing a mixture of

argon and helium gas. It was known that by keeping the inflation pressure unchanged and increasing the inner volume of the bottle, the reduced wall thickness would lead to a change of the gas mass flow curve progression of the inflator.

To analyze the impact of this change on the entire system, the active locality (see Chapter 8 and the Appendix) of the element mass flow curve progression was modeled in the network (shown in Figure 10-6).

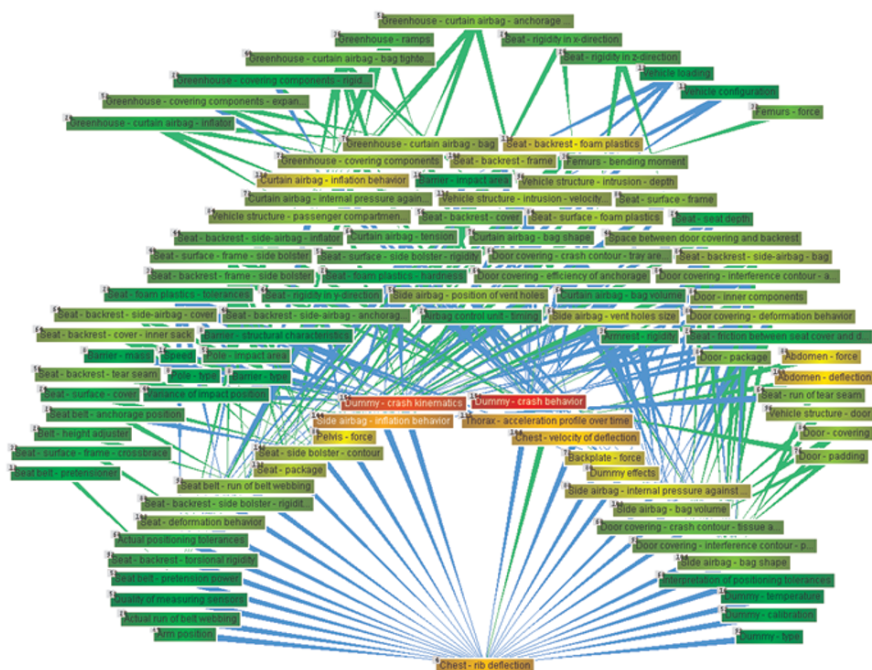
To evaluate the severity of impact, the elements influenced by the mass flow curve progression were colored according to their activity value. Figure 10-6 shows highly active elements (colored red) that could possibly be affected by the planned change. Due to their high activity values these elements can spread changes to a multitude of further elements in the considered system. Therefore, in the next step these elements were investigated in detail. The effects of such system changes were generally unknown before executing the analysis and measures of change management, which relied mainly on the knowledge and experience of several experts. With the impact analysis on hand, an overview of the effects could be conducted in a systematic and time-saving manner.

The checklist-like character of the impact analysis ensures that design decisions are performed and completed at an extremely high level of proficiency. This supports change management in a very efficient way. The developer obtains a deeper understanding of the possible consequences when implementing a change. In particular, it opens up the possibility of systematically identifying the impacts of changes of critical elements. A conclusion can be quickly reached as to whether a change is highly critical or not (and required resources, for example, can then be better planned).

Trace-back analysis

The trace-back analysis represents another highly important method to efficiently handle complexity. This method was initially adopted in the use case at hand to identify the causes of existing problems. Starting from the last element influenced in a dependency chain, a reverse analysis goes backwards step by step through the chain of impact. Elements that remain constant can be hidden from view. A summary of possible problem causes is finally obtained.

A decisive question in the use case of the lateral protection system concerned the causes of problems with the criterion rib deflection. As a result of the structure analysis, the parameter rib deflection was already identified as an extremely critical element in the entire network. All of the side-impact rating procedures showed a more or less intensive linkage to this parameter. Therefore, the trace-back analysis was carried out for the parameter rib deflection. As depicted in Figure 10-7, rib deflection is directly influenced by a variety of other elements. This becomes obvious by modeling the passive locality of the element. Furthermore, the influencing parameters themselves are influenced by other elements. The coloring of the elements indicates their passive sum. In this way highly influenced parameters become visible.



Edge coloring:

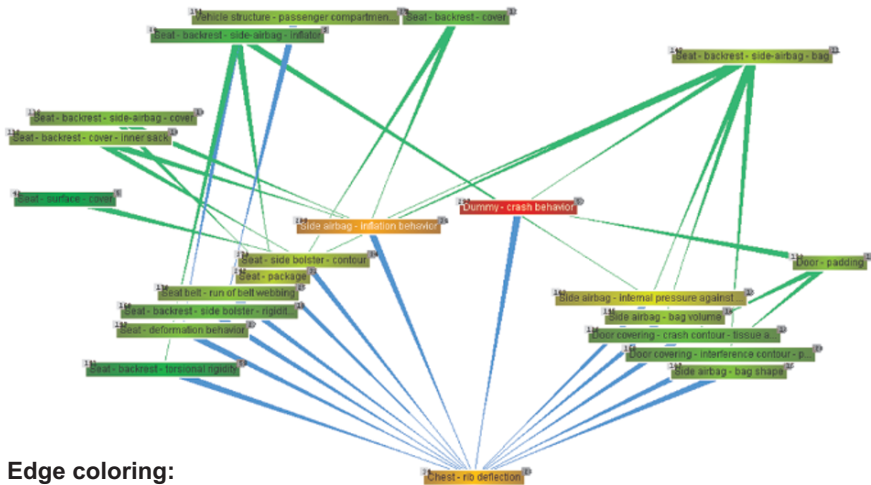
- Blue: dependency
- Green: reverse dependency exists

Node coloring:

- High (red)—low (green) passive sum (referring to the entire system)

Fig. 10-7. Trace-back analysis for the system element “Chest – rib deflection”

With this knowledge on hand it was possible to systematically process the indicated parameters influencing the rib deflection, and the cause of the problem could be rapidly identified in a practical application scenario.



Edge coloring:

- **Blue**: dependency
- **Green**: reverse dependency exists

Node coloring:

- High (red)—low (green) passive sum (referring to the entire system)

Fig. 10-8. Remaining set screws in trace-back hierarchy after hiding “untouchable” elements from view

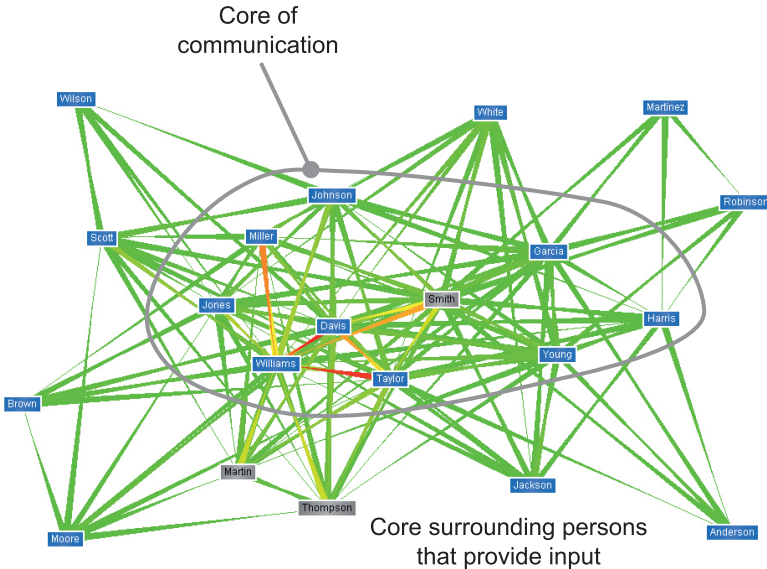
The trace-back analysis was also used as a kind of “set-screw analysis” in the application of safety systems. In the development process, the degrees of freedom for adapting system elements depend on the actual milestone or point in time. At a specific point in time, before the start of production, certain parameters can no longer be adapted, because production facilities are already fixed or physical subsystems are already produced. The system behavior under these boundary conditions can be represented with the help of the set-screw analysis. Elements that can no longer be modified at this time can be hidden from view in the identified trace-back hierarchy. Only a smaller set of parameters remains, which can serve as “set screws” in order to solve the actual problem. Figure 10-8 shows the remaining set screws in the problem case of rib deflection. The scope of elements is much smaller and therefore better manageable than all the probable influencing elements obtained by the preceding trace-back analysis.

Optimization of the communication network

Another problem represented the question of how dependencies within the component network affect the people network. In other words, how people were linked to each other through their interaction with physical system components was not

fully understood. For this reason, a matrix of indirect people interrelations was computed within the framework of the “deduction of indirect dependencies” (see Chapter 7 and Section 10.4). The resulting network provided information about the linkages between people due to their work on different but mutually linked components. It indicated that if one developer implements changes to his dedicated component, the second developer has to react, because his component probably needs to be adapted, too.

Figure 10-9 shows the computed network of developers as a force-directed graph. It can be seen from the depiction that an area of communication has formed in the center. This can be interpreted to mean that usually communication has to pass via the same subset of developers. This computed network allows one to see how the people structure overlaps the existing organizational structure due to component dependencies. The rearrangement of people in the existing organizational units would be the next issue to investigate. For example, the close and necessary integration of suppliers in the communication process is conspicuous in this case.



Edge coloring:

- High (red) — low (green) need for communication between persons

Node coloring:

- Blue: person from internal department of OEM
- Grey: person from supplier

Fig. 10-9. Network of people deduced from change dependencies of components

The work undertaken led to a considerably increased understanding of the safety system considered. Previously unknown dependencies and correlations became visible and controllable when depicted in this way, and measures for adaptation could be more efficiently managed. This led to an increased reliability in the development processes while simultaneously saving resources.

10.6.2 Improved system design

Besides the methods of controlling complexity, a further step consists of designing the product structures to become more robust in order to reduce the impact of required modifications in general. Several structure attributes (clusters, circular paths, hierarchies, etc.) can be considered in order to achieve this.

Various subsystems are involved in the lateral safety system. The fewer dependencies that exist between the individual subsystems, the fewer impact changes will spread through the whole system. The objective was to generate independent subsystems (to a degree that is possible) in order to design a system that would be more robust in general. Changes would then be allowed to have some impact on the delimited modules; if at all possible, however, the effects should not spread to the entire structure.

Figure 10-10 shows the change dependencies between components that play a role in the lateral safety system. Within the framework of the structure analysis, the matrix was reordered using cluster algorithms. Figure 10-10 shows that various clusters can be identified: seat surface, seat backrest, door and greenhouse, including belt. The clusters are to some degree in mutual dependency.

A significant dependency was identified between the clusters' seat surface and the greenhouse, including the belt. Here the potential was highlighted to separate the two clusters completely by eliminating only one bidirectional dependency between two elements. The technical solution to eliminate the dependency consisted of integrating the belt connection into the seat.

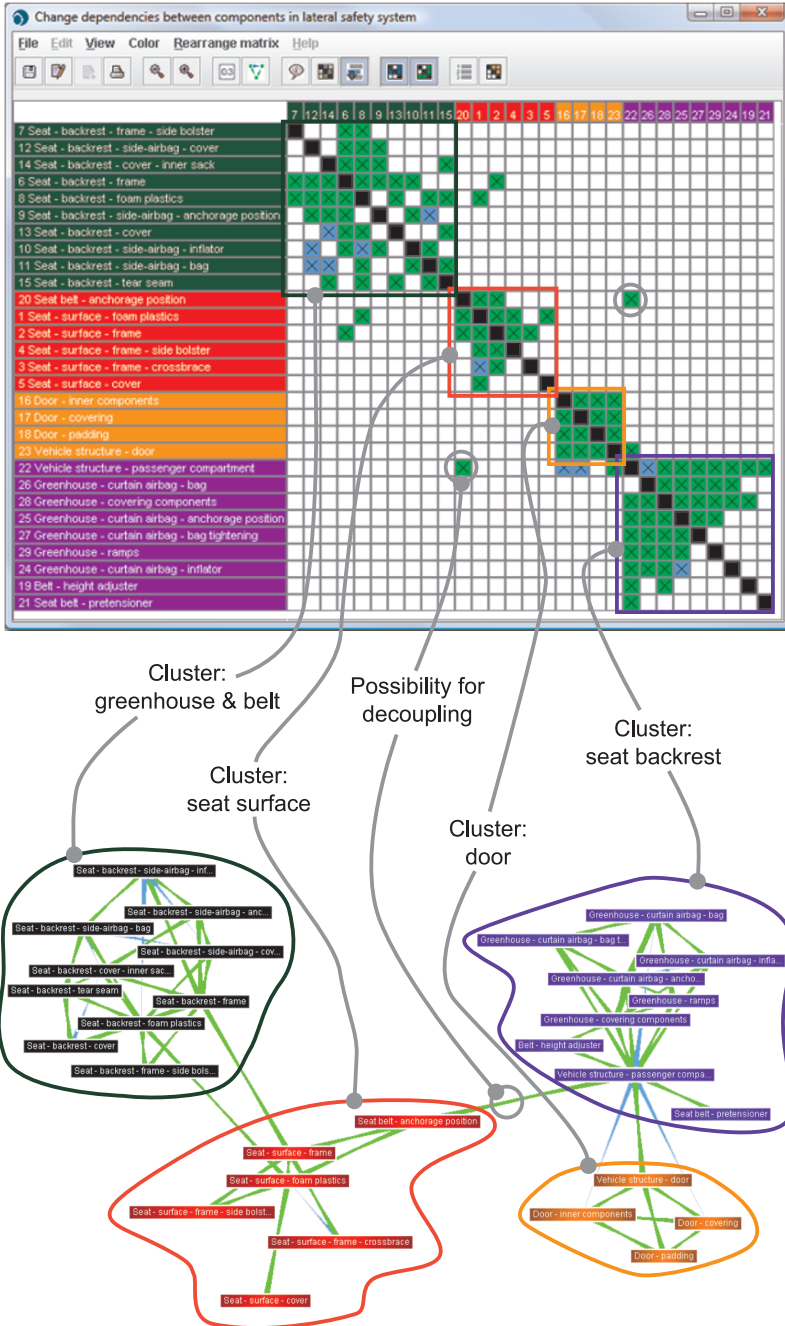


Fig. 10-10. Improving robustness of product structure

Structural Complexity Management

An Approach for the Field of Product Design

Lindemann, U.; Maurer, M.; Braun, Th.

2009, X, 240 p. 127 illus., 85 illus. in color., Hardcover

ISBN: 978-3-540-87888-9