# AN ITERATIVE QUESTION-ANSWER DRIVEN PROCESS TO SUPPORT PRODUCT DESIGN

Kjell Andersson

Machine Design, KTH
SE- 100 44 Stockholm
E-mail: kan@md.kth.se

*Keywords: Modeling, Simulation, Question-Answer, Design Process*

**Abstract**

Product development is often described as an iterative process of finding solutions that match specific requirements. The many dimensions of this process include time, organization, product-specific elements such as the level of abstraction and detail, and analysis to verify the product's properties.

Many types of software tools are used to generate and visualize the concept shape. These include CAD (computer-aided design) tools; tools to simulate and verify product properties, such as FE (finite element analysis) and MBS (multibody systems); and tools for handling product data such as PDM (product data management). This paper focus on the effective use of simulation software such as FE and MBS tools to support the process of verifying that a product meets the formulated requirements. The simulation software can be used for such things as selecting alternative solutions or as a final check or optimization of a solution concept. Its can be used even more effectively if it is supported by a framework for handling the information created during the verification process.

This paper presents a proposal to a question-answer driven process for verification of product requirements. This process is based on the framework presented by the author in [Andersson 2004a, 2004b]. The process is illustrated in a modeling and simulation scenario of a lifting unit in a wheel loader from Volvo CE. This scenario focus on modeling and simulation activities and how these can be supported in a question-answer driven process that investigates the behavior of the lifting unit.

## 1. Introduction

The potential of using modern commercial analysis software have not been yet fully accounted for in industry. It is still often used solely as a basis for verification of product properties late in the design process. One aspect that is often not considered is the large amount of knowledge that is documented together with the analysis results. If this knowledge can be structured in a way that allows for reuse, it can, e.g. be used in the case of redesign of an existing product for a new customer or during an evaluation situated when comparing different product concept, where some of the modelling objects can be valuable to reuse. It can also be used as a support or education tool for less experienced engineers.

Reuse of simulation models is also considered to be a crucial issue that can increase the use of simulation tools in industry and thus help to a better understanding of products behaviour, which also will contribute to design of better products. In order to be able to reuse simulation models and other models used during the verification of product requirements it must be easy to find these models when we need them. This implies that we must structure and store these models together with their metadata describing the modelling context and the intent of the model.

The focus of this paper is to discuss an environment where a question-answer driven process is proposed to enable a flexible and stimulating environment for testing ideas as well as verifying product properties. This also includes a description of the different objects that represent a verification loop from a stated requirements attribute in the requirements specification to a decision basis for the next gate in the development process. The objects presented here are all implemented as XML objects and links between objects are represented as hyperlinks.

An important property of a verification process is traceability, i.e., the possibility to trace what objects that have been created and used during a verification loop. Another important property is the ability to be searchable such that the designer or the team can search for earlier created objects. To make an efficient search we need to specify the context in which they have been formulated, to enable some level of reasoning in the search process.

## 2. Product Development environment

Engineering work is to find solutions or answers to problems during product development. These involves both synthesis and analysis activities and are most often supported by computer tools. These tools often work in different environments and it is difficult to share data between these tools. This complicates the possibilities to investigate and play around with different solutions. An integrated environment where metamodels are used to establish the integrating layer between different software tools can improve the situation and solve at least some of these problems.

The focus of this paper is to discuss an environment where a question-answer driven process is proposed to enable a flexible and stimulating environment for testing ideas as well as verifying product properties.

During the product development a number of different model structures are being developed. Working with verification of product properties mainly involves structures of requirements, parts and behavior models associated to parts. This process is best characterized as a question-answer driven process where the different activities provide connections and relate these structures to each other.

In figure 1 from [Andersson 2004a], the relation between a generic design process model and a stage-gate model is illustrated. In this figure the verification process is illustrated as an activity that originating from the requirement specification, a design concept and available data in databases perform a number of activities and produces a decision basis as a result.

The activity "Investigate problem" in figure 1 is the center of the question-answer driven process where the verification takes place and the relations between requirement specifications, design concept and behavior models are created. In this process we have also introduced some additional models, i.e. *problem, model specification, simulation model, answer* and *decision basis.*

The reason for this is to enable a more detailed description of the verification process and to make it possible to trace how a certain requirement have been verified, how is the problem question formulated, which subsystem models have been used to configure a behavior model etc.

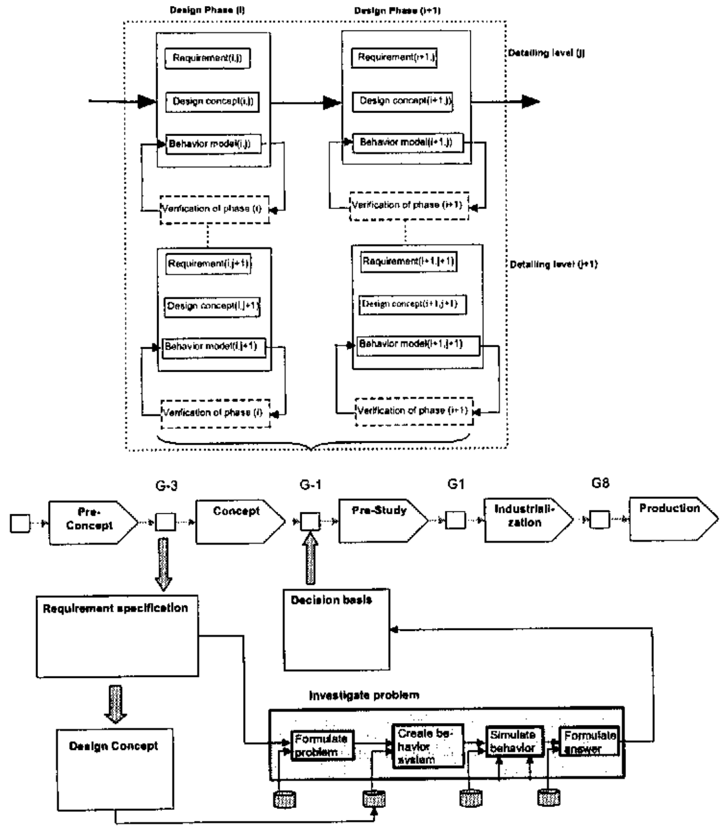


**Figure 1. Relations between a generic design process model and a stage-gate model [Andersson 2004a]**

## 3. Behavior modeling and simulation during product development

The need for estimating the behavior of a product concept exists during the whole product development process. As soon as you have a concept solution, there is a need to estimate the behavior [Lindemann, 1998]. In the early design phases, a rough estimate may be sufficient though in later phases more detailed and sophisticated models are needed and as a final verification a prototype testing may be needed. Figure 1 schematically illustrates how behavior modeling and simulation can be included as part of a general design process model and how this model can be related to a stage-gate model.

The approach to behavior modeling presented in this paper is the one that we use in the VISP project as well as in the earlier MOSAIC project [Andersson, 1998], [Sellgren, 1998]. This approach is based on treating the product as a technical system as suggested by Hubka [Hubka 1984], which means that we can divide it into a number of different subsystems In order to make any analysis at all, the main parts or characteristics must be known. As soon as we can identify some parts in the product, we can separate these and treat them as being modules or systems. A characterizing thing for systems is that they can be described by what is within the system boundary and how it is interacting with other systems via the interfaces. An interesting area to identify for each module is where it can interact e.g. is connected with other modules, i.e. the mating features for that module. The actual interface feature between two modules is treated as a relation that must consists of at least two mating features.

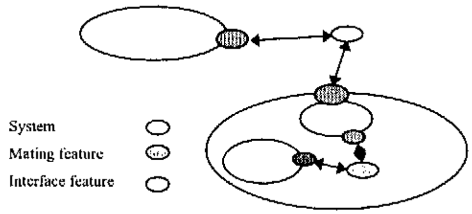


**Figure 2. The concepts of System, Mating Feature and Interface feature**

## 4. A question-answer driven process

A question-answer driven process is a good characterization of the verification activity where calculated product properties are compared to the required ones.

Of this activity we will concentrate on the parts of the verification activity that have a key impact on the decision made by the management at each gate. In figure 1, two main type of activities are schematically described, generation of design concept and analysis of this concept with respect to the formulated demands in the requirement specification. Of these two activities we assume that we have a well-defined concept and that we concentrate on the activities involved in the investigation of properties and behavior of this concept.

This investigation can be seen as a verification loop for each requirement attribute as illustrated in figure 3. This verification process is initiated by the requirement specification and is ending up in a decision basis for the project management.

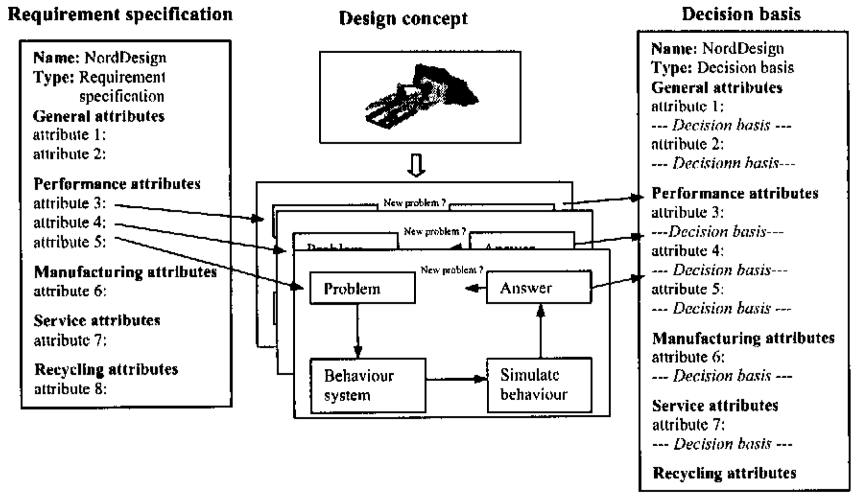


**Figure 3. A loop for verification of requirement attributes.**

The results of the activities involved in verification of formulated demands and wishes in the requirement specification are documented in a decision basis. This decision basis, the requirement specification and the design concept constitute the main parts of the design model, which is a subset of the total product model.

### 4.1 The problem Investigation matrix

The verification loop is initiated by an attribute in the requirement specification. However, each loop may yield results that can contribute to the verification of other attributes. Thus before starting a new loop by formulating a new problem, it is important to know whether any earlier results can contribute to this verification. But there has been a lack of any simple method or tool for visualizing these relations. To address this lack, I have developed a problem investigation (PI) matrix (see figure 4). In this matrix the requirements attributes are listed on the left-hand side and the problem formulations are listed at the top. The relation between a problem formulation and an attribute is marked in the matrix. A letter "I" means that this is the attribute initiated this problem formulation, and a letter "C" means that that the answer to this problem formulation contributes to the verification of this attribute.

| | | Problem | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Problem_Id1 | Problem_Id2 | Problem_Id3 | Problem_Id4 | Problem_Id5 | Problem_Id6 | Problem_Id7 | Problem_Id8 | Problem_Id9 | Problem_Id10 |
| Attribute | attribute_Id1 | I | | | | | | | | | |
| | attribute_Id2 | | I | | | | | | | | |
| | attribute_Id3 | | | I | | | | | | | |
| | attribute_Id4 | C | | | I | | | | | | |
| | attribute_Id5 | | | | | I | | | | | |
| | attribute_Id6 | | | | | | I | | | | |
| | attribute_Id7 | | | | | | | I | | | |
| | attribute_Id8 | C | | | | | | | I | | |
| | attribute_Id9 | | | | | | | | | I | |
| | attribute_Id10 | | | | | | | | | | I |

**Figure 4. The Problem Investigation (PI) matrix**

This method aims to visualize which analysis that have been made and if it contribute to verification of attributes other than the one initiated the problem. It can also support the search for verification of earlier product variants by using this method for illustrating dependencies and links to the objects in the matrix. Information that is interesting both when looking at earlier products as well as during development of a new product, can be to show e.g., objects that can be traced from the problem object, or result of the formulated problem, i.e. the answer document.

### 4.2 Tracing dependencies

The discussion about a tool, the PI matrix, for visualizing relations between attributes and problems automatically leads to the question of how to trace these relationships. Sutinen [Sutinen 2000] discusses three basic techniques that can be used for management and maintenance of traceability information. These are:

- traceability tables,
- traceability lists,
- automated traceability links.

Of these three, the PI matrix falls into the first category. The described objects contain sufficient meta information in order to use this technique to display the PI matrix. As a convenient way of interacting with the PI matrix the problem fields can contain links to the describing objects. A request, which falls into the third category above, to show a network graph of objects related to a problem object in the PI matrix can then in the general case result in a graph similar to figure 5. The dashed lines in the lower right in figure 5 indicates that if some information is missing for solving the initial problem, a new problem is formulated that must be solved before the initial problem. In the lower left part of figure 5, the dashed lines indicates that solving one problem may lead to new questions and problems that have to be taken care of before an answer can be formulated for the initial problem. The dashed lines further illustrate the iterative nature of this evaluation and verification process.

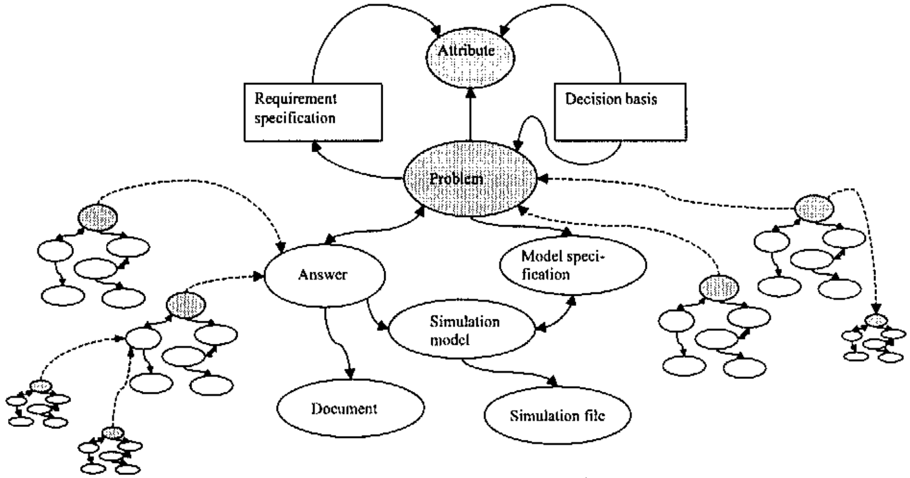


**Figure 5. A network graph illustrating traceability links**

## 5. Example: A wheel loader from Volvo CE

The presented question-answer driven process will now be used in a modeling and simulation scenario of a lifting unit in a wheel loader from Volvo CE. This scenario focus on the modeling and simulation activities and how these can be supported in an iterative question-answer driven process of investigation the behavior of the lifting unit.

To start with we assume that we have a situation as illustrated in figure 6. We will now follow the activities leading from the requirement specification to the decision basis for verifying the demand on work envelope for the wheel loader. In figure 6, this demand is specified by the attribute "Work envelope".

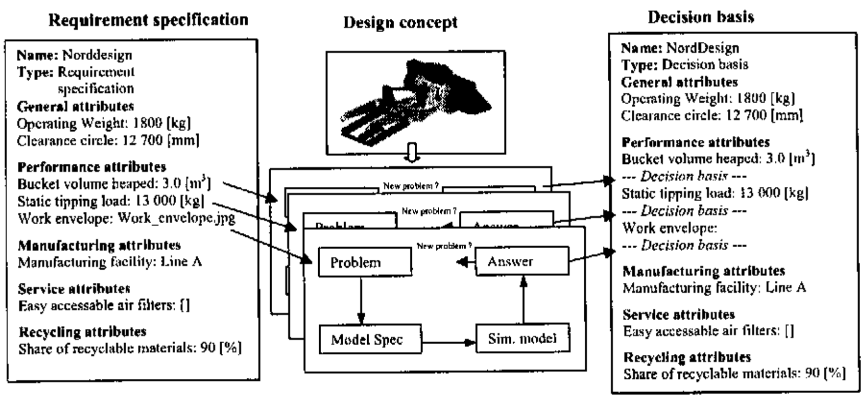


**Figure 6. Verification loop for the lifting unit.**

The first thing to in this in this process is to define the problem in terms of a question which in this case can be formulated as: " What is the static work envelope for full vehicle with lifting unit L110?" We also have to define what attribute this problem applies and also in which requirement specification. This is resulting in a problem object shown in figure 7(left). Note that the references to the model specification and answer object are filled in later in this verification process, i.e. figure 7 (left) shows the problem object and a graph (right) after one verification loop has been completed.

| | |
|---|---|
| Name | Problem_WE_031128 |
| Context | Lifting Unit |
| Description | What is the static work envelope for full vehicle with lifting unit L110 |
| Applies to | Req_spec_031128 |
| Attribute | Work_envelope_031128 |
| Model specification | Model_specification_031128 |
| Problem answer | Answer_WE_031201 |

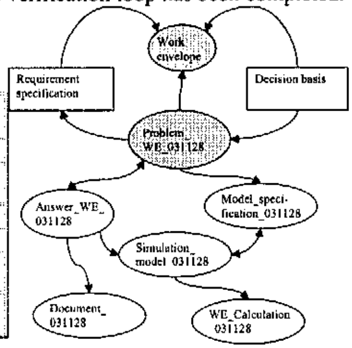


**Figure 7. A problem object (left) and a graph (right) of all objects created during one verification loop.**

Next step in this process is to define a model specification where we actually have to decide how to compose the system model that we will use to simulate the actual behavior property and compare it to the wanted value in the requirement specification. One of the first things we have to decide is which simulation environment that we want to use to perform the simulations. In this case we have decided to use ADAMS from MSC Software Inc. Then we have to select to subsystem models that should be used for the actual analysis. Here we have utilized the systems view of behavior simulation which means that we have predefined a number of subsystem models of each part as well as their connection points (mating features). In figure 8 two alternative behavior models of the lifting arms of the lifting unit is shown. The one at the left is a rigid ADAMS model and at the right a flexible ADAMS model based on an imported FE beam model. This is an example of the choices the engineer has at a certain modeling situation.

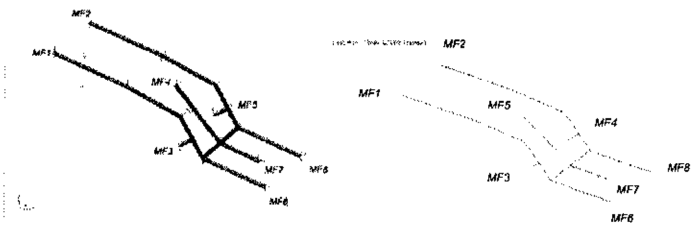


**Figure 8. Two different MBS models of the lifting arms, one rigid and one flexible.**

After selecting submodels the engineer also have to choose how to connect the different submodels to each other. Defining the interface features that connect the mating features of the different parts makes this connection. In this case the interface features are realized by joints, see figure 9.

| | |
|---|---|
| Name | Model_Specification_040213 |
| Context | Lifting Unit |
| Description | Model specification for calculation static work envelope for full vehicle with lifting unit L110 |
| Problem | Problem_WE_031178 |
| Target application | ADAMS |
| Submodels | {Lift_frame,Bucket,Lift_cyl,Tip_cyl,CrossB, GDF,GH,HIJ,BI} |
| Connections | {REV1,REV2,REV3,REV4,REV5,REV6,REV7,REV8, REV9,REV10,REV11,REV12,REV13,REV14,FIX1,FIX2} |

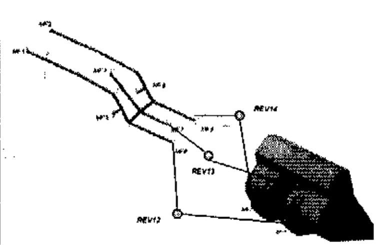

**Figure 9. Model specification object (left) and definition of interface features (right).**

Based on this model specification a command file containing the definition of a system model with all specified submodels and connections is generated. The ADAMS software to perform the wanted simulation of work envelope for this concept then uses this file. Before this analysis can be made a number of modeling issues have to be taken care of, e.g. modeling of stroke restriction on lift and tip cylinders. The resulting ADAMS model and the result of the investigation of work envelope is shown in figure 10.

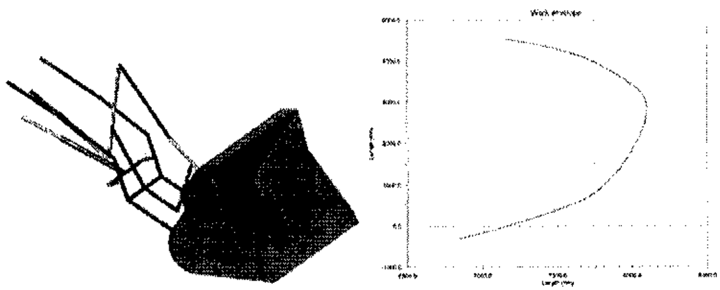

**Figure 10. ADAMS model of the lifting unit and the calculated work envelope.**

The final part of the question answer process (see figure 6) is the activity to formulate an answer to the formulated problem question. The result of this activity is a problem answer formulated in a document with graphs and drawings. This is represented in an instance of a "problem answer" object. In this activity we also have to consider if we have all information we need to verify this requirement or if more information is needed. We might find it necessary in this case to know how the behavior of a dynamic model of the lifting unit and how this will effect the work envelope. This give rises to a definition of a new problem question that needs to be investigated. This can either be created as a follow-up question (see the dotted lines connected to the answer object in figure 5) to the original or as a totally new problem associated to the *work envelope* attribute.

## 6. Summary and Conclusions

Product development is a very complex task that involves many people of different professions, working together during a long period of time, sharing the same goal or temporary goal. A common way to deal with complexity is to simplify, to concentrate on what is considered to be the most important properties. In this paper I have chosen to focus on the activities that, during product development deals with the verification of product requirements.

In earlier papers [Andersson 2004a, 2004b], I have presented a proposal to an information framework that can support a requirement driven product development process. The base for this is a design process model that is capable to describe problem statements, model specifications, simulation models and problem answers as separate objects. This design process model is based on the work by [Malmqvist 2000] and [Andersson 1997] and enables a fine granularity level of information with traceability on an object-to-object level between the attributes in the requirements specification and the estimated product properties.

This paper presents a proposal to a question-answer driven process for verification of product requirements, based on the framework discussed above. This is illustrated in a modeling and simulation scenario of a lifting unit in a wheel loader from Volvo CE. The scenario focus on the modeling and simulation activities and how these can be supported in a question-answer driven process of investigation the behavior of the lifting unit.

The results so far seem promising and can be seen as a step towards a more systematic approach to deal with verification of product properties. This opens up opportunities to reuse old knowledge in terms of earlier defined problems, simulation models being configured etc. A challenging task that can be seen as the next step in this approach is to investigate how to use knowledge based system to handle the created models and to utilize them for a decision support system.

**Acknowledgements**

The work presented here was performed in VISP research project and financially supported by the Swedish Agency for Innovation Systems (VINNOVA).

**References**

*Andersson K., "Modeling and Simulation as a Control Means for Product Development", 6th International Conference on Management of Technology, MOT97, Gothenburg, June 1997.*

*Andersson K., Sellgren U., "Modeling and simulation of physical behavior of complex products", , Produktmodeller '98, Linköping November 1998*

*Andersson, K.,, "A design process model for behavior simulations of complex products", ASME:s DETC 99, Las Vegas September 1999.*

*Andersson, K., "Simulation of a tunnel drilling sequence to determine loads on a rock drilling equipment", ADAMS User Conference, Rome 2000.*

*Andersson, K., "A Requirement Driven Product Development Process" 7th Workshop on Product Structuring – Product Platform Development, Chalmers University of Technology, Göteborg, Sweden 2004.*

*Andersson, K., "Simulation in product design: An iterative question-answer driven process" to be presented at Design 2004, Dubrovnik, Croatia, May 2004.*

*Hubka, V., " Theory of Technical Systems", ISBN 3-540-17451-6, Springer-Verlag 1984.*

*Lindemann, U., Statter, R., "Industrial application of the method "Early determination of product properties", Proceedings of DETC'98, Atlanta Georgia USA 1998.*

*Malmqvist J., "Implementing Requirements Management – A Task for Specialized Software Tools or PDM Systems?", Proceedings NordDesign 2000, DTU, Copenhagen 2000.*

*Sellgren, U., Andersson, K., " MOSAIC - a Framework and a Methodology for Behavior Modeling of Complex Systems", Produktmodeller '98, Linköping November 1998.*

*Sutinen K., Almefelt L., Malmqvist J., "Implementation of requirements traceability in systems engineering tools", Proceedings of Product Models 2000, Linköping 2000.*