INTERNATIONAL DESIGN CONFERENCE - DESIGN 2004
Dubrovnik, May 18 - 21, 2004.

DESIGN 2004

SIMULATION IN PRODUCT DESIGN: AN ITERATIVE QUESTION-ANSWER DRIVEN PROCESS

K. Andersson

Keywords: modelling, simulation, requirements, design process

1. Introduction

Product development is often described as an iterative process to find solutions that fulfil a given requirement specification. This process spans over many different dimensions, e.g. time, organisation and more direct specific product dimensions as abstraction level, detailing level and analysis for verification of product properties. Product properties of the finalised product should meet the targets set up in the requirements specification. However this is not a straightforward task and is often handled as a formulated problem or question that is sent to an analysis expert to investigate. This investigation is transforming the problem statement into a model specification, which, among other things, is depending on the current design phase, abstraction level and the overall importance of the requested properties. This model specification is then the basis for the configuration of a simulation model, e.g. based on pre made models of different subsystems followed by the actual analysis that can be used to answer the question.

A requirements-driven product and process model is discussed by Malmqvist [Malmqvist 2000], which is based on the chromosome product model by Andreasen [Andreasen 1992]. One of the key issues in the product model discussed by Malmqvist is the requirement management, RM, system. Further the granularity of information in the RM system is a key characteristic that determines what traceability that can be obtained. The granularity of information can be either object granularity or document granularity. Malmqvist further defines three types of traceability that can be supported in a RM system, .i.e. document to document, object to document and object to object.
In a study by Almefeldt et al. [Almefeldt 2003] on requirements management in the automotive industry showed that the procedures that are practiced in industry are being influenced and harmonised to those described in academic literature. One finding in study is that the follow-up of requirements and their fulfilment tends to be more problematic than management of the requirement specification itself.

For management of maintenance of traceability information there are three basic techniques that can be used according to Sutinen [Sutinen 2000]. These are:

- traceability tables
- traceability lists
- automated traceability links.

Ramesh [Ramesh 97] discusses a framework for model traceability and an intelligent decision support system that utilises this framework for reasoning in order to support model development and maintenance activities in software development.

In this paper I will discuss the need for traceability of the models that document the verification process initiated by a stated requirement in the requirement specification and resulting in a decision basis. For this process the product model need to be extended to handle a number of new object types, e.g. problem statements and model specifications. Further, the product model should support requirement management that can to have traceability on the object to object level or at least object to document level, to enable a sufficient follow-up of the formulated requirements.

2. The design process model

To start with we need to describe the process in which we want to put a further functionality, i.e. an improved traceability of requirements and verification of requirements. As a starting point we can use the general design process model below.

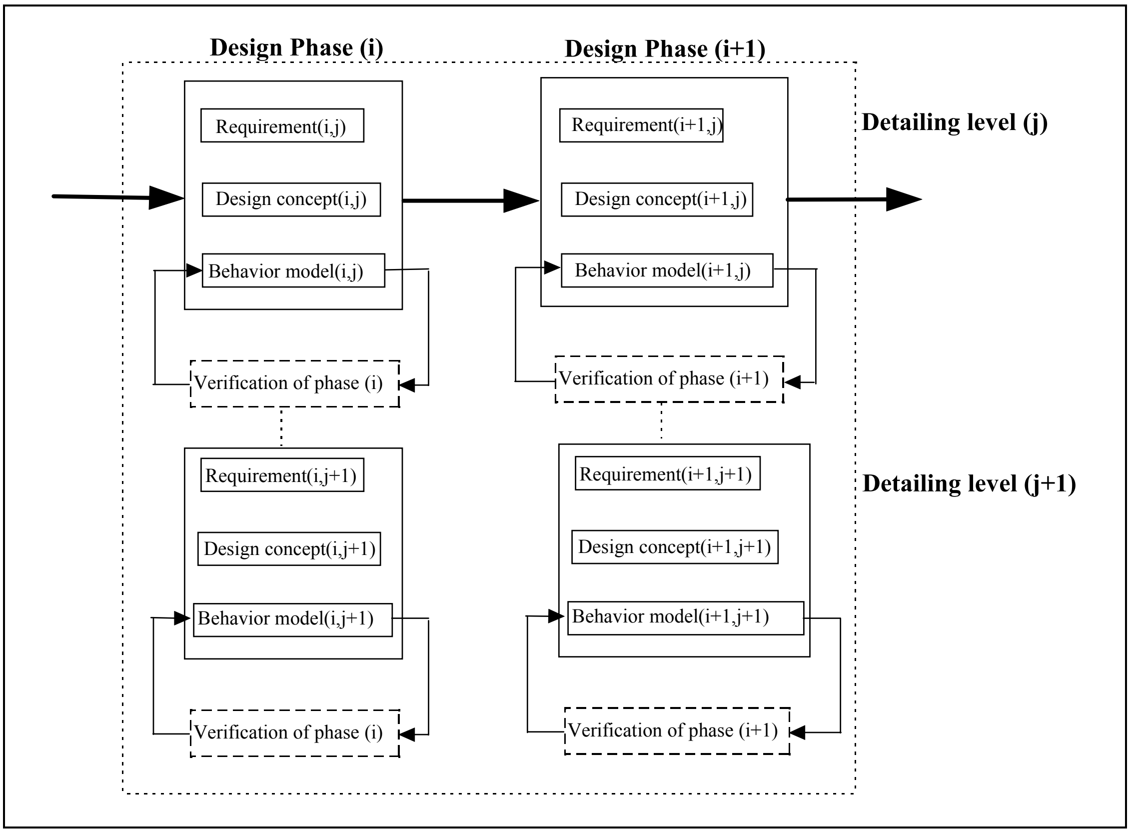

Figure 1. A generic design process model [Andersson 97]

This is a generic model and in order to capture the verification of requirements we need some additional functionality. A first step in this direction is to relate the generic process model in figure 1 to a more specific activity oriented stage-gate model as shown in figure 2. Figure 2 also shows the information flow, where the decision at a specific gate is resulting in a requirement specification that will guide the design work in the descending design phase.

This model can be interpreted, as the result of the decision at gate G-3 is the mission to further develop a product concept according to a specific requirement specification. This is to be done during the concept phase in the figure above. From the verification and analysis point of view this means that we have to find out if the requirements are met and on what basis. This is investigated in the activity

"Investigate problem" in figure 2 and is resulting in a decision basis for the next gate in the design process.

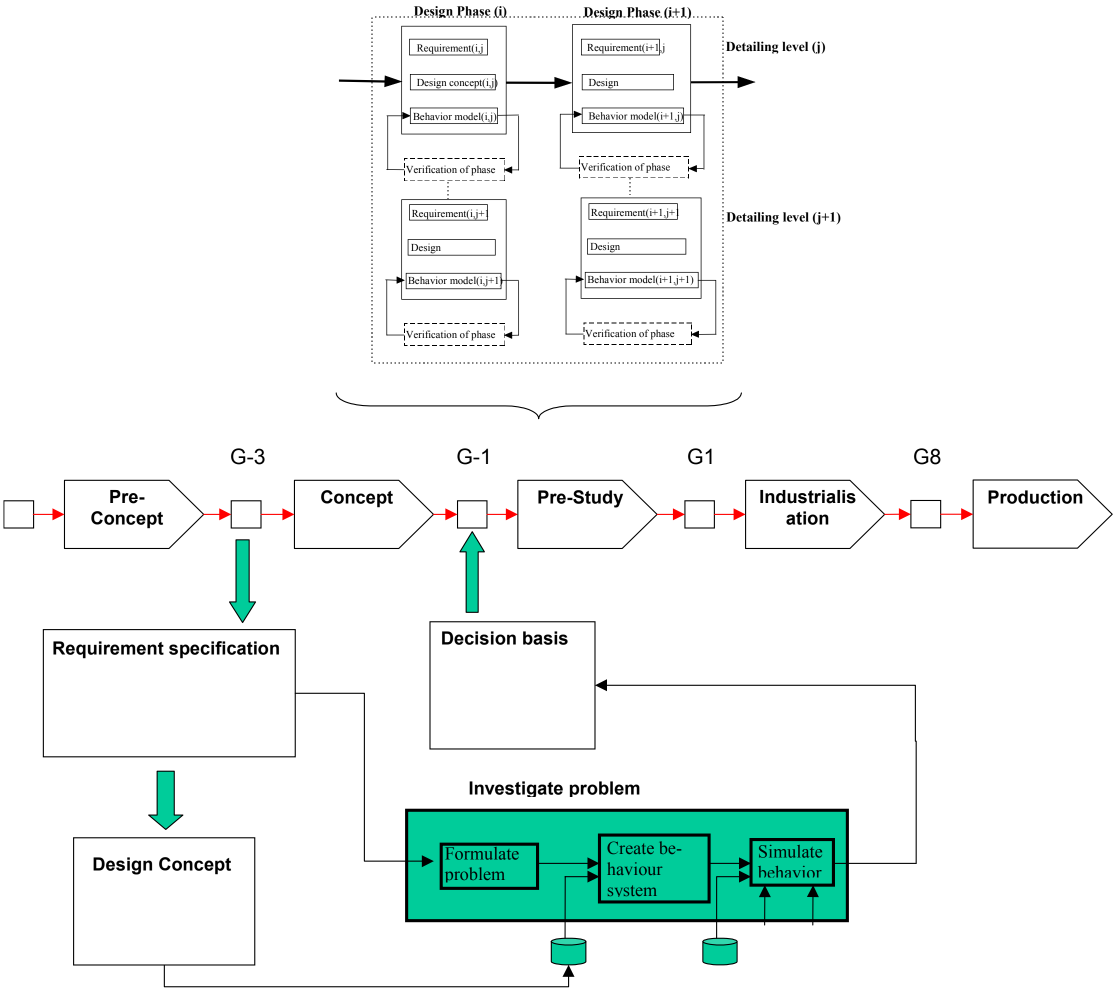

Figure 2. Relation between the generic design process model and a stage-gate model

2.1 Verification of requirements

The series of activities involved in the 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 (figure 3), which is a subset of the total product model.

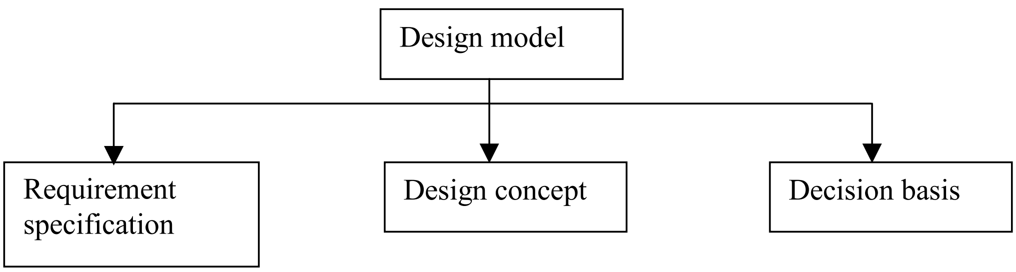

Figure 3. Main components of a design model

The base for the verification is what in figure 4 is the named activity “Investigate problem”. This activity takes the requirement specification as input for the first sub activity “Formulate problem”.

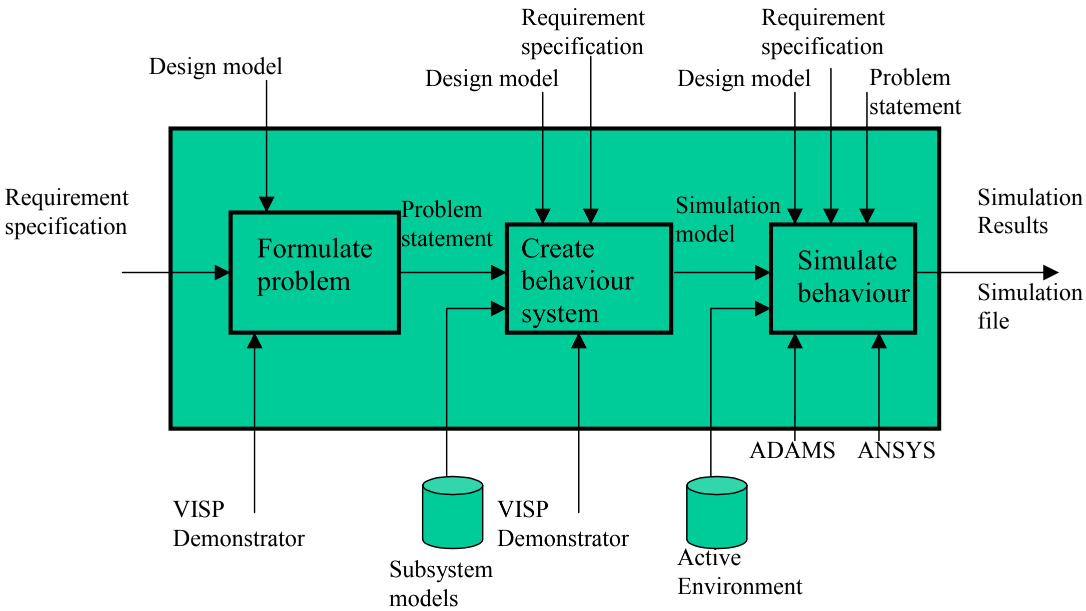

Figure 4. The activity “Investigate problem”

2.2 Formulate problem

This activity is the first in a sequence of activities being performed during this phase with the focus to explore the properties of the design concept. More precisely this activity means that the designer will define the problem, supported by e.g. a web-based interface where the problem is formulated as a question. This information is then stored in an object as an instance of the “problem” object.

2.3 Create behaviour system

This second activity is initiated by the formulated question and starts with a control if all conditions are known needed to solve the problem and answer the question. If this not is the case an additional problem is formulated. This additional problem needs to be solved first so that all necessary information is available before solving the initial problem. After that we need to make an model specification where we state what type of information we need to get from the simulation, e.g. if we need the dynamic or static properties of the product concept.
Next in turn is to configure the system model, according to the model specification, that can be used to explore the properties of the product concept. This configuration utilizes pre made subsystem models stored in a modelling database; see [Andersson 99], [Andersson 00].

2.4 Simulate behaviour

This third activity takes the configured model put the additional constraint on the model; based on the active environment it will work within and performs one or a series of analysis. This also includes importing the simulation model into the selected analysis tool, e.g. ADAMS, where the additional conditions are defined.

3. Example: Wheel Loader from Volvo CE

The presented process model will now be used in a modelling and simulation scenario of a lifting unit in a wheel loader from Volvo CE. This scenario focus on the modelling and simulation activities and how these can be supported in an iterative question-answer driven process of investigation the behaviour of the lifting unit. To start with we assume that we have a situation as illustrated in figure 5 below. 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 5, this demand is specified by the attribute “Work envelope”.

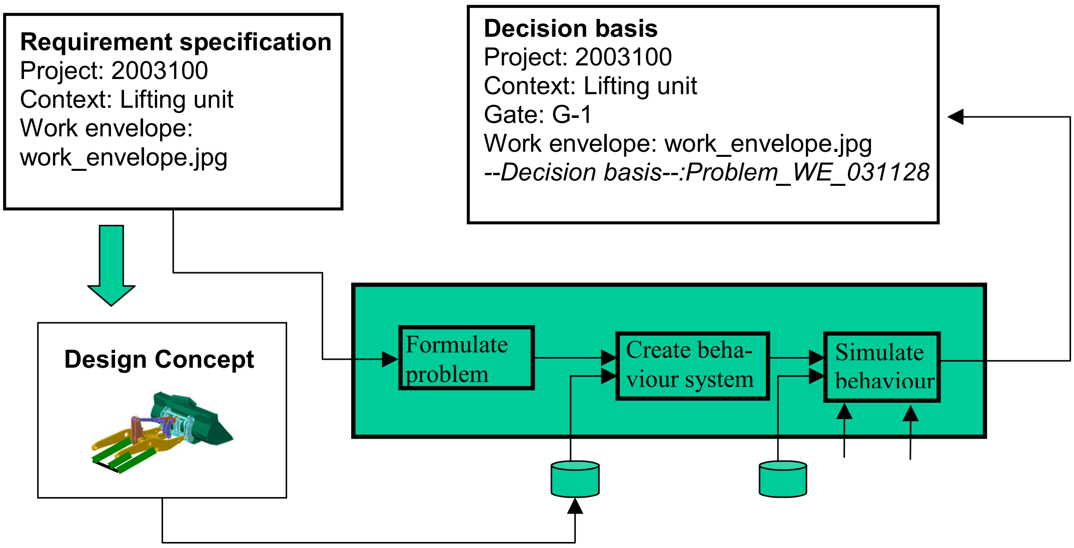

Figure 5. The verification loop for the wheel loader example

During the verification loop the product model is evolving. In this paper a subset of the product model is represented as a design model. The design model of the lifting unit is shown in figure 6, where only the main relations, consists_of (solid) and applies_to (dashed) are shown.

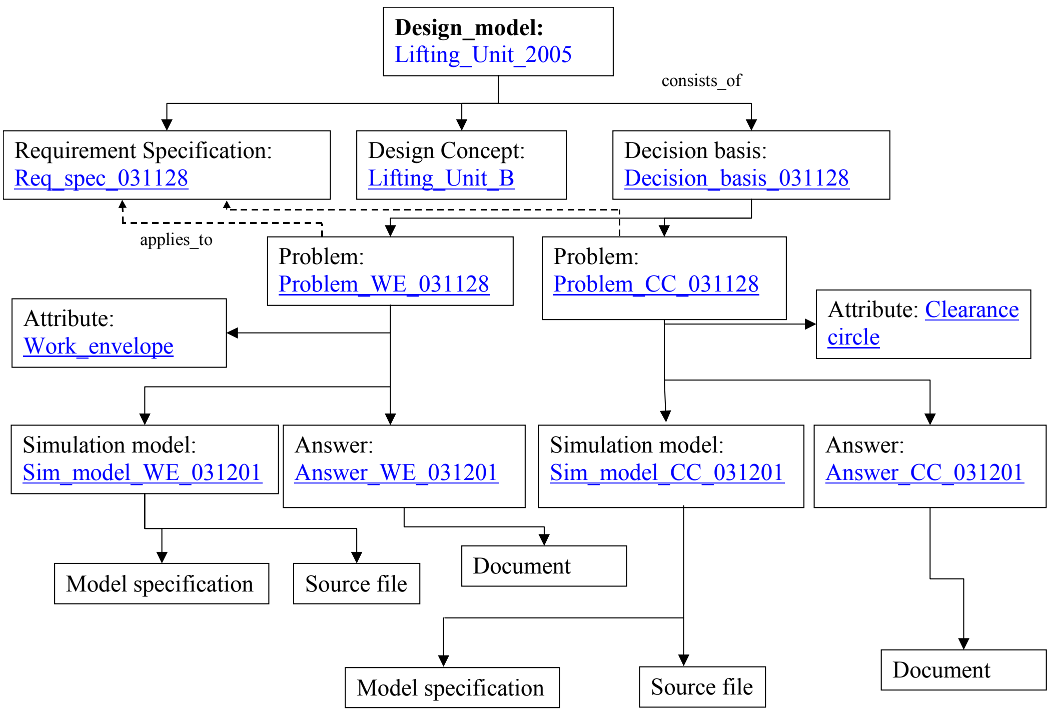

Figure 6. A design model of the lifting unit

3.1 Formulate problem

In this first verification step we have to define the problem in terms of a question description and what attribute it applies to in the requirement specification. In this example we focus on the verification of the wanted work envelope for the lifting unit. The object describing the problem is shown in figure 6 and 7. Note that the attribute is represented as a separate object as shown in figure 7. We also have to define the context for this problem. This information is stored in a problem object where we also add references to a simulation model created to solve this problem as well a reference to the answer to this question, see figure 7.

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
Simulation model	Simulation_model_031128
Problem answer	Answer_WE_031201

Name	Work_envelope_031128
Context	Lifting Unit
Description	Working envelope for full vehicle with lifting unit L110
Value	See graph below
Image	Graph

Figure 7. A problem description object (left) and a description of the requirement attribute "work envelope" (right)

3.2 Create behaviour system

Once we have created a problem description, the next step is to define a model specification, which defines what simulation submodels we use and how these should be connected. We also have to define what simulation tool that should be used to simulate the system behaviour. In this example we define that a rigid ADAMS model is sufficient in order to get a rough estimate in this early design phase.
Then, selecting pre-made ADAMS subsystem models from a modelling database performs the configuration of the system model. In these subsystem models we have separated to connection interface to other subsystem models, see [Andersson 99], [Andersson 00]. This enables us to explicitly define the connections between the subsystems. This activity is resulting in a model specification object, see figure 8.

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_031128
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}

Name	Simulation_model_031128
Context	Lifting Unit
Description	Calculation static work envelope for full vehicle with lifting unit L110
Model specification	Model_specification_031128
Simulation file	WE_Calculation_031128

Figure 8. A model specification object for calculation of work envelope (left) and the simulation model object (right)

3.3 Simulate behaviour

Next, the configured simulation model will be loaded into the simulation software, in this case ADAMS, where additional constraints need to be added. In this example with the lifting unit we have to define how to control the movement by defining restrictions on max- and min lengths of the cylinders. The outer boundary of the working envelope can then be obtained by using an external force dragging the bucket to its maximum reach positions. Figure 8 illustrates the simulation model object for this example and figure 9 shows the ADAMS model for this calculation together with a plot of the resulting work envelope. The simulation model object in figure 8 contains references to the model specification as well as the ADAMS file, which contains the results data.

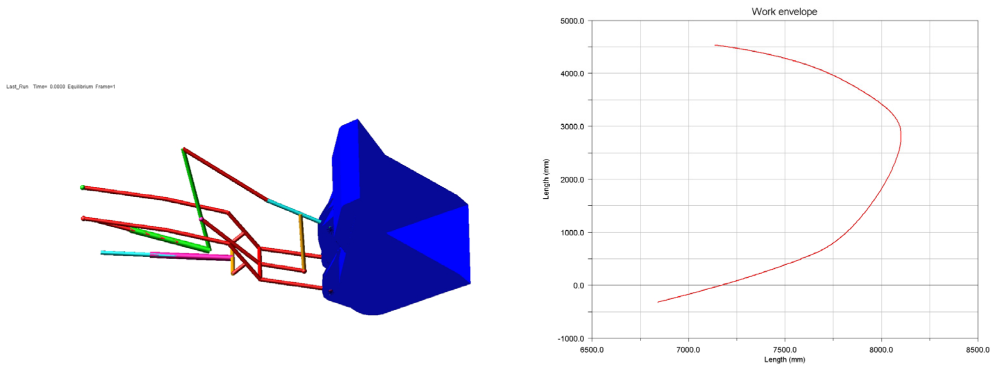

Figure 9. An illustration of ADAMS simulation model (left) and the simulated work envelope (right)

As a final activity after performing the behaviour simulations is to interpret the results of the simulations and to formulate an answer to the posed question in the problem definition.
This answer is e.g. formulated in a text-based document, with tables or graphs from the analysis tool, where the conclusions of the analysis should be clearly stated.

3.4 Decision basis

The Investigation of product properties of the actual design concept is an iterative process where a problem description object is created for each attribute in the requirement specification that have to be verified during a specific development phase.

Name	Decision_basis_031128
Description	Decision basis for lifting unit
General attributes	
Operating Weight	18 000 [kg]
---*Decision basis*---	
Clearance Circle	12 700 [mm]
---*Decision basis*---	Problem_CC_031128
Performance attributes	
Bucket volume, heaped	3,0 [m3]
Static tipping load, straight	13 000 [kg]
Static tipping load, at 35 deg turn	12 000 [kg]
Static tipping load, at full turn	11 500 [kg]
Breakout force	166 [kN]
Digging Cycle time	15 [S]
Work envelope	Work_envelope.jpg
---*Decision basis*---	Problem_WE_031128
Manufacturing attributes	
Manufacturing facility	Line A

Figure 10. Decision basis for the lifting unit

This iterative process is resulting in a decision basis that is successively filled in, where references to problem description objects serve as a decision basis for each attribute, see figure 10. This iterative process is illustrated as a loop in figure 5 where only part of the requirement specification and the decision basis is shown.

4. Summary, conclusions

In order to support a question-answer driven simulation process, a design process model that is capable to describe problem statements, model specifications, simulation models and problem answers as separate objects is introduced. This is based on the work by Malmqvist [Malmqvist 2000] and Andersson [Andersson 97] and enables a granularity level of information that allows traceability on an object-to-object level between the attributes in the requirements specification and the estimated product properties. This also enables a traceablity and reuse of partial result created during the verification of a specific requirements attribute as well as a possibility to study the effects that changes in the requirements specification have on product properties.

A methodology for an iterative question driven simulation process using this design process model has been illustrated in a modeling and simulation scenario of a lifting unit in a wheel loader from Volvo CE. In this scenario the iterative question-answer driven process of investigating the behaviour of the lifting unit is described as well as the resulting decision basis for the next decision point in the stage gate model.

Acknowledgements

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

References

Almefelt L., Andersson F., Nilsson P., Malmqvist J., "Exploring requirements management in the automotive industry", Proceedings of ICED´03, Stockholm 2003.

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.,,"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.

Andreasen M. M., "Designing on a "Designers Workbench" (DWB)" Proceedings of the 9th WDK Workshop, Rigi, Switzerland 1992.

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

Ramesh B., "Representing and reasoning with traceability in model life cycle management", Annals of Operations Research, Vol. 75, pp. 123-145, 1997.

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

Kjell Andersson, Docent, Associate professor
KTH, Machine design
SE- 100 44 Stockholm, Sweden
Telephone: +46 8 790 6374, Telefax: +46 8 20 2287
E-mail: kan@md.kth.se