Supporting Collaboration in an Extended Enterprise with the Connector View on Enterprise Models

Anders Carstensen[1], Lennart Holmberg [2], and Kurt Sandkuhl[1]

[1] School of Engineering at Jönköping University,
P.O. Box 1026, 55111 Jönköping, Sweden
{anders.carstensen,kurt.sandkuhl}@jth.hj.se
[2] Kongsberg Automotive
P.O. Box 504, SE-565 28 Mullsjö, Sweden
lennart.holmberg@ka-group.com

Abstract. The work presented in this paper aims at contributing to enterprise interoperability. The focus is on collaboration in extended enterprises as general application field and active knowledge models as underlying technology. The starting point for the paper is an application case from automotive industry, where active knowledge models were developed for essential product development tasks. The paper concerns the development of a connector between these active knowledge models in order to support collaboration between two partners. The focus is on the operationalising phase of the connector development and on experiences. One result from this work is to propose three levels for the modeling, depending on the maturity level of the collaboration - searching for a partner, modeling existing collaboration and enhancing collaboration through detailed modeling.

Keywords: Enterprise model, extended enterprise, collaborative work, connector view, enterprise interoperability.

1 Introduction and Background

Enterprise interoperability has been a research subject since more than 20 years and still offers numerous challenges for the scientific community. Researchers have been examining the interoperability challenges on various levels (e.g. from the business models to technical details [17]) and from different perspectives (like those described by ODP [10]). The spectrum of solutions proposed spans form general reference models or frameworks (see [18] for an overview) to service description languages and protocols for heterogeneous environments.

The work presented in this paper aims at contributing to enterprise interoperability and focuses on (1) collaboration in extended enterprises as general application field, (2) enterprise knowledge modeling as underlying technology, and (3) networked manufacturing in the specific use case considered. An extended enterprise is a dynamic networked organization, which is created ad-hoc to reach a defined objective using the resources of the participating enterprises. The focus on collaboration in extended

J. Stirna and A. Persson (Eds.): PoEM 2008, LNBIP 15, pp. 111–126, 2008.

enterprises indicates that not easily automatable routine activities are in the center of our interest, but complex processes involving several actors in different companies and their coordination. We take advantage of progress in the field of active knowledge modeling and propose to use such models for analyzing collaboration requirements, designing solutions in a participative manner involving relevant stakeholders, and operationalising these solutions. One important characteristic of collaboration in networked manufacturing is the weight of product knowledge in the enterprise knowledge models, i.e. knowledge about product structure, design concepts, configuration possibilities is essential for the collaboration.

The article is structured as follows: the remaining part of this section introduces the application context of this work and the subject active knowledge modeling. Section 2 summarizes previous work on the approach of connector views. Section 3 introduces first steps to operationalising of the connector view approach including material from the industrial case. Section 4 contains a discussion and related work. Future work and conclusions are presented in section 5.

1.1 MAPPER - Project

This paper is based on work in the EU-FP6 project MAPPER[1] (Model-adapted Process and Product Engineering). MAPPER had a runtime from autumn 2005 until spring 2008 and aimed at enabling fast and flexible manufacturing in networked enterprises by providing methodology, infrastructure and reusable services for collaborative engineering. One of the MAPPER use cases focused on distributed product development involving different subsidiaries of an automotive supplier. The collaboration of this automotive supplier with a sub-supplier forms the context for the work presented in this paper.

Product development in this context includes elicitation of system requirements based on customer requirements, development of functional, design of logical and technical architecture, co-design of material, electrical and mechanical components, integration testing and production planning. The process is geographically distributed, as it involves the sub-supplier and engineers and specialists at several locations of the automotive supplier. The purpose of the enterprise knowledge modeling in this use case was to capture the relevant product knowledge and process knowledge required for collaborative engineering. In particular, the knowledge model has to enable flexible collaboration processes between supplier and sub-supplier, i.e. an integration of both partners on work process, product model and collaboration service level is crucial.

1.2 Active Knowledge Modeling (AKM)

Enterprise knowledge modeling aims at capturing reusable knowledge of processes and products in knowledge architectures supporting work execution [2]. These architectures form the basis for model-based solutions, which often are represented as active knowledge models (AKM) [3]. An essential characteristic of active models vs. passive models is that “the model must be dynamic, users must be supported in

[1] See [1] for more information about MAPPER.

changing the model to fit their local reality, enabling tailoring of the system's behavior" [4]. Furthermore, the model must "first and foremost be available for the user in the operational information system at execution time. Second, the model must automatically influence the behavior of the computerized work support system or workplace. Third, the model must be dynamically extended and adapted; users must be supported in changing the model to fit their local needs, enabling tailoring of the work environment's behavior" [6, p. 5].

In MAPPER, the active knowledge models included the POPS* perspectives, which are mutually reflective [5]:

- process perspective (P): work processes and tasks,
- organization perspective (O): roles involved in the processes
- product perspective (P): components, configuration possibilities and dependencies of the product family,
- systems perspective (S): IT systems supporting processes and product development,
- further perspectives (*) depend on the purpose of the modeling project.

The overall modelling process was inspired by the C3S3P methodology. C3S3P is based on work in several EU projects, like EXTERNAL [7] and ATHENA[2]. C3S3P distinguishes between seven stages called concept study, scaffolding, scoping, solution modeling, platform integration, piloting in real projects and performance monitoring and management. More information about C3S3P is provided in [8] and [6].

In order to support the application of active knowledge models in everyday industrial practice an IT-platform was developed consisting of the enterprise modeling tool METIS[3], an execution environment, modeling templates and model-generated work places. In the remaining part of this paper, this platform will be called AKM-platform[4]. The most important part in the AKM-platform for the end user are model-generated and role-specific work places, which allow for capturing and modifying product knowledge during product development in collaboration with partners.

2 Previous Work

The work presented in this paper builds to a large extent on previous work presented in [9], which will be summarized in this section.

The starting point for our work is the use case briefly introduced in section 1.1. In this context, two companies (partner A and partner B) are setting up a new collaboration. Partner A is active as a supplier towards the automotive manufacturing industry, while Partner B is a supplier of wire products for Partner A. In the use case Partner A and B want to improve their collaboration concerning wire solutions for seat-heating. In this case the wire is a material for producing the seat heating product at partner A, while the wire at the same time is a product offered by Partner B. For the planned improvement at partner A, several negotiable solutions exist. These include different materials with different properties and price-levels. Hence the collaboration situation is not trivial and includes different departments and roles at each partner.

[2] http://www.athena-ip.org/

[3] http://www.troux.com. METIS was recently renamed to Troux Architect.

[4] http://www.akmodeling.com

The purpose of the work described in [9] was to model the collaboration between the two partners. Starting point were existing active knowledge models, which capture organizational knowledge in collaborative engineering using the POPS* perspectives, for example for tasks like "establish material specification" or "develop new test method". The main result of the previous modeling activity was an additional active knowledge model with the connector view, common for all collaborating partners, and a supporting view for each partner. The *connector viewpoint* defines the content of the connector view and a methodology to design the connector view.[5] Table 1 shows a definition of the connector viewpoint and the associated methodology. By using the connector view several advantages can be gained:

- "Existing models for the involved companies need not be changed, in the initial stages;
- Stakeholders have a natural place to relate collaborative elements;
- It is not necessary to align existing enterprise models according to one specific partner." ([9]).

The Reference Model for Open Distributed Processing (RM-ODP) is an ISO standard which among other things defines five different viewpoints for developing interoperability in distributed systems. These viewpoints are: the enterprise viewpoint, the information viewpoint, the computational viewpoint, the engineering viewpoint and the technological viewpoint [11]. In our use case we defined supporting views that consists either of elements that already are existing in the partners enterprise model or of triggering events that need to be added. Thus the supporting views can be seen as part of the enterprise views of the partners, while the connector viewpoint extends the set of viewpoints given in RM-ODP.

As described above we have developed a connector view common for both partner A and partner B, with a supporting view for each partner. In the connector view we have defined several different collaboration elements (23 to be exact). Each such collaboration element relates to one or several elements in each partner's supporting view and also to one or several *Information objects* in an *Information view*, common for both partners[6]. The collaboration elements define activities that emanate from the collaboration and thus cannot be said to belong to any partner's enterprise models and is not owned by a specific partner. Likewise, the information objects shared between the collaborating partners do not belong to just one of the partners, even though maintenance (such as storage etc.) of the information is a concern for the respective partner. By this approach we want to address the mutual need to explore what information and product knowledge is essential in the collaboration.

The first parts of the methodology in the connector viewpoint defines how to build (or establish) the connector view (see stage 1 to stage 3 in table 1) while the last part defines *Operationalising* the connector view (see stage 4 in table 1). In the previous work [9] *Operationalising* the connector view was not performed, and is a subject of this paper.

[5] More about viewpoint can be found in [10].

[6] The information viewpoint is one of the viewpoints of RM-ODP and should give a logical object-based representation of the distributed data and the constraints and possible manipulation of the data [11].

Due to the complexity of the operationalisation stage, the following sections will not cover this completely, but rather position the landmarks for this stage. The methodology for defining the connector view is a work in progress and requires additional use cases.

Table 1. Definition of the connector viewpoint

<table>
<tr><td colspan="3">The connector viewpoint is an integrative viewpoint on different enterprise models exposing the intersection of these models including:
▪ Objectives of the collaboration between the enterprises in question;
▪ Overlapping processes or tasks;
▪ Information shared or exchanged in these processes;
▪ Resources involved in the overlapping processes or supporting the collaboration, like IT-systems or machinery;
▪ Roles involved in the overlapping processes or tasks and – if required their competences.
A view based on the connector viewpoint may include <i>collaboration elements</i>, which identify subjects of organizational or technical change (transient nature) when the collaboration is implemented. These elements are not owned by any of the collaborating partners, and may correspond towards supporting objects.</td></tr>
<tr><td>Stage</td><td>Step</td><td>Methodology description</td></tr>
<tr><td>1</td><td>a</td><td>Model goals and problems. It is important to find out from each partner what goals they want to meet with the system integration and what the problems are in meeting these goals.</td></tr>
<tr><td>1</td><td>b</td><td>Identify existing partner models reflecting the information usage. These can be the existing enterprise models that are relevant for the case to be modelled at the specific partner.</td></tr>
<tr><td>1</td><td>c</td><td>Identify collaboration elements in the connector view. E.g. how to connect tasks belonging to different collaborating partners, decisions to be taken, exchange of certain information etc.</td></tr>
<tr><td>2</td><td></td><td>Identify information necessary for collaboration elements in the connector view. Identified information objects are related to the collaboration elements in the connector view, (see 4b).</td></tr>
<tr><td>3</td><td>a</td><td>Identify supporting objects that relate to the collaboration elements. These are extensions of existing partner models that are relevant tasks or other objects in the enterprise for supporting the communication through the collaboration elements in the connector view.</td></tr>
<tr><td>3</td><td>b</td><td>Identify resources connected to supporting objects. Such as: roles, documents and systems as sources for information.</td></tr>
<tr><td>4</td><td>a</td><td>Relate collaboration elements through the supporting objects to the partner models. Collaboration elements are related to supporting objects and supporting objects to objects in the partner models.</td></tr>
<tr><td>4</td><td>b</td><td>Operationalise the connector view<b>.</b> Describe information usage in connector view. This includes defining the rules for how the information should be interoperable. It may include such things as: How terminology matches between systems, triggering mechanisms for connecting and exchanging information etc. In this step a re-evaluation of the initially stated goals and problems should be done in order to really focus on what is important when setting up the functionality of the connector view.</td></tr>
</table>

3 Towards Operationalising the Connector View

3.1 Describing the Collaboration (Elements)

The first step in operationalising the connector view is to define in a collaboration-description the correspondences between elements in the connector view and elements in the respective companies' supporting views. A question of interest is how the information objects related to the collaboration element is used in these correspondences? Due to the limited time given in the project for this analysis, only five collaboration elements have so far been included in the description. One such collaboration element included is "Meet to communicate requirements and decide on test methods", shown in figure 1. In the figure the relationships between the elements in the supporting views and the collaboration element and also between the information objects in the information view and the collaboration element are shown. The related elements are listed in tables 2 to 4. For this particular collaboration element the following collaboration description was written:

"At [*partner A*] requirements are specified and to get input for the requirement [*partner B*] is asked for technical support, this is an iterative procedure. When the requirements have been developed they are sent to [*partner B*]. [*partner B*] checks the possibilities of existing products in their portfolio, which meets the requirements on the wire. If no available solution exists an investigation of new development will be started. Requirement on the product (the wire) is important for determining which test method to use, but it is also necessary to discuss requirements on the test method as such. In order to meet the requirements from [*partner A*], [*partner B*] has to search for available test methods and this may trigger an investigation of new development of test method."

The written collaboration-description is an initial part of stage 4 b (see table 1) in the methodology for developing the connector view. The purpose is to analyze how the partners want to develop their collaboration and thus elaborate the specifics of the collaboration elements. There are several supporting elements at both partners which trigger the collaboration element and the specification of the collaboration element (in the collaboration-description) gives some directions of how these supporting elements act together.

The collaboration element is on a high level of conceptualization and we can state that there is no strict procedural order between the activities that trigger the collaboration element. This is illustrated by partner A, where we have (in accordance with stage 4a in the methodology described in table 1) associated the supporting elements with tasks in the active knowledge models forming the starting point for our work (see section 2). There are several elements in these models which in one way or the other may use the supporting object and thereby contribute to triggering the collaboration element. E.g. for the supporting object "Ask for technical support, suggestions of material" the model of "Establish material specification" has 4 associated tasks, "Internal testing" has 2 tasks and "Develop new test method" has 12 associated tasks. This indicates that there is no strict procedural order in which the supporting elements trigger the collaboration element. This seems also to be case for the other four described collaboration elements. Although there can be some activities that need to be performed before others there is often alternative or iterative paths where the order can not be strictly predicted in advance.

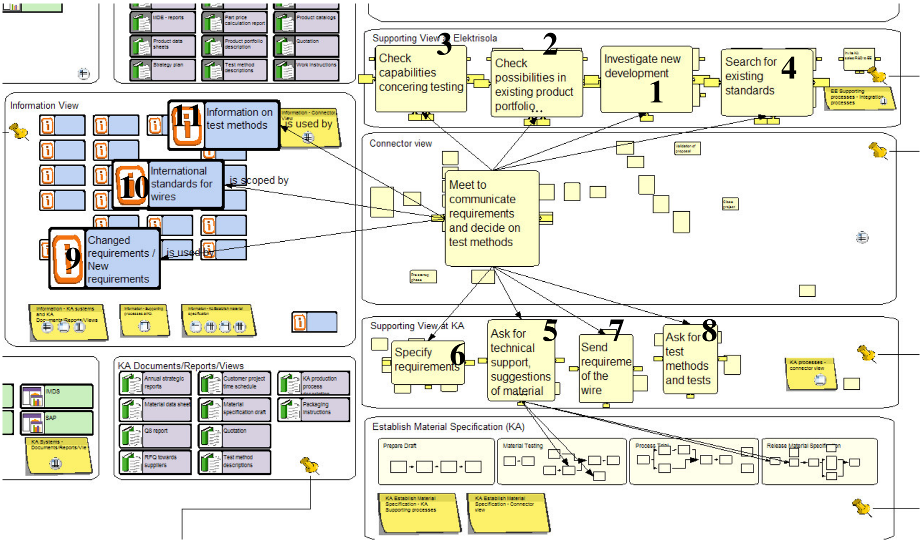
Supporting View at Elektrisola
Check capabilities concering testing
3
Check possibilities in existing product portfolio...
2
Investigate new development
1
Search for existing standards
4
EE Supporting processes - Integration processes
Information View
Information on test methods
11
Information - Connector View
is used by
International standards for wires
10
is scoped by
Changed requirements / New requirements
9
is used by
Information - KA systems and KA Documents/Reports/Views
Connector view
Meet to communicate requirements and decide on test methods
Supporting View at KA
Specify requirements
6
Ask for technical support, suggestions of material
5
Send requireme of the wire
7
Ask for test methods and tests
8
KA processes - connector view
KA Documents/Reports/Views
Annual strategic reports
Customer project time schedule
KA production process
Material data sheet
Material specification draft
Packaging instructions
Q8 report
Quotation
RFQ towards suppliers
Test method descriptions
IMDS
SAP
KA Systems - Documents/Reports/Views
MDE - reports
Part price calculation report
Product catalogs
Product data sheets
Product portfolio description
Quotation
Strategy plan
Test method descriptions
Work instructions
Establish Material Specification (KA)
Prepare Draft
Material Testing
Process Testing
Release Material Specification
KA Establish Material Specification - KA Supporting processes
KA Establish Material Specification - Connector view

Fig. 1. On the right side, the connector view is shown in the middle with the supporting view for partner B above and the supporting view for partner A beneath. On the left side the information view. Some of the elements in the supporting views and information view are related to the collaboration element "Meet to communicate requirements and decide on test methods". These elements have been enlarged to make them more readable in the figure. The numbers reference the corresponding rows in tables 2 to 4 for the modeled supporting- and information elements.

Table 2. Elements in the supporting view for partner B, related to the collaboration element "Meet to communicate requirements and decide on test methods"

	Name	Description
1	Investigate new development	To think about new solution together with the team at [*partner B*]
2	Check possibilities in existing product portfolio	Compare requirements with properties of existing products
3	Check capabilities concerning testing	Test methods, test equipment, human resources and competences and experiences. Includes a meeting with QA/central lab, production and plant manager.
4	Search for existing standards	International wire standards (focused on the product and test methods), material safety data sheets, IMDS information, general data sheet of components, data sheet of specific solution.

Table 3. Elements in the supporting view for partner A, related to the collaboration element "Meet to communicate requirements and decide on test methods"

	Name	Description
5	Ask for technical support, suggestions of material	Initial communication concerning requirements and possible technical solutions
6	Specify requirements	Conclude on important properties and their requested values
7	Send requirements of the wire	The material specification is send usually by e-mail
8	Ask for test methods and tests	Ask the supplier for recommended test methods for the important properties

Table 4. Information object in information view related to the collaboration element "Meet to communicate requirements and decide on test methods"

	Name	Description
9	Changed requirements / New requirements	Can be find in the up-dated material specification
10	International standards for wires	Can be both standards concerning the product or test methods for the product
11	Information on test methods	Can be both inline test methods and offline test methods (tests made in the lab like tensile strength and bending tests). This can be further divided in existing test methods and new test methods developed either at [*partner A*] or at [*partner B*].

3.2 Further Analysis of the Connector View

In order to further refine and detail the collaboration elements in the connector view we decided to analyze the collaboration description by applying the IRTV-methodology from [6]. We have made this analysis on the specific collaboration element "Meet to communicate requirements and decide on test methods". Accordingly we modeled the four different dimensions: Information, Role, Task and View (IRTV), using the tool Metis and the AKM-platform. This is an iterative work, starting with modeling the Information and the Role dimensions, as seen in figures 2 and 3.

The application case is still under development and we will briefly present only some of the details from our work. It has to be mentioned that the AKM-platform is still under development and our work has been done using a prototype installation which does not have the full capacity that the platform is supposed to have when it is completed.

Stakeholders representing the different roles have been involved in the modeling at specific occasions. Between these meetings the modeling facilitator has prepared a solution, based on previous discussions, to be evaluated and further discussed. Information structures that should be possible to capture in the workplace are modeled in the information dimension. Such an information structure is "Component Specification",

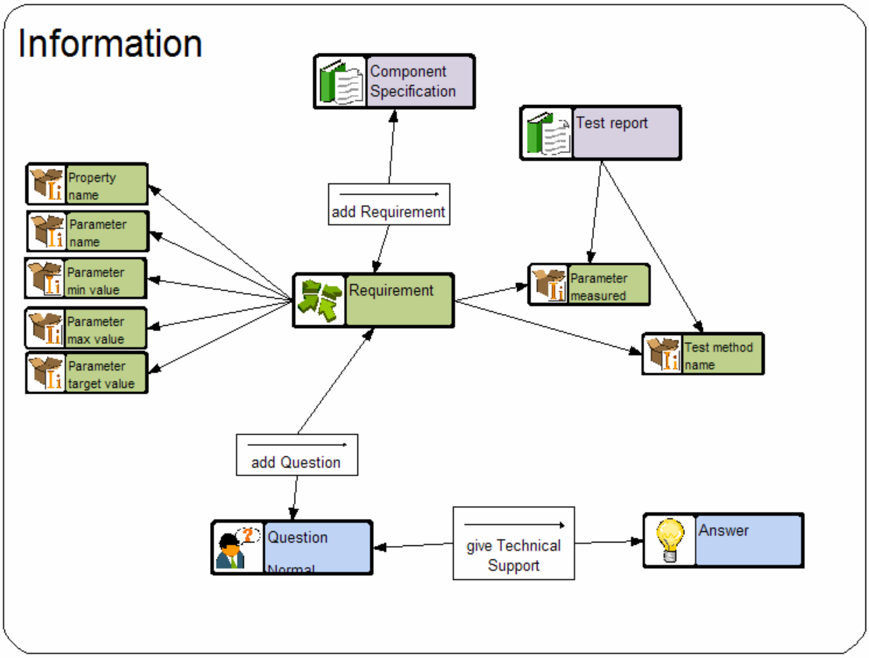

Fig. 2. The initially modeled Information dimension

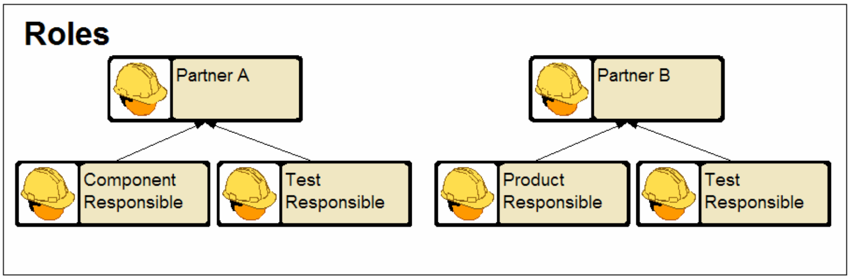

Fig. 3. The initially modeled Role dimension

which contains different "Requirement"-objects. A specific requirement-object contains some "Property"-objects such as "Property name", "Parameter name", "Parameter min value", etc. The information objects captured in the Information dimension relate to roles captured in the Role dimension through tasks modeled in the Task dimension. A specific task is implemented by a view in the View-dimension and defines how the involved information objects should be used by the task. The task and view dimensions from our use case are shown in figs. 4 and 5. The modeled tasks are those that concern the specific collaboration element and the involved roles.

Modeling the four different dimensions give other perspectives on how to operationalise the collaboration elements. In order to model the workplace for specific stakeholders at partner A and partner B, we are forced to think about questions like:

What are the main objectives with their work? Which information objects are they interested in modeling? What tasks are they interested in using? How can they use a modeling tool for performing their work? Etc. This will automatically involve the product that the stakeholder is working with, since this is a primary concern for the stakeholders.

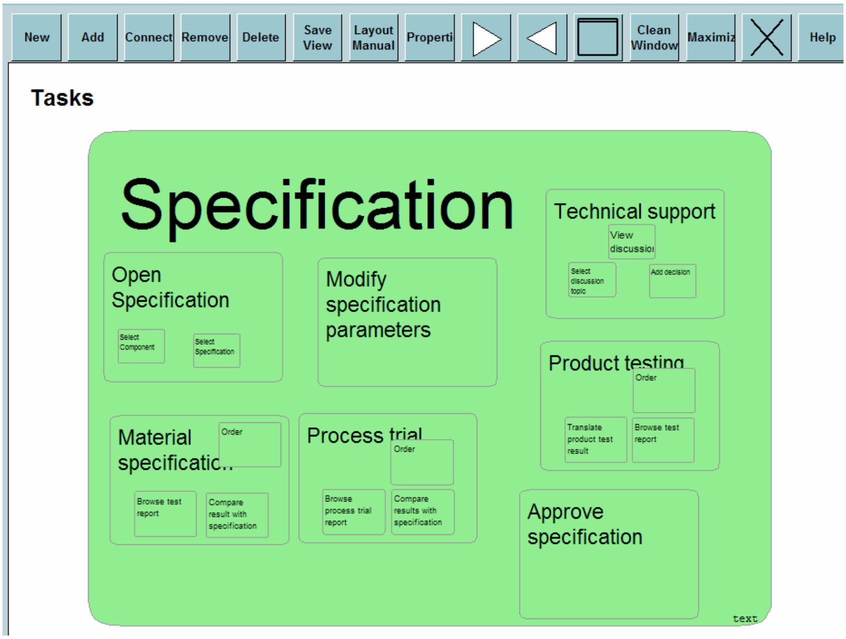

Fig. 4. Showing the modeled Task dimension

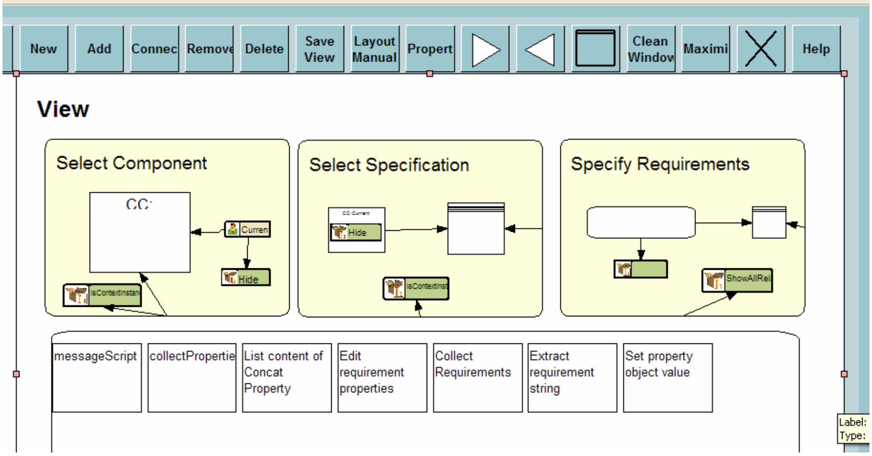

Fig. 5. Showing the modeled View dimension. The seven boxes in the lower part of the figure contains the added scripts.

The *component responsible* at partner A manages the component specification, a document which consists of several material specification documents. Document should not be taken in a literal sense since some documents, in the use case, will exist as models. It is actually a discussion point as such what information should exist as documents (in a literal sense) and which information should exist only as models. We will not take that discussion further here, albeit it is an important interoperability issue how to make document bound information available in the IT-systems. A specific material specification document contains several requirements which can be added by both the *component responsible* and the *test responsible* at partner A as well as by the *product responsible* and the *test responsible* at partner B. This requires that the modeled workplace can be used from both partner A and partner B. An interesting issue is the ownership of the information. Some requirements will be partly added by both partners. It is also a matter of the degree of maturity the documents have reached. In order to add requirements it is necessary for the *test responsible* at both partner A and partner B to perform tests on materials. These tests will add measured values and will require a specific test-method. The test method can be an in-house test-method created at partner A or partner B or it can be required from available sources. Sometimes questions arise when partner A performs tests and it is then necessary to pose these questions to partner B, who is responsible for delivering the material that is tested. These questions can have to do with the material as such or with the test-method used.

Several points of collaboration have been indicated: Entering of requirements from both partners; Investigation of available test-methods required from the other partner; The need to ask technical questions that arises during the testing. Specific tasks were modeled in order to implement these interactions as a workplace to be used by both partners. Such a task is e.g. to open and select a component and a specific requirement. The modeled tasks that are shown in fig. 4 become selectable choices in a menu in the workplace (see fig. 6). In the use case we have model-generated views (see fig. 5) that correspond to the tasks. For the task “Select component” it will be possible to select the component from a list of available components and for the task “Select specification” it is possible to select the specific requirement from the list of requirements belonging to the selected component. The task “Modify specification parameters” opens a view with possibility to add or modify components and requirements for the selected requirement.

By using the workplace information that belongs to the requirements for a specific product will be modeled. The models generated this way will be *active* in the sense we described earlier, they are available for the user and is actively updated and used.

Unfortunately, due to limits in the AKM-platform regarding work place generation, not all generation tasks could be done solely by modeling of views. It became necessary to add scripts to perform some specific operations. This requires that the programmer become acquainted with the metis-programming API. The modeled workplace is a prototype which in its present state is incomplete and need to be further developed in order to be put into operation.

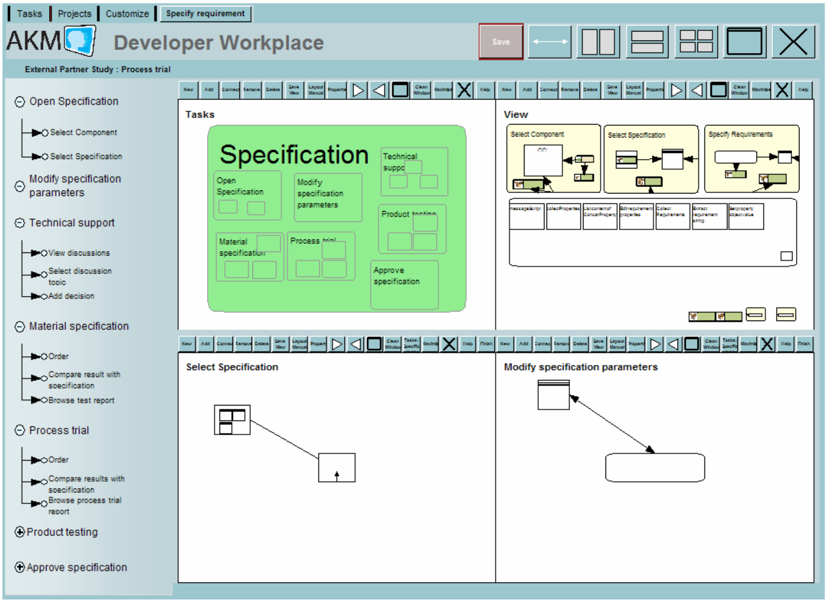

Fig. 6. The user workplace as it looks when you run it inside the workplace for generating workplaces. On the left side you can see the menu from which the user can select the tasks to perform. This menu is generated from the tasks modeled within the main task “Specification”. When the user selects a task a window with the specific view is opened, much like the view shown in figs. 4 and 5. By using the small buttons “New”, ”Add” etc. (barely visible in this figure, but visible in figs. 4 and 5) the user can modify the model.

4 Discussion and Related Work

Different approaches that are relevant for the use case can be mentioned, such as Service Oriented Architectures (SOA), Model Driven Architecture (MDA) and Semantic Web. The following citation summarizes these approaches in a nutshell: “Though many of these approaches show promising results, they do not represent a holistic approach for capturing and nurturing enterprise knowledge” [6, p. 164]. It might however be seen as a paradox that collaboration between different approaches is necessary for solving interoperability problems. Process modeling is used for specifying how things should be performed, i.e. a procedural expression of actions. This can be put into contrast to business models that express what should be done. In process modeling as such it is many times difficult to see the logical reasons behind a specific design [12]. P. Johannesson proposes in the paper [12] that a combination of business and process modeling is fruitful since this places the process models on a declarative ground. Likewise in enterprise modeling it is important to perform goal modeling and to combine this with other types of models like process models.

Web services poses interesting aspects such as ways to combine different web services in order to create electronic business processes. Business Process Execution Language WS-BPEL (or BPEL4WS) and Web Service Choreography Definition

Language (WS-CDL) are evolving standards for designing web services. WS-BPEL can be used for defining business processes that can integrate user- as well as application activities. It does so by invoking already defined web services and by placing a process layer that defines how these web services are invoked. In WS-BPEL processes can be of two types: Executable processes with the purpose to "model the actual behavior of a participant in business interaction"; abstract processes with the purpose to define protocols for message exchange. The abstract processes, which are not executable, serve the function of defining a visible behavior while hiding the actual implementation. The reason for this is to make it possible for participants to not reveal sensitive business processes [13], however collaboration will always require some degree of openness in order to establish and maintain the collaboration.

The Object Management Group (OMG) is specifying a standard Business Process Definition Metamodel (BPDM), with the purpose to achieve integration between different ways of performing business process modeling. They are proposing two different views to be used: *Orchestration* and *Choreography*. Orchestration is about how to internally realize the activities in a web service and choreography is about how to combine collaborating services to achieve a business process [14]. Dijkman and Dumas have proposed four different views: the *choreography*, *orchestration*, *Interface behavior* and *Provider behavior* views. The *Interface behavior* and *Provider behavior* views cover what is necessary to model in order to receive and send messages between the collaborating web services [15]. Much of the described methodologies for modeling business processes are however perceived by the users as too technical and difficult to use in the initial modeling stages. The link to the goals for achieving collaboration does not become obvious. Alternative modeling strategies have been proposed, such as modeling views for capturing milestones and scenarios, which later are converted to executable models [16].

WS-BPEL and Choreography modeling has many interesting capabilities that can be combined with our approach of the connector view. Our purpose is however not only to create a model for delivering messages between the different stakeholders of the collaboration partners. It is to create active models which are dynamically changeable by the user. Active Knowledge Modeling can serve two purposes in our use case. First of all it can deliver a workplace for the collaboration between several partners. Secondly it can deliver a way of modeling an active connector view, which can be changed according to the collaboration needs. The described use case has concentrated only on the first of these challenges.

In our use case we have investigated the collaboration between two companies that already have some business relationship that they want to improve. It is reasonable that the companies, before they reach this level of maturity, have established contacts with collaborating partners. Although it can not be seen as a process which is done once and for all because the collaborating partners may change. Once the collaborating partner(s) have been found it is necessary to build the ways of how to collaborate. In this level it becomes necessary to meet and discuss matters for collaboration. This is as far as collaboration reaches in many situations today. It could also be possible to reach a higher level of maturity by integrating available information sources, which today are used in manual way even though IT is used on both sides.

5 Conclusions and Future Work

In this paper we have described work in progress for performing the last part, named *Operationalising*, in our methodology for developing the connector view. This work has given us some insights about the modeling work which the methodology requires.

In our work, we experienced it as useful to think in different levels for the use of the connector viewpoint:

Level 1: The companies try to describe how they want to start their collaboration. This includes finding a suitable collaboration partner and the initial steps to take.

Level 2: The companies want to develop and improve an already existing business relationship.

Level 3: The companies want to model their collaboration and implement it as executable models that are able to integrate information and software systems.

Most of the work we have presented in this paper can be considered to be on level 2. There is a distinction between how to proceed with the modeling and what becomes important to model on the different levels. On level 1 the companies need to establish their interface which declares and defines their need of collaboration. There is no need to reveal any detailed information about business interaction or products. On level 2 it becomes necessary to model the collaboration and information objects more thoroughly and describe the scenarios for how to perform the collaboration. On level 3 it is a necessity that the connector view models, in detail, all interfaces between applications which communicate information. All the stages in the connector view methodology are applicable on all three levels, although Operationalising is a more challenging task on level 3.

In our work with the connector view we have shown that there is no strict procedural order between the tasks which are part of the collaboration, either it is an activity that generates information or it is an act that directly triggers an act of collaboration. By this insight it is easy to understand statements such as “it is often difficult to understand the reasons behind a certain process design and what consequences alternative designs would have” [12 p. 131]. No doubt it is a major step to grasp the logic that is necessary for choreography modeling which assumes the existence of clearly defined process steps in order to combine the different web services. We believe that web service technologies such as WS-BPEL and WS-CDL are well suited to carry out the work necessary on what we called “level 3”, i.e. establishing connections between collaborating partner when there exists a clear picture of how the collaborating partner want to act. Before this modeling phase it is necessary to strengthen the analysis and we believe that the connector viewpoint can be helpful in defining the complex web of connections that constitute a collaboration. Using the IRTV-methodology can partly be considered as moving into a modeling on level 3. Our experiences concerning this part of the modeling are:

- Modeling as such had a catalytic effect on collaboration and helped to reach a common understanding between the collaborating partners.
- Modeling executable models is an underestimated task and there are many challenges in future work to manage this successfully.

There are still several open issues that require more work. Our intention is to continue the work we have started with the described use case. Some of these open issues to work further with are:

- How to complete the definition and implementation of the operationalisation stage in the connector viewpoint.
- How to extend the connector view to several collaborating partners,
- How to manage exchange of one partner with another partner,
- How to establish the connection with a suitable partner on level 1,
- How to manage the complexity where a more detailed connector view on level 3 is developed,
- How to reuse existing models between the execution environments at the different partners
- How to deal with information that exist as documents (in a literal sense) versus information that exist as models,
- How to deal with the ownership of the information.

References

1. Johnsen, S., et al.: Model-based adaptive Product and Process Engineering. In: Rabe, M., Mihok, P. (eds.) Ambient Intelligence Technologies for the Product Lifecycle: Results and Perspectives from European Research. Fraunhofer IRB Verlag (2007)
2. Lillehagen, F., Karlsen, D.: Visual Extended Enterprise Engineering embedding Knowledge Management, Systems Engineering and Work Execution. In: IEMC 1999 - IFIP International Enterprise Modeling Conference, Verdal, Norway (1999)
3. Krogstie, J., Jørgensen, H.D.: Interactive Models for Supporting Networked Organizations. In: Persson, A., Stirna, J. (eds.) CAiSE 2004, LNCS, vol. 3084, pp. 550–563. Springer, Heidelberg (2004)
4. Jørgensen, H.D., Krogstie, J.: Active Models for Dynamic Networked Organisations. In: Dittrich, K.R., Geppert, A., Norrie, M.C. (eds.) CAiSE 2001, LNCS, vol. 2068. Springer, Heidelberg (2001)
5. Lillehagen, F.: The Foundations of AKM Technology. In: Proceedings 10th International Conference on Concurrent Engineering (CE) Conference, Madeira, Portugal (2003)
6. Lillehagen, F., Krogstie, J.: Active Knowledge Modeling of Enterprises. Springer, Heidelberg (2008)
7. Krogstie, J., Lillehagen, F., Karlsen, D., Ohren, O., Strømseng, K., Thue Lie, F.: Extended Enterprise Methodology. Deliverable 2 in the EXTERNAL project (2000), `http://research.dnv.com/external/deliverables.html`
8. Stirna, J., Persson, A., Sandkuhl, K.: Participative Enterprise Modeling: Experiences and Recommendations. In: Krogstie, J., Opdahl, A.L., Sindre, G. (eds.) CAiSE 2007 and WES 2007. LNCS, vol. 4495, pp. 546–560. Springer, Heidelberg (2007)
9. Carstensen, A., Sandkuhl, K., Holmberg, L.: Towards a Methodology for Modeling Interoperability Between Collaborating Enterprises. In: Proceedings from: 10th International Conference on Enterprise Information Systems, Barcelona (2008)
10. ISO/IEC, IEEE Std 1471-2000: Recommended practice for Architectural Description of Software Intensive Systems. ISO/IEC 42010 IEEE:24 (2007)
11. Putman, J.R.: Architecting with RM-ODP. Prentice Hall, Upper Saddle River (2001)

12. Johannesson, P.: The Role of Business Models in Enterprise Modelling. In: Krogstie, J., Opdahl, A.L., Brinkkemper, S. (eds.) Conceptual Modelling in Information Systems Engineering, pp. 123–140. Springer, Heidelberg (2007)
13. OASIS: Web Services Business Process Execution Language version 2.0, Public Review Draft, August 23 (2006), `http://docs.oasis-open.org/wsbpel/2.0/`
14. OMG, Business Process Definition Metamodel (BPDM) Beta 1, OMG Document Number: dtc/07-07-01 (2007), `http://www.omg.org/cgi-bin/doc?dtc/2007-07-01`
15. Dijkman, R., Dumas, M.: Service-Oriented Design: A Multi-Viewpoint Approach. International Journal of Cooperative Information Systems 13(4), 337–368 (2004)
16. Barros, A., Decker, G., Dumas, M.: Multi-staged and Multi-viewpoint Service Choreography Modelling. Technical Report 4668, Queensland University of Technology, Brisbane, Australia (2006)
17. Linthicum, D.S.: Next generation Application Integration: From Simple Information to Web Services (2003) ISBN 0-201-84456-7
18. Vernadat, F.: Enterprise Modelling and Integration: principles and applications. Chapman & Hall, London (1996)