IT SUPPORT FOR THE CREATION AND VALIDATION OF REQUIREMENTS SPECIFICATIONS – WITH A CASE STUDY FOR ENERGY EFFICIENCY

Thomas Reichel, Gudula Rünger, Daniel Steger, Haibin Xu
Chemnitz University of Technology, Germany

ABSTRACT
Complete, clear, and valid requirements specifications are the foundation for a successful product development. For technical products, these specifications have to take into account several factors, including customer needs, the market, governmental regulations, international standards, manufacturer-specific policies, and, lately, the environmental impact of products. To cope properly with all these factors IT support is highly recommended. This article proposes a workflow that supports the creation and validation of requirements specifications as well as the collaboration between domain experts. The workflow utilizes existing approaches of requirements management in the engineering domain and can be used for IT based validation of completeness, integrity, and consistency of requirements specifications. A reference architecture for an IT solution implementing the workflow and the corresponding IT support is proposed. In a case study, the workflow is applied to customer needs of an energy-efficient milling machine.

Keywords: Requirements specification, checklist, workflow, clarification, energy efficiency

1 INTRODUCTION
The creation of clear and valid product requirements is one of the first steps in a product development process. Product requirements should satisfactorily represent the customer needs as well as other factors that have an impact on the product. These factors can include the market, suppliers, governmental regulations, international standards, and manufacturer-specific policies [1]. Lately, the environmental impact as well as energy and resource efficiency also play an import role for new technical products and have to be considered in the product requirements [2].
The transformation of customer needs and other factors into well-defined requirements is denoted as clarification [1]. This transformation involves a specification and a refinement of customer needs that may be vague and ambiguous. The requirements resulting can be compiled into a so-called requirements specification, which describes all necessary physical parameters, constraints, and properties of a product. A requirements specification enables the manufacturer to create a conceptual design of the product and forms the basis of the business relation between the customer and the manufacturer. The specification can be revised by the manufacturer at any time in the product development process, i.e., before the actual manufacturing. Existing needs or requirements have to be revised, for example, if a competing product with similar functionalities is put on the market or new technologies arise. The development of technical products, and hence the clarification, is a collaborative task of experts of different domains, such as mechanical engineering, electrical engineering, legal department, controlling, and marketing. This collaborative development is denoted as integrated product development [3].
During the clarification, domain experts handle requirements related to their own domain and they form interdisciplinary groups to clarify requirements that are related to multiple domains. For instance, the requirement “increase energy efficiency” of a machine tool concerns the mechanical domain (e.g., working principles) and the electrical domain (e.g., electric control), as well as regulations (e.g., material compliance, ISO/TR 14062[1][4]) and the controlling domain (e.g., cost of mechanical or electrical modifications).

[1] ISO/TR 14062: describes concepts and current practices relating to the integration of environmental aspects into product design and development.

Since a requirements specification is the basis for further development steps, errors in the customer needs and the specification should be detected during the clarification to avoid expensive changes afterwards. Therefore, a validation of the requirements is necessary [5]. In this article, the following validation criteria for requirements specifications are discussed:

- *Completeness:* A specification should contain all necessary requirements to characterize an entire product to be developed. Such a specification contains requirements for all life cycle phases of the product. However, completeness does not imply that a specification is fixed after the clarification or contains all technical details to actually manufacture a product.
- *Integrity:* A specification should represent all customer needs in a reasonable manner. Integrity ensures that the customer and the manufacturer have the same understanding of the product to be developed.
- *Consistency:* A specification is consistent if no conflicts between individual requirements exist. If customer needs have to be clarified by multiple domain experts, consistency ensures that the specification does not contain contradictions between resulting requirements.

These criteria are necessary, since meeting all requirements, complying with the customer needs, and avoiding contradictory requirements are mandatory for the development of a product. However, these criteria alone are not sufficient. Wilson et al. [5], for example, propose further criteria, such as correctness and traceability, which should also be considered during the clarification.

To validate completeness, integrity, and consistency of a requirements specification, IT support is helpful for the development of complex technical products in a short release cycle. In this article, a workflow for the creation and validation of a requirements specification for a complex product is proposed, which supports the collaboration between domain experts involved. The workflow structures the clarification and coordinates domain experts so that they can handle a large number of customer needs. Since this workflow can also be utilized to change an existing specification, it also supports the change management of requirements. The workflow can be executed by a standardized workflow management system, which is part of the reference architecture proposed in this article. The reference architecture manages the data of customer needs and requirements, supports the validation of completeness, integrity, and consistency of requirements specifications, and provides different views on customer needs and requirements for specific domain experts.

The next sections are organized as follows: Section 2 describes fundamentals of the clarification in the engineering domain and discusses limitations of existing approaches. Related work is discussed in Sect. 3. Section 4 specifies the workflow for the clarification in detail and proposes a reference architecture for the corresponding IT support. Section 5 presents an example clarification of customer needs for a milling machine. Finally, Sect. 6 concludes.

2 CLARIFICATION OF REQUIREMENTS IN THE ENGINEERING DOMAIN

A clarification is usually initiated after an informal description of the customer needs, e.g., a statement of work, is available. This description may originate from direct orders of customers or from requests of the manufacturer's planning department [1]. The description of customer needs contains important technical aspects and major use cases of a product to be developed. However, the description often includes incomplete or vague statements and may even contain contradictory demands or wishes. Therefore, the clarification has to refine and to complete these customer needs in order to create a specification that is the basis for subsequent development steps, such as the conceptual design and the embodiment design of a product [1]. In this section, the following fundamentals of the clarification for technical products are presented: the integrated product development (IPE), the validation of the completeness of specifications by checklists, the validation of the integrity of specifications, the usage of workflows in the engineering domain, and selected methods of the quality management.

Ehrlenspiel [3] proposes the approach of an IPE that incorporates all stakeholders of the product development, such as different company departments, customers, and suppliers, into a collaborative development process. In this process, design decisions are made by all stakeholders concerned and, thus, inconsistent design decisions should be avoided. The workflow proposed in this article represents an implementation of the IPE approach for the process of requirements clarification.

In the engineering domain, checklists are used as working aids that identify all relevant aspects of specific issues. To support the validation of the completeness of a requirements specification, example checklists are provided by [1] and [3]. These checklists contain general aspects of a technical product and its development in a hierarchical or tabular form, as shown in Figure 1. A specification is

Figure 1. Checklists for requirements specifications in the engineering domain (extracts). Left: Hierarchical checklist by Ehrlenspiel [3]. Right: Tabular checklist by Pahl/Beitz [1].

considered to be complete, if it properly considers all items of such a checklist. However, this kind of validation of completeness requires a checklist that contains all mandatory and optional aspects of the product to be developed. To increase the applicability and significance these example checklists should be adapted for a specific product type, e.g., machine tools, see [1].

A manual process to validate the completeness of a specification is time consuming, because for each requirement it has to be decided, whether it belongs to a checklist item or not. Using an IT system can reduce the efforts necessary to handle a large number of requirements and checklist items. Besides the validation of completeness, a validation of the integrity of a specification is necessary during the clarification. To validate the integrity, the customer has to decide whether a specification properly represents the demands and wishes concerning the product to be developed. Therefore, the customer has to be involved in the requirements clarification. An IT system can support this collaboration by workflows.

Since the creation and validation of requirements specifications is an iterative process that requires collaboration [1], workflows can be used to support the clarification. A change process of a document is one example process that is implemented as a workflow in existing IT systems for product development. During the change process, modifications of a document by employees require the management to review and to approve or to reject the modifications. Workflows are controlled by a workflow management system that provides a user interface (e.g., a web interface), which lists all tasks assigned to a user (e.g., to review or to change a document). For a document-related task, the user is able to upload or download documents and mark documents as changed. The reference architecture proposed in Sect. 4.6 also utilizes such a workflow management system.

Methods that preserve the quality of a product in the entire product life cycle should also be applied in product development. Examples are the value analysis, the conjoint analysis, the failure mode and effects analysis (FMEA), and quality function deployment (QFD). These methods may be utilized as part of the workflow for the creation and validation of a requirements specification. This utilization is not considered in this article.

3 RELATED WORK

Methodologies for the product development process, such as the axiomatic design [6], and Gero's FBS model [7], use clear and valid requirements specifications as the starting point of the development of products. Both methodologies describe necessary properties of the specifications, but do not explain how they are created and validated in a collaborative environment. In contrast, this article proposes an implementation for the iterative creation and validation of specifications, including the corresponding collaboraton of domain experts. A checklist for the validation of completeness of specifications for the axiomatic design approach has been proposed in [8]. However, additional criteria, such as integrity and consistency, are not considered.

Since the management of customer needs and requirements in the IT domain is similar to the engineering domain, approaches for the clarification in the IT domain can be adapted for the engineering domain. The Rational Unified Process [9] (RUP) provides procedure models for the development of complex software systems. The RUP distinguishes between functional (i.e., technical) and nonfunctional (e.g., governmental regulations) requirements, which are gathered independently. Both types of requirements are unified in a so-called Software Requirements Specification. Similar to the roles proposed in this article, the roles *project manager* (System Analyst), *domain expert*

(Requirement Specifier), and *customer relation* (Requirements Reviewer) are involved in the requirements analysis of RUP.
Another approach is the HOOD Capability Model [10], which proposes procedure models for the requirements management of complex technical systems. The HOOD Model emphasizes the management of relations between customer needs and requirements for the maintenance, the reuse, and the further development of products.
Quality function deployment (QFD) [11] provides a methodology to trace requirement priorities, relationships between customer needs and requirements, and requirement conflicts during the product development process. QFD supports the integration of different company departments and a separation between requirements and the corresponding product solution. However, QFD neither specifies a methodology to transform customer needs into requirements nor does it support the validation of completeness and integrity of a specification [1,3].
Alexander [12] proposes a taxonomy to identify stakeholders of a product development process. This taxonomy is graphically represented by an onion diagram with concentric circles that represent different spheres of influence on the product, such as the facility and the organization. This taxonomy can be utilized to identify possible stakeholders, which can be represented by the roles in the clarification workflow proposed.
Apart from procedure models and methods, there exist IT systems to support the requirements management process. Wnuk et al. specify essential aspects for an implementation of such an IT system [13]. Rational DOORS Web Access[2] is a web-based system that implements the requirements management of the RUP. Similar to the approach proposed in this article, DOORS uses workflows to change requirements specifications, but does not organize the entire clarification as workflow. Since DOORS focuses on the development of software systems, the utilization of checklists, as proposed in this article, is not supported. A validation of requirements specifications is done by software test cases. IT support for the validation of completeness and consistency during clarification is missing.
To integrate requirements management with the product data management of technical products, Windchill RequirementsLink[3] supports the linkage of requirements with product structures to trace requirements. RequirementsLink provides a collaborative requirements management system, however, a validation of the criteria as proposed in this article is not supported.

4 WORKFLOW APPROACH FOR THE CREATION AND VALIDATION OF A REQUIREMENTS SPECIFICATION

The workflow to create a requirements specification is usually based on informal approaches, for example proposed by [1,3,6], that emphasize the collaboration between a project manager and engineers. However, IT support for a validation of a specification are not part of these approaches. In this section, a workflow is proposed that manages the collaboration between all stakeholders of the clarification and supports the validation of correctness, integrity, and consistency in a software system.

4.1 Workflow for the creation and validation of a requirements specification

The starting point of the workflow is a catalogue of customer needs, e.g., a statement of work. In three phases, the customer needs are incrementally transformed into a requirements specification. In the first phase (*creation*), the customer needs are clarified by design engineers with different expertise. The engineers refine and complete the customer needs and create a draft of the requirements specification. Customer needs and resulting requirements are linked to related checklist items. This is used for the validation of the specification draft in the second phase (*validation*). If the draft cannot meet the validation criteria, the specification is revised by the engineers. The process of (re-)creation and validation is repeated until the specification complies with all validation criteria. The requirements specification is released in the last phase (*finalization*).
The stakeholders (also called roles) of the workflow are depicted at the left side of Figure 2 and Figure 3. The role *customer relation* represents a contact person that can clarify all questions about the customer needs. The *project manager* corresponds to the engineer-in-chief who is responsible for the requirements clarification. Several *domain experts* join the workflow to clarify customer needs in their domain. The role *domain expert* may correspond to a single person, but can also stand for a group of

[2] http://www-01.ibm.com/software/awdtools/doors/webaccess/
[3] http://www.ptc.com/products/windchill/requirementslink

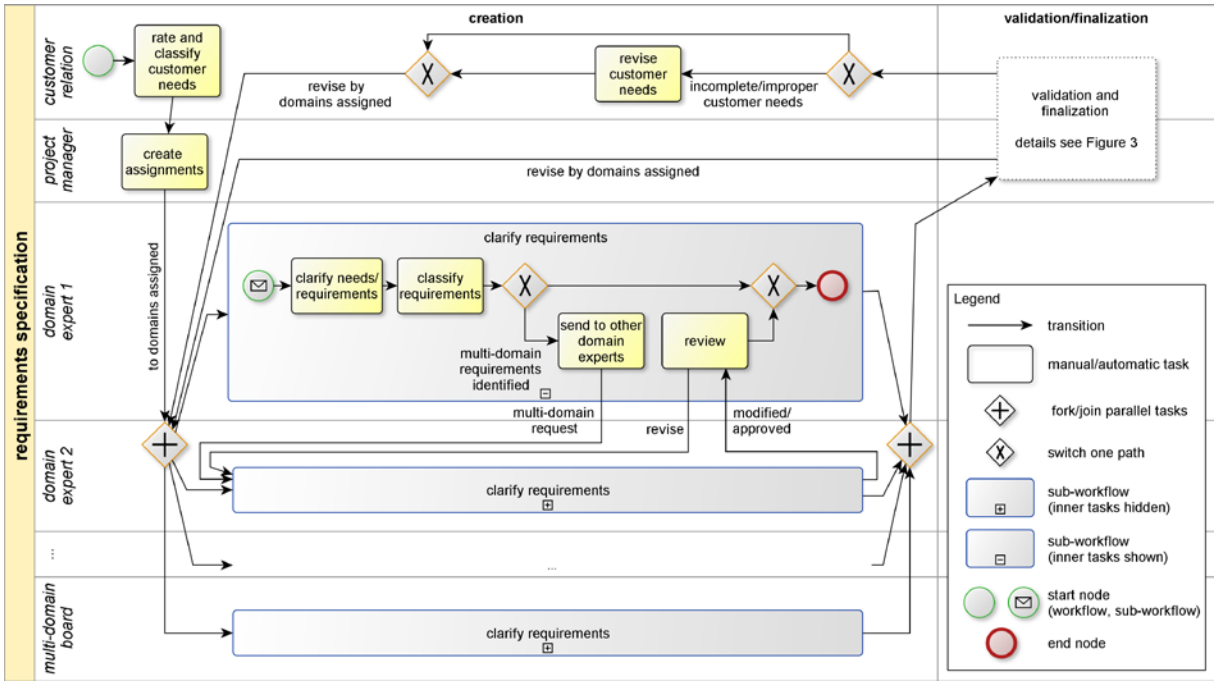

Figure 2. Workflow for the creation and validation of a requirements specification. The creation phase is depicted in detail.

engineers of the same domain. A group of experts of different domains can form a *multi-domain board*. This board is responsible for clarifying multi-domain requirements. The workflow supports the assignment of tasks to appropriate stakeholders to avoid broadcast communication. A stakeholder is informed or asked to perform a task only if that stakeholder is affected.

The IT system stores the relations between customer needs, requirements, checklist items, and stakeholders of the workflow in a data model. The data model is stored in the relational database PostgreSQL[4]. In the validation phase, these relations can be used to evaluate the validation criteria. For example, the relations between customer needs and resulting requirements have to be examined to validate the integrity of a specification. In the next subsections, the phases of the workflow and the corresponding IT support in these phases are described in more detail.

4.2 Creation phase of a requirements specification

The creation phase of the workflow starts with the rating and the classification of the customer needs by the stakeholder with the role *customer relation* (task *rate and classify customer needs* in Figure 2). In this task, the needs are distinguished into customer demands and wishes. The rating can be utilized to prioritize related requirements, which is not discussed in this article. In the task *rate and classify customer needs* the stakeholder with the role *customer relation* classifies the customer needs by linking them to appropriate checklist items. For example, the feed speed required for a milling machine is linked to an item “technical requirement/operation”. The classification of customer needs is also used to separate the needs into domains, such as technical, economic, organizational, and juridical domain. This separation is exploited in the next task *create assignments* of the *project manager*, who assigns the customer needs to *domain experts*. For example, all customer needs linked to the checklist item “technical requirement/operation” can be assigned to experts of the technical domain, e.g., design engineers. A customer need has to be assigned to a single *domain expert*, if the need refers to a single domain. However, a customer need may refer to multiple domains, for example the need “increase energy efficiency” refers to a technical as well as an economic domain. Multiple domain needs may indicate a conflict of interests, such as the energy consumption of a product and its manufacturing costs in the need “increase energy efficiency”. Customer needs that refer to multiple domains have to be assigned to a *multi-domain board*. In this board, the *domain experts* involved should discuss and clarify this need together.

[4] http://www.postgresql.org/

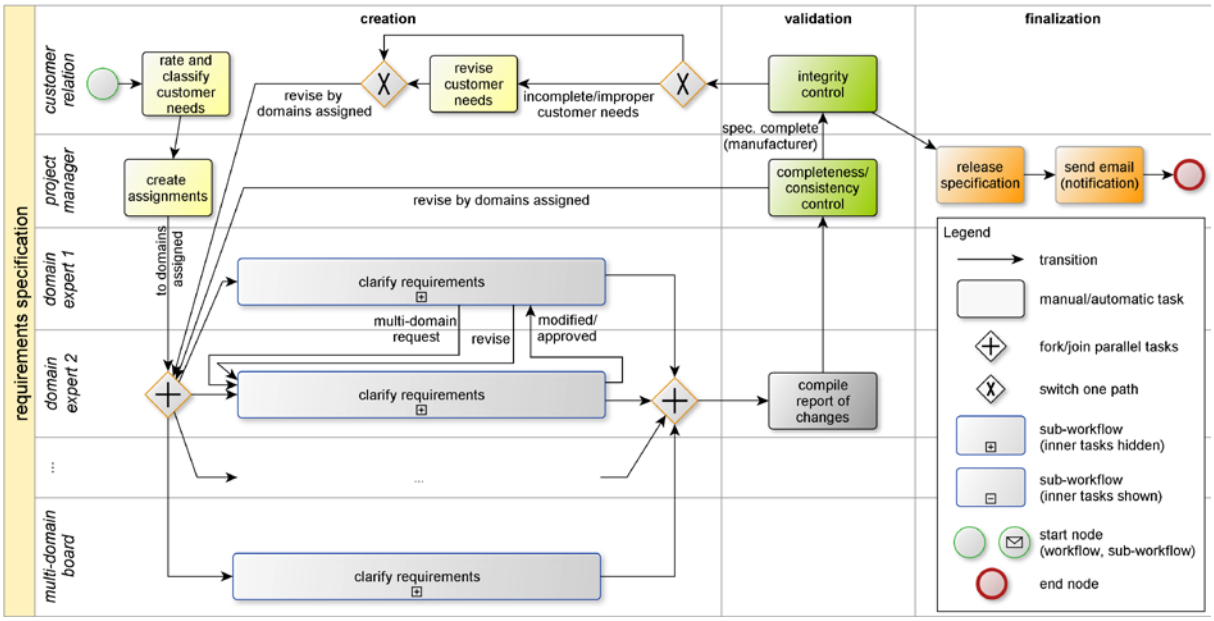

Figure 3. Workflow for the creation and validation of a requirements specification. The validation and finalization phases are depicted in detail.

When all customer needs are assigned to *domain experts* and/or *multi-domain boards*, all experts start to clarify the needs assigned (sub-workflow *clarify requirements*). The sub-workflows clarifying the needs assigned are executed concurrently. The first task of the sub-workflow is to refine and complete the needs assigned (*clarify needs/requirements*). This is a manual task performed by *domain experts* that may incorporate the stakeholder with the role *customer relation* to examine the actual demand of the customer. In this case, the experts and the customer jointly clarify the needs assigned. A customer need results in a single requirement or in multiple requirements and multiple similar customer needs may result in a single requirement. Since the clarification task is customized for each need, the IT support for this task can deliver existing requirements that are linked to the same checklist items as the customer need. These existing requirements may be reused in the current specification. In the next task, *classify requirements*, the *domain expert* creates links between the requirements and related checklist items. This task is similar to the first task of the entire workflow (*rate and classify customer needs*). The links between requirements and checklist items are stored in the data model. They are utilized later to validate the completeness of the requirements specification.

During the clarification sub-workflow the *domain expert* (or the *multi-domain board*) has the option to consult other experts by a so-called *multi-domain request*. This request starts new sub-workflows (*clarify requirements*) assigned to the *domain experts* consulted. The experts consulted may change or enhance the requirements assigned (*clarify needs/requirements*) and alter checklist items (*classify requirements*) linked to the requirements. After the *domain experts* consulted have finished their sub-workflows assigned, the requirements modified are returned to the original expert. This original *domain expert* reviews the requirements modified and may reject (*revise* the requirements by the expert consulted) or accept the changes (finish the sub-workflow). Each request and reply of a consultation can include a text message that specifies the changes required or the modifications realized. In addition to this message, other kinds of documents may be attached, e.g., sketches.

4.3 Validation phase of a requirements specification

The validation of the requirements specification starts after all *domain experts* have finished the *clarify requirements* sub-workflows assigned by the *project manager*. The first task of the validation phase is the task *compile report of changes* (see Figure 3). With this task, the IT system supports the *project manager* by automatically generating a so-called “report of changes”, which contains all requirements modified and checklist items that are linked to these requirements. Therefore, the report contains relevant parts of the entire specification that need a review by the *project manager*.

To validate the completeness of the requirements specification, the *project manager* has to examine all checklist items and the related requirements of the report of changes in the task *completeness/*

consistency control. Since the checklist contains all mandatory aspects for a product to be developed, the *project manager* has to decide whether the requirements linked to a checklist item properly represent this item. If an item is properly represented, the *project manager* marks the item as "complete". Otherwise the manager sends the requirements concerned to revision. For each *domain expert* who is assigned to a requirement in revision, a new sub-workflow *clarify needs/requirements* is started automatically after the *project manager* has finished the task *completeness/consistency control*. The specification is complete from the manufacturer's point of view if all checklist items are marked as "complete".

In the task *integrity control*, the stakeholder with the role *customer relation* validates the integrity of the complete specification. To achieve this, the stakeholder examines all requirements modified as given in the report of changes and decides whether a customer need is properly clarified. Since the customer needs and the requirements resulting are stored in the data model, a list of customer needs and the corresponding requirements are provided by the IT system. If a customer need is properly clarified, the stakeholder with the role *customer relation* marks the need as "clarified". It is examined once again, when the requirements of this need are modified. Similarly to the previous task *completeness/consistency control*, the stakeholder with the role *customer relation* may select requirements to be revised by *domain experts*. In addition to this task, a customer need can be modified if the stakeholder with the role *customer relation* identifies an incorrect need (task *revise customer need*). However, this task is optional. When all customer needs are marked as "clarified", the *finalization* phase of the specification starts.

The consistency of the requirements specification is implicitly validated by the *multi-domain boards* and the consultation of additional *domain experts* via *multi-domain requests*. Since a customer need that refers to multiple domains is clarified by appropriate *domain experts*, a contradicting specification can be avoided. The consistency depends on the detection of all domains referred to by a customer need. In addition to the validation by *multi-domain boards*, the *project manager* has to check the consistency of requirements that are not clarified by a *multi-domain board* (task *completeness/ consistency control*). Contradicting requirements can be detected by an examination of all requirements linked to a specific checklist item.

4.4 Finalization phase of a requirements specification

In the last phase (*finalization*), the specification is released and can be utilized for subsequent product development steps, such as the conceptual design and the embodiment design. The specification is issued to the stakeholders of the workflow as well as the management of the customer and the manufacturer (*notification*). This task is supported by the automatic generation of the specification document based on the stored requirements, ratings, etc.

4.5 Using the workflow as a change process

Besides the creation and validation of a requirements specification, the change management is essential to preserve the quality of a specification and, hence, the quality of the product. Changes to a specification may occur in each product life cycle phase, for example the refinement during the embodiment design or the correction of a detected fault in operation. The clarification workflow introduced in Sect. 4.1 can be adapted for an implementation of a change workflow that preserves completeness, integrity, and consistency of a specification modified.

The change workflow starts with the task *completeness/consistency control* (see Figure 3). In this task, the *project manager* selects all requirements that should be modified. Based on these requirements, new *clarify needs/requirements* sub-workflows are started and appropriate tasks are assigned to *domain experts*. The experts that revise the requirements may be exchanged by the *project manager*. The validation and finalization phases in the change workflow are the same as in the clarification workflow.

For the development of new products, a requirement of an existing specification can be reused. This may include checklist items linked and *domain experts* that revised this requirement earlier. Additionally, copying a specific requirement to a new specification creates a link to the existing requirement. If the existing requirement is changed, this link can be utilized to propagate the modification to the requirement copied.

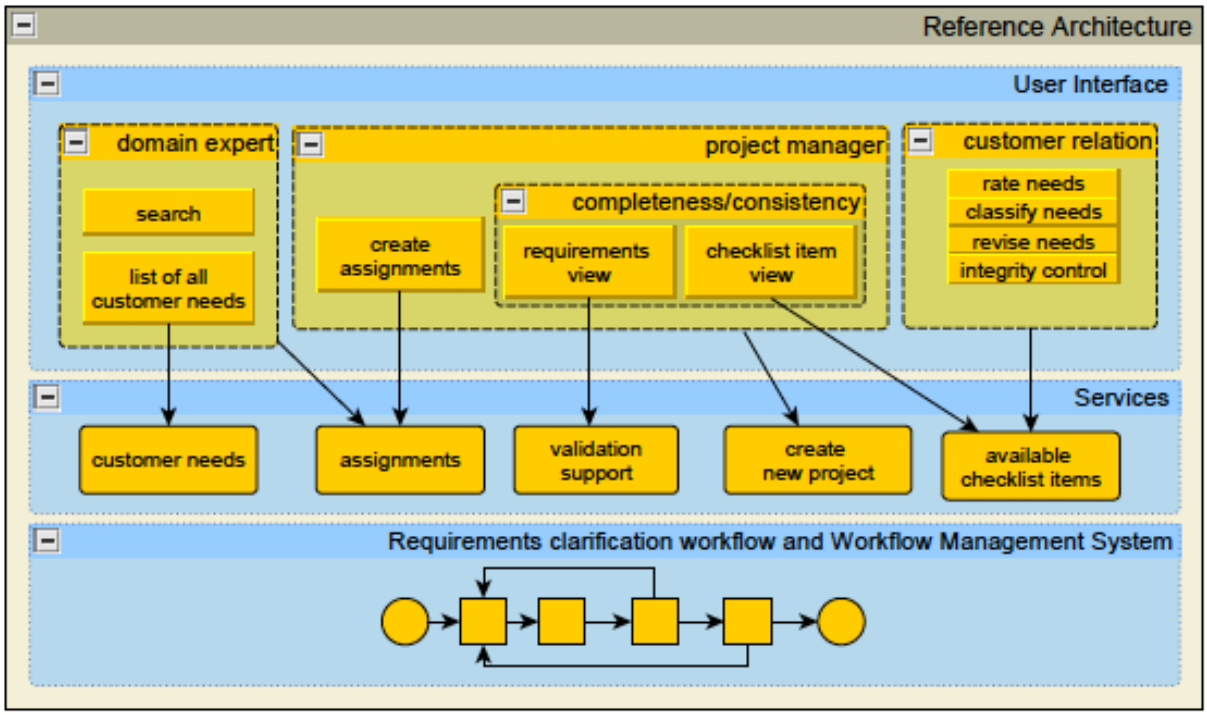

Figure 4. Reference architecture for an implementation of the clarification workflow and the IT support proposed based on an enterprise application server architecture.

4.6 Reference architecture

The reference architecture of the clarification workflow and the IT support proposed is based on an enterprise application server architecture, e.g. the JBoss Enterprise Application Server[5]. Such an architecture provides frameworks to develop and to execute web-based applications as well as frameworks for the design and the execution of workflows (e.g., jBPM[6] for JBoss). Figure 4 depicts the reference architecture for an implementation of the workflow and the corresponding IT support. The web-based *User Interface*, located in the top layer, provides different views for each task of the stakeholders in the clarification workflow. One example is the view to assign customer needs to *domain experts* by the *project manager* (*create assignments* in layer *User Interface*). Each functionality provided, e.g., to get the current *list of all customer needs*, is implemented by services located in the services layer (e.g., *customer needs*). The services are linked with the workflow. The workflow is located in the bottom layer. This workflow is executed by a workflow management system that is responsible to control the sequence of tasks specified by the workflow.

5 EXAMPLE REQUIREMENT CREATION AND VALIDATION

In this section, the workflow for the creation and validation of a requirements specification (Sect. 4) is applied to the development of an energy-efficient milling machine. In Figure 5, the customer needs “increase energy efficiency” and “delivery date” (the input of the workflow) are depicted at the left side, while the requirements “reduce energy consumption” and “speedup cutting process” (the output of the workflow) are located at the right side. The checklist items (based on [3]) represent a subset of factors that influence the development of the milling machine. In the following, the terms of Figure 2 as well as of Figure 3 are utilized for the description of the workflow.

The workflow starts with the rating and the classification of the given customer needs by the stakeholder with the role *customer relation*. As proposed by [14], the energy efficiency of a machine tool can be calculated as the ratio of the output of the machine (*output*), e.g., the volume of parts processed, and the amount of energy invested ($energy_{in}$), for example the electrical energy or the energy equivalent of compressed air. The classification of the need “increase energy efficiency” results in two related checklist items: process throughput and ISO/TR 14062 (relations **a)** in Figure 5). Since the need is related to a “guidelines” domain as well as a “technical” domain, the *project*

[5] http://jboss.org
[6] http://jboss.org/jbpm

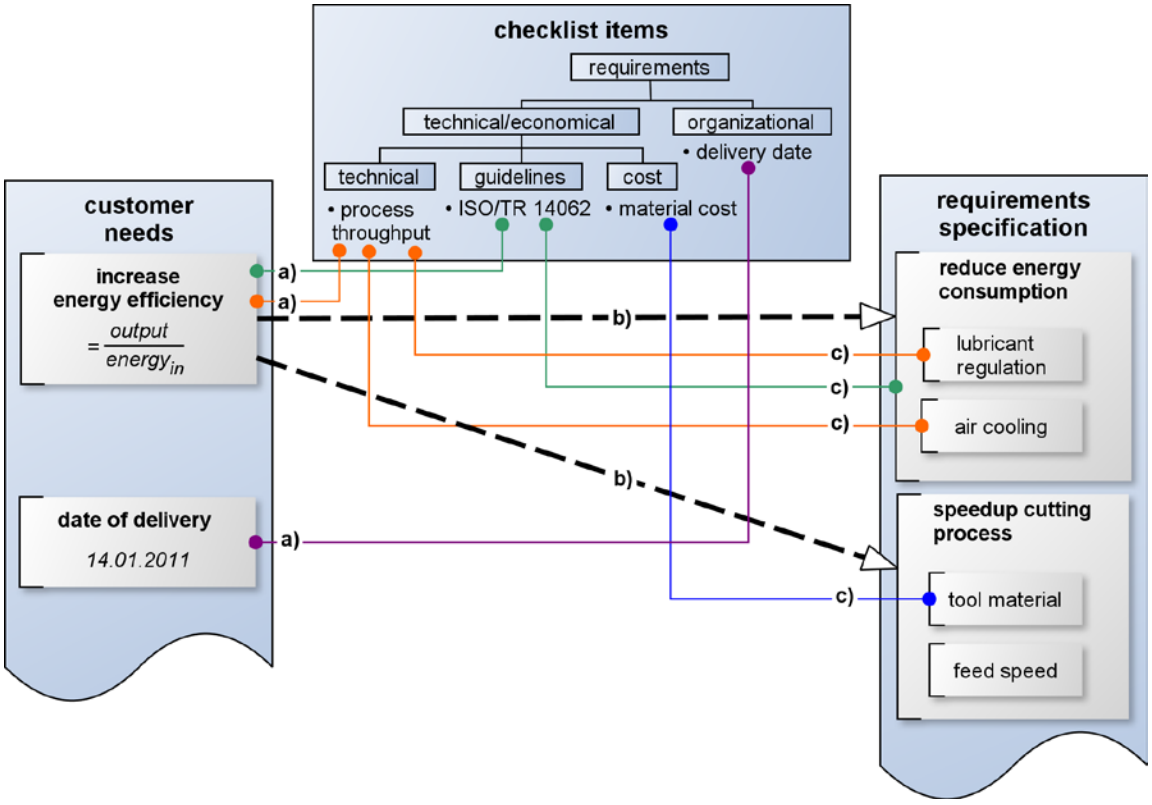

Figure 5. Refinement and assignment of checklist items for the customer needs "increase energy efficiency" and "date of delivery" of a milling machine.

manager assigns this need to two *domain experts* (in task *create assignments* of Figure 2), who have to clarify the need jointly in a *multi-domain board* (MDB).

This assignment starts a new sub-workflow (*clarify requirements*) for the MDB. In the example, the energy efficiency of the milling machine can be improved by two options: Either by an increase of the output of the machine (with unchanged $energy_{in}$) or by a decrease of the energy invested (with unchanged *output*). The MDB chooses both options by creating the requirements "reduce energy consumption" and "speedup cutting process" in the task *clarify needs/requirements*. Details are attached to both requirements. For example, in order to speedup the cutting process, an improvement of the tool material used and an increase of the feed speed is proposed. With this clarification task the MDB generates the relations **b)** between the customer need and the requirements resulting. After the generation of the requirements, the MDB has to link related checklist items to these requirements and/or their details (task *classify requirements*). The relations **c)** are added in this task.

When all customer needs are clarified, the *project manager* has to validate the completeness of the specification (task *completeness/consistency control* of Figure 3) based on the report of changes that initially contains all requirements. For this validation, the *project manager* examines all checklist items and decides whether they are properly represented by the requirements linked. In the example, the *project manager* identifies that the item *delivery date* is not represented by the requirements. Therefore, a revision of this customer need is necessary.

The validation of the integrity is performed after the *project manager* has validated the completeness and the consistency of all requirements. In the task *integrity control*, the stakeholder with the role *customer relation* decides whether all customer needs and requirements fulfill the demands and wishes of the customer.

6 CONCLUSION

This article proposes a workflow and the corresponding IT support for the creation and validation of requirements specifications and outlines a reference architecture providing the corresponding IT support. The approach simplifies the collaboration between domain experts and provides a methodical validation of completeness, integrity, and consistency of requirements specifications from both, the customers and manufacturers point of view. This validation of requirements is necessary in larger

industrial enterprises as well as in small companies and consultancies in order to fulfill constraints in time, cost, and scope for the development of a product. The proposed case study demonstrates the necessity of the validation during the clarification.
Future work focuses on an implementation of the reference architecture proposed in this article and on the integration of the implementation into the IT system proposed in [15]. The integration into this IT system, which supports entire development process, is mandatory to trace requirements in all product life cycle phases.

ACKNOWLEDGEMENT

The Cluster of Excellence "Energy-Efficient Product and Process Innovation in Production Engineering" (eniPROD ©) is funded by the European Union (European Regional Development Fund) and the Free State of Saxony.

REFERENCES

[1] Pahl G., Beitz W., Feldhusen J. and Grote K.-H. *Engineering Design. A Systematic Approach*, 2007 (Springer London).
[2] Niemann J., Tichkiewitch S. and Westkämpfer E. (Eds.) *Design of Sustainable Product Life Cycles*, 2009 (Springer Berlin Heidelberg).
[3] Ehrlenspiel K. *Integrierte Produktentwicklung. Denkabläufe, Methodeneinsatz, Zusammenarbeit*, 2009 (Hanser Fachbuch).
[4] International Organization for Standardisation *ISO/TR 14062:2002 Environmental management – Integrating environmental aspects into product design and development*, 2002.
[5] Wilson W.M., Rosenberg L.H. and Hyatt L.E. Automated Quality Analysis of Natural Language Requirement Specifications. In *14th Annual Pacific Northwest Software Quality Conference, Portland*, 1996, pp. 795-825.
[6] Suh P.S. *Axiomatic Design. Advances and Applications.*, 2001 (Oxford University Press New York).
[7] Gero J.S. Design Prototypes: A Knowledge Representation Schema for Design, *AI Magazine*, 1990, 4(11), pp. 26–36.
[8] *IEEE P1220 Standard for Systems Engineering*, 1994 (IEEE Standards Dept New York).
[9] Kruchten P. *Rational Unified Process: An Introduction*, 2003 (Addison-Wesley Longman, Amsterdam).
[10] Hood C., Wiedemann S., Fichtinger S. and Pautz U. *Requirements Management: Interface Between Requirements Development and all other Engineering Processes*, 2007 (Springer, Berlin).
[11] Akao Y. *Quality Function Deployment: Integrating Customer Requirements into Product Design*, 1990 (Productivity Press Cambridge, MA).
[12] Alexander I.F. A Taxonomy of Stakeholders: Human Roles in System Development. In *International Journal of Technology and Human Interaction*, Vol. 1, 2005.
[13] Wnuk K., Regnell B. and Schrewelius C. Architecting and Coordinating Thousands of Requirements - An Industrial Case Study. In *Proceedings of the 15th International Working Conference on Requirements Engineering: Foundation for Software Quality*, 2009, pp. 118-123.
[14] Neugebauer R., Frieß U., Paetzold J., Wabner M. and Richter M. Approach for the Development of Energy-efficient Machine Tools, *Journal of Machine Engineering*, 2009, 9(2), pp. 51–62.
[15] Rünger G., Götze U., Putz M., Bierer A., Lorenz S., Reichel T., Steger D., Wenzel K. and Xu H. IT Systems Integration for the Development of Energy-Efficient Products. In *Proc. of the 1st International Colloquium of the Cluster of Excellence eniPROD*, 2010, pp. 677-696.

Contact: Thomas Reichel
Chemnitz University of Technology
Department of Computer Science, Professorship Programming and Parallelism
09111 Chemnitz
Germany
Tel: Int +49 371 53136443
Fax: Int +49 371 531836443
Email: thomr@cs.tu-chemnitz.de