5th International Conference on Science, Technology and Management
India International Centre, New Delhi (ICSTM-16)
30th July 2016, www.conferenceworld.in ISBN: 978-93-86171-00-9

THE TOTAL QUALITY MANAGEMENT SYSTEM ENSURING DEVELOPMENT OF SOFTWARE PROCESSES IN BUSINESS SECTOR

Mohd. Faisal khan[1], Dr. Debaprayag Chaudhury[2],

[1]PhD[*], Research Scholar, Deptt. of Computer Science & IT,
E.C. Road, AMET University, Chennai, Tamil Nadu (India)
[2]Approved Ph.D. External Guide, Deptt. Of Computer Science & IT,
E.C. Road, AMET University, Chennai, Tamil Nadu (India)

ABSTRACT

The Software development is a very complex process in which many organizational units, persons, systems and artefacts are involved. In companies that exceed a certain size the business processes become difficult to handle and the quality of the product can decrease. A process-oriented view on software development is increasingly popular witnessed by current publications on the software development process and the application of modern quality management systems. The ISO 9000 norm describes a total quality management system that provides a process oriented view on general production processes and is widely used in many industries. In this research paper we suggest a systematic way of describing and developing software processes that contain certain desirable properties and fulfil quality management demands according to requirements. Therefore, the design pattern approach known from object-oriented software engineering will be adapted to the modelling of business processes. Using this approach, the requirements of the ISO 9000 norm can be translated to software development process on patterns. These patterns can be used in modelling or reengineering the business processes of a software company in a way that the processes fulfil important properties of quality management systems and improve the overall quality of the resulting software product. Additionally, finding quality management patterns in software development processes can indicate the existence of a functioning quality management system to certification authorities and customers.

Keywords: TQM, Software Development, Patterns, Quality.

I INTRODUCTION

The development of software is a business process comparable to business processes in otherIndustries or sectors. The general wish to improve the quality of business processes can not only be found in the "old economy" service and producing industries, also software companies follow this trend by spending reasonable effort in advances in the software development process according to the market requirements. A possible combination of this effort together with the advances in modern object-oriented Software design and modelling can be found in [1]. Many software development companies use different languages to describe their business

processes and workflows in order to improve their efficiency and quality system. These languages have in common that they describe a set of activities connected by a control flow and information about artifacts and responsibilities, yet they are lacking means to model higher level structures and abstractions of concepts like, for example, quality management properties. A small example can illustrate the problems. A business unit produces individually customized software units according to the customer's specification or customer's requirements. As these software units will be delivered directly to a key account customer reaching high quality requirements is essential for the maintaining the sales and popularity. This makes it necessary to include quality control into the business process of producing these software units. This quality control usually consists of regular audits (verifying, validating, monitoring or testing activities according to predefined quality objectives [ISO 9001]) combined with reporting to the quality management department to be able to review and develop the software development process as whole requirements. How can a software development process be designed to include all necessary properties and activities for this kind of quality control? How can these properties and activities be abstracted from the production process itself?

In object-oriented software systems could in contrast to that properties be inherited from a class or an interface Quality Control which defines all necessary methods. With the languages available to model business processes this is not possible. In this work we want to present a way to describe higher-level structures of processes that can be communicated and used in modelling or re-engineering business processes or system and will help to ensure that certain requirements are met by the business process. Furthermore, it will give an idea how it can be validated that necessary properties for a functional quality management system based on ISO 9000 are present in the business processes.

Section 2 discusses business processes in general and ways to describe them. Section 3 gives a brief overview of quality management System. In section 4 it will be shown how patterns can be applied to business processes. Section 5 formulates a template for the description of business process patterns. Finally, section 6 shows different ways of applying a pattern to a concrete business process and how important aspects of quality management are implemented into a business process by applying a quality management pattern using a short example. The paper ends with a conclusion and an outlook on future work.

II BUSINESS PROCESSES

Businesses create on their value to the stakeholder in the value chain, which describes the transformation of goods or information in physical state, place or time performed by different organizational entities of the company which are working in different field [2]. The basic outline of all the processes involved in the value creation has called *business logic*. So business logic denotes the way of the fundamental transformation of goods or service common to a certain industry is performed but it is also individual in the sense that every company or industry is distinct from its competitors, which gives it a competitive advantage or disadvantage in the market and growth [2]. Going more into detail the business logic and the value chain is formed by a number of different business processes for the different fields of operation. A business process could describe operations like the production of a certain goods, the procurement of raw materials, the acceptance of a customer's call to the hotline etc. It consists of *activities* performed by human or automated actors. These activities are the basic

building blocks of the process. In the business process and workflow management literature there exists an established way of handling to the description of business processes and workflows by dividing it into a number of *aspects*. Aspects mean in this context different sets of properties that do not overlap (in other words: are orthogonal to each other) [3, 4].

- The *functional aspect* describes what has to be done to perform an activity.
- The *organizational aspect* describes who or what automated system performs an activity and gives information on the relationships between organizational entities.
- The *informational aspect* describes what information is needed and produced by performing an activity denotes the structure and flow of data.
- The *behavioural aspect* describes the order, in which activities are performed and the control structures that connect them to form a whole process.

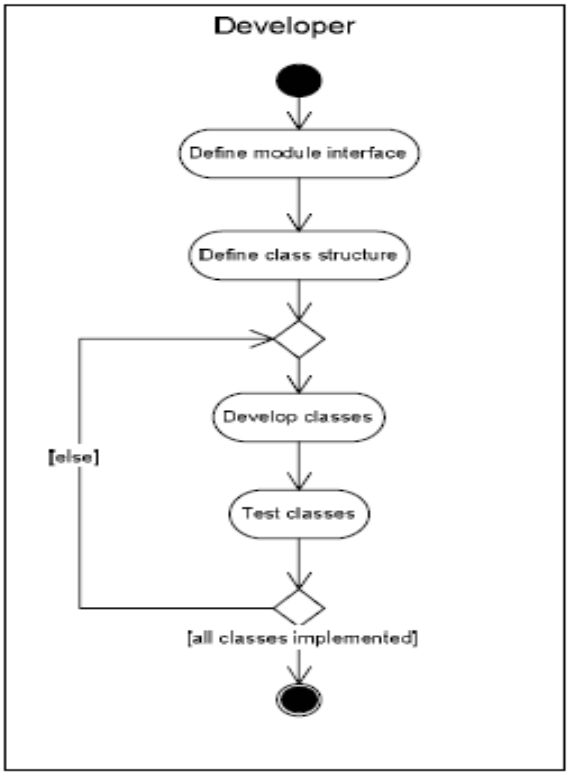

Figure 1: An example of a software development business process

The understanding of a business process in this context is rather that of a business workflow with the difference that automated execution (e. g. on a workflow management system) is neither mandatory nor intended. Therefore, a process consists of activities that are interconnected by a control structure. Activities are performed by organizational units including persons and machines in defined roles. Artifacts are objects that can flow between activities and be exchanged between organizational units. In this work we want to present business processes as UML activity diagrams [8, 9]. Figure 1 shows an example of a software development process. The next section gives a brief overview of quality management as a source for business process patterns.

III QUALITY MANAGEMENT

The complexity of modern products like software systems increases continuously as well as the complexity of modern organizations and businesses system. That makes it necessary to design business processes that contain structures and properties which allow the handling of complexity and possible errors resulting from it. Total Quality Management (TQM) systems provide a systematic and process oriented way to design business processes that reduce complexity and ensure improve quality throughout the whole production process. The quality management movement emerged in the Japanese car manufacturer industry in the 1950ies. In order to gain market shares on the highly competitive American car market Just- In-Time production was introduced to reduce costs of extensive stock capacities and enable faster production cycles, therefore increasing the efficiency of the car production and allowing cheaper market prices. Zero-Error production was a prerequisite for Just-In-Time production, because suppliers and producers are tied very closely together and low stock capacities make them dependent on each other. Minor errors can therefore lead to halt the whole production process [Oess89]. In the following decades quality management has gained momentum and spread out into American and European companies, at first mainly in the producing industries. The American military and space technology issued in 1959 a set of rules for quality assessments system called “Quality program requirements” as MIL-Q-9858. At the end of the 70ies the Technical Committee TC 176 “Quality Management and Quality Assurance” of the “International Organization for Standardization” was found on behalf of a German initiative. Many parts of the MIL-Q-9858 together with other developments lead to the creation of the ISO 9000 norm for quality management, which is now the most popular TQM system world-wide.

IV APPLYING PATTERNS TO BUSINESS PROCESSES

The requirements or need on business processes described in the quality management system norm ISO 9000 are used in this work as the source for business process patterns. Generally, patterns can be used for the two different main purposes: forward and reverse engineering. The forward engineering approach means that a pattern will be used to construct a business process from the beginning whereas the reverse engineering approach wants to analyse an existing process in order to find or match patterns and properties already existing in a business workflow.

4.1 Patterns in Business Process Construction

Patterns describe solutions for problems in a context. Early works on design patterns have been made in different contexts like for example architecture [5]. The methodology was adopted in the software engineering world because it happens frequently in software design projects that similar design problems occur over and over again in different fashions. Software Industry is also very competitive and challenging to stay on Quality & cost. Design Patterns provide proven solutions to common problems. The book of Gamma, Helm, Johnson and Vlissides [6] is probably the best known collection of design patterns in software engineering.

The application of design patterns does not only provide solutions to frequent design problems but has further advantages and solution. Most important is the fact, that applying patterns gives a well-defined structure to the

constructed design and fulfil the expectation. Patterns can be identified by their individual names and their functionality can be looked up in the pattern catalogue. If one developer wants to explain how a solution works he could say to the other something like “it works by the XXX pattern” and therefore complex structure become tractable. By that way design knowledge can easily be communicated. Yet the application of patterns allows a great deal of flexibility: another main advantage of the pattern approach is that the structures of the solution provided by a number of patterns can usually be intermixed and combined. This makes it possible to handle different demands on a complex system simultaneously. Using patterns, a number of business logic and technical aspects can be integrated into the design at the same time without losing overall perspective. System independent design can be performed by restricting the design solely on business logic patterns and apply technical structures or patterns later in the design process. Finally, a pattern catalogue is a concise way of representing design knowledge and good practice so that building up pattern catalogues for business processes is a good way to formulate and communicate knowledge about business process design.

4.2 Patterns in Business Process Analysis

The Quality Management Example In This Work Is Also An Example For The Value Of Pattern Recognition In Existing Processes. A company that desires to be certified ISO 9000 compliant needs to verify to the certification authority that the business processes to fulfil certain properties defined in the norm. With respect to the aspect of quality control such properties can be for example that quality objectives are defined before the production process takes place, that the quality of the product has to be assessed according to the defined objectives afterwards and that a quality management department exists which is provided with the necessary information to develop the quality management system itself. All these aspects can be brought together in a pattern called “Quality control”. Analysing an existing business process and finding a quality management pattern in this process means that certain quality management properties are fulfilled and therefore pattern matching can be used to prove to a certification authority or customers that important parts of a quality management system are implemented. The next section describes a systematic way of writing down business process patterns.

V A Pattern Description Language for Business Processes

As soon as the number of patterns used increases, a catalogue of patterns has to be formulatedin a thorough and concise way to gain all advantages of the pattern approach. Applicable patterns should be easy to find in a pattern catalogue, to compare and to cope with. This is usually achieved by defining a pattern description template that contains the three basic elements problem, context and solution together with a number of other sections according to the application domain. Each pattern in the catalogue is described using that template and each section of the template contains different properties of the pattern.

Templates such as those used be Gamma et al. [6] and Meszaros [7] are specifically designed for the description of patterns in software engineering and cannot be directly applied to business processes. This makes it necessary as a first step to define a template that meets the special needs of the application domain. Concerning business processes and workflows as application domain, the above mentioned behavioural, organizational and

informational aspects need to find their equivalent in the template. Figure 2 shows a proposal for a business process template.

The section "Name" is important to be able to communicate about a pattern and therefore making it part of the design knowledge. A short "Description" of the pattern gives a brief :

Name *The pattern name describes the most important property of the pattern in a few words.*

Description *A short description of the pattern and the problem that is solved.*

This section should contain only a few words.

Application Context *describes the requirements for the application of the pattern.*

Organizational Context *This section shows the assumptions about the organizational structure in which a pattern should be applied.*

Problem *A detailed description of the problem including the forces that lead to the creation of the pattern and to its application. Optionally containing an example scenario. Solution A full description of the structure of the solution. The section could contain an activity diagram, a Petri net etc.*

Result describes the resulting context after the application of the pattern. This could be the fulfilment of requirements or invariants or the creation of certain data.

Reasons for choosing the given solution.

Related Patterns with similar content or that work together with the actual one.

Figure 2: A business process pattern template

overview to be able to find an appropriate pattern quickly. The section "Application Context" has to be consulted in order to see if a chosen pattern is really applicable in the given situation. A specialty concerning patterns for business processes is the section "Organizational Context". A business process pattern usually requires the existence of certain organizational units that are responsible for performing activities. In this section all information about such organizational units is put together.

The sections "Problem "and "Solution" are self-explaining. In connection with business processes the section "Result" usually contains a list of documents or other data units that were created by the execution of the process and therefore pays attention to the informational aspect. The sections "Reasons" and "Related Patterns" provide information that is useful for choosing a pattern for a given problem.

VI EXAMPLE

To illustrate the description of patterns using the template defined in the preceding section and the concepts of modelling business processes with patterns we want to give an example for a business process pattern derived from the requirements of the ISO 9000 norm. A central part of a quality management system is quality control. This means that the result of a production process has to be compared to predefined quality objectives by a supervisor. The next subsection is the description of the pattern "Quality Control" derived from the ISO 9000 norm.

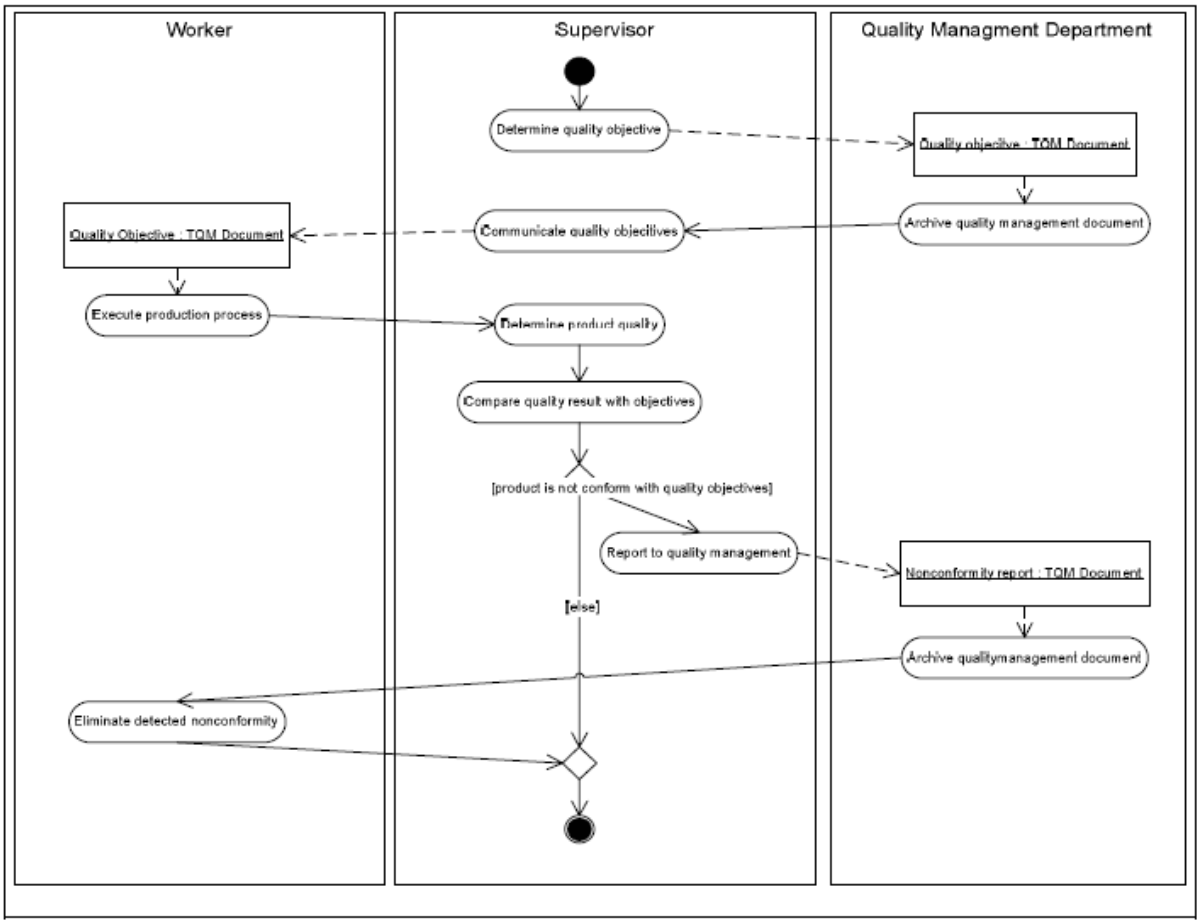

Figure 3: Activity diagram as part of the section "Solution" of the example pattern: "Quality control "

6.1. Example Pattern

Name: Quality Control

Description: How can a constant high level of quality be reached?

Application Context: Products and services are the result of the execution of business processes.

Organizational Context: Products are produced by production workers. A supervisor controls the products and is responsible for the production workers. The organization has a quality management department that is responsible for the quality management system as a whole.

Problem: The quality of the products and services shall be adequate to fulfil the needs of the customer. No product shall reach the customer that does not fulfil the quality requirements. Information shall be collected that allows the identification of repeated errors and their cause.

Documents are created and archived that prove to the customer that the product quality has been tested and confirmed.

Solution: Quality tests are highly necessary for processes in which products for external customers are produced. The quality objectives have to be determined before the production process starts. The quality objectives have to be communicated to the production workers. Quality nonconformities have to be eliminated. All quality information has to be collected and

sent to the quality management department for statistical analysis, systemic improvements etc.

Result: Quality is defined according to the customers needs. Products reaching the customer are quality tested. Information about quality objectives and test results are available at the quality management department.

Reasons: Regular defined quality test are the basic building block of a quality management system. Early-stage quality objective definition is often omitted or forgotten but necessary for a functional quality management system. The application of the pattern reminds the business process modeler of including all necessary activities for quality control. Further treatment of the quality management documents shall not be part of this pattern. Figure 3 shows the solution provided by the pattern "Quality Control" in an activity diagram.

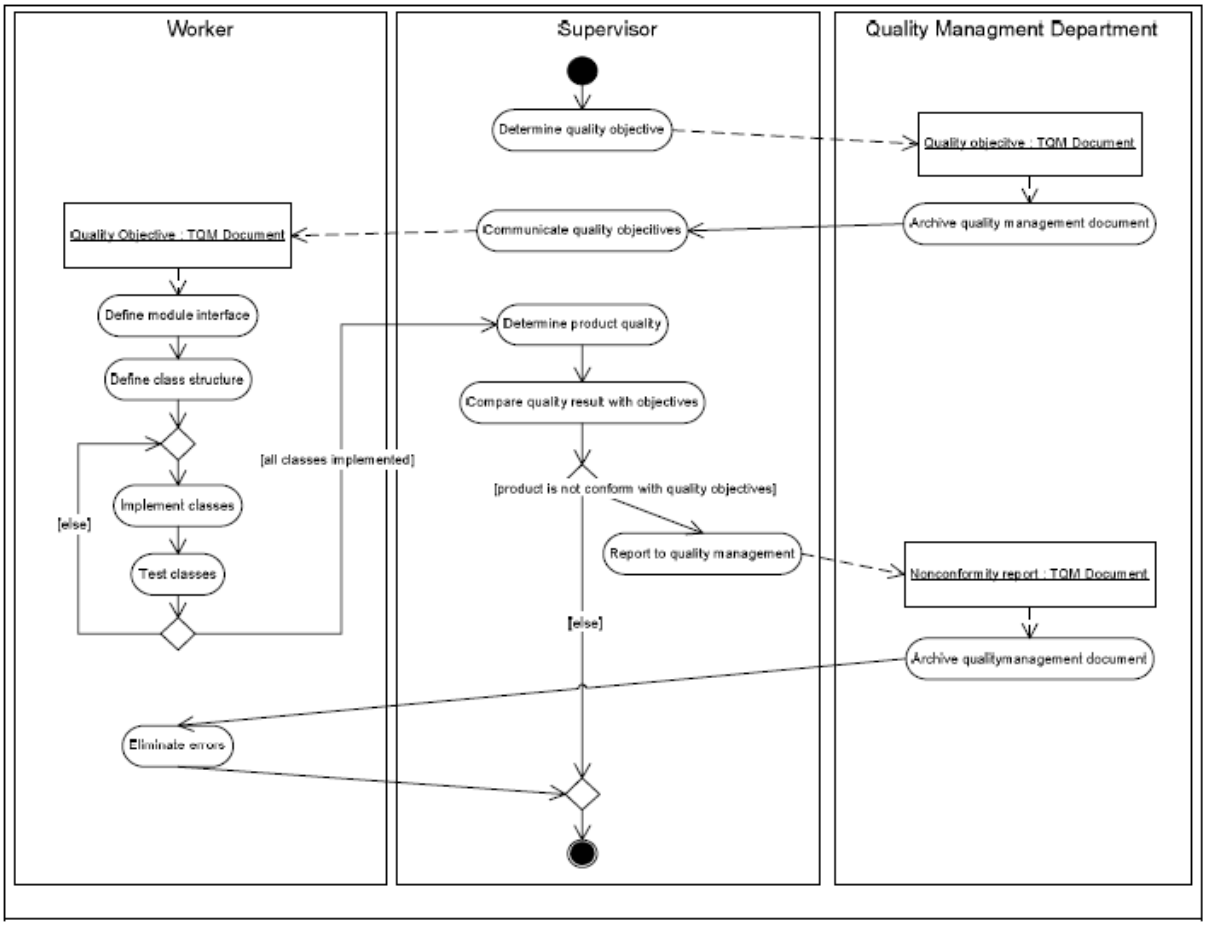

Figure 4: Simple implementation of the pattern "Quality Control"

6.2 Application of the Pattern

Now this pattern can be applied to construct a new process out of an existing process by adding quality control. Figure 1 shows an example business process describing an incremental software development process. Two different ways of applying the pattern to the business process shall be presented. Figure 4 shows that the software production process from figure 1 is included into the pattern process replacing and therefore refining the activity "Execute production process". The activities in the pattern are defined as neutrally as possible to

make it applicable in different contexts. In this refinement process some activities are concretized according to the application domain as for example “Debug code and eliminate errors”.

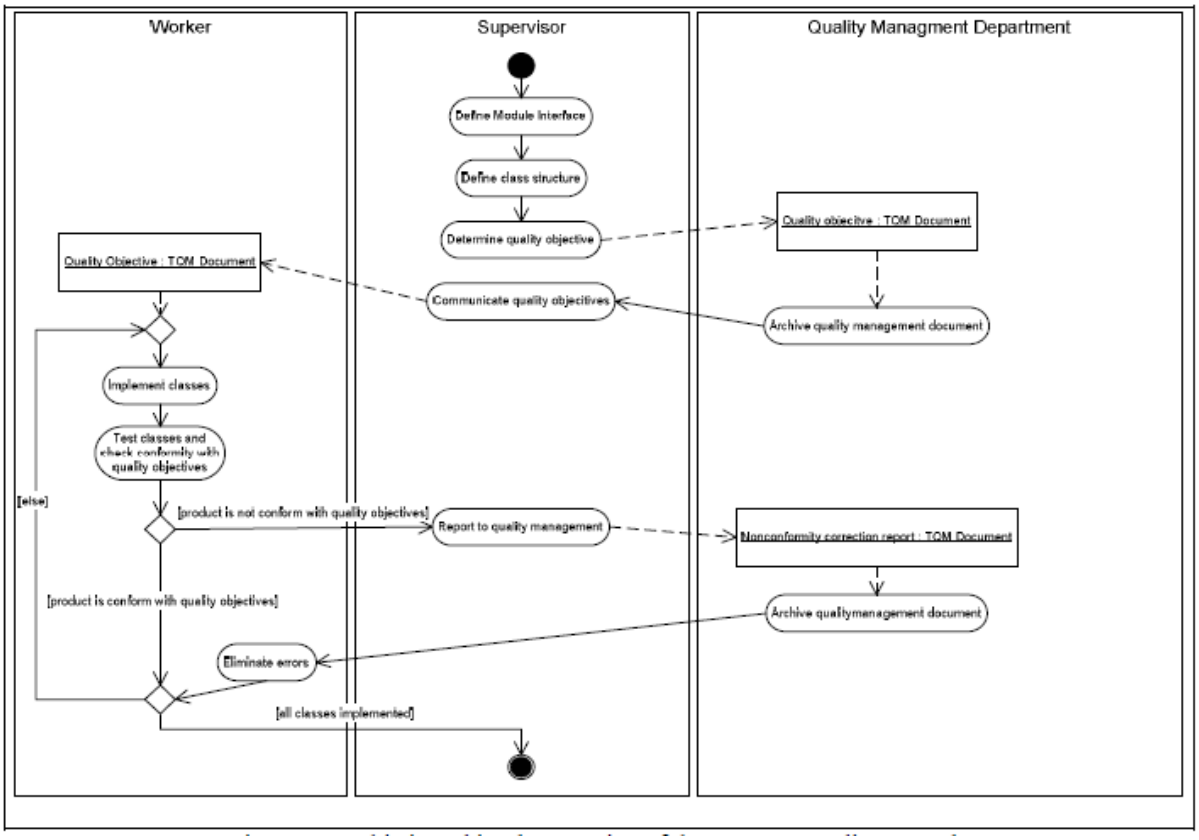

Figure 5: Sophisticated implementation of the pattern “Quality control”

Figure 5 shows a more sophisticated way of implementing the pattern into the production process. It exploits the fact that the production process contains some kind of quality control itself by the repeated testing of the implemented classes. The process in figure 4 relates to the software production process as a black box and is therefore an example of simple refinement. In contrast to that the process in figure 5 shows the flexibility of the pattern approach by the fact that far more complex applications of a pattern are possible. This process shows a rather tight connection between quality management and the developing process. Here the pattern is intermixed with the production process.

Both results of the business process modelling process as shown in figure 4 and figure 5 are possible ways of refinement and the application of a pattern in a technical sense. Both fulfil the quality management requirements as stated in the “Problem” section of the pattern. It is a question of managerial decision if a company prefers a rather loose or tight coupling of the quality management system into the development respectively production process. These two examples show how differently and flexibly process patterns can describe high-level structures and properties and how they can be included into concrete production processes in order to fulfil certain requirements, e. g. those of quality management.

VII CONCLUSION

In this paper we could show that the pattern approach can be beneficial not only in object oriented software design but also in business process design. The example of quality management patterns indicates that these benefits are not restricted to forward engineering of business processes but that the pattern approach also be helpful in analysing properties of given processes. Future work in this area is intended to describe how pattern matching in a business process can be defined and how the existence of a pattern in a process can be validated on a more formal basis.

REFERENCE

1. I. Jacobson, G. Booch, J. Rumbaugh. The Unified Process. Addison-Wesley, Reading, Mass., 2001.
2. M. E. Porter. Competitive Advantage – creating and sustaining superior performance. Free Press, New York, 1998
3. B. Curtis, M. I. Kellner, and J. Over. Process modeling. Communications of the ACM, 35(9), 1992.
4. S. Jablonski, C. Bussler. Workflow Management: Modeling Concepts, Architecture and Implementation. International Thomson Computer Press, London, 1996.
5. C. Alexander, S. Ishikawa, M. Silverstein, M. Jacobson, I. Fiksdahl-King, S. Angel. A Pattern Language. Oxford University Press, New York, 1977.
6. E. Gamma, R., R. Johnson, and J. Vlissides. Design Patterns – Elements of Reusable Object-Oriented Software. Addison-Wesley, Reading, Massachusetts, 1995.
7. G. Meszaros. A Pattern Language for Pattern Writing, http://www.hillside.net/patterns/writing/patternwritingpaper.htm. November 2002
8. M. Fowler, K. Scott. UML distilled – a brief guide to the standard object modeling language. Addison-Wesley, Reading, Mass., 2000
9. Object Management Group. OMG Unified Modeling Language Specification, Version 1.4, ftp://ftp.omg.org/pub/docs/formal/01-09.67.pdf. September 2001