Process-Oriented Requirements Management

Sebastian Graf
PROMATIS software GmbH
Ettlingen

Keywords

Business Process Management, BPM, modeling, simulation, reference models, workflow, BPEL, monitoring, round trip, risk management

Summary

In the context of the implementation of increasingly complex IT systems a solid high quality and complete requirements analysis is a must. No one will dispute this statement seriously and yet the requirements analysis process and in particular, the process-oriented analysis is an unloved stepchild in many projects. When asked about the reasons, we always encounter the same familiar suspects: no time, no budget, no tools, no trained resources and the frequently asked question on where the benefit of a process-oriented analysis lies for the project. If we trace the various phases of projects, we notice that errors are propagated from the analysis into the design and implementation, leading to costly rework scenarios. To address this problem masterly, the available development tools have now summarily been ordered a round trip, therefore creating a technology to redirect project-artifacts from the development or design phase back to the level of analysis. This is based on the philosophy that software developers will correct analytical errors, which have made it into the implementation, and the results of the correction can then be transported back via reverse engineering, back to the level of analysis. Granted, such concepts may be helpful to a limited extent, but there is no substitute to the realization that the royal road to a successful project leads just yet over a high-quality analysis. This paper describes the key aspects to pay attention to in a process-oriented requirements analysis in order to not having to rely on a supposedly curing round trip.
# Table of Content

1. Why IT Projects fail ........................................................................................................ 3

2. Why a Round Trip is not a Cure against Failure......................................................... 5

3. Using BPM Tools wisely ............................................................................................... 7
  3.1 Selecting the appropriate tools .................................................................................... 7
  3.2 The process-oriented approach ................................................................................... 8
  3.3 Knowledge Bases / Reference Models ........................................................................ 8
  3.4 Simulating a business process .................................................................................... 8
  3.5 Key indicator based monitoring .................................................................................. 9
1 Why IT Projects fail

They actually exist - IT projects that fail. And if one believes the renowned Standish Group (http://www.standishgroup.com), then this is by no means a marginal phenomenon: According to the CHAOS report by the Standish Group in the years 2004 to 2008, an average of 20% of all IT projects failed, 48% barely escaped the collapse, and only 32% were successfully completed. For 2010, the report even predicts an increase of failing projects at a total of 32% (see Figure 1).

![Project Success according to the CHAOS Report](image)

Figure 1: CHAOS Report 1994 - 2010, Standish Group
The reasons for failure stated in the report also speak for themselves: In 13% of the cases an incomplete requirements analysis was identified as the main cause, in 12% non-business users were not involved to a sufficient extent, in 10% unrealistic assumptions were made, and in 9% ever-changing demands were not even detected. If one adds up these numbers, it becomes clear that deficiencies in the requirements analysis add up to 44% of the reasons for the failure of a project. Lack of technical expertise, however, was identified in only 4% of the cases as the cause (see Figure 2).

Figure 2: Details CHAOS report, Standish Group

The conclusion from these downright depressing figures can only be the fact that the key to project success has to be found in a high quality requirements analysis. For this reason, the topic BPM (Business Process Management) has justifiably experienced a renaissance, which was unthinkable only a few years ago.
2 Why a Round Trip is not a Cure against Failure

Many software vendors, including Oracle, have BPM tools now firmly entrenched in their product portfolio and propagate BPM as a fundamental and integral part of software development. It is even suggested that these BPM tools are equipped with bi-directional interfaces that allow the exchange of analytical artifacts with the appropriate development tools in both directions. In the Oracle environment for example the BPA Suite, the BPM Suite or Horus® are available for analysis and the SOA Suite for development, and many users are also quite willing to use these products in their projects.

But what is often forgotten: The simple fact that a requirement analysis does not only require appropriate tools, but also a matching approach and well-trained analysts, which by their tool skills and experience are in a position to ensure that high-quality results emerge from the analysis phase. Frequently one notes therefore in the implementation phase of a project that generated development artifacts based on the results of requirements analysis, such as BPEL processes, are not feasible, and at worst these processes do not even meet the technical requirements. To address this widespread problem, the relevant tool manufacturers have a built-in round trip in their tools, with which subsequent corrections to the development artifacts can be redirected to the level of analysis. Thus, it should be possible to correct errors from the analysis relatively easily and with little effort in the development phase and then carry out adjustments on the analysis level. That this is a fatally wrong way and by no means assures safety in the project soon becomes clear when one considers this approach in detail.

The following problems make it clear very quickly, that the above-described round trip method is not suitable to subsequently correct problems that are due to an insufficient analysis:

1. The round trip method is based on the assumption that a software developer recognizes in the development process that the development blueprints derived from the analysis artifacts are incorrect. This assumption is very critical because software developers have only very rarely and only in small projects all the necessary technical know-how to perform this review. Typically, these errors are not recognized by the development team and occur only apparent in the test phase, when users experience the system for the first time.

2. The method assumes that the developer carries out the necessary corrections on the implementation level. But how is that to be done? This would mean that the developer, equipped for example with BPEL processes, performs a correction in these processes together with the business users. This approach does sound rather practical, however, one should take into account, contrary to all popular opinions, that BPEL processes (in the opinion of the author similar for BPMN processes) are very technical artifacts, which are in no way suitable for a purposeful dialogue with a business user or even on a management level.

3. Finally, the method is based on the assumption that it is possible for different models from the analysis and the implementation phase to be transformed nearly lossless into each other. And of course, this transformation should also be possible in a bi-directional manner. It is the recurring dream of the common IT community, to represent
two formal description methods, which are equipped with different syntax and semantics, generatively together. However, experience shows very clearly, that with such transformations a loss is to be expected, and the more different the two procedures are, the greater the expected losses. For example, considering the conversion of event-driven process chains (EPC), as they are used inter alia in the BPA Suite, into BPEL processes, no one can speak of a complete conversion.

Ultimately, therefore, the following two scenarios are possible for projects:

1. In the implementation phase, only a few discrepancies come to light (due to high-quality analysis). In this case, it is certainly not a critical factor if the available tools of the analysis and implementation phase do not provide a round trip, and if the few necessary adjustments have to be done manually on both levels.

2. In the implementation, it is actually found that large parts of the generated blueprints are technically wrong (which is due to dramatic failures in the analysis). In this case, the project is in great difficulties and is a candidate to be found in the next Standish report in the category "failed projects". A round trip will not change that.
3 Using BPM Tools wisely

Therefore, the key question is not whether it makes sense to use BPM methods and tools in software projects or not, rather, the crucial question is how BPM methods are used wisely to minimize the relevant project risks.

3.1 Selecting the appropriate tools

The use of a BPM tool usually starts with the selection of the appropriate tool. Even at this early stage, basic issues have to be considered, and a one-vendor strategy is not always the best guide. Rather, the following considerations should be taken into account:

- Can the tool map all of the information needed for the rest of the project? All available tools on the market allow mapping procedure models and organizational structures. Moreover, it may be necessary to be able to describe the business objects that are the subject of the process, in a structured manner, in order to be able to derive i.e. data models and service interfaces subsequently. Other artifacts that may be of great interest in documenting the requirements are, for example: Business rules, roles, and resources, key indicators, risks, objectives and achievements, to name a few.

- How detailed can each of the artifacts be described? Here, a closer look on the appropriate tool is worthwhile. Example: Is the activity of a process only represented by a graphical object in which a descriptive text can be deposited? Or would it possible to deposit much more detailed information, such as cost, time, design rules and such?

- How strongly are the individual model types linked? Here, an assessment is necessary on whether it is possible to combine artifacts from one model type with the artifacts from a different model type. One example is the linking of objects from the procedure model with objects from the object model. The description of objects in the procedure model is typically only very superficial. Usually only one graphical object is defined, which is provided with a comment that describes the structure of the object in prose. If later, however, a service interface or database structure needs to be derived from this process, it is essential to describe this object in an object model in detail and well structured.

- How easy is it to use the tool? The use of a BPM tool stands or falls with the acceptance by users. Here, less is often more. The most comprehensive package is useless if the users have no expertise in the operation and if it is not clear how to deal with the tool, in order to capture all requirements on one hand, and on the other side, to ensure that the results of the analysis phase can be really used in other phases of the project.

- Which interfaces are provided with the tool? Here, it has to be clarified before the project start, which artifacts are expected by the development team after the analysis and how they can be extracted from the BPM tool. Most of the BPM tools
that are currently available on the market are able to generate BPEL blueprints, for example. Only a few tools can actually generate BPMN or are, for example, able to transform analytical object models into UML models or physical database models.

### 3.2 The process-oriented approach

After the appropriate tool is identified, it must be ensured that the tool is also used profitably. It is often required that tool-users do some rethinking. In many BPM projects, one can observe that BPM tools are abused to document requirements in a purely functional and therefore inarticulate manner. Once this aberration is chosen, a project failure can be forecast at an early phase. This method is very alluring in SOA projects, by the way, where the danger exists at an early project phase to think in services and not in processes. With the use of BPM methods, it must be clear to all involved that the technical process is the central object in the requirement analysis, no ifs or buts. Very often one hears the reasoning that a process really does not exist, that one would only like to develop a system that does this first and then that, and if the event X occurs, then one has to react in a certain manner. Unaware, the astute critic has already begun to describe a process, however. Furthermore, it should also be noted that business users who are required to play a fundamental role in the analysis phase, as only these persons have knowledge of business procedures (what is often forgotten or even ignored by IT specialists), think exclusively in processes, either consciously or unconsciously. Therefore, if one wants to acquire a complete picture of the requirements, one should refrain from discussions with users about functions or services.

### 3.3 Knowledge Bases / Reference Models

The larger the project, which is to be supported by BPM methods, the greater the required effort will become in the process analysis. This is the case in large-scale Enterprise Resource Management Projects. For such projects, it pays to look out for so-called knowledge bases or reference models. Some BPM tool manufacturers, such as Horus software GmbH, have recognized that a BPM tool alone is no guarantee for the successful use of this tool. In many cases, the allotment of appropriate content is much more important, which helps the user in taking the first steps with the tool, without having to start from "a blank sheet of paper".

Furthermore, the expertise of innumerable projects is hidden within good Knowledge Bases, which can prove to be an invaluable asset for your project.

### 3.4 Simulating a business process

Unfortunately, the topic of simulation still plays a minor role in the IT branch to this day. It almost seems as if the guild of software engineers wanted to take a special position here, compared to their colleagues: For an automobile manufacturer it is a matter of course that a new vehicle will have traveled millions of miles in a simulator before it is even built. A civil engineer building a huge bridge will of course seek a simulators advice when it comes to the question of what would happen next if the bridge were to be subject to an
earthquake. In addition, we all welcome the fact that Lufthansa pilots are trained in a simulator first, before being allowed to fly a real airplane. However, why does the medium simulation not come to use in the IT industry - in the sector, of all places, which develops the corresponding simulators for all other branches of industry? In most cases, the specialist users and the software developers do not understand what the advantages of a simulation of business processes consist of; these are actually obvious:

1. One can prove the logical propriety and integrity of a process with a simulator. It is a prerequisite that the modeling tool used is based on a formal description method, which allows such an examination. Ensuring the integrity of processes in practice is therefore one of the largest problems. Process branches frequently cause significant additional efforts and expenses in the implementation and are forgotten in the analysis. As a rule, specialist users first think of the standard process. Exceptions or special cases are easily forgotten. Mostly because these are underestimated by the low appearance probability. A simulator can help the modeler discover process model errors in the analysis.

2. The simulation can predict the behavior of SLA (Service Level Agreement) process relevant performance criteria. More statements can be made regarding cycle times, flow-rate, waiting times, critical paths, resource constraints and the like - and this even before the first line of code had been programmed, and not just when the system is already in production and various SLA parameters have been broken. Particularly with SOA projects, it has shown that the SLA-relevant requirements for an application are mostly underestimated or not even assessed in the analysis, with the result that in the production environment it is discovered that the implemented solution is not in a position to meet the required flow characteristics. This fact would have been easy to identify in advance with a simulation.

3. Looking at the available BPM tools on the market, you then realize very quickly that few manufacturers offer genuinely useful simulators. If simulation is used for a BPM project, then this issue should play a major role in product selection. Some tools will falsely offer an animator instead of a simulator. The difference here is that an animator is indeed able to describe the dynamics of a process through animation, but in contrast to a real simulator no profound analysis of the simulation runs are possible. With a good simulator it is also expected that weaknesses in the processes will be detected automatically and corresponding instructions given on how the processes can be improved.

4. Finally, it must be ensured that the simulator in question is not only able to simulate "Hello World" processes. This is especially important when not only the logical propriety of a process is to be verified, but also if it concerns examining the relevant performance criteria of a business process over a long simulation period (it is necessary in certain cases to simulate business processes over several months).

### 3.5 Key indicator based monitoring

If the BPM-based implementation of a process is complete and the process is in the production environment, then it is extremely important to monitor the real behavior of the processes in real time by way of key indicator monitoring in order to close the gap between simulation and the real world. At the same time it must be distinguished between a purely
technical and a specialized monitoring. The technical monitoring is used primarily for process operators to ensure compliance with relevant SLA properties. Oracle provides the suitable tools in this instance with BAM (Business Activity Monitoring) and the Enterprise Manager.

Far more important is the specialized monitoring. Key performance indicators are monitored here, which typically are provided to the specialist users in so-called Balanced Scorecards or Management Performance Cockpits. These key performance indicators must already exist in the analysis phase of the examinations. If this is not the case, the system will indeed go into operation after implementation, however the specialist users and management are, as far as the evaluation of current processes in terms of business performance management, flying blind. Critical conditions, bottlenecks and slow-building problems cannot be detected. In the documentation of key performance figures within the analysis phase, it must be ensured that the origin of the measured values on which the figures are based are clearly defined, and that individual key performance indicators are linked by way of hierarchy formation and aggregation. Subsequently, these key hierarchies must be depicted in appropriate monitoring tools. Unfortunately, currently there are no BPM tools are available that would allow the automatic transfer of key hierarchies into monitoring tools, which would prove to be a very rewarding activity for the producers of BPM tools.

If in the analysis one has fully recorded the performance indicators as part of the process models, and then implemented these into a monitoring tool, it is possible that information regarding the real behavior of these key performance indicators can once again flow into the analysis models in a refinement step. In this context, one often speaks of closed-loop monitoring in the BPM community, although here all tool manufacturers are still far from an operative implementation. This replay of information concerning key performance indicators at an analytical level provides eminent assistance in the refinement of analytical models, thus allowing an improved adaptation of process models to the conditions of the actual processes used. Furthermore, this is a basic requirement if it becomes necessary to react fast and flexibly to changes in the process landscape. It is exactly this consistent and coherent approach to the continual optimization of business processes, in terms of a process control system and in the opinion of the author, to which preference should clearly be afforded in contrast to a rudimentary round trip process between analysis and design or implementation tools.
Conclusion:
BPM is back - many software manufacturers have complemented their portfolio with the appropriate tools. And yet, the number of failed IT projects increases year by year, as obviously dramatic mistakes are being made in the requirements analysis, in spite of BPM. These seemingly incompatible facts clearly show that although many companies have BPM tools at their disposal, they do not use the tools at all or use them in an imprudent manner in terms of quality improvement. Often the use of BPM methods is limited to the analysis phase of a project in which useless mounds of paperwork are created that cannot be utilized in the further course of the project. To make matters worse, many discussions on the topic of BPM lose themselves in totally irrelevant technical details. Rather than to discuss the implementation of efficient and high quality processes, high value technical model transformations are discussed, that on closer inspection turn out to be inconsequential. BPM, if sensibly applied, can make a considerable contribution to the successful execution of a software project. Good tools and highly qualified BPM specialists are a prerequisite, but primarily a strategic approach that ensures, while establishing BPM in all project phases up to the startup of an IT system, that efficient business processes and their proper implementation are the focus of the project.

Note
The listed products are trademarked and belong to the respective owners. Effective date of documentation: September 2010

PROMATIS software GmbH
Pforzheimer Str. 160
76275 Ettlingen
Ph . +49 7243 2179-0
Fax +49 7243 2179-99
info@promatis.de
www.promatis.de