

foryouandyourcustomers shows you why Business Process Modelling (BPM) should not be the first step in a definition of requirements.

'BPM considered harmful'
by Stefan Berner,
1 January 2017

BPM considered harmful

Stefan Berner

1 January 2017

Original article appeared in:

Lecture Notes in Informatics: Modellierung 2016 (page 181) ISBN
978-3-88579-648-0

Abstract

Business Process Modelling (BPM) plays a major role in the early stages of software development – too large, in the author's opinion. Almost all procedural methods recommend starting with BPM modelling. Almost all business analysts use BPM as a starting point, and as the simplest route to knowledge and understanding of a business world.

Without a common understanding of the terms used and their relevance to an environment, neither future users nor computer scientists can model effective business processes. BPM is not the best way to arrive at this common understanding. BPM has other serious disadvantages that make it unsuitable as a starting point for an IT project.

This paper reveals the disadvantages of using BPM too early. Furthermore, it also demonstrates how a better understanding of the world to be modelled can be obtained before process modelling.

Keywords: BPM, process model, information model, software quality

Introduction

With his famous *letter to the editor*, 'Go To Statement Considered Harmful' [2], Dijkstra launched a controversial discussion which finally led to the acceptance of structured programming. It is this meaning of the expression 'harmful' that has been adopted in the title of the present paper. The following observations from a practitioner's viewpoint are intended to move the discussion forward. What role should Business Process Modelling play in a software project? Is it really the best starting point and approach to a software solution?

BPM has certain disadvantages. The way it is typically used does not result in the best software quality. The observations listed below, and the hypotheses derived from them, have been confirmed by the author's many years of practical project experience. It is not within my power to verify them independently. This article is intended to motivate people involved in research to test these statements and hypotheses in neutral studies.

Concepts

Business Process Model

«BPM allows to model and present business processes in a user acceptable way

The purpose of the Business Process Model is to describe business processes in a way that customers can understand. At the same time, it is intended to provide a formal template for implementation in the software. A Business Process Model typically unites the following aspects:

- Functions
- Data
- Information
- Events
- Statuses
- Organizational units
- Players
- Outlets
- Media
- Data storage systems and channels
- Interfaces

The Business Process Model describes what stakeholders do with information, and how they do it, from a professional perspective. It describes business process from the users' point of view. Example:

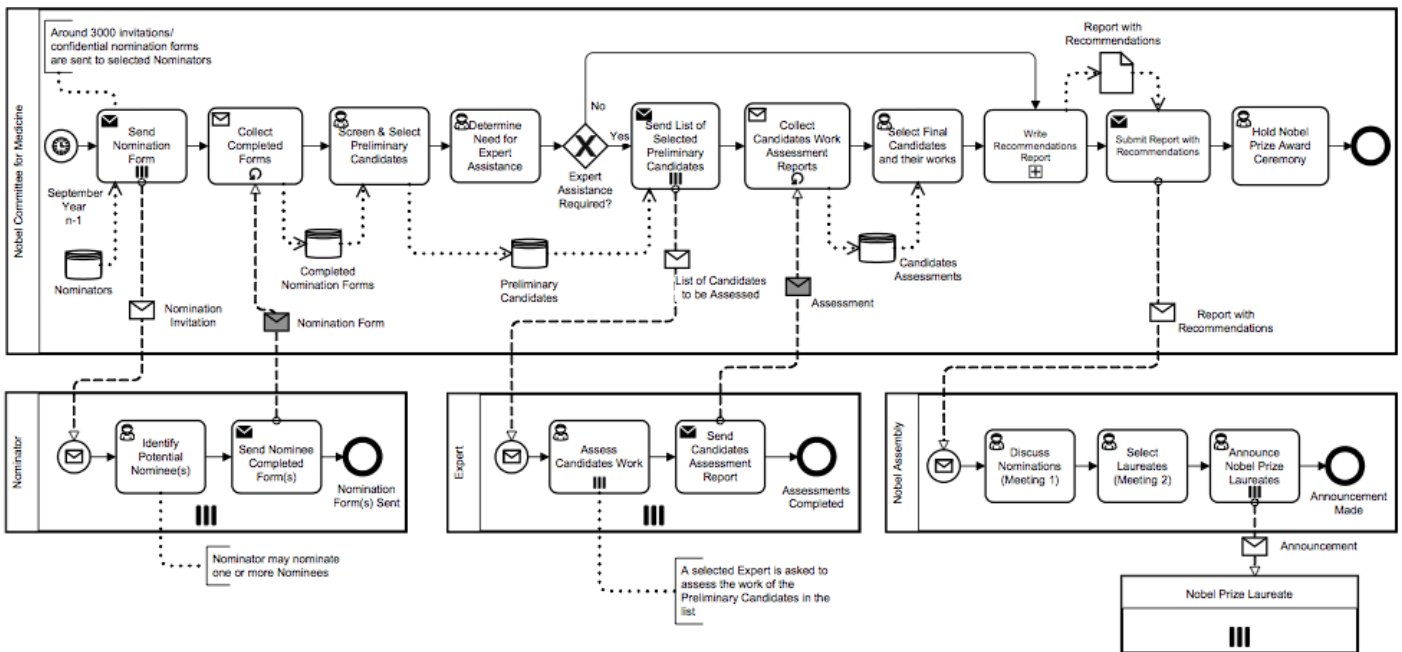


Figure 1: Example of a Business Process Model (see [3])

Information model

'The information model describes all information requirements unambiguously, uniquely and precisely.'

An information model is a description of the information required in an environment (see [1]). It describes entities, along with their attributes and relationships to other entities. It is usually described as a conceptual data model. The term **information model** is used to emphasise the following differences from a data model:

- All names (entities, attributes, relationships) are terms taken exclusively from the business world, and are accepted by all stakeholders without reservation.
- All links between entities are described by a verb, in both directions.
- No technical terms appear (e.g. database, tables, data types, keys, constraints, XML etc.)

The information model describes what things (entities) actually *have to do with one another* in functional terms – how they are connected structurally. It describes the possible, consistent and relevant statuses of a system – its static structure.

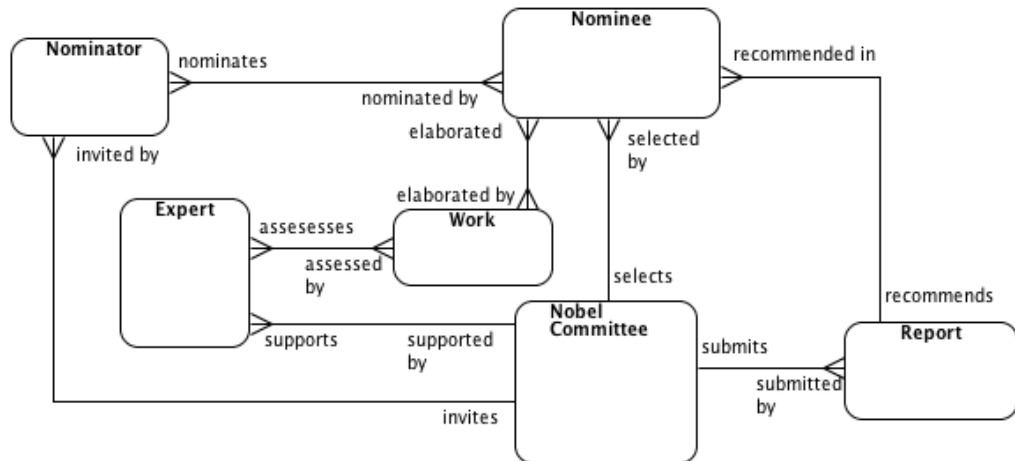


Figure 2: Example of an information model

The use of the term *data model* in connection with business modelling is deliberately avoided. Data represent information in storable, technologically manageable elements. Data values *per se* are meaningless. Users use the data values. They use information that results from interpretation of the data values. The information model is an explicit record of the mandatory functional interpretation of the data in an environment.

Criticisms of BPM Incorrect sequence

*'Criticism 1:
BPM does not start from
the requirements and
goals of the tasks.'*

A process transforms a system from one consistent status to another. Processes are never the destination; they are always the path. Processes describe *how* a change of status is effected. They do not describe the statuses as such. The Business Process Model describes what stakeholders *do* with things (entities), and how they do it. It describes how the system is transformed from one consistent status to another by means of an activity, with the aid of (technological) tools.

In order to design a good process, the starting point and final goal must be identified. On one hand there are the triggers (events) and the initial status, on the other the expected final status of the information manipulated. Without knowing what the aim of a process is, there is little point in describing it. Consistent statuses are described by functional, static system knowledge – the information model mentioned above.

In practice, a process model is usually created as the first stage in the description of a system. The reason for this will be explained in more detail below. In the second stage, the information used or created in this model is then combined to form a data model. A complete, precise description of the information structure from the users' perspective, independent of the processes, is hardly

ever encountered in practice. The result of this lack of consideration for the terms used and an inadequate understanding of the business world is that data models usually excessively reflect the structure of the processes and the perspectives of the developers. With this procedure, structural functional problems and inconsistencies are only identified late in the development process (when the data model is implemented or the module is created, or during tests). By this time there is no willingness to change the processes, since they have usually run their course and are fixed. This results in architecture problems on the data and program level that have to be 'compensated for'. Such a procedure is certainly less than ideal.

Too complex

*'Criticism 2:
BPM unites so many
aspects in one diagram
that it is too complex for
the average reader.'*

Because of the multitude of aspects represented in a Business Process Model, the document and its creation become very complex. When modelling individual processes, it is extremely difficult to develop a coherent structural description of the whole system at the same time. Such a task requires an above-average capacity for abstraction, and for many people this is too demanding.

In addition, the complexity of the model limits its usefulness as an overview document. For most stakeholders, it is very difficult to read. Some tasks can be followed easily ('What happens when there is a warning?'). With a Business Process Model, gaining an overview of the whole system can only be achieved with difficulty.

Legacy

*'Criticism 3:
BPM often tries to
improve the outdated
solutions of the old
system.'*

Process models are typically created in cooperation with the users of the existing systems. In most cases, this involves starting out with existing systems and trying to improve them. The existing processes were created in response to the requirements of the environment at that time (organisation, technology, communication channels, media disruptions, etc.) Many processes are the way they are because this was the best (most pragmatic) solution *under the circumstances prevailing at that time*. Over the years, these processes have determined the working methods of the people who work with them. Many employees have familiarised themselves with these processes and internalised them as the basis of their work. These processes *are* their work. From their point of view, business procedures are what the existing processes represent. A good example of this is the Swiss postcode. It was introduced in 1964 to improve the manual processes of sorting and distributing post. Its (hierarchical) structure was oriented towards geographical regions, economic centres and the main transport axes (above all the railways). Rejected by the population when it was introduced, over time it transformed itself into a cultural asset that affirmed their identity. With today's techniques of optical address recognition and automatic sorting, etc., it has become an inefficient relic. Despite this, it continues to be used and applied in processes relating to addressing.

Better processes are created when the purely functional aims of existing processes (=information statuses) are extracted, and the way to achieve this is redesigned while taking present framework conditions into account.

Ambiguous term definitions

*'Criticism 4:
BPM usually adopts the
loaded terms of the old
system indiscriminately.'*

When modelling business processes, business areas are always viewed and modelled from the point of view of individual processes, and therefore of individual players or organisational units. The terms that are used to describe the information required or created (entities) are always interpreted from the point of view of the particular person one is speaking to at the time. Example: Everyone

knows what a 'product' is. In an application that deals with the entire product cycle, the meaning of the term changes from department to department. In the production department, a 'product' is something different (i.e. has different properties and relationships) from a 'product' in purchasing, marketing, logistics or sales.

What is required is a definition and description of all collectively used terms that is independent of the players and process descriptions. Without these definitions, there is a grave danger that sloppy nomenclature and definitions will be adopted. The result is a semantically inaccurate architecture.

Modelling the processes too early makes the required discussion of uniform terms and definitions difficult. The focus on individual processes each time implies a specific interpretation of the terms. As a result, misunderstandings in communication between departments and IT – or also between different departments – often go unnoticed during the business analysis.

Causes and solution

Causes

Why do almost all developers, in all stages of their software development, start with the processes and derive the required data from them? Why is this *incorrect sequence* so popular? At least since Niklaus Wirth's *Algorithms and Data Structures* (1975), we know that a good data structure can result in simpler algorithms.

'Describing processes is closer to our experience and easier than describing static structures.'

It seems to be easier to describe processes (paths, routes) than static structures (maps). The vast majority of people prefer to describe routes (see [4]). Interestingly, by contrast it is easier to get an idea of an environment with the aid of a map than to navigate through this landscape using a description of a tour. This asymmetry can be explained by the fact that a map presents more information in a smaller space, and routes almost come about by themselves when one understands the underlying structure. Documenting information in concentrated form is time-consuming. This extra work is repaid by the time saved in reading it. Goethe described this circumstance as follows: 'Please excuse the length of this letter, I had no time to express myself briefly!'

'Finding uniform, clearly defined terms is one of the hardest tasks in the field of digitalisation.'

Uniform definitions and usages of terminology are hard to find and implement. They presuppose that the differences are recognised, and that the players involved are prepared to adapt their use of terminology, and therefore their way of speaking. Unambiguous terms can only be worked out in discussions about content and understanding. This often requires professional decisions about the correct meaning of a term, or the one that makes most sense in this environment. In meetings with professional representatives, these discussions are preferably avoided. They mean a lot of work, and IT staff and users shy away from the expense of their possible consequences. It is superficially easier (= involves less work) to discuss processes. The context and possible interpretations are clearer, and fewer discussions are needed.

'Starting with a description of the information creates a solid basis for all subsequent tasks.'

It requires a great deal of effort to identify, extract and change the various aspects of existing processes. It is intellectually challenging, and requires good specialist knowledge. Often, resistance from users and their managers has to be overcome (see above example of the postcode). The result of avoiding difficult tasks in this way is that too many superfluous and outmoded aspects of the existing business processes are incorporated into the new solution.

The attempt to avoid expenditure (of thought), as well as the reluctance to revise one's own understanding of things, result in sloppy terms and data structures and, therefore, to less than ideal 'legacy' processes that depend heavily on the past.

Approach to a solution

'BPM must be simplified and must solve one aspect after another.'

The overall task of BPM as a whole must be subdivided. Structural criteria (system boundaries, media disruptions, organisational units, programming environments etc.) are often used to break down processes into sub-tasks. The disadvantage with them is that, with each change to the environment, the modular structure of the solution can become inefficient or incorrect. The most stable aspect of systems are the actual subject-specific semantics. For example, the information structures and basic professional procedures of double-entry book-keeping have hardly changed since they were introduced in the 14th century.

The most volatile aspects are technologies and media. Structuring the solution according to semantic aspects results in systems with long-lasting stability. To create a stable, simple solution, the following are essential:

- Aspects with different rates of change must be documented in separate working stages and documents.
- Process requirements and results (events, starting status and target status) must be modelled and described before the processes.
- Aspects whose purpose is to communicate with customers or specialist users must be separated from aspects relating to technical matters. (Each document may only contain elements that the target audience can understand and considers relevant.)

The following procedure is tried and tested with BPM. First, the business events and possible system statuses (information model) are established. It is a necessary requirement for modelling good processes that these two documents can, and should, be worked out independently of process considerations. The next step is to describe the stable, purely professional aspects of the business processes. This should be done as independently as possible from any technology, organisation or media (disruptions), etc.

It is only after these overview data from a professional perspective are available that it is possible to create good operational processes that combine all aspects. This last step integrates the professional requirements (business events, information structure, professional description of processes) into the current environment (organisation, technology, system landscape).

Figures 3 and 4 on the next page show the information needs and professional process steps of the development process in rudimentary form.

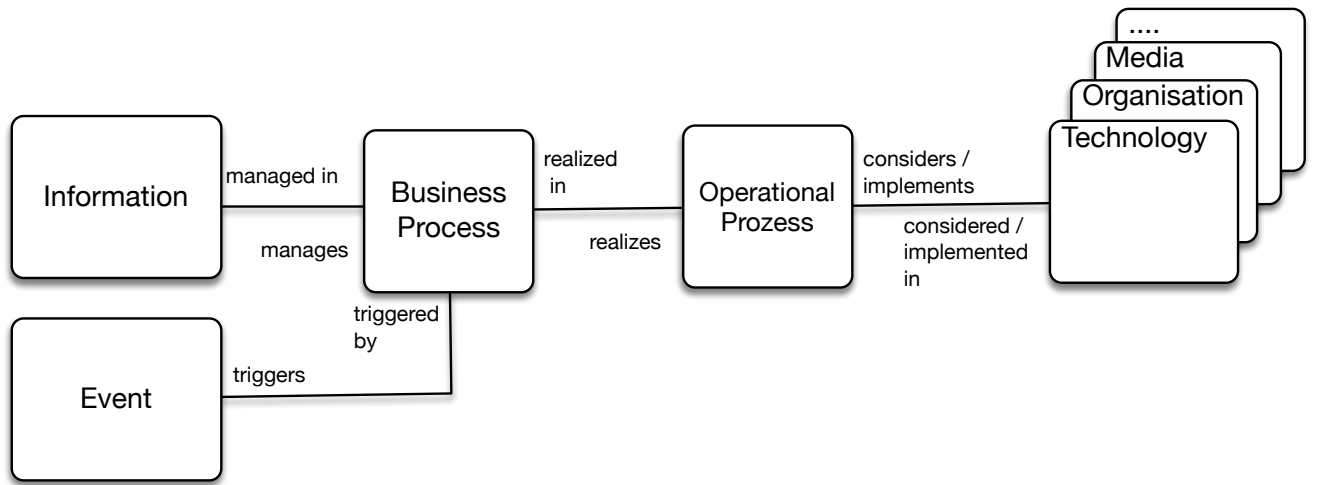


Figure 3: Sketch of an information model of the development process

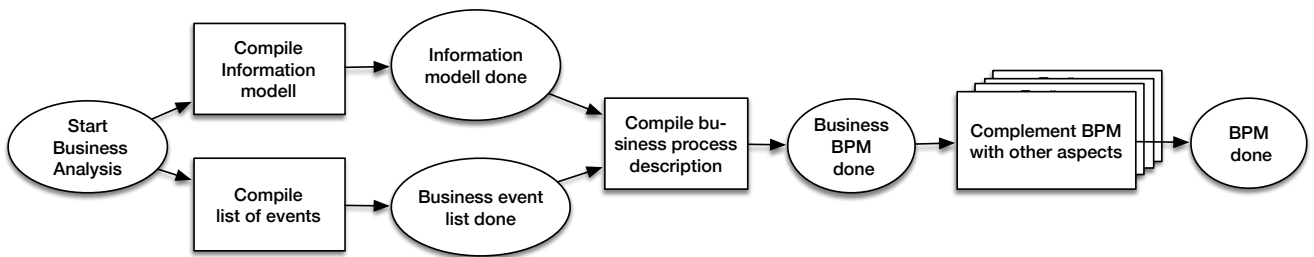


Figure 4: Sketch of a professional process model of the development process

'BPM is not a bad method. It is normally used incorrectly, and therefore does not produce the best results.'

Conclusion

BPM is not harmful *per se*. In practice, it is used incorrectly, or at the wrong time. Too many aspects are worked out in the wrong sequence and with the wrong focus. The result of this is that

- the overview documents produced are not suitable for the end users.
- many business analysts, managers and professional people are over-stretched by the tasks involved.
- the resulting data models do not optimally model professional semantics.
- too many *legacy* aspects are incorporated into the new solution
- many modules become too complex.
- new technologies, media and forms of organisation are not used with sufficient efficiency.

The solution approach described in the previous section has been successfully applied in several projects. Modelling the processes was vastly simplified. Compared with the old solution, many processes became simpler. The above list of negatives was extrapolated from the experience of several projects. My hypothesis is that changing from BPM to **information modelling as the first step** in the development process reduces these disadvantages, or prevents them. I hope that the present considerations will motivate people to work according to this approach, or to communicate their working methods and results, so that enough examples will emerge to enable the advantages of this procedure to be more objectively confirmed.

Bibliography

[1]

Berner Stefan: *Information Modelling – A Method method for improving understanding an accuracy in your collaboration*, ISBN: 978-3-7281-3943-6 vdf Hochschulverlag, Zurich 2019

[2]

Dijkstra, Edsger W.: 'Go To Statement Considered Harmful', *Communications of the ACM*, Volume 11 Issue 3, 1968, pp. 147–148

[3]

BPMN 2.0 by Example: Version 1.0, 2010, www.omg.org/spec/BPMN/2.0/examples/PDF/10-06-02.pdfPage26

Linde, Charlotte and Labov, William: 'Spatial Networks as a Site for the Study of Language and Thought', *Language*, Vol. 51, No. 4, 1975, pp. 924–939