1

Business Process Management

The recent progress of Business Process Management (BPM) is reflected by the figures of the related industry. Wintergreen Research estimates that the international market for BPM-related software and services accounted for more than USD $1 billion in 2005 with a tendency towards rapid growth in the subsequent couple of years [457]. The relevance of business process modeling to general management initiatives has been previously studied in the 1990s [28]. Today, Gartner finds that organizations that had the best results in implementing business process management spent more than 40 percent of the total project time on discovery and construction of their initial process model [265]. As a consequence, Gartner considers Business Process Modeling to be among the Top 10 Strategic Technologies for 2008.

Despite the plethora of popular and academic textbooks [164, 95, 196, 378, 27, 380, 248, 7, 9, 257, 49, 213, 233, 170, 407, 415, 199, 405, 447, 227, 408] as well as international professional and academic conference series such as the BPM conference [13, 106, 5, 115, 23], there are several fundamental problems that remain unsolved by current approaches. A particular problem is the lack of research regarding the definition of good design. What few contributions there are reveal an incomplete understanding of quality aspects. Business process modeling as a sub-discipline of BPM faces a particular problem in that modelers who have little background in formal methods often design models without understanding the full implications of their specification (see [336]). As a consequence, process models designed on a business level can rarely be reused on an execution level since they often suffer from formal errors such as deadlocks. Formal errors can, however, be identified algorithmically with verification techniques. In contrast, inconsistencies between the real-world business process and the process model can only be detected by talking to stakeholders. The focus of this book will be on formal errors. Since the costs of errors increase exponentially over the development life cycle [306], it is of paramount importance that errors are discovered as early as possible. A large amount of work has been conducted in an attempt to resolve this weak understanding by providing formal verification techniques, simulation tools, and animation concepts. Several of these approaches cannot be applied, however, if the business process modeling language in use is not specified appropriately. Furthermore, this research area does not address

J. Mendling: Metrics for Process Models, LNBIP 6, pp. 1–15, 2008.

the root of the problem: as long as we do not understand why people introduce errors in a process model, we will never be able to improve the design process.

This chapter provides an overview of business process management and business process modeling. Section 1.1 elaborates on the background of business process management through a historical classification of seminal work. Section 1.2 defines business process management and illustrates the business process management life cycle. Section 1.3 discusses modeling from a general information systems point of view and derives a definition for business process modeling. Section 1.4 distinguishes between formal verification and external validation of business process models and emphasizes the need to understand why formal errors are introduced in business process models. Finally, Section 1.5 concludes the chapter with a summary.

1.1 History of Business Process Management

In the last couple of years, there has been a growing interest in business process management from industry as well as from business administration and information systems research. In essence, business process management deals with the efficient coordination of business activities within and between companies. As such, it can be related to several seminal works on economics and business administration. *Henri Fayol*, one of the founders of modern organization theory, recommended a *subdivision of labor* in order to increase productivity [122, p.20]. *Adam Smith* had already illustrated its potential benefits by analyzing pin production [406]. Subdivision of labor, however, requires *coordination* between subtasks. Business process management is concerned with coordination mechanisms in order to leverage the efficient creation of goods and services in a production system based on such subdivision of labor. The *individual tasks* and the *coordination between them* are, therefore, subject to optimization efforts. *Frederick Taylor* advocated the creation of an optimal work environment based on scientific methods to leverage the most efficient way of performing individual work steps. In the optimization of each step, he proposed to "select the quickest way", to "eliminate all false movements, slow movements, and useless movements" and to "collect into one series the quickest and best movements" [421, p.61]. The efficient coordination of business processes is demonstrated by the innovation of the assembly line system: its inventor *Henry Ford* proudly praised the production cycle of only 81 hours "from the mine to the finished machine" in his factories to illustrate the efficiency of the concept [130, p.105].

In academia, *Nordsieck* was one of the first to distinguish between structural and process organization [321, 322]. He described several types of workflow diagrams for things such as subdivision and distribution of labor, sequencing of activities, or task assignment [321]. In his work, *Nordsieck* identifies the order of work steps and the temporal relationship of tasks as the subject of process analysis with the overall goal of integrating these steps [322] and distinguishes between five levels of automation: free course of work, contents bound course of work, order bound course of work, temporally bound course of work, and beat bound course of work [322].

The decades after World War II saw a discussion about the potential of information systems to facilitate automation of office work [246, 80]. In the 1950s, these ideas seemed still quite visionary [311]. Later, in the early 1970s, it became apparent that information systems would indeed become a new design dimension in an organizational setting (see [145, 166, 148]), but research of the time mainly focused on the structural aspects (such as the relational data model [82] and query languages that later evolved to SQL [29, 30, 76]) without paying much attention to behavioral aspects such as processes. At that time, the logic of business processes used to be hard-coded into applications and were, therefore, difficult to change [189, 310]. Prototypes for office automation during the late 1970s were the starting point for a more explicit control over the flow of information and the coordination of tasks. The basic idea was to build electronic forms for clerical work that was normally handled via paper. In his doctoral thesis, *Zisman* [472, 471] used Petri nets [335] to specify the clerical work steps of an office agent and introduced a respective prototype system called SCOOP. A comparable approach was presented by *Ellis* [117], who modelled office procedures as Information Control Nets, a special kind of Petri nets consisting of activities, precedence constraints, and information repositories. An overview of further work on office automation is provided in [118].

Although the business importance of processes received some attention in the 1980s [338] and new innovations were introduced in information system support of processes (e.g. system support for communication processes [455] based on speech act theory introduced by [34, 390]), it was only in the early 1990s that workflow management prevailed as a new technology to support business processes. An increasing number of commercial vendors of workflow management systems benefited from new business administration concepts and ideas such as process innovation [95] and business process reengineering [164]. On the other hand, these business programs relied heavily on information system technology, especially workflow systems, in order to establish new and more efficient ways of doing business. In the 1990s, the application of workflow systems, in particular those supporting information systems integration processes, profited from open communication standards and distributed systems technology that both contributed to interoperability with other systems [139]. The Workflow Management Coalition (WfMC), founded in 1993, is of special importance for this improvement [185]. The historical overview of office automation and workflow systems given in [310, p.93] illustrates this breakthrough nicely. This period also saw an increase in scientific publications on workflow technology and process specification (see [119, 139, 75, 196, 386, 327, 326, 346, 3, 453, 248]) and intra-enterprise processes remained the major focus of business process management through until the end of the 1990s [97].

Since the advent of the eXtended Markup Language (XML) and web services technology, application scenarios for business process integration have become much easier to implement in an inter-enterprise setting. Current standardization efforts mainly address interoperability issues related to such scenarios (see [292, 280, 277]). The common industry interest in facilitating the integration of interorganizational processes leverages the specification of standards for web service composition (e.g. the Business Process Execution Language for Web Services

(BPEL) [91, 26, 24]), for web service choreography (the Web Service Choreography Description Language (WS-CDL) [209]), or for interorganizational processes based on ebXML and related standards (see [183] for an overview). The integration of composition and choreography languages is currently one of the main research topics in this area [270, 451].

Today business process management is an important research area that combines insights from business administration, organization theory, computer science and computer supported cooperative work. It remains a considerable market for software vendors, IT service providers and business consultants.

1.2 Definition of Business Process Management

Since the beginning of organization theory, several definitions for business processes have been proposed. As early as the 1930s, *Nordsieck* described a business process as a sequence of activities producing an output. By this definition an activity is the smallest separable unit of work performed by a work subject [322, pp.27-29]. Along these lines *Becker and Kugeler* [48] propose the following definition:

> "A process is a completely closed, timely and logical sequence of activities which are required to work on a process-oriented business object. Such a process-oriented object can be, for example, an invoice, a purchase order or a specimen. A business process is a special process that is directed by the business objectives of a company and by the business environment. Essential features of a business process are interfaces to the business partners of the company (e.g. customers, suppliers)."

As *Davenport* puts it [95, p.5], a "process is thus a specific ordering of work activities across time and place, with a beginning, an end, and clearly identified inputs and outputs: a structure for action." *Van der Aalst and Van Hee* add that the order of the activities is determined by a set of conditions [9, p.4]. It is important to distinguish here between the business process and several individual cases. Consider a business process such as car production. This process produces cars as an output. The production of one individual car that is sold to customer John Smith is one case. Accordingly, each case can be distinguished from other cases and a business process can be regarded as a class of similar cases [9].

Several categorization schemes were proposed in relation to business processes and information systems support. As an extension of *Porter*'s value chain model (see [338]), *Van der Aalst and Van Hee* distinguish between production, support, and managerial processes [9, p.9]. *Production processes* create products and services of a company that are sold to customers. These processes are of paramount importance as they generate income for the company. *Support processes* establish an environment in which the production processes go smoothly. Therefore, they do not only include maintenance activities, but also marketing and finance. *Managerial processes* direct and coordinate production and support processes. They are primarily concerned with defining goals, preconditions and constraints for the other processes. *Leymann and*

Roller provide a classification scheme[1] for processes based on their *business value* and their *degree of repetition* [248]. They use the term *"production process"* to refer to those processes that have both a high business value and a high degree of repetition. *Administrative processes* are also highly repetitive but of little business value. Furthermore, *collaborative processes* are highly valuable but hardly repeatable. Finally, *ad hoc processes* are neither repetitive nor valuable. *Leymann and Roller* conclude that information systems support should focus on production processes. In particular, workflow management systems are discussed as a suitable tool. Further definitions and classifications can be found, for example, in [264, 251, 114].

Business process management can be defined as the set of all management activities related to business processes. In essence, the management activities related to business processes can be idealistically arranged in a life cycle. Business process management life cycle models have been described in [9, 310, 114]. In the remainder of this section, we mainly follow the life cycle proposed in [310, pp.82-87] because it not only includes activities but also artifacts, and because it consolidates the life cycle models for business process management reported in [176, 134, 420, 317]. This life cycle shares the activities analysis, design and implementation with the general process of information systems development identified by [448]. The life cycle comprises the management activities of analysis, design, implementation, enactment, monitoring and evaluation. The solid arcs represent the typical order of these activities (see Figure 1.1). Organizations differ in the level of sophistication in which they support these phases and the smooth transition between them. A related model of business process management maturity is discussed in [363].

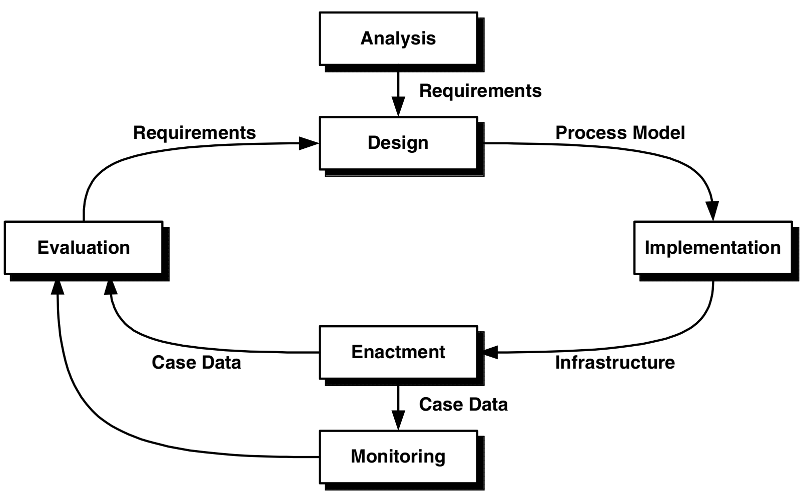

Figure 1.1. Business process management life cycle

[1] The authors refer to the GIGA group who originally introduced the scheme.

Analysis: The business process management life cycle begins with an analysis activity (see Figure 1.1). This analysis covers both the environment of the process and the organization structure. The output of this step is a set of requirements for the business process such as performance goals or intentions [352].

Design: These requirements drive the subsequent design activity. The design includes the identification of process activities, the definition of their order, the assignment of resources to activities and the definition of the organization structure. These different aspects of process design are typically formalized as a business process model [93, 139, 381, 9]. This model can be tested in a simulation if it meets the design requirements.[2]

Implementation: The process model is then taken as input for implementation. In this phase, the infrastructure for the business process is set up. This includes training of staff, provision of a dedicated work infrastructure or the technical implementation and configuration of software. If the process execution is to be supported by dedicated information systems, the process model is used as a blueprint for the implementation.

Enactment: As soon as the implementation is completed, the actual enactment of the process can begin. In this phase the dedicated infrastructure is used to handle individual cases covered by the business process. The enactment produces information such as consumption of time, resources and materials for each handled case. This data can be used as input for two subsequent activities: monitoring and evaluation.

Monitoring is a continuous activity that is performed with respect to each individual case. Depending on process metrics, for instance maximum waiting time for a certain process activity, monitoring triggers respective counteractions if such a metric indicates a problematic situation.

Evaluation, on the other hand, considers case data on an aggregated level. The performance results are compared with the original requirements and sources of further improvement are discussed. Evaluation thus leads to new requirements that are taken as input in the next turn of the business process management life cycle.

The business process management life cycle reveals that business process models play an important role in the design, implementation and enactment phases, especially when information systems support the process enactment. As a result, they are valuable resources for continuous process improvement, quality management, compliance management, knowledge management, end-user training, ERP system selection, and software implementation [165, 93, 95, 159, 359]. Current market research supports this relevance: approximately 90% of participating companies in a survey conducted or considered business process modeling [333]. This trend is partially motivated by new legislation including the Basel II recommendations on banking laws

[2] Note that zur Muehlen considers simulation as a separate activity related to evaluation [310, p.86] but this view neglects the fact that simulation is always done to evaluate different design alternatives.

and regulations and the Sarbanes-Oxley Act in the United States. In practice, software tools play a decisive role in performing the various management activities in an efficient and effective manner. There are several commercial and academic tools which support different life cycle activities (see [9, Ch.5]). The Workflow Management Coalition has proposed 5 interfaces in a reference model in order to link these tools [184]. The availability of tools is critical to the modeling of business processes in a correct and consistent way.

1.3 Definition of Business Process Modeling

Before defining business process modeling, we need to discuss the term "modeling" in a more general manner. *Nordsieck* has emphasized that "the utilization of symbols enables the model not only to replace or to complement natural language for the representation of complex matters, but to reveal the notion of the subject matter often in a more comprehensive way as with any other form of representation" [321, p.3]. The most important features of a model are brevity, clarity, precision and its graphic quality [321, p.3]. *Stachowiak* defines a model as the result of a simplifying mapping from reality that serves a specific purpose [414]. According to this definition, there are three important qualities a model should possess: First, a mapping that establishes a representation of natural or artificial originals that can be models itself; second, only those attributes of the original that are considered relevant are mapped to the model while the rest are skipped. Therefore, the model provides an abstraction in terms of a homomorphism in a mathematical sense [232]. Finally, the model is used by the modeler in place of the original at a certain point in time and for a certain purpose. This means that a model always involves pragmatics [343].

A weakness of *Stachowiak*'s concept of a model is that it implies an epistemological position of positivism.[3] This is criticized in [388], where the authors propose an alternative position based on insights from critical realism and constructivism.[4] This position regards a model as a "result of a construct done by a modeler" [388, p.243]. As such, it is heavily influenced by the subjective perception of the modeler. This makes modeling a non-deterministic task (see [293]) that requires standards in order to achieve a certain level of inter-subjectivity. The Guidelines of Modeling (GoM) [50, 388, 51] define principles that serve this standardization purpose. They are applicable for either epistemological positions or positivism and constructivism because both the choice for a certain homomorphism (positivist position) and the perception of the modeler (constructivist position) introduce subjective elements.

The Guidelines of Modeling (GoM) [50, 388] include six particular principles for achieving inter-subjectivity of models. The first three define necessary preconditions for the quality of models (correctness, relevance, and economic efficiency) and the other three are optional (clarity, comparability, and systematic design).

[3] Positivism is the philosophical theory that establishes sensual experience as the single object of human knowledge.

[4] In contrast to positivism, constructivism regards all knowledge as constructed. Therefore, there is nothing like objective knowledge or reality.

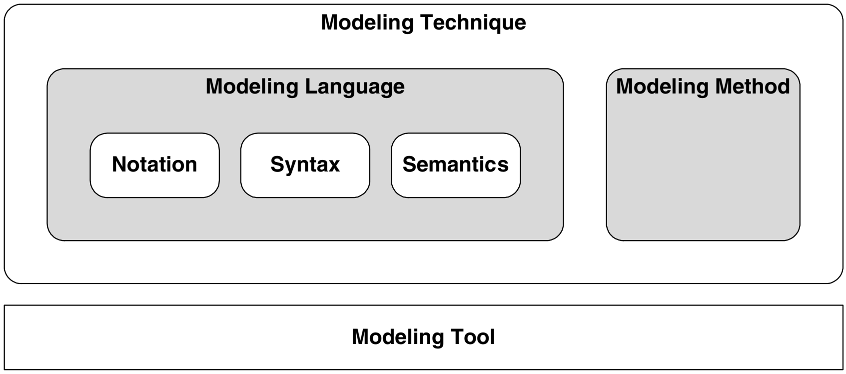

Figure 1.2. Concepts of a modeling technique

Correctness: A model must be syntactically correct. This requirement demands the usage of allowed modeling primitives and their combination according to predefined rules. A model must also be semantically correct. It must, therefore, be formally correct and consistent with (the perception of) the real world.

Relevance: This criterion demands that only interesting parts of the universe of discourse are reflected in the model. It is, therefore, related to the notion of completeness as proposed in [46].

Economic Efficiency: This guideline introduces a trade-off between benefits and costs of putting the other criteria into practice. For example, semantic correctness might be neglected to a certain extent if achieving it is prohibitively expensive.

Clarity: This is a highly subjective guideline demanding that the model must be understood by the model user. It is primarily related to layout conventions or the complexity of the model.

Comparability demands consistent utilization of a set of guidelines in a modeling project. It refers to naming conventions amongst other things.

Systematic Design: This guideline demands a clear separation between models in different views (e.g. statical aspects and behavioral aspects) and defined mechanisms to integrate them.

Following this line of argument, the explicit definition of a modeling technique appears to be a useful means to address several of these guidelines. A *modeling technique* consists of two interrelated parts: a modeling language and a modeling method[5] (see Figure 1.2). The *modeling language* consists of three parts: syntax, semantics and, optionally, at least one notation. The *syntax* provides a set of constructs and a set of rules how these constructs can be combined. A synonym is

[5] Several authors use heterogeneous terminology to refer to modeling techniques. Our concept of a modeling language is similar to grammar in [448, 449, 450] who also use the term method with the same meaning. In [207], a modeling method is called "procedure" while the term "method" is used to define a composition of modeling technique plus related algorithms.

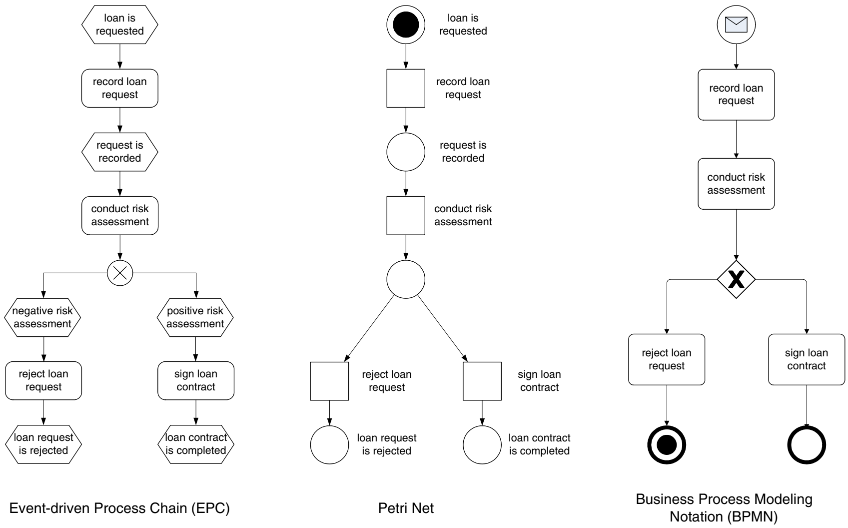

Figure 1.3. Examples of process models in different modeling languages

modeling grammar [448, 449, 450]. *Semantics* bind the constructs defined in the syntax to a meaning. This can be done in a mathematical way, for example by using formal ontologies or operational semantics. The *notation* defines a set of graphical symbols that are utilized for the visualization of models [207]. As an example, Figure 1.3 shows the same loan approval business process in different modeling notations: namely Event-driven Process Chains (EPCs), Petri nets and Business Process Modeling Notation (BPMN). The *modeling method* defines procedures by which a modeling language can be used [450]. The result of applying the modeling method is a model that complies with a specific modeling language[6]. Consider entity-relationship diagrams (ERDs) as defined in [77]. Since they define a modeling language and a respective modeling method, ERDs are a modeling technique. Entities and Relationships are syntax elements of its language. They are used to capture certain semantics of a universe of discourse. The notation represents entities as rectangles and relationships as arcs connecting such rectangles and carrying a diamond in the middle. Respective procedures, like looking for nouns and verbs in documents, define the modeling method. In practice, *modeling tools* are of crucial importance for the application of a modeling technique: they support the specification of models, the redundancy controlled administration of models, multi-user collaboration and model reuse via interfaces to other tools [359]. A recent comparison of business process modeling tools is reported in [25].

There are different approaches to providing a foundation for the correctness and relevance of what is to be put into a process model (see [398]). The following paragraph sketches ontology, speech act theory, the workflow patterns, and metamodeling as four alternative foundations. These four approaches are chosen as examples for

[6] Instead of model, *Wand and Weber* use the term “script” (cf. [448, 449, 450]).

their wide-spread application in information systems research. Further foundations and evaluation techniques for modeling languages are discussed in [398].

- *Ontology* is the study of being. It seeks to describe what is in the world in terms of entities, categories and relationships. It is a prominent sub-discipline of philosophy. *Wand and Weber* were among the first to adopt ontology for a foundation of information systems modeling (see [448, 449]). They make two basic assumptions: as information systems reflect what is in the real world they should also be modelled with a language that is capable of representing real-world entities; and that the ontology proposed by *Bunge* [66] is a useful basis for describing the real world. The so-called *Bunge-Wand-Weber* (BWW) model proposed by *Wand and Weber* includes a set of representation constructs that are deemed necessary and sufficient for describing real-world things including their properties and behavior. These constructs should thus be available for modeling a specific domain and fulfilling certain consistency criteria [449]. An overview of applications of the BWW model is given in [345]. For examples of other ontological models refer to [450, 158]. Recently, ontology languages such as OWL [263] have become popular for defining domain ontologies to be used as a component of the semantic web [54].
- *Speech act theory* is a philosophy of language first proposed by *Austin* [34] and subsequently refined by *Searle* [390]. It emphasizes that language is not only used to make statements about the world that are true or false but also utilized to do something. A priest, for example, performs a speech act when he pronounces a couple husband and wife. The language action perspective has extended this view after determining that speech acts do not appear in isolation, but that they are frequently part of a larger conversation [455]. *Johannesson* uses this insight to provide a foundation for information systems modeling based on conversations built from speech acts [200]. Coming from the identification of such conversations, *Johannesson* derives consistent structural and behavioral models. Both the foundations in ontology and in speech act theory have in common that they imply two levels of modeling: a general level that is based on abstract entities that the respective theory or philosophy identifies, and a concrete level where the modeler identifies instances of these abstract entities in his modeling domain.
- *Workflow Patterns:* Business process models capture different aspects such as activities, control flow, organizational entities, functional goals and information consumed and generated by activities [93, 461, 107, 208]. The heterogeneity of business process modeling languages (see [292]) has motivated research into generic patterns that need to be described in a model. The work by *Van der Aalst, Ter Hofstede, et al.* identifies different patterns for control flow [12, 368], data [371], resources [370], exception handling [369] and instantiation [98]. These patterns have been used in various evaluations of process modeling languages. For an overview refer to [367].
- *Metamodeling* frees modeling from philosophical assumptions by extending the subject of the modeling process to the general level. The philosophical theory of this level, such as an ontology, is replaced by a metamodel. The difference to

an ontological foundation is that a metamodel does not claim any epistemological validity. Essentially, the metamodel identifies the abstract entities that can be used in the process of designing models. In other words, the metamodel represents the modeling language (see [31, 207, 232]). The flexibility gained from this meta-principle comes at the cost of relativism; as a metamodel is meta relative to a model, it is a model itself. Therefore a metamodel can also be defined for the metamodel and it is called metametamodel. This regression can be continued ad infinitum without ever reaching an epistemological ground.[7] Most modeling frameworks define three or four modeling levels (see UML's Meta Object Facility [329], CASE Data Interchange Format (CDIF) [129] or Graph Exchange Language (GXL) [456]). The definition of a modeling language based on a metamodel is more often used than the explicit reference to a philosophical position. Examples of metamodeling can be found in [332, 331, 129, 378, 380, 32, 31, 33]. Several tools like MetaEdit [410, 212], Protegé [323] or ADONIS [203] support metamodeling in such a way that modeling languages can be easily defined by the user. For the application of the meta principle in other contexts refer to [315, 419].

The meta-hierarchy provides a means to distinguish different kinds of models. A model can never be a metamodel by itself, however; it can only be relative to the model for which it defines the modeling language. Models can also be distinguished depending on the mapping mechanism [419, p.21]: *Non-linguistic* models capture some real-world aspects as material artifacts or as pictures. *Linguistic* models can be representational, verbal, logistic or mathematical. Focusing on business administration, *Kosiol* distinguishes descriptive models, explanatory models and decision models [226]. Descriptive models capture objects of a certain area of discourse and represent them in a structured way. Beyond that, explanatory models define dependency relationships between nomological hypotheses. These serve as empirically valid general laws to explain real-world phenomena. Finally, decision models support the deduction of actions: this involves the availability of a description model to formalize the setting of the decision, a set of goals that constraint the design situation and a set of decision parameters.

The terms *business process model*, *business process modeling language*, and *business process modeling* can thus be defined as follows:

- A *business process model* is the result of mapping a business process. This business process can be either a real-world business process as perceived by a modeler or a conceptualized business process.
- *Business process modeling* is the human activity of creating a business process model. Business process modeling involves an abstraction from the real-world business process because it serves a certain modeling purpose. Therefore, only those aspects relevant to the modeling purpose are included in the process model.

[7] This negation of a theoretical foundation of a modeling language has some similarities with approaches that emphasize that models are not mappings from the real world but products of negotiations between different stakeholders, as in [181, 402].

- *Business process modeling languages* guide the procedure of business process modeling by offering a predefined set of elements and relationships for business processes. A business process modeling language can be specified using a meta-model. In conjunction with a respective method, it establishes a business process modeling technique.

This definition requires some explanation. In contrast to [414], it does not claim that the business process model is an abstraction and serves a purpose. These attributions involve some problems about whether a model always has to be abstract or to serve a purpose. Instead, the procedure of business process modeling is characterized in such a way that it is guided by abstraction and a purpose in mind. This is important as a model is not just a "representation of a real-world system" (as *Wand and Weber* put it [448, p.123]), but a design artifact in the sense of *Hevner et al.* [180] that itself becomes part of the real world as soon as it is created. Beyond this, business process models can be characterized as linguistic models that are mainly representational and mathematical. The representational aspect points to the visual notation of a business process modeling language, while the mathematical notion refers to the formal syntax and semantics. In practice, business process models are often used for documentation purposes [96]. They can, therefore, be regarded as descriptive models for organization and information systems engineers. They also serve as explanatory and decision models for the people who are involved in the actual processing of cases. In this book, the focus is on the descriptive nature of business process models.

1.4 Business Process Modeling and Errors

It is a fundamental insight of software engineering that design errors should be detected as early as possible (see [60, 450, 306]). The later that errors are detected, the more work must be redone and the more design effort has been wasted. This also holds for the consecutive steps of analysis, design, and implementation in the business process management life cycle (see [360, 361, 336]). In the design phase, process models are typically created with semi-formal business process modeling languages while formal executable models are needed for the implementation. This problem is often referred to as the gap between business process design and implementation phase (see [312]). Therefore, the Guidelines of Process Modeling stress correctness as the most important quality attribute of business process models [51].

In order to provide a better understanding of potential errors in business process models, it is proposed to adapt the information modeling process as identified by *Frederiks and Van der Weide* [132]. This process can also serve as a framework for discussing business process modeling in the analysis and design phase of the business process management life cycle. Furthermore, it covers several steps to provide quality assurance in the modeling phase which is of paramount importance for the success of modeling projects (see [360, 361]). Figure 1.4 gives a business process modeling process mainly inspired by [132] and consisting of eight steps. In accordance with *Van Hee et al.* [174], it is proposed to first verify the process model (Step 6) before validating it (Step 7-8).

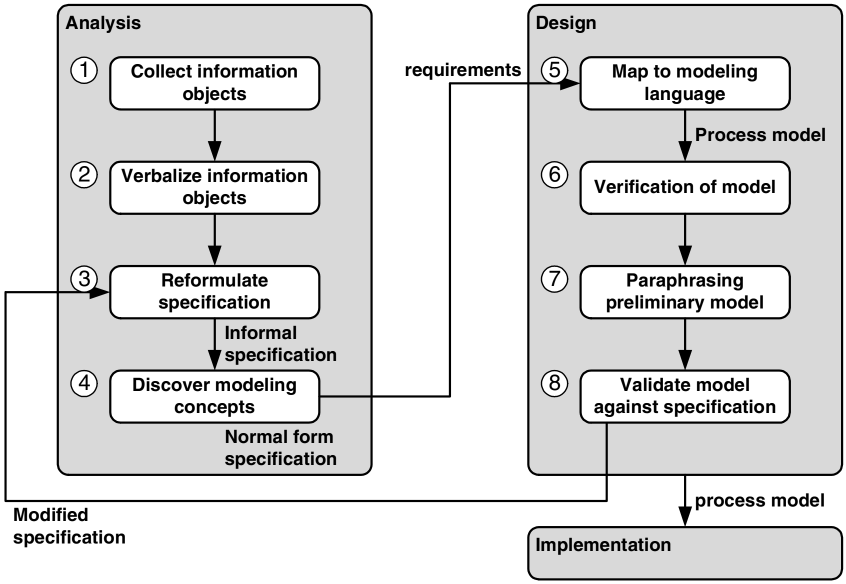

Figure 1.4. Business process modeling process in detail, adapted from [132].

The business process modeling process starts with collecting information objects relevant to the domain (*Step 1*). Such information objects include documents, diagrams, pictures and interview recordings. In *Step 2*, these different inputs are verbalized to text that serves as a unifying format. This text is rearranged according to some general guideline of how to express facts (*Step 3*) yielding an informal specification. The following step (*Step 4*) takes this informal specification as a basis to discover modeling concepts from and to produce a normalized specification. This normal form specification is then mapped to constructs of the process modeling language (*Step 5*) in order to create a business process model. These models have to be verified for internal correctness (*Step 6*) before they can be translated back to natural language (*Step 7*) in order to validate them against the specification (*Step 8*). In Steps 6-8 the order of activities follows the proposal of *Van Hee et al.* [174]. It is a good idea to first verify the internal correctness of a model before validating it against the specification, as this prevents incorrect models from being unnecessarily validated.

The business process modeling process points to two categories of potential errors based on the distinction of verification and validation. This distinction follows the terminology of the Petri nets community (see *Valmari* [436, pp.444]), the conceptual modeling community (see *Hoppenbrouwers, Proper, and Van der Weide* [187]) and the software engineering community (see *Boehm* [59], *Sommerville* [413]). Different terms for similar concepts are used in *Soffer and Wand* [412].

- *Verification* addresses both the general properties of a model and the satisfaction of a given formula by a model. Related to the first aspect, formal correctness criteria play an important role in process modeling. Several criteria have been proposed including soundness for Workflow nets [2], relaxed soundness [101] or well-structuredness (see [102] for a comparison). The second aspect is the subject of model checking and involves issues like separation of duty constraints, which can be verified, for example, by using linear temporal logic (LTL) (see [337]).
- *Validation* addresses the consistency of the model within the universe of discourse. As it is an external correctness criterion, it is more difficult and more ambiguous to decide. While verification typically relies on an algorithmic analysis of the process model, validation requires the consultation of the specification and discussion with business process stakeholders. SEQUAL can be used as a conceptual framework to validate different quality aspects of a model [250, 228].

In this book, we will refer to *formal errors* in connection with the internal correctness of business process models. Formal errors can be identified via verification. Furthermore, we use the term *inconsistencies* to refer to a mismatch of model and specification. Inconsistencies are identified by validation. Generally speaking, *error detection* is related to both verification and validation [436, p.445]. We also focus on *error detection related to verification* and, in particular, to the question which combination of model elements affects the verification of a correctness criterion for a business process model.

While there has been empirical work on different aspects of conceptual modeling [399, 39, 256, 138], little such work has been conducted on formal errors of business process models in practice. One reason for this is that large repositories of business process models capture specific and valuable real-world business knowledge of industrial or consulting companies. Confidentiality concerns present a serious problem for academia since practical modeling experience can hardly be reflected in a purely theoretical way. *Thomas* [422] calls this the "dilemma" of modeling research. One case of a model that is, at least partially, publicly available is the SAP Reference Model. It has been described in [92, 211] and is referred to in many research papers (see [127, 235, 281, 362, 427]). The extensive database of this reference model contains almost 10,000 sub-models, 604 of them being non-trivial EPCs [92, 211]. The verification of these EPC models has shown that there are several formal errors in the models (see [473, 109, 110, 275]). In [275] the authors identify a lower bound for the number of errors of 34 (5.6%) using the relaxed soundness criterion. In another survey, *Gruhn and Laue* [154] analyze a collection of 285 EPCs mainly taken from master theses and scientific publications. From these 285 models 30% had trivial errors and another 7% had non-trivial errors. These first contributions highlight that errors are indeed an issue in business process models.

1.5 Summary

In this chapter, we discussed the backgrounds of business process management and defined important terms related to it. We also sketched the importance of business process modeling for the business process management life cycle. Since process models are created in the early design phase, they should be free from errors in order to avoid expensive rework and iterations in subsequent phases. In the following chapters, we concentrate on Event-driven Process Chains (EPCs) which are frequently used for business process modeling. Based on a formal semantics definition, we identify verification techniques to detect errors.