

# Chapter 5

## Process Discovery



*All truths are easy to understand once they are discovered;  
the point is to discover them.*  
Galileo Galilei (1564–1642)

In the previous two chapters we learned how to create process models. However, our starting point was often a textual description of the process, which is hardly available in practice, at least the first time a model is created for a given business process. There are various methods that we can use to create process models by inferring information about the business processes within an organization, e.g., by interviewing process participants or by observing how they operate in practice. By the same token, it is important to ensure that a model is not only syntactically correct, but that it also accurately reflects the actual business process being modeled. To this end, we need to thoroughly understand the operations of a business process, as well as to possess the modeling skills to represent a business process in a high-quality BPMN model. These two types of skills are hardly unified in the same person. Hence, multiple stakeholders with different and complementary skills are typically involved in the construction of a process model.

In this chapter, we first present the challenges faced by the stakeholders involved in the lead-up to a process model. Then, we discuss methods to facilitate effective communication and information gathering in this setting. Given the information gathered in this way, we show step-by-step how to construct a process model and what quality criteria should be checked before the model can be accepted as an authoritative representation of a business process.

### 5.1 The Setting of Process Discovery

Process discovery is defined as the act of gathering information about an existing process and organizing it in terms of an as-is process model. This definition emphasizes gathering and organizing information. This means that process discovery is

a much broader activity than process modeling. Clearly, modeling is a part of this activity. The problem is though that modeling can only start once enough information has been put together. And indeed, gathering information often proves to be cumbersome and time-consuming in practice. Therefore, we need to first define a setting in which information can be gathered effectively. In order to address these issues, we can describe four tasks of process discovery:

1. Defining the setting: dedicated to assembling a team in a company that will be responsible for working on the process.
2. Gathering information: concerned with building an understanding of the process. Different discovery methods can be used to acquire information on a process.
3. Conducting the modeling task: deals with the actual creation of the process model. A modeling method gives guidance for mapping out the process in a systematic way.
4. Assuring process model quality: aims to guarantee that the resulting process model meets different quality criteria. This task is important for establishing trust in the process model.

Once the setting has been defined, the remaining three tasks are often performed in an iterative manner, i.e., as we gather information about a given process, we create a draft model and assure that this is of good quality. In the following, we discuss the key roles involved in process discovery.

### ***5.1.1 Process Analyst Versus Domain Expert***

Two roles are fundamental in a process discovery project: the *process analyst* and the *domain expert*. One or more process analysts are commonly responsible for gathering information about a given business process, and driving the modeling task, under the leadership of the process owner. As such, a process analyst must be familiar with process modeling languages such as BPMN and be skilled at gathering and organizing process-related information. However, process analysts are hardly knowledgeable of all the details of the process in question.

*Example 5.1* Let us consider the following two modeling tasks:

- Modeling the process for ordering books through an online bookstore, from the perspective of the customer.
- Modeling the same process from the perspective of the bookstore.

If you have already learned how to model business processes by the help of this book, you should be able to complete the first modeling task above. The reason is that quite likely you will be familiar with this process as you have already ordered a book online, through your preferred bookstore. The case is likely to be different for the second modeling task: you will only be able to complete this task if you have worked for an online bookstore, which is less common. □

Ultimately, the participation in a business process from behind the scenes, i.e., from the perspective of the company that delivers a service or produces a product via a given process, is what determines whether or not we are intimately familiar with that process. In practice, process analysts are supposed to model business processes which they have experienced neither as process participants nor as customers. So they have to gather an extensive amount of information about the process in order to understand how it works from the inside, by consulting with those who are involved in its performance on a daily basis, i.e., the domain experts.

A domain expert is thus any individual who has an intimate knowledge of how a process or specific tasks within that process are performed. Typically, this is a process participant, but it can also be the process owner or an operational manager who coordinates a team of process participants. External roles such as partners, suppliers, and customers of the process should also be consulted as domain experts, since they can offer a complementary view on the same process, though their knowledge of the process would undoubtedly be confined to their limited exposure to it. On the downside, domain experts are not proficient in process modeling. In some companies, domain experts even refuse to discuss process models, because they do not feel comfortable explaining their involvement in the process before a process model. As a consequence, they often rely on process analysts for organizing their process knowledge in terms of a process model.

Such difference in modeling skills between process analysts and domain experts results from a different exposure to practical modeling and to modeling training. Many companies use training programs to improve the modeling skills of domain experts. Such training is a prerequisite for modeling initiatives where process participants are expected to model their own processes. On the other hand, there are BPM consultancy companies that specialize in particular industry domains such as auditing, finance, or mining. It is an advantage when BPM consultants who also have domain expertise can be assigned to process modeling projects.

It is the task of the *process owner* to secure the commitment and involvement of both analysts and domain experts during the definition of the setting of process discovery. The number and type of process analysts and domain experts to involve depends on the complexity of the process in question. In the rest of this section we will elaborate on this, starting with the three challenges of process discovery.

**Exercise 5.1** You are the manager of a consulting company and you need to hire a person for the newly signed BPM project with an online bookstore. Consider the following two profiles; who would you hire as a process analyst?

- Mike Miller has ten years of work experience with an online retailer. He has worked in different teams involved with the order-to-cash process of the online retailer.
- Sara Smith has five years of experience working as a process analyst in the banking sector. She is familiar with two different process modeling languages and with several modeling tools.

### 5.1.2 *Three Process Discovery Challenges*

The fact that modeling knowledge and domain knowledge are often available in different persons gives rise to three essential challenges of process discovery, namely fragmented process knowledge, thinking in cases, and lack of familiarity with process modeling languages.

The first challenge of process discovery relates to *fragmented process knowledge*. Business processes are a set of related tasks. Nowadays, however, due to specialization and division of labor, hardly all the tasks of a process will be performed by the same resource. Rather, different tasks will be assigned to different specialized resources. This has the consequence that a process analyst must gather information about a given process by talking with different domain experts who are responsible for the various tasks in the process. Typically, domain experts have an abstract understanding of the overall process and a very detailed understanding of their own tasks. This makes it difficult to puzzle the different views together. In particular, one domain expert might have a different idea about what output has to be expected from an upstream task than the person that actually works on it. Potential conflicts in the provided information have to be resolved. It is also often the case that the rules of the process are not explicitly defined in detail. In those situations, domain experts may operate under diverging assumptions, which may not be consistent with each other. Fragmented process knowledge is one of the reasons why process discovery requires several iterations. Having received input from all relevant domain experts, the process analyst must make proposals for resolving inconsistencies, which again requires feedback from, and eventually approval from, the domain experts, before obtaining the final endorsement from the process owner.

The second challenge stems from the fact that domain experts typically *think of processes on a case level*. Domain experts will find it easy to describe the tasks they conducted for one specific process instance, but they might have problems responding to general questions about how a process works in the general way. Process analysts often get answers like “You cannot really generalize, every case is different” or “We can never do anything exactly in the same way, there are so many special conditions to answer such a question”. It is indeed the role of the process analyst to organize and abstract the pieces of information provided by the domain expert in such a way that a systematically defined process model can emerge. Therefore, it is required to formulate questions on specific aspects of the process for the domain experts, e.g., what happens if certain conditions do or do not hold, if a given outcome is achieved, or if certain deadlines are not met. In this way, the process analyst can reverse-engineer the conditions that govern the routing decisions of a business process.

The third challenge of process discovery is a result of the fact that domain experts are typically *not familiar with business process modeling languages*. This observation already gave rise to the distinction between domain experts and process analysts. In this context, the problem is not only that domain experts are hardly trained to create process models themselves, but also that they are not trained to read

process models that others have created. This lack of training can encumber the act of seeking feedback on a draft process model. In this situation, it is typically not appropriate to show the model to the domain expert and ask for corrections. Even if domain experts understand the activity labels well, they would often not understand the routing constructs of a modeling language like BPMN. Therefore, the process analyst has to explain the content of a process model in detail, for example by translating the model to natural language. Domain experts will feel at ease in commenting on natural-language explanations of the process, pointing out aspects that need modification or further clarification according to their understanding of the process.

The box “Profile of an Expert Process Analyst” describes what makes a process analyst an expert.

**Exercise 5.2** Consider the order-to-cash process of your preferred online bookstore and assume you have access to three internal resources: a customer relationship manager (who handles sales and reclaims), a warehouse worker (who looks after shipments), and a financial officer (who raises invoices and collects payments). As a process analyst, what questions do you need to ask these domain experts to be able to obtain a complete and systematic view of this process?

*Hint.* Think of the different exposure to this process that the three resources have and of the possible conditions, process outcomes, and exceptions that they may have experienced while executing this process.

### PROFILE OF AN EXPERT PROCESS ANALYST

The skills of a process analyst play an important role in process discovery. *Expert* process analysts can be described based on a set of general dispositions, their actual behavior in a BPM project, and in terms of the quality of the process models resulting from their efforts.

Research on expertise in the general area of system analysis and design has found that there are certain personal traits that are likely to be observed in expert process analysts. One of the ways to describe personality is the so-called *Five Factor Model* developed in psychological research. In essence, this model describes five psychological factors, namely *openness* (appreciating art, emotion, and adventure), *conscientiousness* (tendency to self-discipline, achievement, and planning), *extraversion* (being positive, energetic, and seeking company), *agreeableness* (being compassionate and cooperative), and *neuroticism* (being anxious, depressed, and vulnerable). These factors have also been studied regarding their connection with expert analysts. Expert analysts appear to be strong both in terms of conscientiousness and extraversion. Indeed, process discovery projects require the conscientious planning and coordination of interviews or workshops with various domain experts in a limited period of time. Further, process discovery projects are sometimes

(continued)

subject to internal politics within the organization, in situations where the agenda of different process stakeholders is not thoroughly clear, where they might fear losing their position, or even where they have conflicting agendas. In such environments, it is valuable to have an energetic and extraverted process analyst who can create a positive atmosphere for working on the project, and seek compromise between the involved parties, conscious of the interests at stake.

Process discovery in general belongs to the category of ill-defined problems. This means that at the beginning of a process discovery project it might not be very clear which domain experts have to be contacted, what documentation should be utilized, and what agenda the different stakeholders might have in mind. The way expert analysts navigate through a project is strongly influenced by their experience with former projects. Thus, there is a strong difference between the way novices conduct problem understanding and solving and the way expert analysts do. In terms of problem understanding, it has been observed that expert analysts approach a project in terms of what things need to be achieved. Novices lack this clear goal orientation and try to approach things in a bottom-up way. This means, they often start by investigating material that is easily accessible and talk to persons that respond readily. Experts work in a different way. They have an explicit set of triggers and heuristics available from experience with prior projects. They tend to pay specific attention to the following aspects:

- *Getting the right people on board.* If you need to talk to a given process participant, make sure their immediate superior is on board and that the process participant knows that their hierarchy backs their involvement in the process discovery effort.
- *Having a set of working hypotheses on how the process is structured at different levels of detail.* In order to progress with the project, it is important to have a short and precise set of working hypotheses, which you can then validate. Prepare an extensive set of questions and assumptions to be discussed with domain experts, e.g., in interviews or workshops.
- *Identifying patterns in the information provided by domain experts.* These can be used for constructing parts of a process model. Such pieces of information typically refer to specific control-flow structures. For instance, statements about certain activities being alternative, exclusive, or subject to certain conditions often point to the use of XOR gateways. In a similar way, statements about activities being independent from one another, or sometimes being in one or another order, often point to the use of AND gateways. It is often easy for expert analysts to sketch out processes by combining these patterns.

(continued)

- *Paying attention to model quality.* Easy-to-understand models help better engage relevant stakeholders and are also valuable for the analyst throughout the creation of the model itself. Experts also use the right level of abstraction. For example, you should not show a very detailed model to a manager. The importance of layout is apparent from the fact that expert analysts often take a great deal of time in arranging the various model elements neatly, so as to render the model more readable.

## 5.2 Process Discovery Methods

As we now have an idea of the tasks process analysts have to perform, of their capabilities, and of what limitations they have to keep in mind when interacting with domain experts, we turn to different methods for gathering information about a process. We distinguish three classes of discovery methods, namely evidence-based discovery, interview-based discovery, and workshop-based discovery. They have relative strengths and weaknesses, which we will discuss subsequently.

### 5.2.1 Evidence-Based Discovery

Various pieces of evidence are typically available for studying how an existing process works. We discuss three evidence-based methods: document analysis, observation, and automated process discovery.

**Document analysis:** Document analysis exploits the fact that there is usually documentation available that can be related to an existing business process. In the ideal scenario, this can take the form of process descriptions, which are available from previous modeling exercises. Other document types include internal policies, organization charts, employment plans, quality certificate reports, glossaries and handbooks, user forms, data and system models, work instructions, and work profiles. However, there are potential issues with document analysis. First, most of the documentation that is available about the operations of a company is not readily organized in a process-oriented way. Think of an organization chart, for instance. It defines the organizational units and positions, and is helpful to identify a potential set of process stakeholders. For example, in case of our online bookstore, it might reveal that the sales department, the logistics department, and the financial department are likely to be involved in the order-to-cash process. Second, the level of granularity of the documentation might not be appropriate. While an organization chart draws rather an abstract picture of a company, there are often many documents that summarize parts of a process on

a too fine-grained level. Many companies document detailed work instructions for tasks, and work profiles for positions. These are typically too detailed for modeling a business process at a conceptual level. So we may need to abstract from, or refine, the information found in these documents, to get the required information to model our business process. Third, many of the documents are only partially trustworthy. For a process discovery project, it is important to identify how a process works in reality. Many documents do not necessarily capture reality. Some of them are outdated and some state how things should work idealistically (normative documents), and not how people actually do their work. The advantage of document analysis is that a process analyst can use the available documentation to get familiar with certain parts of a process and its environment, and also to formulate hypotheses. This method is helpful before talking to domain experts. On the downside, an analyst has to keep in mind that documents do not necessarily reflect the reality of the process.

**Observation:** If we use observation as a method of discovery, we directly follow the processing of individual cases in order to get an understanding of how the process works. As a process analyst, we can either play the active role of a customer of the process or the passive role of an observer. As part of the *active customer role*, we trigger the execution of a process and record the steps that are executed and the set of choices that are offered. For instance, in the case of our online bookstore, we can create a new book order and keep track of which activities are performed at the side of the retailer. This provides a good understanding of the boundaries of the process and its essential milestones. However, we will only see those parts of the process that require interaction with the customer. All back-office processing remains a black box. The role of a *passive observer* is more appropriate for understanding the end-to-end process, but it requires access to the people and sites where the process is being performed. Usually, such access needs the approval of the managers and supervisors of the corresponding teams, and not all sites can be accessed (think of an offshore oil platform). Furthermore, there might be a potential issue with people acting differently, because they are aware of being observed. People usually change their behavior under observation in such a way that they work faster and more diligently. This is important to remember when execution times have to be estimated, and possible exceptions to the normal process flow have to be identified, as part of the modeling task. However, discovery based on observation has the advantage that it reveals how a process is conducted in reality as of today, which is in contrast to document analysis that typically captures the past.

**Automated process discovery:** *Automated process discovery* is a method that uses *event logs*, i.e. process execution data stored by common enterprise systems available in an organization, to automatically discover a model of the business process that is supported by these systems. Think for example of a claims management system in an insurance company. This system does not necessarily have an explicit definition of the claims handling process it supports by means of a process model. With automated process discovery, though, it is possible

to automatically extract a model of this process even if the process is hard-coded in the system. The advantage of this method is threefold. First, event logs capture the actual execution of business processes, so we obtain an objective representation of the process, as opposed to relying on our understanding of it as a result of observing the process directly or consulting with domain experts. Second, event logs often record a rich set of process-related information beyond the tasks that have been performed, including for example task timestamps and the resources that executed these tasks. We can use this input to enrich our models with performance information such as activity durations and frequencies, or by discovering alternative views of the process, e.g., the social network of how process resources interact with each other. Third, this method is not limited by the confines of a given enterprise system, so it can be used to reconstruct end-to-end processes that span different systems, which would be hard to do by accessing each system in isolation. Consider for example the meter-to-cash process of a utility company from the moment a customer's service consumption is measured to the time the customer is billed. This process may be supported by two different systems: an ERP system to handle the measurement and a CRM system for the billing part. By creating a unified event log from process execution data out of these two systems, it would be possible to trace the entire meter-to-cash process, hence going beyond the boundaries of individual systems. A limitation is that event logs are not always available, and when they are, they sometimes only record some tasks in the process and not others (i.e., there are gaps in the log), or contain noise and other logging errors. Also, depending on the granularity in which process information has been logged, the resulting models may be too low-level and so hard to understand. We will introduce concrete techniques for automated process discovery and further discuss advantages and limitations within the process monitoring phase in Section 11.4.

**Exercise 5.3** As a process analyst working for the University of Newtown, you have been engaged by Mark Johnson, the process owner of the student admission process, in a project that aims at improving this process. In order to model the as-is process, you start by collecting relevant information about this process. The available documentation includes the organization chart of the Office of the Deputy Vice-Chancellor (DVC)<sup>1</sup> for Student Affairs where Mark's team sits, the UML class diagram of the student admission system which supports this process, and a set of relevant organizational policies that you extract from the university's Web pages. These documents are reported in Figures 5.1, 5.2, 5.3.

Based on this documentation, formulate initial hypotheses on how the student admission process works. Next, identify the relevant domain experts to interview and their supervisors whom you should seek approval from.

---

<sup>1</sup>The Deputy Vice-Chancellor is one of the most senior academic positions at a college or university. Depending on the country, this position is variously called Vice-President, Vice-Rector, or Provost.

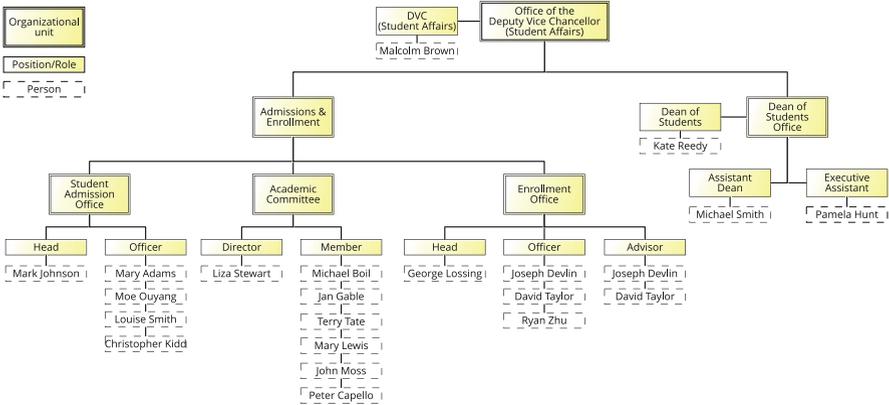


Fig. 5.1 Organization chart of the Office of the DVC (Student Affairs)

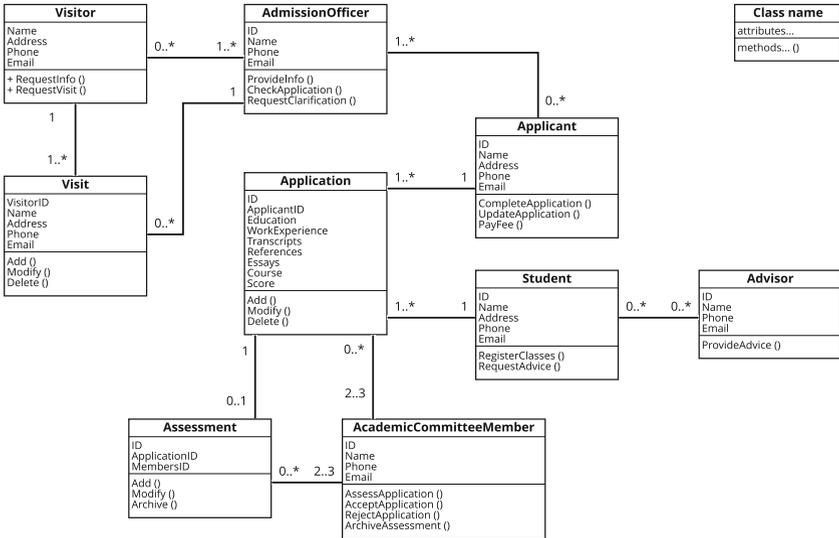


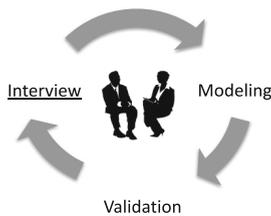
Fig. 5.2 Extract of the UML class diagram of the student admission system

### 5.2.2 Interview-Based Discovery

*Interview-based discovery* aims at interviewing domain experts to inquire about how a process is executed. Figure 5.4 illustrates the typical phases of the interview method. First, those involved in the process of interest are approached for an initial interview. We mentioned that process knowledge tends to be fragmented due to specialization and division of labor (first discovery challenge). For this reason, we need to interview multiple domain experts. Since at the time of the interviews we

- *An applicant is admitted if:*
  - *their prior education is consistent with the study area of the selected course*
  - *the submitted essay is not plagiarized and is of good quality*
  - *the score of the prior degree is at least 70 out of 100 (standard 100-point scale)*
  - *the two reference letters are satisfactory.*
- *Successful applicants must accept the offer within four weeks from notification.*

**Fig. 5.3** Organizational policies for student admission



**Fig. 5.4** Phases of the interview method

may not yet grasp the details of the involvement of different domain experts in the process, it may be required to discover the process step-by-step, and as we learn the latter, plan interviews with additional people.

We can use two strategies for conducting an interview: (i) starting from the process outcomes (e.g., an order being fulfilled), we work our way *backwards* until we reach the process triggers (e.g., the receipt of a purchase order); or (ii) starting from the triggers, we proceed *forward* until we reach the process outcomes. Conducting interviews in a forward manner enables us to elicit process knowledge from the interviewee by naturally following the flow of processing in the order of how it unfolds. This is particularly helpful for understanding which decisions are taken at which stage. Following the process backward can also be helpful. For example, some domain experts may find it easy to identify the possible outcomes of a process or of an activity (e.g., an order fulfilled or rejected), and from that retrieve what is required to get to that outcome by traversing the process backward (e.g., the payment and the delivery notice are needed for an order to be fulfilled). Both strategies, downstream and upstream, are important when interviewing domain experts. With each interviewee, it must be clarified what input is expected from prior upstream activities, what decisions are taken, what is produced as output of their activities, and to what resource it is then forwarded.

When conducting an interview, it is more effective to balance between a structured and a free-form interview approach. For example, considering a 1-h interview, one may spend the first 45 min to go through a list of predefined questions

to validate current hypotheses (structured part), and use the remaining 15 min to let the interviewee discuss any concern or aspect of the process they believe to be relevant (free-form part). *Free-form* interviews have the advantage of enabling domain experts to discuss the process at a level of detail that they find appropriate, which may lead to uncovering certain aspects of the process previously disregarded. *Structured* interviews, in contrast, allow us to validate our hypotheses but may create in the interviewee a feeling of running through a checklist, with the effect of holding back important information one is not explicitly asked about. In fact, a recurrent pitfall is that when asked how a given process or activity is performed, the interviewee tends to describe the normal way of processing. Thus, exceptions tend to be neglected. In other words, the interview ends up covering only the “sunny-day” scenario. One way to prevent this pitfall is to explicitly ask questions about the “rainy-day” scenarios. For example, one may ask: “How did you handle your most difficult customer?”, “What was the most difficult case you worked on?”, “What happens if the customer does not reply on time?”. To formulate these questions, it is handy to think of the possible exceptions that may arise in a process (internal, external, or activity timeouts) and of their nature (business vs. technology fault). This can help to uncover exceptions and more generally process variants that while not necessarily frequent have a sufficient impact on the process to be worth documenting. For example, in an order-to-cash process, one may ask a sales officer what happens if the ordered items are out-of-stock (internal business exception), or if the customer decides to cancel the order (external business exception), or if the ERP system that checks stock levels is unresponsive (internal technology exception).

Coming back to the phases in Figure 5.4, after an initial interview, we can construct a process model offline (second phase), based on our interview notes or recordings. Due to domain experts thinking on a case level (second discovery challenge), we must be able to abstract information on individual cases from the interviewees, in order to construct meaningful process models. Once the model has been created, we need to validate it with the domain experts (third phase) to make sure that it correctly reflects their view (we will talk more about validation later in this chapter). To validate a model, we may need to translate this into natural language, due to domain experts not being familiar with process modeling languages (third discovery challenge). Validation typically leads to the need of interviewing the person again to clarify certain parts of the process. A second iteration of the cycle in Figure 5.4 is often enough to get the approval of the interviewee. However, especially for complex processes, more than two iterations may be required.

In summary, interview-based discovery offers a rich and detailed picture of the process and its participants. Interviewing multiple process participants (also for the same role) has the potential to reveal inconsistent perceptions that different domain experts may have on how a particular process operates. It also helps the process analyst to understand the process in detail. However, it is a labor-intensive discovery method which requires ample time of different individuals.

**Exercise 5.4** After collecting relevant information on the student admission process (see Exercise 5.3 on page 167), you interviewed some representatives for the two roles involved in this process: Mary Adams and Louise Smith as student admission officers, and Peter Capello as a member of the academic committee (Mark Johnson, the process owner, confirmed that the enrollment office is not involved in this process). The relevant parts of the interview transcripts are provided below.

Student admission officer (Mary Adams):

“My process starts when I receive an application for admission. First, I check the completeness of this document. If the application is incomplete, I need to send a request for clarification back to the applicant. Otherwise, I forward it to the academic committee. I then receive a response from the academic committee which can be either of the below:

- A notification of acceptance from the academic committee. In this case, I prepare a letter of offer and send it to the applicant via post to collect his or her signature. Most of the times, I receive a signed offer back from the applicant, but sometimes I don’t.
- A notification of rejection. In this case, I send a rejection letter to the applicant via ordinary post.

The problem is that the academic committee is often too slow to reply. I wonder if these academics are just too overloaded with work to care about student admissions. . . .”

Student admission officer (Louise Smith):

“When I get a fresh application, it’s important that this contains all the required information, including name, address, phone number, and email address of the applicant. Unfortunately, the Web portal has many bugs and sometimes lets through incomplete applications, which are a nightmare to rectify! This means going back and forth with the student at least a couple of times. Anyway, once the application is complete, I pass it to a member of the academic committee using our internal student admission system—the same that collects applications via the Web portal. Most of the times, the member of the academic committee replies with a notification of acceptance, in which case I need to prepare a letter of offer and send it to the applicant via post. Our policies are such that applicants must reply within four weeks. In fact, we are flooded by applications, so if they do not hurry to reply, we will offer the place to someone else. Sometimes I receive a notification of rejection. Well, in this case I formulate a rejection letter and send it to the applicant via post”.

Member of academic committee (Peter Capello):

“When I receive an application from the admission officer, I assess its quality. I extract the grade of the applicant from his or her previous degree and convert it to a standard score based on a conversion table. The score must be at least 70%; otherwise the student is out. Next, I perform a plagiarism check of the essay contained in the application using our plagiarism detection software. Most of the times, the essay is plagiarism-free. If so, I proceed to read it and assign it a score. Finally, I read the two reference letters attached to the application. There’s a lot you can learn from a reference letter. Often there are subtle messages that the referee wants you to get, like “This is a great student, but I’ve had better ones”. In any case, based on the score, quality of the essay, and reference letters, if I deem that the applicant is qualified, I send an acceptance notification to the admission officer, otherwise I send a rejection notification. In either case, I archive the results of my assessment in my database. Ah, I communicate with the student admission office using our internal student admission system. A piece of junk, sometimes messages get lost and I have to send them again, if I’m lucky to find it out!”

Next, you took an active role in observing how this process works by acting as the applicant. Using a fake identity (in agreement with the process owner), you triggered this process several times by submitting various applications via the Web portal. After this, you came up with the following observations.

#### Applicant:

To apply for admission, the applicant needs to prepare an admission application and submit it to the university via a Web portal. The application must include academic transcripts, an essay, and two reference letters. The applicant will then receive a response from an admission officer via ordinary mail, which can be:

- A letter of offer. In this case, the applicant needs to sign the letter of offer and return it to the admission officer via post within four weeks.
- A rejection letter. In this case, the applicant does not do anything further and the process is finished.
- A request for clarification from the admission officer. This is an email notification. In this case, the applicant provides the required documentation to the admission officer by submitting an updated application through the same Web portal used for the initial submission, and then gets a response that is the letter of offer, the rejection letter, or again a request for clarification.

Using the information above, create a draft BPMN model of the as-is student admission process. This draft will then be validated with the people that have been interviewed, before sign-off by the process owner. Make appropriate assumptions.

### 5.2.3 *Workshop-Based Discovery*

*Workshop-based discovery* also offers the opportunity to get a rich understanding of the business process, with the advantage of resolving inconsistent views between domain experts more quickly than interviews. In fact, in contrast to interviews, a workshop involves multiple process participants at the same time. Moreover, it may include two further roles in place of or in addition to the business analyst. A *facilitator* may be invited to coordinate the verbal contributions of the participants, while a *process modeler* may be invited to create the process model as the workshop unfolds. If these two figures are available, the business analyst monitors the workshop and acts as a *scribe* taking note of all relevant concerns that may require further investigation. For example, if a passage of the process may not be clear, but the discussion moves elsewhere, the scribe would record this point to come back to it later, in order not to stop the flow of thoughts. In small contexts, facilitator and modeler may coincide with the process analyst, but in contexts where the business process in question is complex and where a large number of participants are attending, it is advisable to tap into these extra roles, budget allowing.

In terms of effort, an end-to-end detailed process model can hardly be completed within a single workshop session. Typically, three to five sessions of no more than

3 h each are required to complete the modeling effort, including consolidating the model between sessions to ensure a high level of quality.

The involvement of multiple domain experts requires diligent scheduling and preparation. The sessions should be scheduled well in advance to ensure the simultaneous availability of domain experts with different involvement in the business process. This includes at least one representative of each role participating in the process (e.g., a customer relationship manager, a warehouse worker, and a financial officer for the order-to-cash process of our online bookstore example). It is useful to also involve technical staff managing the systems supporting the process, even if these people are not directly involved in the process (e.g., the system administrator of the ERP system used to automatically check stock availability). It is further advisable that the project sponsor (typically the process owner) also participates, at least in the first session, to stress the importance of the project. In any case, there should not be more than ten to twelve domain experts per session, otherwise there will not be enough time for each to take parole. If the process is available in multiple variants (e.g., distributed geographically or per product), it is better to discover each variant in a separate session to avoid confusion between variants. This is also the case if there is a need to create a consolidated as-is process model for all variants, as this consolidated model may be achieved off-line after the various sessions.

At the beginning of the first workshop session the analyst should reset expectations and illustrate the format of the workshop. Participants may have a different understanding of the goals of the workshop, so it is important to clarify objectives (what process should be discovered), importance (how this project contributes to the company's strategy), and scope (how deep the process modeling should go). In the first workshop session it can also be beneficial to take a lightweight yet participative approach to process modeling. One technique to engage workshop participants is to ask them to collectively build a rough model of the process (a sketch) using sticky notes on a wall. The facilitator starts with a pad of sticky notes. Each sticky note is meant to represent a task or event. The group begins with discussing how the process typically starts, i.e., what its possible triggers are and what tasks are performed next. The facilitator then writes the name of the start event on a sticky note and posts it on the wall. Then they ask what can happen next. The participants start mentioning one or more possible tasks. The facilitator writes these down on new sticky notes and posts these on the wall, organizing them for example from left to right or top to bottom to capture the temporal order of tasks. In this exercise, no lines are drawn between the tasks and no gateways are discovered. The purpose is to build a sequence of process tasks. Sometimes, participants disagree on whether something is one or two tasks. If the disagreement cannot be resolved, the two tasks can be written down as two sticky notes bundled together, hence forming a composite activity, e.g., in certain processes tasks "Prepare invoice" and "Post invoice" may be done by the same resource, hence they form a sub-process "Handle invoice". In general, it is important to avoid too much deliberation in order to keep the workshop moving. The facilitator also needs to pay attention to the fact that the tasks being posted are at the same level of granularity. When people start mentioning

micro-steps, e.g., “Put the document on a fax machine”, the facilitator should lift the level of abstraction back to a conceptual process model level. In the end, this exercise leads to a sketch process model that the process analyst can take as input to construct an initial BPMN model after the workshop session. This can be done during the session if a process modeler is available.

At the beginning of the second session, the analyst may provide the participants with a quick introduction to the core set of BPMN elements (start and end events, activities, XOR and AND gateways) in order to show the model that has been prepared as a result of the first workshop session. This model, which can be shown on a whiteboard or directly in a modeling tool through a beamer, can be used to frame the discussion with the aim of validating the current understanding of the business process. It is important, however, not to get lost in the details of the modeling notation to avoid steering attention away from the actual discovery effort.

Workshop-based process discovery requires an organized facilitation and an atmosphere of openness. The facilitator must ensure that parole is balanced between the different participants. This means on the one hand restricting the speech time of talkative participants and, on the other hand, encouraging more introverted participants to express their perspective. Moreover, an atmosphere of openness is indispensable to everybody’s participation.

*Example 5.2* Consider the following two companies.

Company A is young, founded three years ago, and has grown rapidly to a current toll of one hundred employees. Company B is owned by the state and operates in a domain with extensive health and security regulations. How might these different characteristics influence workshop-based discovery?

An atmosphere of openness is influenced by the culture of the company. In organizations with a strongly emphasized hierarchy, ideas and critique may be held back, so it may be difficult for domain experts to express their view openly if their boss is also present. In contrast, if creativity and independent thinking are appreciated, participants are more likely to feel at ease with uttering their ideas and issues, even in the presence of their boss. In our example, it might be the case that the young dynamic company has a more open culture than the company with extensive regulations. This must be taken into account when organizing a workshop. □

It is the responsibility of the analyst to carefully choose the participants depending on the organizational culture. Further, it is the responsibility of the facilitator to always try to stimulate constructive interactions among participants, while remaining neutral when diverging opinions arise. Meantime, while criticism should be allowed, the facilitator should keep negative comments from participants at a minimum to avoid creating unnecessary attrition between them. The facilitator should also challenge viewpoints until a consolidated opinion on the process is formed.

**Exercise 5.5** Consider the complaints that have emerged from the interviews reported in Exercise 5.4 (page 170). As a facilitator, what questions would you ask the various participants to investigate these further in a workshop?

### 5.2.4 Strengths and Weaknesses

The different methods of process discovery have each strengths and weaknesses. These can be discussed in terms of objectivity, richness, time consumption, and immediacy of feedback (see Table 5.1).

- *Objectivity*: Evidence-based discovery methods typically provide the best level of objectivity. Existing documents, existing logs, and observation provide an unbiased account of how a process works. Interview-based and workshop-based discovery both have to rely on the descriptions and interpretations of domain experts who are involved with the process. This bears the risk that those persons may have perceptions and ideas of how the process operates, which may be partially incorrect. Even worse, domain experts may opportunistically hide relevant information about the process from the analyst. This may be the case if the process discovery project happens in a political environment where groups of process stakeholders fear loss of power, loss of influence, or loss of position.
- *Richness*: While interview-based and workshop-based discovery methods show some weaknesses in terms of objectivity, they can provide rich insights into the process. Domain experts involved in interviews and workshops are a good source to clarify reasons why a process is set up as it is. Evidence-based methods might show issues that need to be discussed and raise questions, but they often do not provide an answer. Talking to domain experts also offers a view into the history of the process and the surrounding organization. This is important for understanding which stakeholders have which agenda. Evidence-based discovery methods sometimes provide insights into strategic considerations about a process when they are documented in white papers, but they hardly allow conclusions about the personal agendas of the different stakeholders.
- *Time consumption*: Discovery methods differ in the amount of time they require. While documentation around a particular process can easily be made available to a process analyst, it is much more time-consuming to conduct interviews and workshops. While interview-based discovery requires several feedback iterations, it is difficult to schedule a workshop session with various domain experts at the same time, especially on short notice. Automated process discovery often involves a significant amount of time for extracting, reformatting, and filtering event logs. Passive observation also requires coordination and approval

**Table 5.1** Relative strengths and weaknesses of process discovery methods

Aspect	Evidence-based	Interviews	Workshops
Objectivity	High	Medium-high	Medium-high
Richness	Medium	High	High
Time consumption	Low-medium	Medium	Medium
Immediacy of feedback	Low	High	High

time. Thus, it is a good idea to start with document analysis, since documentation can often be made accessible on short notice.

- *Immediacy of feedback*: Those methods that directly build on the conversation and interaction with domain experts are best for getting immediate feedback. Workshop-based discovery is best in this regard since inconsistent perceptions about the operation of a process can directly be resolved by the involved parties. Interviews offer the opportunity to ask questions whenever process-related aspects are unclear. However, not all issues can be resolved directly with a single domain expert. Evidence-based discovery methods raise various questions about how a process works. These questions can often only be answered by talking to domain experts.

The above strengths and weaknesses are summarized in Table 5.2. Since each discovery method has strengths and weaknesses, we recommend employing a mixture of them in a discovery project, if budget allows. The process analyst typically starts with documentation that is readily available. It is essential to organize the project in such a way that the information can be gathered from the relevant domain experts in an efficient and effective way. Interviews and workshops have to be scheduled during the usual work time of domain experts. Thus, experts

**Table 5.2** Summary of strengths and weaknesses per discovery method

Method	Strengths	Weaknesses
Document analysis	<ul style="list-style-type: none"> <li>• Structured information</li> <li>• Independent from stakeholders availability</li> </ul>	<ul style="list-style-type: none"> <li>• Outdated material</li> <li>• Wrong level of abstraction</li> </ul>
Observation	<ul style="list-style-type: none"> <li>• Context-rich insights</li> </ul>	<ul style="list-style-type: none"> <li>• Potentially intrusive</li> <li>• Stakeholders likely to behave differently</li> <li>• Only few cases and not all processes can be observed</li> </ul>
Automated discovery	<ul style="list-style-type: none"> <li>• Extensive set of cases</li> <li>• Objective data</li> </ul>	<ul style="list-style-type: none"> <li>• Potential issue with data quality and level of abstraction</li> <li>• Data may not be available or be available only in part</li> <li>• Data extraction and preparation is time-consuming</li> </ul>
Interviews	<ul style="list-style-type: none"> <li>• Context-rich insights</li> </ul>	<ul style="list-style-type: none"> <li>• Requires sparse time of stakeholders</li> <li>• Time-consuming: several iterations required before sign-off</li> </ul>
Workshops	<ul style="list-style-type: none"> <li>• Context-rich insights</li> <li>• Direct resolution of conflicting views</li> </ul>	<ul style="list-style-type: none"> <li>• Requires simultaneous availability of multiple stakeholders</li> <li>• Time-consuming: multiple sessions typically required</li> </ul>

have to be motivated to participate and involved in such a way that it is the least time-consuming for them. Once issues arise about specific details of a process, it might be required to turn back to evidence-based discovery methods.

*Question* In what situations is it simply not possible to use one or more of the described discovery methods?

There are various circumstances that may restrict the application of different discovery methods. Direct observation may not be possible if the process partially runs in a remote or dangerous environment. For instance, the discovery of an oil-extraction process at an offshore oil platform might belong to this category. There might also be cases where documentation does not exist, for example when a startup company which has gone through a period of rapid growth wants to structure its purchasing process. Lack of input may also be a problem for automated process discovery based on event log data. If the process under consideration is not yet supported by an IT system, or it is only supported in part, there is no data available for the automated discovery of the end-to-end process. In general, interviews are always possible. It might still be a problem though to gain commitment of domain experts to participate in interviews, especially because more than one interview is typically required. Moreover, this may be the case when the process discovery project is subject to company-internal politics and hidden agendas. Workshop-based discovery can be critical in strictly-hierarchical companies with a non-open culture.

**Exercise 5.6** The order-to-cash process of your favorite online bookstore has ten major activities conducted by ten people with five different roles. How much time do you approximately need for creating a process model that is validated by the various stakeholders and approved by the process owner? Consider two scenarios: one in which you run interviews, the other in which you run workshops. You may also use other discovery methods in these two scenarios, in addition to either interviews or workshops. Can you estimate the difference in time effort between the two scenarios? Make appropriate assumptions.

### 5.3 Process Modeling Method

Modeling a business process during process discovery is a complex task. Thus, it is good to follow a predefined procedure in order to approach this task in a systematic way. One way to do so is to work in five steps as follows:

1. Identify the process boundaries
2. Identify activities and events
3. Identify resources and their handoffs
4. Identify the control flow
5. Identify additional elements.

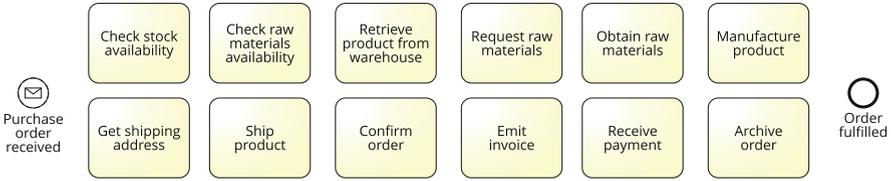
### ***5.3.1 Step 1: Identify the Process Boundaries***

The identification of the process boundaries is essential for understanding the scope of the process. As such, part of this work might have already been done with the definition of a process architecture during the process identification phase. The process boundaries vary depending on the perspective of the party we take. For example, let us consider again the order-to-cash process that we modeled in Chapter 3. Three parties are involved in this process: seller, customer, and supplier (for simplicity, we only consider one supplier rather than two). Let us assume we are a process analyst working for the seller company. Thus, in Step 1 we need to identify the boundaries of this process from the perspective of the seller, which is our party of interest. Technically, this means we need to identify the events that trigger our process and those that signal its completion. One way to do so is to identify the business objects that are required as input and provided as output of the process. Another option, as far as the end events are concerned, is to identify the possible outcomes of the process. For example, our order-to-cash process is triggered by the receipt of a purchase order from the customer (so the input object to the process is a purchase order) and completes with the fulfillment of the order (the final outputs are an invoice and a product, which are required to fulfill the order). Accordingly, we can identify one start message event (purchase order received) and one end event (order fulfilled). These two events mark the boundaries of our process from the perspective of the seller. If the process had negative outcomes, we would model these via terminate end events.

**Exercise 5.7** Identify the process boundaries for the procure-to-pay process described in Exercise 1.7 (page 31).

### ***5.3.2 Step 2: Identify Activities and Events***

The goal of the second step is to identify the main process activities and intermediate events. The advantage of starting with activities in workshops or interviews is that domain experts will be able to articulate what they are doing even if they are not fully aware of the overarching business process. In this step, we also need to identify the events that occur during the process, which we will model with intermediate events in BPMN. Figure 5.5 lists the twelve activities and two events in our order-to-cash example (there are no intermediate events in this example). The initial set of activities and events obtained in this step may undergo revisions, e.g., more activities may be added as we add more details to our model. If the process is too complex, we suggest you only focus on the main activities and intermediate events at this stage, and add the others at a later stage when a deeper understanding of these elements and their relations has been gained.

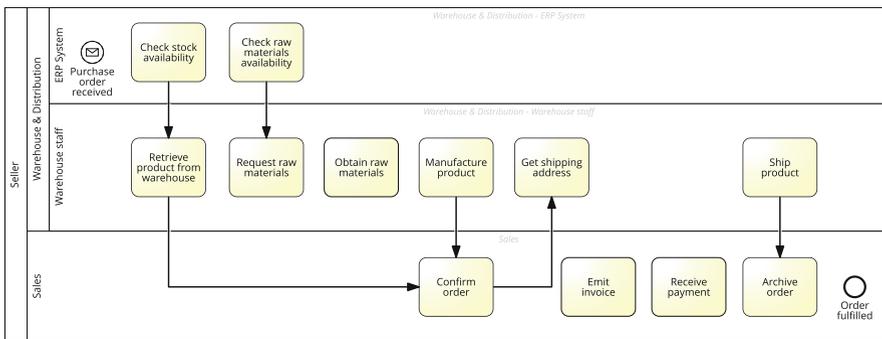


**Fig. 5.5** The activities and events of the order-to-cash process

**Exercise 5.8** Identify the main activities and events for the procure-to-pay process of Exercise 1.7 (page 31).

### 5.3.3 Step 3: Identify Resources and Their Handoffs

Once we have identified the set of main activities and intermediate events, we can turn to the question of *what resource* is responsible for which activity. This information provides the basis for the definition of pools and lanes, as well as for the assignment of activities and events to these elements. At this stage, the order of the activities is not defined yet. Therefore, it is good to first identify those points in the process where work is handed over from one resource to another, e.g., from one department to another. These handoff points are important since a participant being assigned a new task to perform usually has to make assumptions about what has been completed before. Making these assumptions explicit is an essential step in process discovery. Figure 5.6 shows the set of activities and events of the order-to-cash process now being assigned to the lanes of the seller pool, with sequence flows indicating handoffs. The handoff points also help to identify parts of the process that can be studied in isolation from the rest. These parts can be refined into sub-processes by the help of the involved stakeholders. For example, in the order-to-cash



**Fig. 5.6** The activities and events of the order-to-cash process assigned to lanes

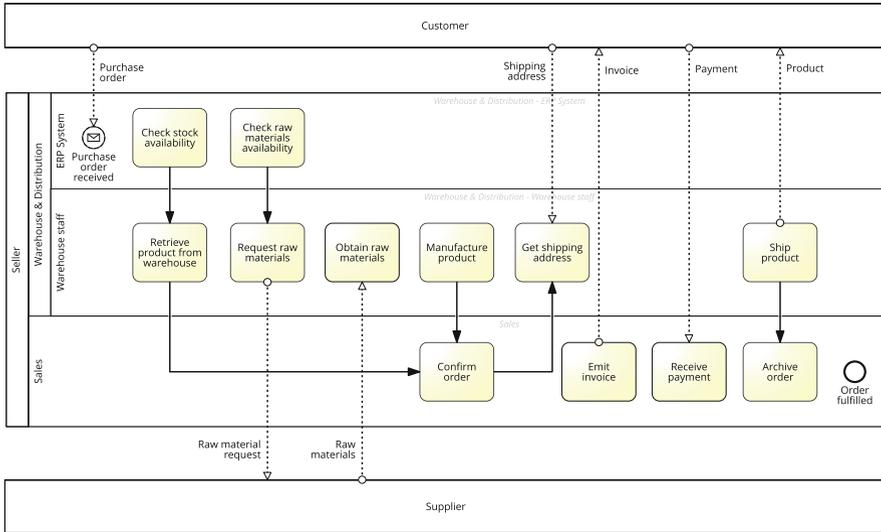


Fig. 5.7 The handoff of work between the seller, the customer, and the supplier

process the acquisition of raw materials (see Figure 4.15 on page 131) could be handled in isolation from the rest of the process, since this part involves the suppliers and personnel from the warehouse & distribution department.

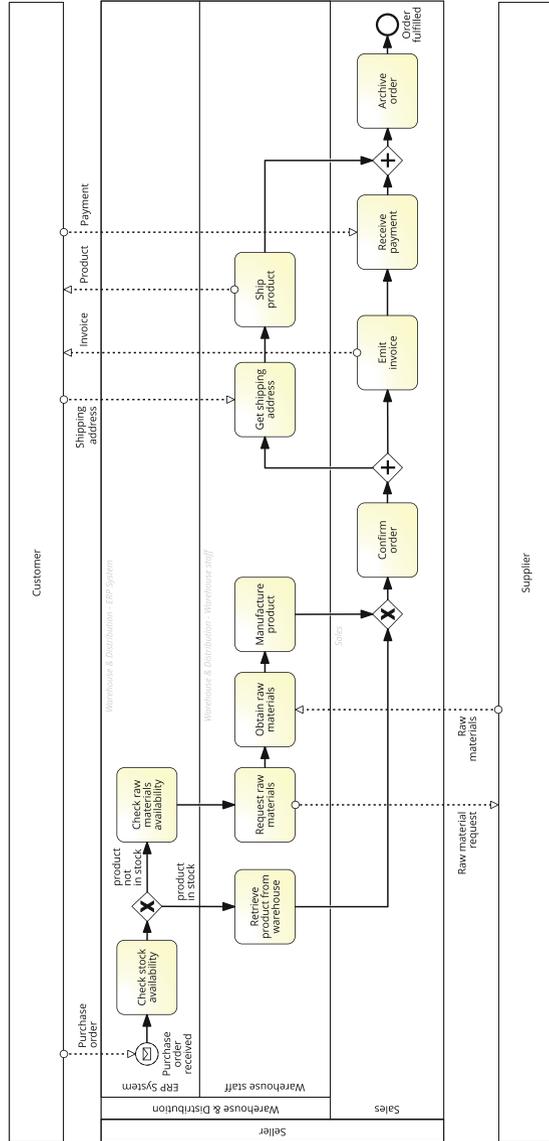
If the process involves external parties such as customers, business partners, or suppliers, we use pools to model these external parties and message flows to capture the handoff between them. For our order-to-cash example we obtain the model in Figure 5.7.

**Exercise 5.9** Using the process description in Exercise 1.7 (page 31), first identify the involved resources; next, assign the activities and events you obtained in Exercise 5.8 to these resources; and finally identify the handoffs.

### 5.3.4 Step 4: Identify the Control Flow

The internal handoffs within our business party of interest, i.e., those that we have represented via sequence flows, define an initial structure for the control flow. In essence, control flow relates to the questions of *when* and *why* activities and events are executed. Technically, we need to identify order dependencies, decision points, concurrent execution of activities and events, and potential rework and repetition. Decision points require the addition of (X)OR-splits and relevant conditions on the sequence flows originating from these splits. Rework and repetition can be modeled with loop structures. Concurrent activities that can be executed independently from each other are linked to AND gateways. Event-based splits are used to react to

**Fig. 5.8** The control flow of the order-to-cash process



decisions taken outside the process. Figure 5.8 shows how order constraints are captured by control-flow arcs in the seller pool of the order-to-cash process. Here we can see that the handoffs that we identified in the previous step have now been refined in more elaborate dependencies.

**Exercise 5.10** Using the process description in Exercise 1.7 (page 31), refine the model you obtained in Exercise 5.9 by defining the full control flow.

### 5.3.5 *Step 5: Identify Additional Elements*

Finally, we can extend the model by capturing the involved business objects and exception handlers based on the purpose of our model. For the objects, this means adding data objects, data stores, and their relations to activities and events via data associations. For the exception handlers, this means using boundary events, exception flows, and compensation handlers. As we mentioned in Chapters 3 and 4, the addition of data elements and exceptions depends on the particular modeling purpose. For example, if the process is meant to be automated, it is desirable to explicitly capture data and exception aspects. We may also add further annotations to support specific application scenarios. For instance, if the model is used for risk analysis or for process cost estimation, we may need to add risk and cost information. In general, which elements to be added depends upon the particular application scenario.

*Question* When should we stop modeling a process?

As discussed in Chapter 3, the level of modeling detail is determined by the particular modeling purpose. During process discovery, the purpose is to gain a sufficient understanding of the process as required to perform the subsequent analysis. Hence, there is no need to document the process in a level of excruciating detail. Unfortunately, though, many organizations fall into the trap of creating very detailed models during process discovery. This may have a negative impact on the overall cost of a BPM project, and most importantly, it will delay the actual improvement of the processes.

**Exercise 5.11** Using the process description in Exercise 1.7 (page 31), refine the model you obtained in Exercise 5.10 by adding business objects and exception handlers.

### 5.3.6 *Summary*

In this section, we illustrated a method for constructing a business process model via a number of incremental steps. This method lends itself well to workshops, as it can be run over multiple workshop sessions. For example, we can do Steps 1–2 in the first workshop session, and then Steps 3, 4, and 5 in a subsequent session each, starting each session by validating the results of the previous step with the workshop participants.

If you are an expert analyst combining a strong knowledge of BPMN with excellent facilitation abilities, you may run this method in an integrated manner. In this alternative, you would model the control flow on-the-fly as you add resources, i.e., you would do Steps 3 and 4 simultaneously.

## 5.4 Process Model Quality Assurance

As discussed, gathering and organizing process-related information in a process model is typically a sequential activity (e.g., we conduct different workshop sessions or interviews), which involves at least one process analyst and multiple domain experts. Therefore, there is a need to assure that the model we produce is of high quality. As shown in Figure 5.9, a process model is subjected to three quality aspects: syntactic, semantic, and pragmatic quality. Verification is the activity of assuring syntactic quality, validation that of assuring semantic quality, and certification that of assuring pragmatic quality. In addition, modeling guidelines and conventions can be used to achieve a high quality right from the start.

### 5.4.1 Syntactic Quality and Verification

*Syntactic quality* relates to the conformance of a process model to the syntactic rules of the modeling language used. We distinguish two types of syntactic rules: structural and behavioral rules. *Structural rules* relate to the way the various model elements are connected with each other while *behavioral rules* relate to the way a process model can be instantiated. Syntactic rules are important because they are meant to increase model understandability and to avoid ambiguity.

Below we list the main structural rules that apply to a BPMN model:

#### 1. Element level:

- Activities: activities must have at least one incoming and one outgoing sequence flow.
- Events:
  - start events must not have incoming sequence flows;

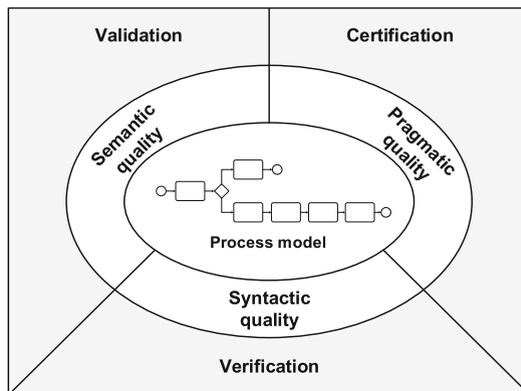


Fig. 5.9 Process model quality aspects and assurance activities

- end events must not have outgoing sequence flows;
  - intermediate events must have at least one incoming and one outgoing sequence flow;
  - only intermediate catching boundary events can be attached to an activity's border.
- Gateways:
    - split gateways must have exactly one incoming and at least two outgoing sequence flows;
    - join gateways must have at least two incoming and exactly one outgoing sequence flows;
    - the outgoing arcs of an (X)OR-split gateway must bear conditions.
  - Flows:
    - a sequence flow must connect two flow nodes (activities, events, and gateways) of the same pool, i.e., sequence flows cannot cross the boundaries of pools;
    - a message flow must connect (an activity or a throwing message event in) one pool with (an activity or a catching message event in) a different pool;
    - a directed data association must connect a data object with an activity or message event, or a data store with an activity, or vice versa;
    - an indirected data association must connect a data object with a sequence flow, or a text annotation with any element.

2. **Model level:** all flow nodes must be on a path from a start to an end event.

The element-level rules restrict the way each model element is used, while the model-level rule ensures that the model is not disconnected. A model is *structurally correct* if it satisfies all the above structural rules. Such rules can be checked by inspecting the graph-based structure of the process model. For example, it is easy to see that the model in Figure 5.10 is structurally incorrect.

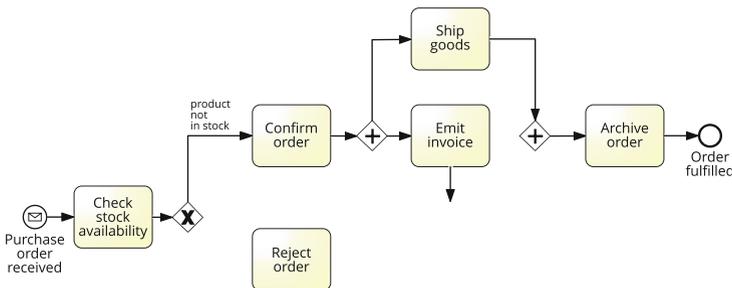
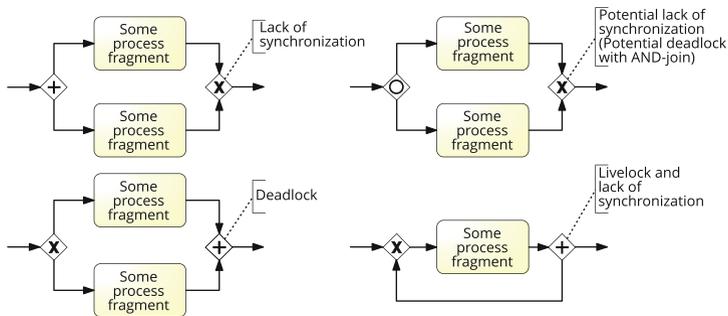


Fig. 5.10 A structurally incorrect process model

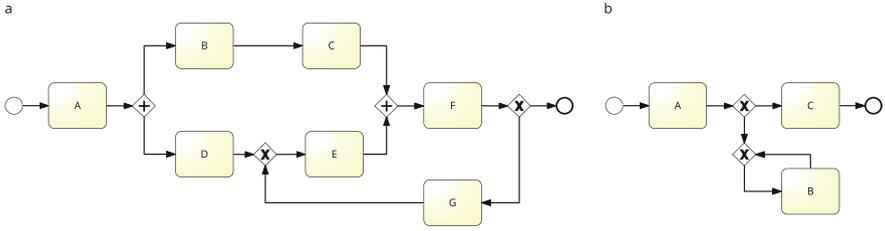
Behavioral rules are required to avoid behavioral anomalies such as deadlocks and livelocks during the execution of a process model. We have already introduced some of these behavioral anomalies in Chapter 3 (see e.g., the discussion on Figure 3.11 on page 88). Let us now take a systematic look at them. A *deadlock* occurs when a running process instance is not able to progress any further once a given state is reached, i.e., a token gets stuck at that state. A *livelock* is another type of behavioral anomaly which occurs when a process instance keeps cycling in a loop. In other words, a token is trapped within a loop structure: it is free to move but only within the loop. For example, this may arise if the condition of a loop always evaluates to true. Both deadlocks and livelocks may prevent tokens from reaching an end event, so the process instance may not be able to complete altogether. Another behavioral anomaly is the *lack of synchronization*. This occurs when two or more tokens are in the same sequence flow because they are not synchronized at some join gateway. Finally, a *dead activity* is an activity that can never be executed in any instance of the process model.

It is easy to see that these behavioral anomalies can arise when mixing a split with a join of a different type in the same block structure, as shown in Figure 5.11. A *block structure* is a single-entry single-exit process model fragment, such that the entry and exit points are two gateways (one split and one join) and each path from one gateway leads to the other gateway. If the split and join match in type no behavioral anomaly can arise, while if the two gateways are different, as in the models of Figure 5.11, this will lead to different behavioral anomalies. Such anomalies, though, can also arise outside of block structures, in which case they are harder to spot. For example, Figure 5.12a shows a model with a deadlock that occurs at the AND-join if activity G is executed. This is because there is an injection of a branch into what would otherwise be a perfect AND-block. A token may come back from this branch and reach the AND-join after E is performed. The AND-join, however, will deadlock because it will never receive a token from C after F has been executed.

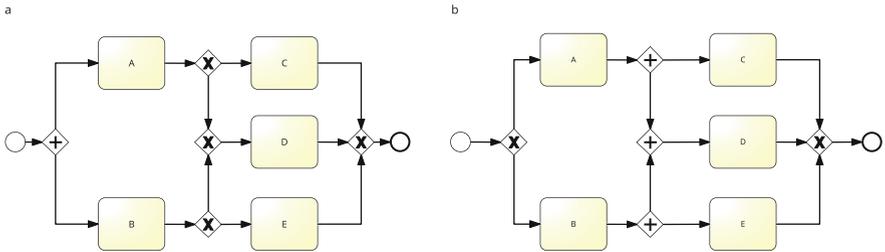
**Exercise 5.12** Have a look at Figure 5.11. What is precisely going wrong in each block structure?



**Fig. 5.11** Common behavioral anomalies in block structures



**Fig. 5.12** A process model with a deadlock (a) and one with a livelock (b)



**Fig. 5.13** A process model with lack of synchronization (a) and one with a dead activity (b)

We say that a process model is *behaviorally correct*, or *sound*, if and only if it satisfies the following behavioral rules:

1. **Option to complete:** any running process instance must eventually complete,
2. **Proper completion:** at the moment of completion, each token of the process instance must be in a different end event,
3. **No dead activities:** any activity can be executed in at least one process instance.

Option to complete implies that there are no deadlocks nor livelocks that prevent the instance from completing, while proper completion implies that there is no lack of synchronization. For example, the model in Figure 5.12a violates the option to complete due to a deadlock if the path via G is chosen. Another example of no option to complete is provided by the model in Figure 5.12b, though this time it is because of a livelock. In addition, this model is structurally incorrect because B is not on a path from a start to an end event. An example of improper completion is given by the model in Figure 5.13a. This model has a lack of synchronization (the last sequence flow will always have two tokens), and in some instances activity D may even be executed twice. Finally, the model in Figure 5.13b violates the no dead activities property, as D can never be executed. Moreover, this model suffers from improper completion, since when the end event is reached a token remains trapped before the AND-join. However, this is not due to lack of synchronization (there can never be two tokens on the same sequence flow), but rather to a deadlock at the AND-join. Thus, while proper completion excludes lack of synchronization, the latter is not the only cause for improper completion.

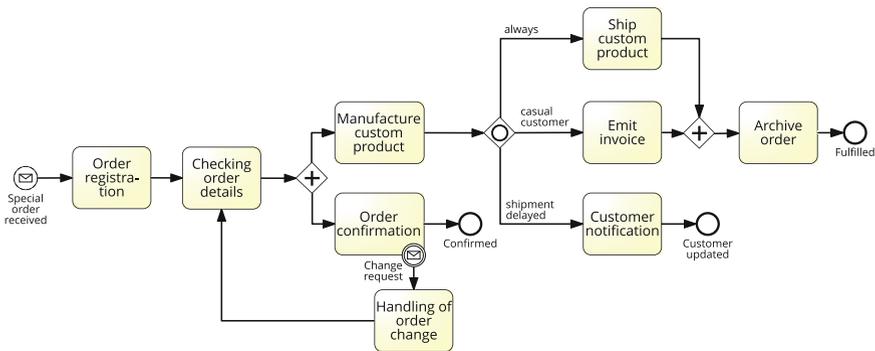
The above definition of *soundness* only takes into account the control flow of a process model. It assumes all input data objects and incoming messages are available when an activity is to be executed, and all output data objects and outgoing messages are produced upon an activity's completion. Properties such as soundness can be checked after a process model is created. Alternatively, a process modeling tool can enforce that a model is correct by design. This can be achieved by allowing only edit operations that preserve structural and behavioral correctness. One easy way to achieve that is to construct models where gateways appear only in block structures and are of matching type (so-called *structured* process models) as the model in Figure 3.12 (see page 90). However, this type of model has limited expressiveness compared to unstructured models, as discussed in Section 4.1 in the context of cycles.

Those parts of a model that cause unsoundness should be reworked. Typically, these parts trigger questions about specific behavior of the process that need to be clarified with domain experts. *Verification* is the activity of checking that a process model is syntactically correct, i.e., that it is both structurally and behaviorally correct. Verification addresses formal properties of a model that can be checked without knowing the corresponding real-world process.

**Exercise 5.13** Which behavioral rules are violated in the model of Figure 5.14? How can this model be made sound?

### 5.4.2 Semantic Quality and Validation

Semantic quality deals with the adherence of a process model to its real-world process. *Validation* is the activity of checking the semantic quality of a model by comparing it with its real-world business process. The particular challenge of validation is that there is no set of formal rules that can be used to easily check semantic quality; rather the focus is on the overall meaning of the model, and



**Fig. 5.14** A process model for fulfilling special orders

therefore, this can only be done by talking to the process participants and by consulting the available documentation.

There are two essential aspects of semantic quality: validity and completeness. *Validity* means that all statements that one can make from the model are correct and relevant to the real-world process. Validity can be assessed by explaining to domain experts how the processing is captured in the model. The domain expert is expected to point out any difference between what the model states and what is possible in reality. *Completeness* means that the model contains all relevant statements about the corresponding business process. Completeness is more difficult to assess. Here, the process analyst has to ask about various alternative processing options at different stages of the process to ensure nothing is missing. For example, the model in Figure 5.8 (see page 181) is incomplete because it does not capture exceptional paths such as that to handle an order cancelation from the customer. It is the job of the process analyst to judge the relevance of these additional elements. This judgement has to be done against the background of the modeling objective, which the process analyst should be familiar with. Let us consider an example to understand the difference between validity and completeness. If a process model for loan assessments states that any financial officer may carry out the task of checking the credit history of a particular applicant while in practice this requires a specific authorization, the model has a semantic problem due to an invalid statement. If the task of checking the credit history is omitted then the model has a semantic problem due to incompleteness.

**Exercise 5.14** What can we say about the semantic quality of the model in Figure 3.9 (page 87)? Refer to the process description in Example 3.5 on page 86.

Validation can be supported by methods like interviews or workshops. Alternatively, there are tools that provide truthfulness by design. This is, for instance, achieved by automatically discovering a process model from an event log, as we will see in Chapter 11. In practice, process models often require the *approval* from the process owner. This approval is a special validation step, since it is an endorsement of the validity and completeness of the model. Beyond that, the approval of the process owner establishes the normative character of the process model at hand. As a consequence, the process model can then be published, used as an input for process analysis and redesign, or archived.

**Exercise 5.15** Consider the model in Figure 5.14 (page 187) with reference to the following process description. Is this model valid and complete? If not, what statements are invalid and what is missing?

When a special order is received, it is first registered and then its details are checked. Next, the order is confirmed and meantime the custom product is manufactured. Once the product has been made, the shipment can be planned. Afterwards, the customer type and shipment status are checked. In fact, if a customer is casual an ad hoc invoice must be emitted, which is not required for ordinary customers. In the latter case, the customer account is simply charged with the costs related to the order fulfillment. Moreover, if the shipment has been delayed, the customer must be updated on the expected delay. Concomitantly to these activities, the custom product is shipped. After the latter activity and after the invoice has been emitted, the process completes with the archival of the order. Any time during

the confirmation of the order and the manufacturing of the respective product, an order change request may be received, in which case any activity must be interrupted to handle the change request. This includes the registration of the order variation and a notification to the customer, after which the process resumes from the order checking.

### 5.4.3 *Pragmatic Quality and Certification*

*Pragmatic quality* relates to the usability of a process model. The particular challenge of pragmatic quality assessment is to anticipate the actual usage of a process model. This aspect focuses on how people interact with a model. For instance, a process model of good pragmatic quality can be checked by testing how well a user understands the model.

*Certification* is the activity of checking the pragmatic quality of a process model by investigating its use. There are several aspects of usability including understandability, maintainability, and learning. *Understandability* relates to how easy it is to read and comprehend a process model. *Maintainability* points to the ease of applying changes to the model. *Learning* relates to how good a process model reveals how its corresponding business process works in reality. There are several characteristics of a model that influence usability including its size, its structural complexity, and its graphical layout.

Certification can be achieved via interviews or experiments with model users, i.e., with those that are meant to use the model in their job, e.g., a process owner using a model for communication purposes or a process analyst using it as input to process analysis and redesign. Alternatively, there are rules that strive to provide usability by design. One of these is *block-structuredness*: a structured process model, besides being always sound, has been shown to often be easier to understand than its unstructured counterpart. As an example, Figure 5.15a shows an unstructured process model while Figure 5.15b shows the structured version of this model, where one activity has been duplicated to avoid crossing arcs, and splits and joins within the same block match in type. This model is semantically equivalent to, yet simpler to understand than that of Figure 5.15a.

There are two essential checks for understandability, and learning. The first one relates to the consistency between visual structure and logical structure. Figure 5.16a and b show the same fragment of the order-to-cash process model. The second model is a rework of the first one in terms of layout, where all elements are laid out following a top-left to bottom-right orientation, and where there are no crossing arcs. Here the elements' position has been changed with the aim to improve the consistency between visual structure and logical structure. Block-structuring a process model, where possible, is also another mechanism to improve this consistency.

The second check is concerned with meaningful labels. Activities, events, and other elements must use labels that follow specific naming conventions. For example, the labels in the model of Figure 5.17 follow inconsistent labeling

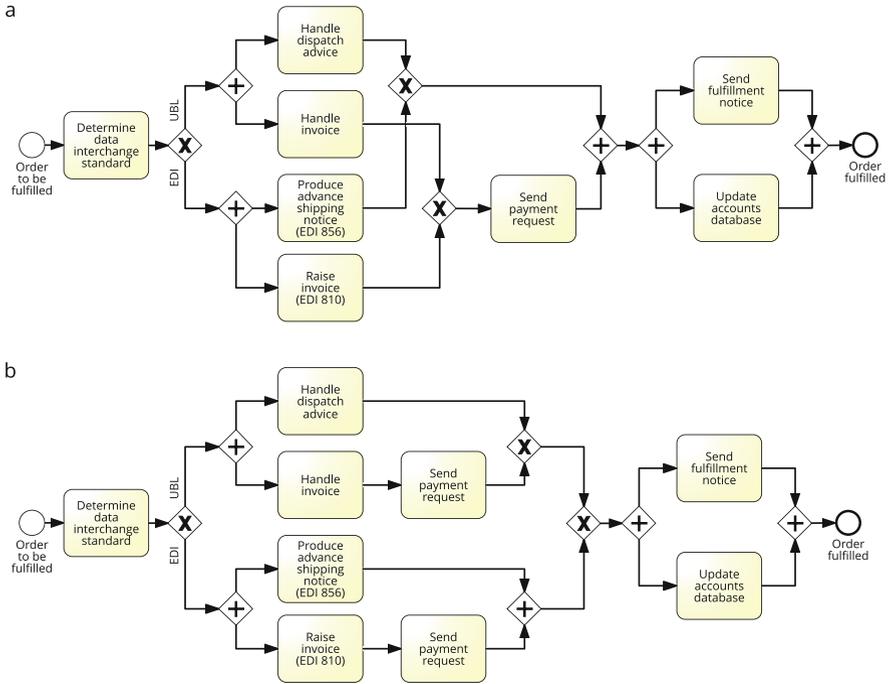


Fig. 5.15 An unstructured process model (a) and its structured counterpart (b).

Acknowledgement This example is taken from [40]

styles and lack the use of a common glossary, resulting in ambiguous meaning which affects the model understandability. Activity “Get approval for expenses” follows the *verb-object* style (imperative verb + business object), which has been shown to be the most effective style for labeling activities. In contrast, activities “Cost planning” and “Recalculation of costs” capture the actions of planning and recalculating as nouns at different positions in the label, following the *action-noun* style. As a result of mixing different labeling styles, the meaning of activity “Plan data transfer” is ambiguous: it could mean either to plan a data transfer or to transfer a record of plan data. In addition, due to the lack of a common glossary, two activities use the term “costs” while another the term “expenses”, though they probably refer to the same thing. Moving to the labels of events and gateways, we see that the label of the end event “Approved” lacks a reference to a business object (it should be “Expenses approved”, following an object-verb style: business object + past participle verb). The XOR-split’s label “Acceptable?” hides the existence of a decision activity “Check plan acceptability”. In fact, as discussed in Section 3.2 (see page 79), it is preferable to avoid labeling (X)OR-split gateways and to use more explicative conditions than “yes” or “no” in the outgoing arcs of the split, e.g., “plan acceptable” and “plan unacceptable”.

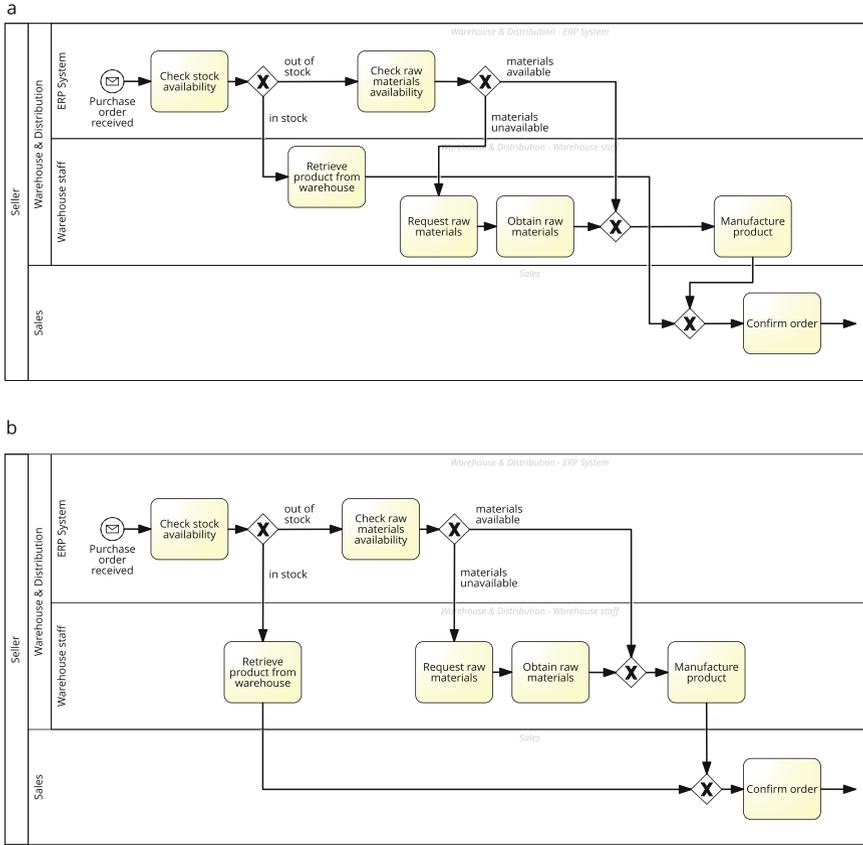


Fig. 5.16 Extract of the order-to-cash process model: with bad layout (a), with good layout (b)

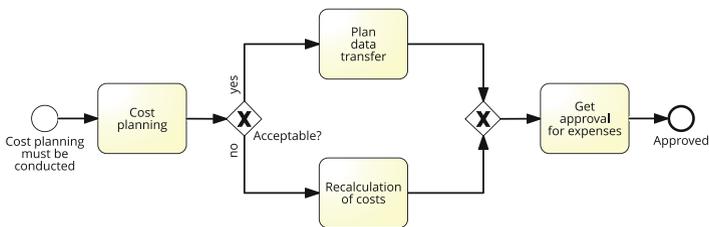


Fig. 5.17 A process model for cost planning.

Acknowledgement This example is taken from [87]

Exercise 5.16 Is the process model in Figure 5.14 (page 187) of good pragmatic quality? If not, how can it be improved?

### 5.4.4 Modeling Guidelines and Conventions

*Modeling guidelines* and *conventions* are an important tool for improving the pragmatic quality of process models. The specific objectives for using modeling guidelines and conventions are manifold: (i) safeguarding model consistency and improving standardization and reuse, especially in the context of large modeling initiatives involving various process analysts; (ii) reducing the dependency on process analysts, who may leave the company at some stage; and (iii) facilitating access to models by non-modeling experts. For example, consider an insurance company that has a BPM team within each line of business (home, motor, commercial). The various BPM teams may follow the same set of modeling guidelines to maximize consistency and reuse across the different insurance services. This way, for example, it will be easier to standardize common parts across all the variants of their claim handling process.

The difference between guidelines and conventions is essentially that the former are suggestions while the latter are mandatory rules. Modeling guidelines and conventions are restrictions to the following aspects of a process model:

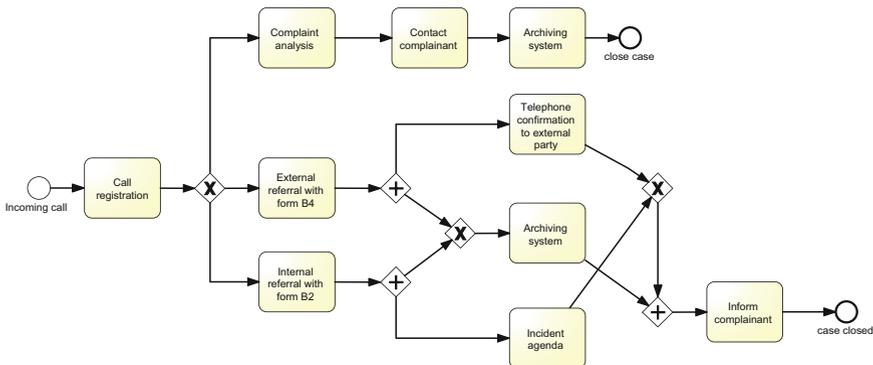
1. **Vocabulary:** avoiding certain elements, e.g., never using event sub-processes.
2. **Structure:** limiting the structure of the model, e.g., setting a threshold on the size or the number of hierarchical layers, or modeling using block-structures only.
3. **Semantics:** avoiding particular element meanings (rarely used), e.g., using boundary events to model business faults only, excluding technology faults.
4. **Appearance:** restricting the model appearance in terms of labels, layout, and notation, e.g., using the verb-noun style to label activities, using terms only taken from a glossary, or modeling with a top-left to bottom-right orientation.

Below we propose a set of modeling guidelines called the *Seven Process Modeling Guidelines (7PMG)*. This set was developed as an amalgamation of insights from available research. Specifically, the analysis of large collections of process models by various researchers identified many syntactical errors as well as complex structures that reduce pragmatic quality. These guidelines are helpful to guide users towards mitigating such problems.

- G1:** *Use as few model elements as possible.* Studies have shown that models of large size tend to be more difficult to understand and have a higher syntactic error rate.
- G2:** *Minimize the routing paths per element.* For each element in a process model, it is possible to determine the number of incoming and outgoing arcs. This summed figure gives an idea of the routing paths through the element. A high number makes it harder to understand the model. Also, the number of syntactic errors in a model seems strongly correlated to the use of model elements with a high number of routing paths.
- G3:** *Use one start event for each trigger and one end event for each outcome.* Empirical studies have established that the number of start and end events is positively connected with an increase in error probability. Models satisfying this requirement are easier to understand.

- G4:** *Model as structured as possible.* Unstructured models are not only more likely to include behavioral anomalies, but they also tend to be harder to understand. Nonetheless, as shown in Section 4.1, it is sometimes not possible or not desirable to turn an unstructured model fragment (e.g., an unstructured cycle) into a structured one. This is why this guideline states “as structured as possible”.
- G5:** *Avoid OR gateways where possible.* Models that have only AND and XOR gateways are less error-prone. This empirical finding is apparently related to the fact that the combinations of choices represented by an OR-split are more difficult to grasp than behavior captured by other gateways. Moreover, the semantics of the OR-join is complex, as it needs to check that each of its incoming branches is active (see Section 3.2.3 on page 86), and as such hampers understandability.
- G6:** *Use verb-object activity labels.* A wide exploration of labeling styles used in process models from practice disclosed the existence of a number of popular styles. From these, model users consider the verb-object style, like “Inform complainant”, as significantly less ambiguous and more useful than action-noun labels (e.g., “Complaint analysis”) or labels that follow neither of these styles (e.g., “Incident agenda”).
- G7:** *Decompose a model with more than 30 elements.* This guideline relates to G1 that is motivated by a positive correlation between size and syntactic errors. For models with more than 30 elements the error probability tends to climb sharply. Thus, large models should be split up into smaller ones. For example, large fragments with a single entry and a single exit can be replaced by a collapsed sub-process activity.

**Exercise 5.17** Consider the process model of Figure 5.18, which captures a business process for handling complaints, as described below. Identify improvements for this model by assessing which of the 7PMG guidelines are not followed. Next, remodel the process based on your observations.



**Fig. 5.18** A process model for handling complaints, as found in practice

A complaint is triggered by a phone call by a complaining customer. It is decided whether the complaint can be handled or whether it has to be referred to an internal or external party. An external referral leads to a telephone confirmation to the external party. An internal referral is added to the incident agenda. If no referral is needed, a complaint analysis is conducted and the complainant is contacted. In either case, the complaint is archived and the case is closed.

Process modeling tools such as Apromore, ARIS or Signavio Process Manager come with a predefined list of modeling guidelines that can be automatically checked against a model, and allow the possibility of defining custom guidelines. For example, one can check that each activity follows a verb-object labeling style or that the model is laid out from top-left to bottom-right. While support for the automated checking of modeling guidelines is common, much less common is the support for the automated checking of the BPMN syntactic rules, where the focus is mostly on structural rules.

## 5.5 Recap

This chapter described how to conduct the different tasks of process discovery: (i) defining the setting, (ii) gathering the required information, (iii) modeling the process, and (iv) assuring model quality. The chapter stressed the complementary skills of process analysts and domain experts. While process analysts are skilled in analyzing and modeling processes, they often lack detailed domain knowledge. In contrast, domain experts have typically limited modeling skills, but a detailed understanding of the part of the process they are involved with. This implies several challenges of process discovery that analysts have to face.

Next, the chapter illustrated different discovery methods. Evidence-based methods typically provide the most objective insight into the execution of the process. However, the immediacy of feedback is low and the richness of the insights can be mediocre. Interviews can be biased towards the perspective of the interviewee, but reveal rich details of the process. Interviews offer a chance to gain direct feedback on process-related matters. Workshops can help to resolve inconsistent views of different domain experts. On the downside, it is difficult to have all required domain experts available at the same time. Budget allowing, we recommend using a mixture of discovery methods based on the specifics of the discovery project.

We then presented a five-step process modeling method. First, we suggest identifying the boundaries of the process in terms of its start and end events. Second, we determine the main activities and events, the different resources involved (internal and external), and their handoff of work. Once this aspect has been clarified, we can determine the full control flow, and complete the model by adding additional elements such as business objects and exception handlers.

In the last section we discussed three measures of quality assurance: syntactic, semantic, and pragmatic quality, and discussed the respective quality assurance activities: verification, validation, and certification. We concluded the chapter by illustrating a set of modeling guidelines that can help improve pragmatic quality.

## 5.6 Solutions to Exercises

**Solution 5.1** Domain knowledge can be very helpful for analyzing processes. It helps to ask the right questions and to build analogies from prior experience. On the other hand, the skills of an experienced process analyst should not be underestimated. These skills are domain-independent and relate to how a process discovery project can be organized. Experienced process analysts are skilled in scoping and driving a project into the right direction. They possess problem-solving skills for handling various critical situations of a process discovery project. There is clearly a trade-off between the two sets of skills. It should be assured that a certain level of process modeling analysis experience is available. If that is not the case for the applying domain expert, the process analyst would be preferred.

**Solution 5.2** To obtain a complete and systematic view of our process, we must overcome two of the three challenges related to domain experts, namely: (i) fragmented process knowledge and (ii) thinking on a case level. To overcome the first challenge, we first need to understand how each of the three domain experts (customer relationship manager, warehouse worker, and financial officer) participates in the process. To this end, we can ask them what tasks they are responsible for, and, for each of these, what inputs are required and what outputs are produced. This will help us understand which handoffs of work exist between them (assuming there is no other resource involved in the process), so as to infer an initial order between their tasks. For example, from this first battery of questions we may realize that the warehouse worker picks books from the warehouse for shipment only upon the receipt of a confirmed order, which is emitted by the customer relationship manager. This suggests a handoff of work between customer relationship manager and warehouse worker.

If inconsistent descriptions of the process emerge out of these initial discussions, we have to ask additional questions to uncover hidden assumptions and conditions underlying these descriptions. For example, the warehouse worker may expect to receive a single confirmed order for all books in a given purchase order, assuming that any shipment should be put on hold until all the ordered books are available. However, the customer may have opted for their books to be shipped in different packages as soon as they become available. In this case, the customer relationship manager may confirm a set of sub-orders (one per package), rather than a single order. To clarify these diverging assumptions between the customer relationship manager and the warehouse worker, we may ask the customer relationship manager about the different shipment options that are available to customers, and assess what implications there should be for the warehouse worker, as opposed to what the latter actually does. This investigation into inconsistent views by the involved stakeholders can already help us identify opportunities for improving the process.

To overcome the second challenge (resources thinking on a case level), we may inquire about exceptions due to *business faults* such as what happens if an order is canceled by the customer, or if an ordered product is unavailable or discontinued. We may also inquire about the existence of *timeouts*, for example by asking if

there is a prescribed timeframe to fulfill an order, and if so, what happens if this deadline is not met. These are examples of questions that help us reason on a process level, because they focus on different conditions and different outcomes, rather than on the case level, i.e., with reference to a specific order. By doing so, we can identify the routing constructs that are required to link all the tasks together and infer the complete control flow. For example, an order is confirmed by the customer relationship manager only if the ordered books are available. If they are not available, the customer is informed accordingly and the order is declined. These two (intermediate) outcomes are mutually exclusive, suggesting the presence of an XOR-split after the stock availability check.

**Solution 5.3** The methods in the classes of the UML class diagram may suggest possible process activities, while the organizational policies may provide the conditions underpinning certain decision activities in the process. Looking at the class diagram, some classes map to organizational roles that participate in our process, such as Applicant, Admission officer, and Academic committee member; other classes map to documents, such as Assessment and Application. However, considering that this class diagram models the functionality of an entire system and that this system likely supports other processes within the university, some of these classes are irrelevant for our specific process. For example, Visitor and Visit probably refer to a process similar to the student admission process, i.e., that for admitting academic visitors to the university.

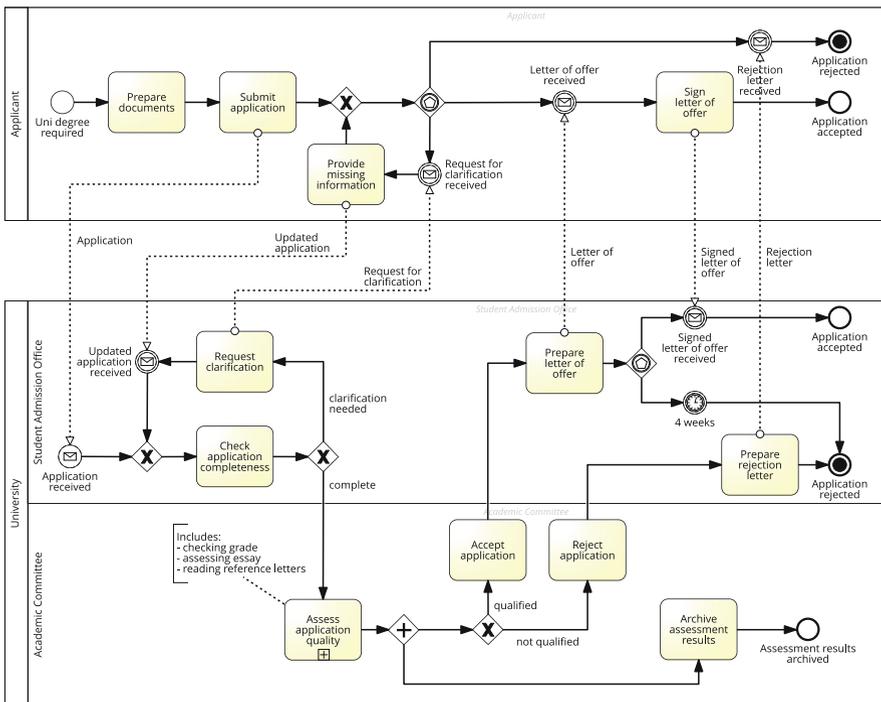
Taking a closer look at the methods for AdmissionOfficer, we can derive three candidate activities for our process: “Provide information”, “Check application”, and “Request clarification”. Which of these are actually activities of our process will have to be assessed by talking directly to an admission officer. Likewise, looking at AcademicCommitteeMember, other candidate activities are “Assess application”, “Accept application”, “Reject application”, and “Archive assessment”. Similar conclusions can be derived from the Applicant class. Observe, however, that not all activities performed by a given participant are reflected in a UML class diagram. This is because some of these activities may be manual or simply not supported by the system in question. Again, this is something that will have to be discussed with domain experts.

Moving to the list of organizational policies, we can infer the conditions underpinning the final decision on an admission application (e.g., based on consistency of prior education and quality of essay). These conditions are probably checked by a member of the academic committee via activity “Assess application”, while via activity “Check application” an admission officer probably checks that all required documents (academic transcripts, essay, reference letters, etc.) are present in the application. If something is missing or unclear they may request more information or documents by performing activity “Request clarification”. In addition, the last policy suggests the presence of a deadline of four weeks for the applicant to accept a letter of offer. We can model this timeout with an event-based XOR-split followed by a timer event (4 weeks) in one branch and an intermediate catching message event on the other, to receive the signed letter of offer. However, from the available

documentation it is not possible to infer which resource will perform this check. It may likely be a student admission officer, if this role handles all communications with the applicant.

Finally, we use the organization chart to determine the persons to interview and their supervisors to ask for permission. Candidates for interview are all officers within the student admission office (with Mark Johnson being their supervisor) and all members of the academic committee (with Liza Stewart being their supervisor). It is not clear at this stage if the enrollment office is involved at all in our process. Probably this office is only relevant to the enrollment process, which follows the admission process and allows students to enroll in particular subjects. Mark can help us figure this out.

**Solution 5.4**



**Solution 5.5** Three complaints emerge from the interviews. Louise Smith complains that the Web portal has bugs and as such it lets through incomplete applications. She points out that rectifying these applications is time-consuming. Peter Capello also complains about technology. He points out to communication issues with the student admission office due to the student admission system losing messages that he needs to resend. He adds that he can only resend messages if he finds out that these went missing, alluding to the fact that sometimes he does not

realize that messages got lost. Also in this case, this problem leads to rework and so to process slowdowns. Finally, Mary Adams laments that the academic committee is too slow to reply.

As discussed, process discovery can provide opportunities to isolate process issues, the impact of which can then be assessed during process analysis. However, before capturing these issues into the model and flagging them for process analysis, it is important to investigate these issues. The purpose is to understand whether they are really issues, in which case they should be captured in the model, or rather sporadic exceptions, which we may neglect to avoid cluttering the model. This can be done during a workshop, where such complaints are discussed directly with all relevant stakeholders. For example, for each complaint we can ask the person who raised it how frequently the particular issue occurs. In our example, we can ask Louise how many times on average she needs to request the applicant to rectify and resubmit his or her application, and when was the last time she did so. If we find out that the issue does not actually happen frequently, or that the last time it occurred was a very long time ago, which suggests that it may have already been fixed (e.g., in a new release of the software), then the issue may not be so important, or it has become irrelevant, and so we may decide not to capture it in the process model.

We can ask the same questions to Peter regarding his communication problem with the student admission office. Interestingly, by bringing everyone at the same table, a workshop can help us to understand the root causes of certain issues. This could be the case for Mary's complaint about the slowness of the academic committee. This issue may likely be caused by the student admission system, which as Peter reported, seems to frequently fail to send messages to the student admission office. So per se this does not depend on the academic committee.

A take-home message from this exercise is that the results of a workshop are not narrowly restricted to the process model that is created, but they extend to the insights gained on process issues, and also provide a forum where process participants can further explore these issues.

**Solution 5.6** This process contains ten major activities that are assigned to five different roles, and there are altogether ten domain experts besides the process owner. We can assume that there will be a kickoff meeting with the process owner and some important domain experts on day one. Furthermore, 1 day might be required to study the available documentation.

*Scenario 1: Interviews.* An interview with one domain expert can take from 2 to 3 h, such that we would be able to meet two persons a day, and document the interview results later in the same day. Let us assume that we meet some persons only once while we seek feedback from important domain experts in two additional interviews. Then, there would be a final approval from the process owner. This adds up to 1 day for the kickoff, one for document study, 5 days for the first iteration interviews, and further 5 days if we assume that we meet five of the ten experts three times. Then, we need maximum 1 day to prepare for the meeting to gain final approval from the process owner, which would be on the following day. If there are no delays and

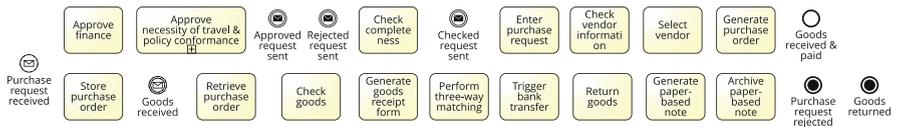
scheduling problems, using document analysis and interviews yields a total of  $2 + 5 + 5 + 2 = 14$  work days as a minimum.

*Scenario 2: Workshop.* Given the relatively low complexity of this process (ten main activities overall, five different roles and ten participants), three workshop sessions of 3 h each should be enough to create a complete process model and validate it. Clearly, this is only feasible assuming that we can simultaneously access at least one representative for each of the ten roles. This will lead to a minimum of five and a maximum of ten people participating in each workshop session, which is feasible (recall that no more than ten to twelve people should attend a workshop to avoid it to become unmanageable). We can use the first session to create a sketch of the model including relevant resources, the second to validate this sketch and identify the main routing constructs, and the final session to refine the control flow and validate the final model. Between each session we can spend 2–3 h to consolidate the results of each session and prepare for the next session. Then, similarly to the first scenario, we need maximum 1 day to prepare for the meeting to gain final approval from the process owner, which would be on the following day. If there are no delays and scheduling problems, using document analysis and workshops yields a total of  $2 + 3 + 2 = 7$  work days as a minimum.

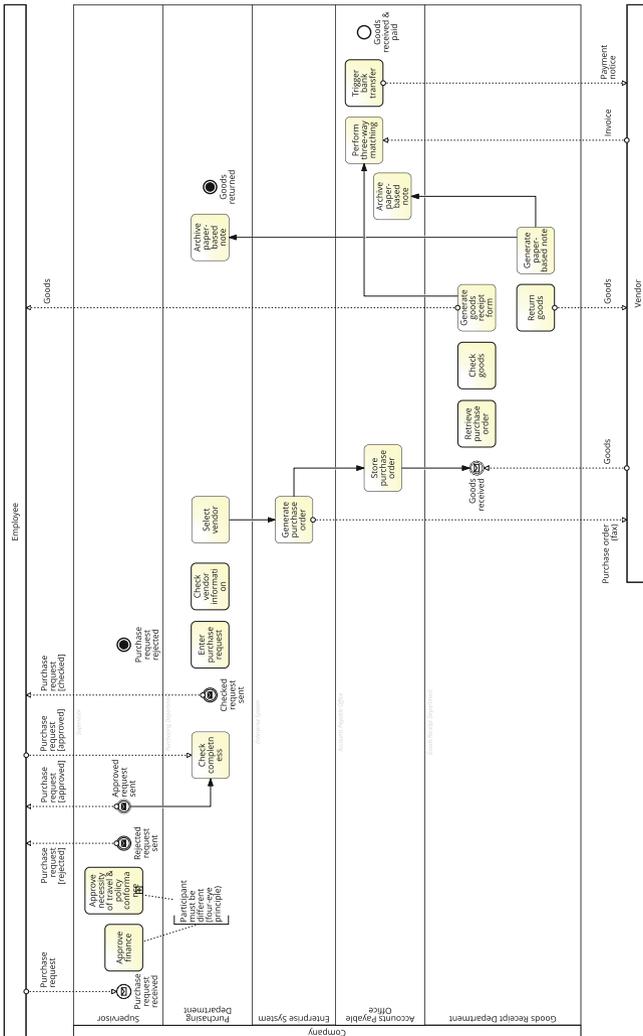
In conclusion, we would take about half the time if conducting workshops instead of interviews.

**Solution 5.7** We take the perspective of the company and consider the employee as the customer. Accordingly, we identify one start event, namely “Request for purchase received”, and three end events, namely “Goods received & paid” (positive outcome), “Purchase request rejected”, and “Goods returned” (negative outcomes).

**Solution 5.8** We identify 16 main activities and four intermediate events. As for the activities, given that the approval for the necessity of purchase and for the conformance to the company’s policies are done by the same supervisor, we can capture these two approvals with a collapsed sub-process. As for the events, we use three intermediate throwing message events to communicate the results of the checks done by the supervisors and by the purchasing department back to the employee, and one intermediate catching message event to model the receipt of the goods.



**Solution 5.9** We identify one pool for the employee, one for the vendor, and one for our company. The latter includes the following lanes: supervisor, purchasing department, enterprise system, accounts payable office, and goods receipt department. In the lane for the supervisor we add a text annotation to specify that a four-eye principle applies to the two approval activities (“Approve finance” and “Approve necessity of purchase & policy conformance”). Activity “Archive paper-based note” is performed by both the purchasing department and by the accounts payable office.







**Solution 5.12** We can observe the following behavioral anomalies:

- On the top-left structure we have a *lack of synchronization*, because an AND-split is followed by an XOR-join. The two tokens created from the AND-split are not synchronized by the XOR-join, leading to two tokens being placed on the arc emanating from the XOR-join.
- On the bottom-left structure we have a *deadlock*. The AND-join requires a token on each of its incoming arcs. However, XOR-split will create a token only on one of its outgoing arcs leading to the process getting stuck at the AND-join, waiting for a second token to arrive.
- On the top-right structure, if the OR-split is followed by an XOR-join we may get a *lack of synchronization*. This occurs if multiple tokens are generated by the OR-split (one per branch). Similarly, there is a potential *deadlock* if the OR-split is followed by an AND-join. The deadlock occurs if the OR-split sends only one token out.
- On the bottom-right structure, there is an XOR-join used as the entry to the loop, while the loop exit is modeled with an AND-split. This has the consequence that one token will remain trapped in the loop at each iteration of the loop, causing a *livelock*. Meantime, each time the AND-split is reached, a second token is created and put on the split's outgoing arc, leading to a *lack of synchronization* (more than one token will be on this latter arc).

**Solution 5.13** This model is unsound because two soundness properties are violated. First, if the exception flow emanating from the boundary event is taken, two tokens will be put on the top branch of the AND-split. This lack of synchronization leads to improper completion, as two tokens will reach the same end event. Second, the model will deadlock at the AND-join if the middle branch out of the OR-split is not taken. This violates the option to complete property.

The model can be made sound by replacing the AND-join with an OR-join and by enclosing the two parallel activities “Manufacture custom product” and “Order confirmation” into an expanded sub-process, to which the boundary event “Change request” is attached. This way, if a request for change is received both the order confirmation and the product manufacturing will be interrupted, preventing any token from proceeding forward if a token is coming back through the exception flow.

**Solution 5.14** Products should be stored in either Amsterdam or Hamburg. However, the model also allows products not to be stored in any of the two warehouses, if the top branch out of each of two XOR-splits is taken. This leads to an invalid statement, so the model is semantically incorrect.

**Solution 5.15** The model is semantically incorrect for the following reasons. First, there are no activities to plan the shipment and to check the customer type and shipment status. This means that any process instance of this model that leads to the fulfillment of an order is invalid.

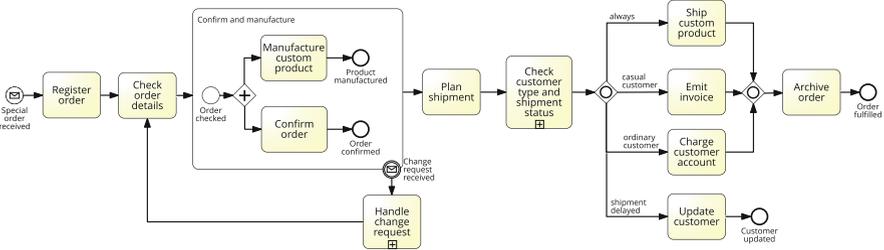
Second, after the handling of a change request, the process should resume from the order checking, but the model suggests that the order registration is

also to be repeated. Moreover, the receipt of a change request only interrupts the order confirmation, but it should also interrupt the manufacturing of the product. Therefore, all instances that lead to a change request are invalid.

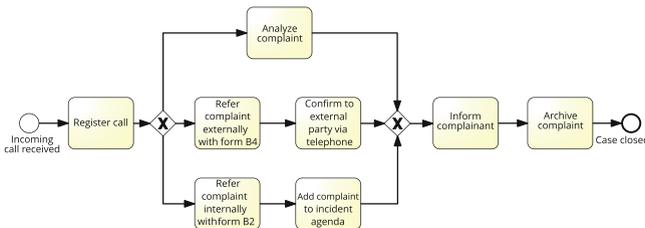
Finally, the model is incomplete as it does not cover the case of ordinary customers, whose account is to be charged before the order can be archived.

**Solution 5.16** This model employs different labeling styles. For example, activities “Order registration” and “Checking order details” follow the action-noun style, while “Ship custom product” and “Emit invoice” follow the verb-object style. Moreover, the label of events “Confirmed” and “Fulfilled” lacks a reference to a business object (the order). The same applies to the boundary message event “Change request”, which in addition lacks the past-participle verb “received”. To improve the pragmatic quality of this model we need to homogenize the various labeling styles, e.g., using a verb-object style for activities and an object-verb style for events. The layout of this model is consistent with a left-to-right orientation, so there is no need to re-layout the model. Taking the results from Solution 5.15 as input, the resulting model is shown in Figure 5.19.

**Solution 5.17** The process model reveals various problems. Several elements with the same name are shown twice (end event and archiving activity), therefore G1 is violated. Also the control structure is very complicated and the model is not structured, violating G4. Finally, several activities do not follow the naming conventions of G6. The model can be reworked to the one in Figure 5.20 which is much simpler, yet semantically equivalent.



**Fig. 5.19** The process model for fulfilling special orders, syntactically and semantically correct, and of high pragmatic quality



**Fig. 5.20** The reworked complaint handling process model

## 5.7 Further Exercises

**Exercise 5.18** As the person responsible for the human resources department of a consultancy company, how would you develop the skills of your junior process analysts?

**Exercise 5.19** As a process analyst, how would you prepare for an interview with a domain expert for the loan assessment process in Solution 3.8 (page 111)? Consider three different domain experts: the process owner, the loan officer, and the financial officer.

**Exercise 5.20** As a process analyst working for a car insurer, you are engaged in a project that aims at improving the company's insurance claim registration process. The first step is to model the as-is process. You have interviewed a few representatives for three key roles participating in this process: a customer service representative from the customer service department, a claims handler from the claims handling department, and a claims manager. The relevant parts of the interview transcripts for each role are provided below.

Customer service representative:

“When I receive a claim from a customer, I first check it for completeness. If it is not complete, I ask the customer to complete the missing information and resubmit the claim. When receiving a complete claim, I register it and send it to the claims handling department. I then wait for a notification from the claims manager that a decision has been taken. After receiving this notification, I send a customer satisfaction survey to the customer. If the customer sends back a completed survey, I first add it to our customer satisfaction database. I then have a closer look at it to evaluate if the overall satisfaction indicated by that customer is at least 5 on a scale from 1 to 10. If it is, my job is done. If it is not, all that is left for me to do is to notify the claims manager. If, after sending out the survey to the customer I do not get a response within two months, I make a *no reply* entry in the customer satisfaction database”.

Claims handler:

“When I receive a claim from the customer service department, I first check whether the claimant has a valid insurance policy. If not, I inform the claimant that the claim is rejected due to an invalid policy. Otherwise, I evaluate the severity of the claim. Based on the outcome of this evaluation, I send relevant forms to the claimant. I also check whether the form is complete. Only if the form is complete, I register the claim in the claims management system. Otherwise, I ask the claimant to update and complete the form. Upon receiving the updated form, I check it again for completeness. After the claim is registered, I start evaluating it as either simple (for minor car accidents) or complex (for major car accidents). When a claim is complex, I need to additionally retrieve the corresponding car accident report from a police reports database. Based on the claim, and on the police report if required, I calculate an initial claim estimate and create an action plan. Finally, I send both the initial claim estimate and the action plan to the claims manager”.

Claims manager:

“After receiving an initial claim estimate and action plan from the claims handling department, I make a final decision. Depending on the outcome of the decision (accept or reject), I notify the customer about my decision. I then update the claim file to record this

decision and notify the customer service that a decision has been taken. After that, there are two possibilities:

- I receive a notification from the customer service that the results of a customer satisfaction survey indicate that the overall satisfaction of the customer is very low (i.e., less than 5). In this case, I retrieve the corresponding survey and claim from our databases. I analyze them thoroughly to identify whether our internal operations could have been done differently, or could be improved in the future to better satisfy our customers. Finally, I send a letter to the claimant to apologise and promise to provide better services in the future.
- I do not hear back from the customer service within two months. In this case, no further action is required from me.”

Next, you took an active role in observing how this process works by acting as the claimant. Using a fake identity (in agreement with the process owner), you triggered this process several times and came up with the following observations.

#### Claimant:

The claimant completes a claims form and submit it to the customer service of the car insurer. Then the claimant has to wait for a response, which can be either of the following:

- Notification from customer service of the approval of my claim; in this case the claimant does not have to do anything further.
- Request from customer service to provide missing information on the forms, in which case the claimant updates the form and resends it to claims handling.
- Rejection from claims handling; in this case the claimant does not proceed any further with his or her claim.

After submitting a completed form to the claims handling department, the claimant waits for the claims manager to send him or her the final decision about the claim. After that, the claimant receives a customer satisfaction survey from the customer service. The claimant may choose to simply ignore this form. He or she may also choose to fill it out (typically the claimant does so when he or she is not satisfied with the service) and sends it back to the customer service. In this case, the claimant may receive a letter of apology from the claims manager within two months; otherwise the claimant is done.

Using the information above, create a draft BPMN model of the as-is claim registration process. This draft will then be validated with the people that have been interviewed before sign-off by the process owner. Make appropriate assumptions.

*Acknowledgement* This exercise is adapted from a similar exercise developed by Wasana Bandara, Queensland University of Technology.

**Exercise 5.21** As a process analyst working for a financial institution, you are engaged in a project that aims at improving the company’s credit application process. The first step is to model the as-is process. You have interviewed a few representatives for three key roles participating in this process: customer service, corporate risk assessor and risk management. The relevant parts of the interview transcripts for each role are provided below.

#### Customer service:

“After I receive a credit application from the customer, I check if the application is complete. If the application is incomplete, I send a request for clarification to the customer. Once I

receive this clarification, I check the application again for completeness. When I assess the application as complete, I pass it on to a corporate risk assessor. I then prepare some further marketing material (e.g., a selection of investment options) for the customer. After that, I will eventually receive one of the following:

- a A notification of approval from the corporate risk assessor,
- b A notification of rejection from the corporate risk assessor, or
- c A request for clarification from the risk manager.

In case of (a), I send a credit approval together with the marketing material to the customer, after which the process is finished for me. In case of (b), I send a credit rejection, after which the process is finished for me. In case of (c), I send a request for clarification to the customer. After receiving the clarification, I pass it on to the risk manager. I will then receive again one of the three documents listed above”.

#### Corporate risk assessor:

“When I receive a credit application from the customer service, I first check it. Afterwards, I send it to the risk manager, from whom I then receive either a notification of approval or a notification of rejection. In both cases, I forward the notification to the customer service, after which the process is finished for me”.

#### Risk manager:

“After receiving a credit application from the corporate risk assessor, I check it for completeness. If it is not complete, I send a request for clarification to the customer service. After the customer service responds with a clarification, I check the credit application again. Once an application successfully passes the completeness check, I assess its content. There are three possible outcomes of this assessment:

- The credit application satisfies our criteria for approval. In this case, I send a notification of approval to the corporate risk assessor. Then I formally authorize the credit in our IT systems, after which the process is finished for me.
- The credit application does not satisfy our criteria for approval. In this case, I send a notification of rejection to the corporate risk assessor, after which the process is finished for me.
- Some information in the application is unclear. In this case, I send a request for clarification to the customer service. After receiving the clarification, I assess the content of the credit application once again. This leads to one of the three outcomes listed here”.

Next, you took an active role in observing how this process works by acting as the customer. Using a fake identity (in agreement with the process owner), you triggered this process several times and came up with the following observations.

#### Customer:

To apply for credit, the customer needs to fill out a credit application and send it to the financial institution. They will eventually get a response, which can be either:

- A credit approval with additional marketing material or a credit rejection. In these two cases, the process is finished for the customer.
- A request for clarification. In this case, the customer can proceed by preparing a clarification and sending it to the financial institution. After that, he or she will get a response that may be a credit approval with additional marketing material, a credit rejection, or again a request for clarification.

Using the information above, create a draft BPMN model of the as-is credit application process. This draft will then be validated with the people that have been interviewed before sign-off by the process owner. Make appropriate assumptions.

*Acknowledgement* This exercise is adapted from a similar exercise developed by Wasana Bandara, Queensland University of Technology.

**Exercise 5.22** How can the model in Figure 5.12a (page 186) be fixed without affecting the cycle, i.e., such that activities F, G, and E all remain in the cycle?

**Exercise 5.23** Consider the process model in Figure 5.21. Does this model suffer from soundness problems? If so, what behavioural rules does it violate? If the model is unsound, how can it be fixed without removing any activity?

**Exercise 5.24** Consider the process model for loan risk assessment of Figure 5.22. Does it suffer from soundness problems? If so, what behavioural rules does it violate? If the model is unsound, how can it be fixed without removing any activity?

**Exercise 5.25** Consider the model in Figure 5.23 with reference to the process for damage compensation described in Exercise 3.16 (page 113). Is this model valid and complete? If not, which statements are invalid and what is missing?

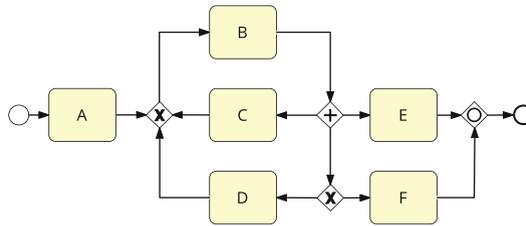


Fig. 5.21 A process model

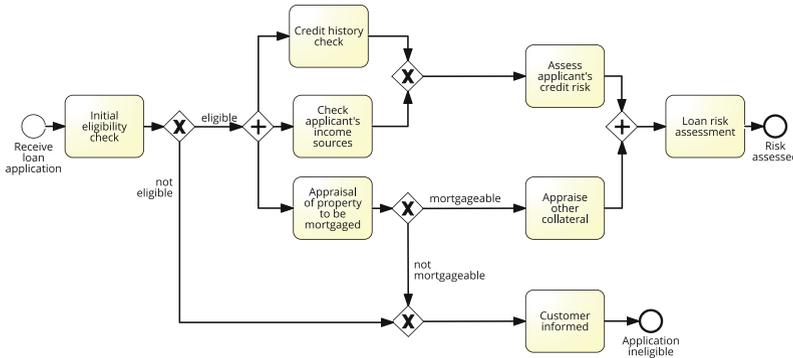


Fig. 5.22 A process model for loan risk assessment

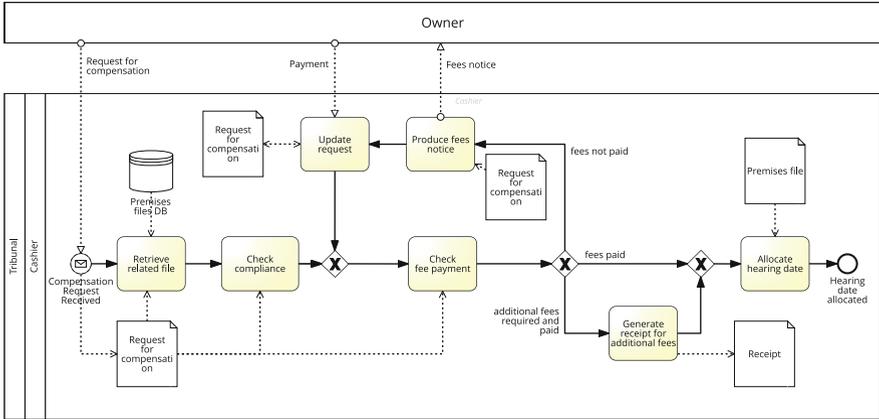


Fig. 5.23 A process model for damage compensation

**Exercise 5.26** Consider the model in Figure 5.24 with reference to the process for handling motor claims described in Exercise 3.20 (page 113). Is this model valid and complete? If not, which statements are invalid and what is missing?

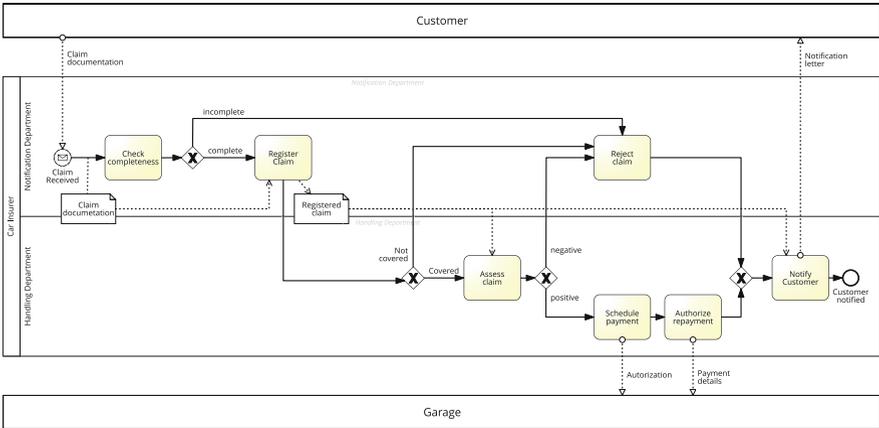


Fig. 5.24 A process model for handling motor claims

**Exercise 5.27** Consider the model in Figure 5.25 with reference to the process for handling claims described in Exercise 3.21. Is this model valid and complete? If not, what statements are invalid and what is missing?

**Exercise 5.28** Propose improved labels where appropriate for the model of Figure 5.22.

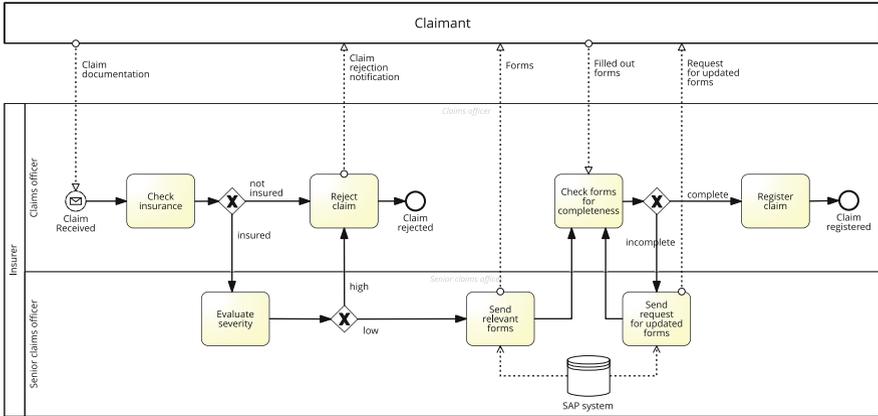


Fig. 5.25 A process model for handling claims

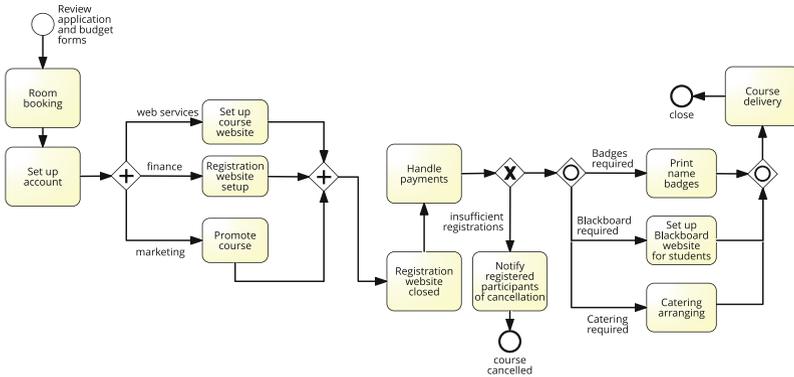


Fig. 5.26 A process model for organizing professional training courses

**Exercise 5.29** Consider the process model of Figure 5.26. This model refers to a process for organizing professional training courses.

1. Is the model semantically correct?
2. What modeling conventions should be enforced to make this model easier to understand and maintain?
3. Rewrite this model by taking into account the observations on semantic and pragmatic quality made from the above two points.

*Hint.* For (1) you do not have any reference process description, so just use common sense.

**Exercise 5.30** Consider the sales campaign process model of Figure 5.27. Describe which 7PMG guidelines can be used to improve this model.

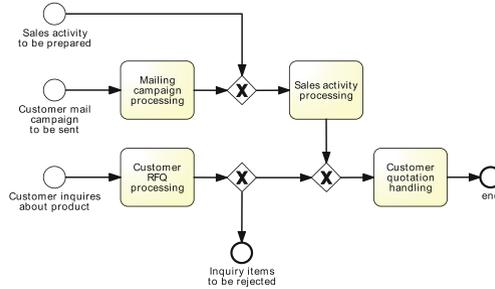


Fig. 5.27 A sales campaign process model

### 5.8 Further Readings

Detailed practical advice on all tasks of process discovery, and specifically information gathering and workshop organization, is provided in the book by Sharp & McDermott [161] and in that by Jeston & Nelis [71]. Other practical advice on workshop organization is offered by Verner [185] and by Stirna et al. [169]. Interview techniques are widely discussed as a social science research method for instance in the book by Berg & Lune [20] or in the book by Seidman [160]. General concerns regarding information gathering are discussed in the area of requirements engineering, for instance in the books by van Lamsweerde [181], Pohl [127], and Dick et al. [36].

Frederiks & van der Weide [48] discuss the skills required from process analysts, particularly when engaging in process discovery efforts. In a similar vein, Schenk et al. [157] and Petre [126] discuss the capabilities that expert process analysts (as opposed to novice ones) generally display when engaging in process discovery, while different facets of the facilitator role are explored by Rosemann et al. [147]. The five-factor personality structure model introduced on page 163 is proposed by Digman [37] and applied to system analyst and development by Clark et al. [26].

In this chapter, we emphasized manual process discovery methods, wherein process models are manually constructed based on information collected from various process stakeholders by means of interviews, workshops, or observation. As mentioned in Section 5.2.1, there is also a whole range of complementary techniques for automated discovery of process models from event logs. These techniques are presented in Chapter 11.

The modeling method introduced in Section 5.3 revolves around the discovery of activities and control-flow relations between activities. This family of approaches is usually called *activity-based modeling* [129]. An alternative approach to process modeling is known as *artifact-centric modeling* [27]. Here the emphasis is not on identifying activities, but *artifacts* (physical or electronic business objects) that are manipulated within a given process, such as a purchase order or an invoice in an order-to-cash process. Once these artifacts have been identified, they are analyzed in terms of the data that they hold and the states they go through during the process.

For example, a purchase order may go through states such as *received*, *confirmed*, *shipped*, and *invoiced*. These states and the transitions between them are called the *artifact lifecycle*. Discovering such lifecycles is the focus of artifact-centric process modeling. Several industrial applications have shown that this approach is particularly suitable for processes that exhibit significant amounts of variation, e.g., variation between business units, geographical regions, or types of customers.

The quality of conceptual models in general, and of process models specifically, has received extensive attention in the research literature. The *Sequal* framework introduced by Lindland et al. adapts semiotic theory, namely the three perspectives of syntax, semantics, and pragmatics, to the evaluation of conceptual model quality [91]. An extended and revised version of this framework is presented in the book by Krogstie [83].

Verification and validation of process models have also received extensive attention in the literature. Mendling [109] for example provides numerous pointers to related research. The verification of Workflow nets, another process modeling language, is specifically investigated by Van der Aalst [2] who connects soundness analysis of process models with formal properties of Petri nets.

In this chapter we listed the main structural rules of BPMN. The complete list of rules can be found in Silver's Method & Style website.<sup>2</sup>

The 7PMG discussed in this chapter originate from [110]. These guidelines build on empirical work on the relation between process model metrics on the one hand and error probability and understandability on the other hand [108, 111, 112, 123, 133, 136, 143, 144], and have been widely used in practice. The 7PMG are one of the available sets of modeling guidelines. For example, another set of guidelines are those by Becker et al. [18]. Moreover, research in the area of process model quality is still developing. So, as insights develop further, it is likely and favorable that these guidelines will be updated and expanded.

As a complement to process modeling guidelines and conventions, it is useful to also keep in mind potential pitfalls to be avoided in process modeling projects. For example, Rosemann [145, 146] draws a list of 22 pitfalls of process modeling, including a potential lack of strategic connection and getting lost in modeling details, to name but a few. His bottom line is that modeling success does not directly equate with business process success.

---

<sup>2</sup><https://methodandstyle.com/the-rules-of-bpmn>.