BPMN 2.0 for Beginners

An overview of BPMN 2.0

BPMN stands for Business Process Modeling Notation. BPMN is very similar to the concept of flowcharting that has been around since the 1980s. Like flowcharting BPMN modeling has the aim of allowing a person to map a workflow in such a way that it can be understood easily by other interested parties. BPMN is a language, and like any language the purpose is to facilitate communication. BPMN is designed to facilitate communication and understanding of business processes.

The “N” part of BPMN stands for “notation.” The graphical notation consists of designated symbols that represent actions, flows, or process behaviors. Visual symbols have been a part of the standard since the OMG released the first version of the BPMN specification in May 2004.

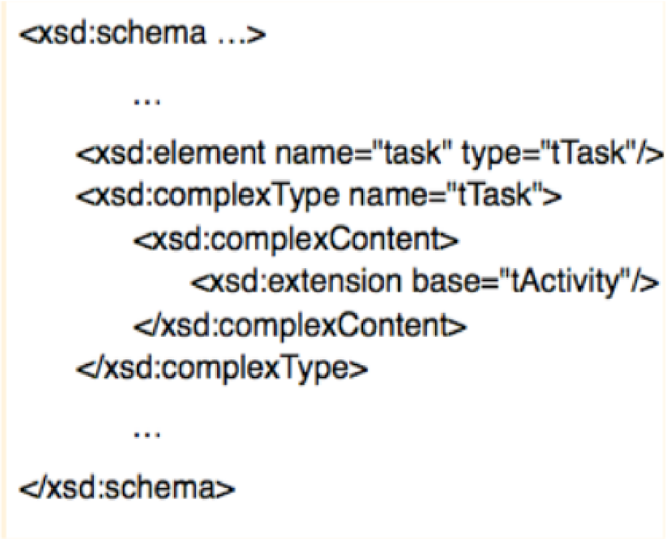
BPMN is not software, and it’s not “owned” by a business, but was developed by the OMG (Object Management Group) as a notation standard that can be understood by business analysts, technical developers, and project managers.

Image 1.0
BPMN is the universal language of process modeling.

The historical problem with flowcharting was that each process modeler could make up his own meanings and interpretations for the diagrams he produced. This had the undesired consequence that not everyone would interpret processes the same way. In other words, process communication was highly imperfect.

BPMN and more specifically BPMN 2.0 solved this problem of ambiguity by introducing a semantic layer in XML which transforms BPMN from just another pretty picture into something much more functional. By adding a defined XML definition to each BPMN object, the BPMN specification assures that each visual process model and all of the objects in the process model can mean only one thing.

Note: All BPMN images and code samples taken from the OMG Specification http://www.omg.org/spec/BPMN/2.0.2/PDF



```
<xsd:schema ...>

        ...

    <xsd:element name="task" type="tTask"/>
    <xsd:complexType name="tTask">
        <xsd:complexContent>
            <xsd:extension base="tActivity"/>
        </xsd:complexContent>
    </xsd:complexType>

        ...

</xsd:schema>
```

Image 1.1
Example Core XML schema.

Thanks to a common syntax many workflow modeling tools and process engines support BPMN. As a result many different software suites can share and edit processes designed with BPMN. Currently, BPMN is the most popular business process modeling language.

Many business professionals are comfortable with using BPMN to visualize workflows, but one of the biggest advantages of BPMN 2.0 is that the people who write code to automate processes understand BPMN 2.0 too. In other words, it's a great way to bridge the gap between business professionals and technical professionals. BPMN 2.0 makes it much easier for modeling tools, process engines, Business Activity Monitoring (BAM) tools, and other tools to exchange processes regardless of the specific vendor tool being implemented.

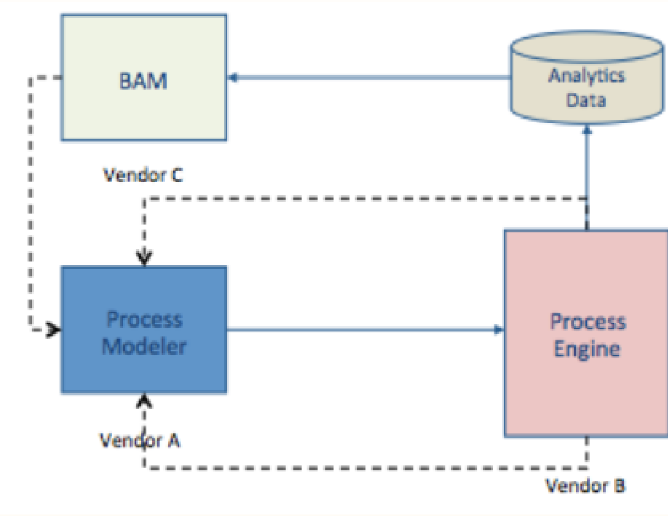


Image 1.2

BPMN 2.0 tries to solve the "round-tripping" problem of allowing process models to be exchanged seamlessly back and forth between process tools from different vendors.

Workflows are essentially what they sound like: the flow of tasks within an organization. Workflows may be completely manual (like the process of making a sandwich), they may be completely automated (like the process of inputting data into equations and making calculations), or they may be a combination of both (like the process of creating, reviewing, and approving a Purchase Request).

Developing workflows is a process of capturing all relevant information that goes into the process: who is involved, what they're responsible for, how tasks are handed off, which tasks are manual and which are automated. Typically, the first stage in business process modeling of workflows is the gathering of relevant information. Next, the process and workflow are modeled. Finally, the new workflow is tested, implemented, monitored, and modified as necessary.

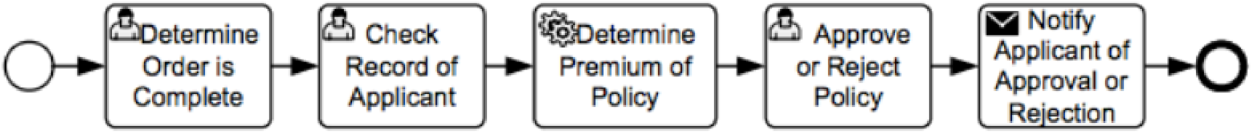

Determine Order is Complete
Check Record of Applicant
Determine Premium of Policy
Approve or Reject Policy
Notify Applicant of Approval or Rejection

Image 1.3

Stakeholders need to be interviewed to understand the true nature of a particular workflow..

Developing workflows is a process of capturing all relevant information that goes into the process: who is involved, what they're responsible for, how tasks are handed off, which tasks are manual and which are automated.

Many critics of BPMN 2.0 complain that BPMN is too complicated to learn. Even if the language itself is designed to be unambiguous by including a singular semantic layer, there are simply too many objects in BPMN for the standard to be useful.

These critics often point to images from the BPMN 2.0 OMG specification such as this matrix of events to emphasize their point. These critics often favor other flavors of modeling tools and methodologies.

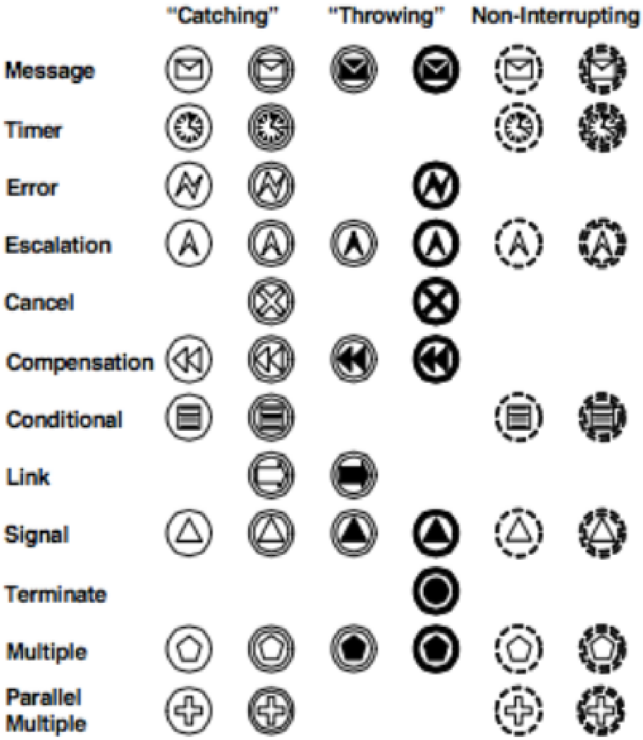


Image 1.4

There are many different possible types of events in BPMN 2.0.

What the critics fail to mention is that most processes do not require the modeler to know the entire specification. In fact, most models don't use more than a handful of the most common process elements.

In fact, BPMN actually consists of only 3 main elements:

1. Events
2. Activities
3. Gateways

Yes, that's right. There are just three main elements in BPMN! Ok, well, let's throw in a fourth element just so we can connect the other three - the **Sequence Flow** (the black lines with arrows that connect everything together).

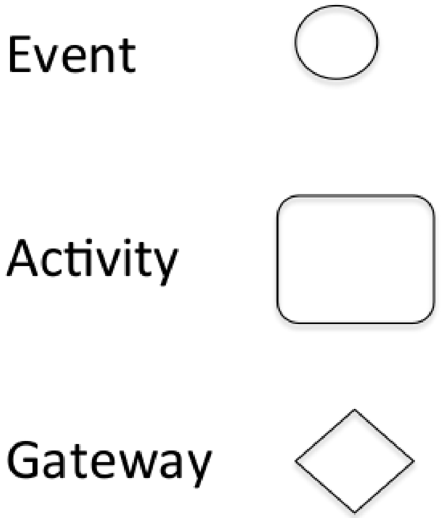


Image 1.5
The 3 Main Modeling Elements in BPMN 2.0.

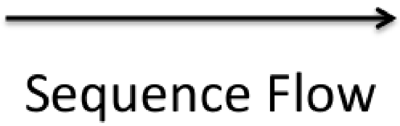


Image 1.6
There are many different possible types of events in BPMN 2.0.

The Descriptive Modeling Subclass is a bit of a mouth full. Some people call these the Level I elements. Basically, these are all the elements you probably need to learn to become relatively proficient in BPMN.

At the very least if your job involves working with processes then you should know these elements:

- Pool and lane
- Sequence flow and message flow
- 3 types of activity: user task, service task, and task (none)
- SubProcess (expanded and collapsed)
- CallActivity
- 2 types of gateway: XOR gateway and parallel gateway
- 3 start events: Message, Timer, and None start
- 2 end events: Message and None
- Data object and data store
- Text annotation and regular data association

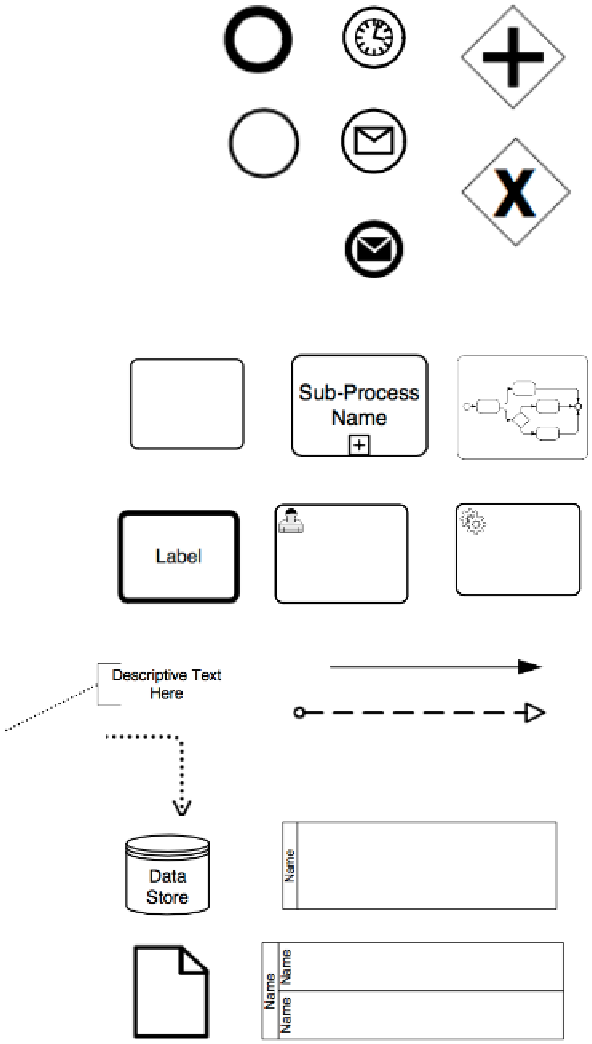


Image 1.7
Descriptive Modeling Subclass Elements.

Let’s take a quick look at the three main types of events:

1. **Start** – indicates where a particular Process will start.
2. **Intermediate** – occur between the start and end events and affect the flow of the process.
3. **End** – indicates where the process ends.

These are pretty simply. The important thing to remember is that there are variants. For example, if you add the picture of an envelop to any of these three events, you would have a message event. You could start a process by the arrival of a message, send a message (or catch a message) with an intermediate event somewhere in the middle of the process, or you could send a message at the end of a process.

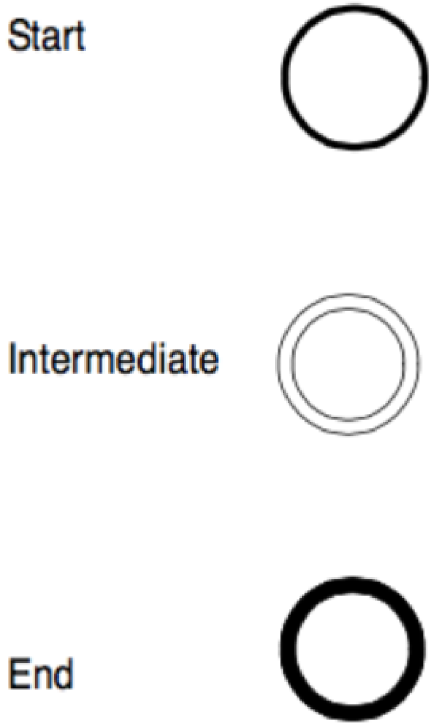


Image 1.8

Events are simple but there can be lots of variants.

An Activity is a generic term for work that an actor performs in a Process. What many people do not realize is that an Activity can be atomic or non-atomic (compound). In other words, an Activity can be a Task or it can be a Sub-process (a process within the main process).

The fun part of activities is that you can also specify various types of activities. For example, tasks can be manual tasks, human tasks, message tasks (have the same function as a message event) script tasks, or service tasks (indicating, perhaps, that work is performed by a web service.

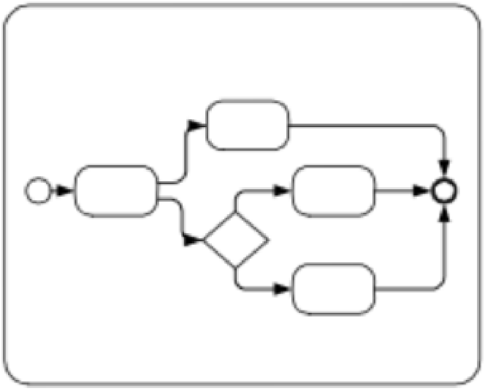


Image 1.10

Expanded Sub-Process.

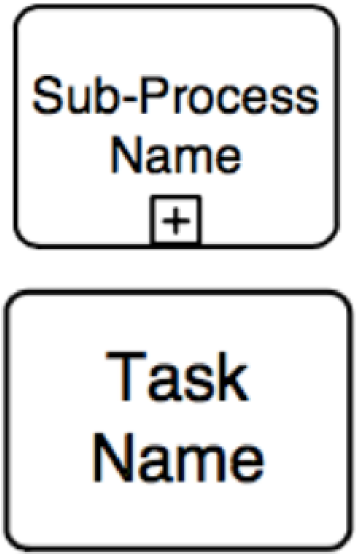


Image 1.9

Activities can be atomic (Task) or non-atomic (Sub-Process).

Events and Activities are pretty easy to understand. The real magic in a process, though, happens at the gateways. Gateways can tell a process to continue exclusively down one path vs. another or instruct participants in the process to begin working in parallel. In still other scenarios, there could be multiple pathways in which some of the pathways only become active under certain circumstances.

There are some pretty funky gateways out there. My personal favorite is the Event-based gateway (also known as a Race Gateway) which basically creates a race between two alternative paths. For example, a company might have a cancelation policy where a customer can cancel an emitted insurance policy within 1 week; otherwise, it will print and mail the policy. In this case, either we get the cancelation within a week or we go down the path of emitting the policy. Wow – that is elegant.

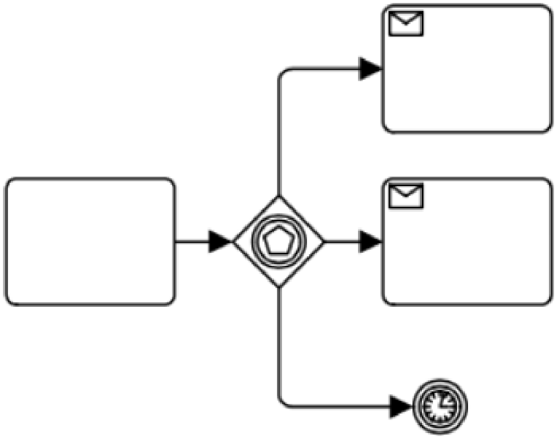


Image 1.11
Event-based Gateways are a difficult to master but elegant feature of BPMN 2.0.

Don't worry if you think the Event-based gateway is difficult; it is. It is also not so common.

Here are the 3 you should know:

- **Exclusive** - This Decision represents a branching point where Alternatives are based on conditional Expressions contained within the *outgoing* Sequence Flows. Only one of the Alternatives will be chosen.
- **Parallel (Fork)** - BPMN uses the term "fork" to refer to the dividing of a path into two or more parallel paths (also known as an AND-Split). It is a place in the Process where activities can be performed concurrently, rather than sequentially.
- **Inclusive -** This Decision represents a branching point where Alternatives are based on conditional Expressions contained within the outgoing Sequence Flows. In some sense it is a grouping of related independent Binary (Yes/No) Decisions. Since each path is independent, all combinations of the paths MAY be taken, from zero to all. However, it should be designed so that at least one path is taken. A Default Condition could be used to ensure that at least one path is taken.

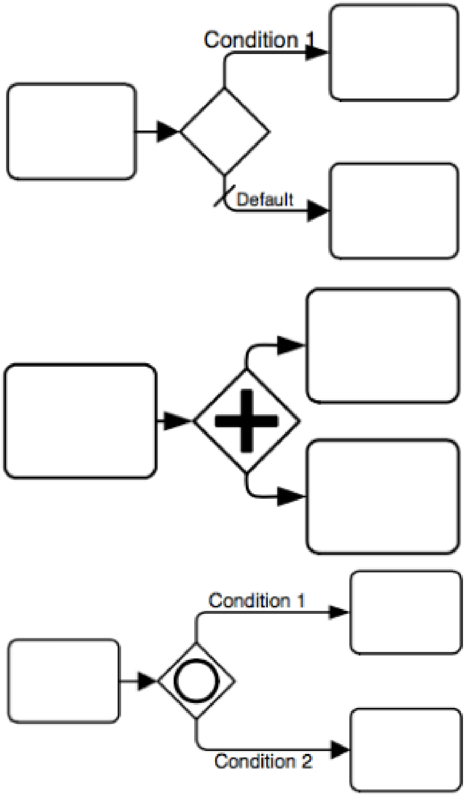


Image 1.12
The 3 Most Common Gateway Types.

It is worth spending another few minutes on another important element in the Descriptive Subclass – Swimlanes, pools, and my favorite, the black box pool.

A **Pool** is the graphical representation of a *Participant* in a Collaboration. It also acts as a “swimlane” and a graphical container for partitioning a set of Activities from other Pools, usually in the context of B2B situations. A Pool MAY have internal details, in the form of the Process that will be executed. Or a Pool MAY have no internal details, i.e., it can be a "black box."

A **Lane** is a sub-partition within a Process, sometimes within a Pool, and will extend the entire length of the Process, either vertically or horizontally. Lanes are used to organize and categorize Activities.

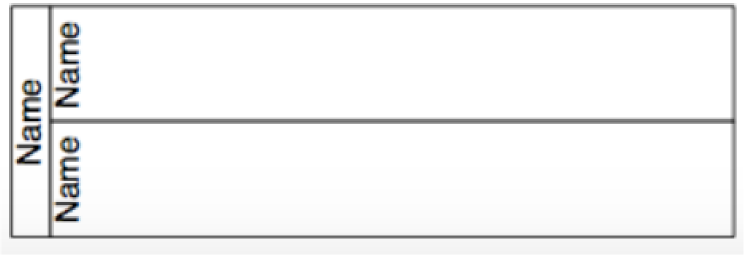


Image 1.13
Pool with 2 lanes.

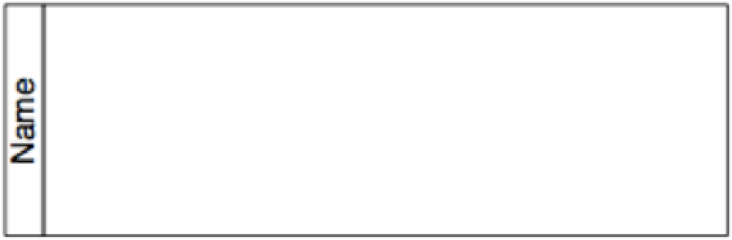


Image 1.14
Empty Pool.

If you get through the Descriptive Element Subclass and you are still hungry for more BPMN, there are some other elements that are fun to learn and quite useful. We will just show you one more in this eBook. For others, head to the OMG specification and read on.

One of the most useful things about BPMN is the exception handling. The simple fact is that most business processes don't follow their "happy path." The happy path is the way the process flows when there is no exception. However, when there is an exception to the workflow, it helps to know how to deal with it.

One way to handle exceptions is with an *Exception Flow. An Exception flow* occurs outside the *normal flow* of the Process and is based upon an Intermediate Event attached to the boundary of an Activity that occurs during the performance of the Process. The exception flow could be based on a timer (Timer Boundary Event), the arrival of a message, or an error. When one of these events happen during the execution of the activity, the activity "throws off" a new path. Pretty cool, right?

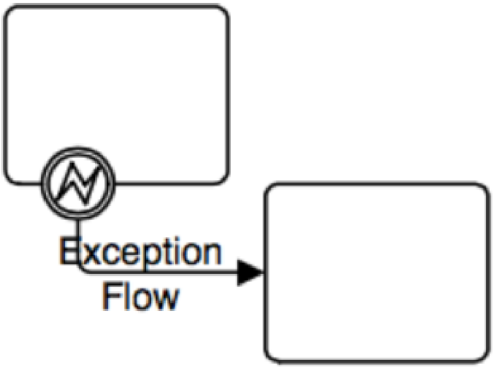


Image 1.15
Exception Handling is elegant in BPMN.

Date: January 2011

WE SET THE STANDARD™

Business Process Model and Notation (BPMN)

Version 2.0

BPMN 2.0 has been around for a few years now, and it has several new features and advantages compared to older versions. Versions prior to 2.0 are less consistent, not as technologically mature, and not as amenable to automation. Many diagrams made under older BPMN versions are outdated, and are far more helpful and meaningful when rendered under BPMN 2.0.

BPMN 2.0 is the most major revision of BPMN so far, with improvements having been done to both visual BPMN elements and “interior” BPMN elements like semantics.

The diagrams rendered under BPMN 2.0 are similar to those created under older versions, but BPMN 2.0 offers a number of new elements as well, including:

- Multiple parallel events
- Escalation events
- Different task types (service, send, receive, etc.)
- Global activities
- Sequential multiple instances of tasks and associated subtasks
- Event-based instantiating gateways
- Event-based sub-processes
- New types of data objects (data input, output, message, collection, etc.)

Shapes and symbols that were used under earlier versions of BPMN are still used under BPMN 2.0. One major change is that BPMN 2.0 adopts the XML interchange format, which makes it easier to turn a model and its notation into an automated, executable process. That means that workflow software vendors using BPMN can more easily take BPMN 2.0 input and automate it.

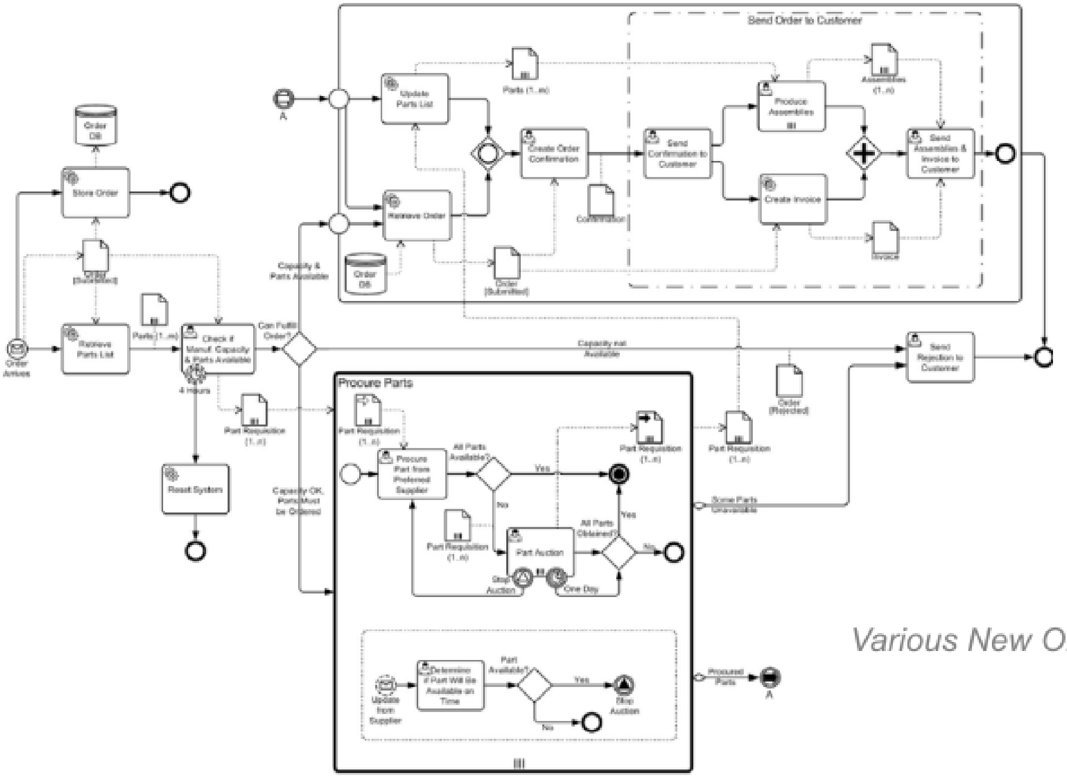


Image 1.16
Various New Object Types in BPMN 2.0.

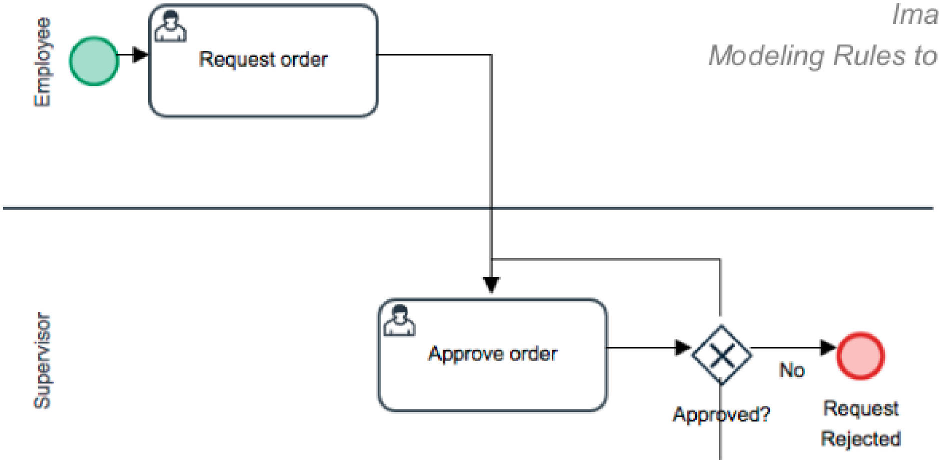
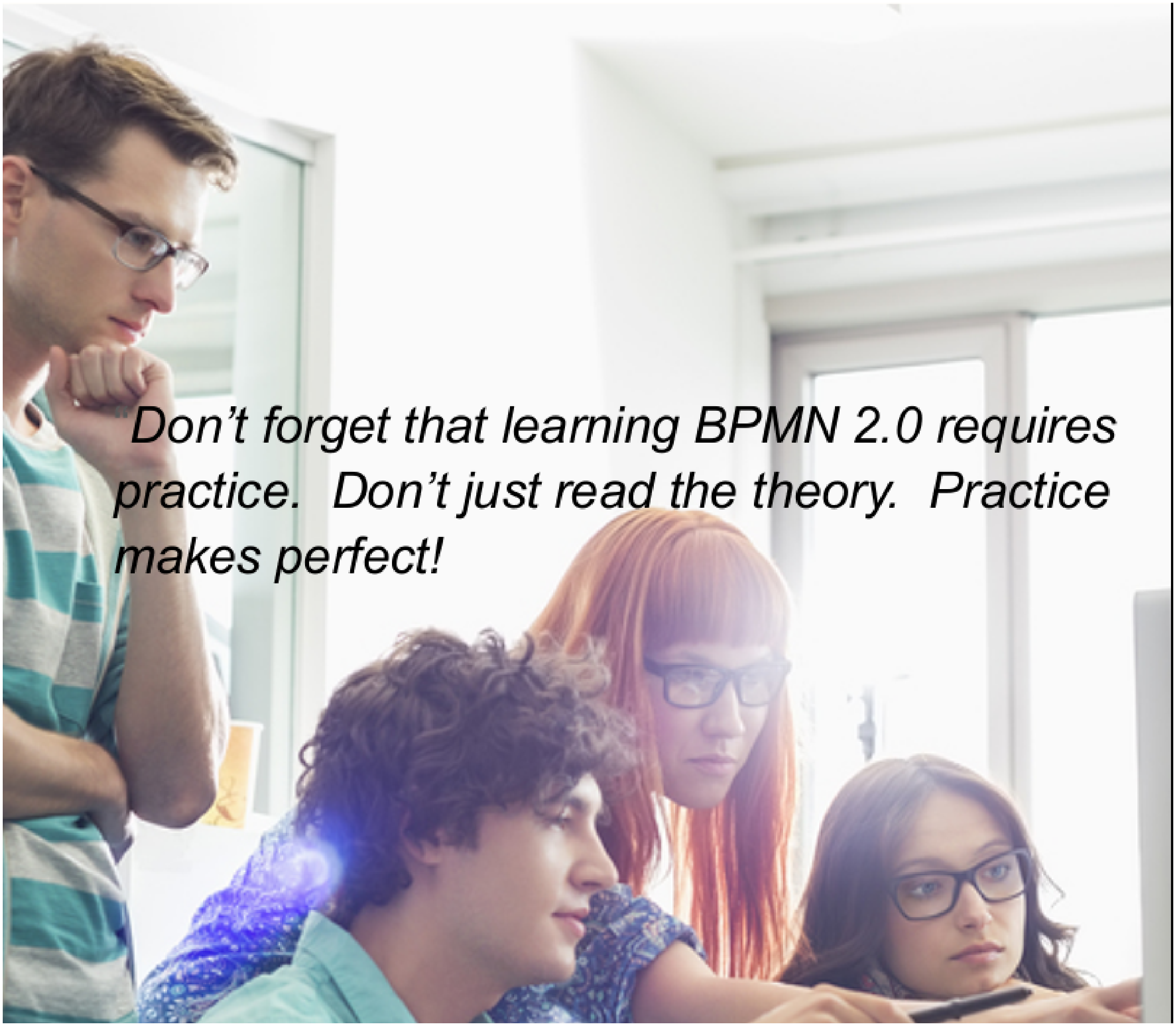


Image 1.17
Modeling Rules to Follow.

You should follow a few best practices when modeling in BPMN 2.0:

1. Make the BPMN diagram accurately reflect process logic, complete with labels on activities, sub-processes, intermediate events, and gateways
2. Show explicitly how exceptions are handled
3. Make models hierarchical (and hence easier to view as a whole) with processes and sub-processes expanding from a top-level diagram
4. Label task types such as “user” (a task a human accomplishes) or “service” (an automated task)
5. Represent actions in a verb-noun format (validate order, verify credit, etc.)
6. Consistently use message flows to indicate business context of activities and events in a workflow
7. Use Black Box Pools to show external participants and their relation to the process



“Don’t forget that learning BPMN 2.0 requires practice. Don’t just read the theory. Practice makes perfect!

In short, the best thing about BPMN 2.0 is that it provides a common language for technical and non-technical users, allowing both types of personnel to have the precision, flexibility, and understanding they need to build processes correctly. Because BPMN 2.0 offers this shared language, collaboration is facilitated, and more new and flexible applications are possible. The system of shared standards makes communication and understanding both within an organization and from one organization to another clearer and easier.

Perhaps your organization is reluctant to try BPMN at all, let alone BPMN 2.0. But there's really no reason to put it off. You don't have to be an expert to make good use of BPMN. Learning the reasoning behind BPMN may require climbing a learning curve, but once you do, a whole new world of workflow automation possibilities opens up. What's more, your technical people will have an easier time translating them into actual automated processes.

Bridging the technical gap is one of the most important functions of BPMN 2.0.

The ProcessMaker Designer allows you to build BPMN 2.0-compliant processes, letting you have a standard notation that will be understood by all your business stakeholders within and outside of your organization. With the ProcessMaker Designer, your business managers, business analysts, and technical developers will all speak the same language, and the “tech gap” between non-technical and technical users narrows considerably.

ProcessMaker is open source and cloud business process management and automation software that offers users power and flexibility. You can use any number of pre-built templates as a starting point, or you can develop workflows from the ground up if you want. Give it a try today.

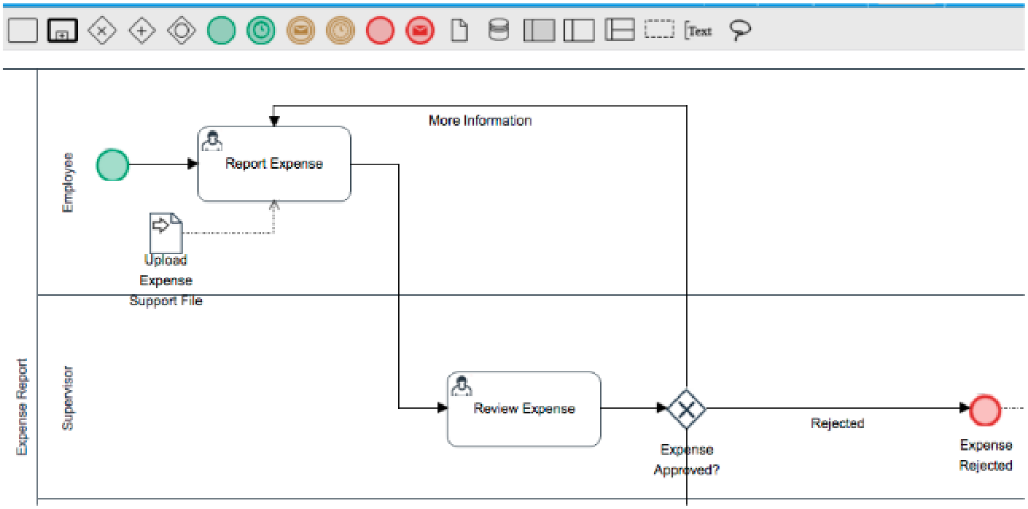


Image 1.18

With ProcessMaker Designer you can build BPMN 2.0 compliant processes.