

Handout zum Fachvortrag BPMN

Business Process Model und Notation

Dragonfly

Business Process Model und Notation:

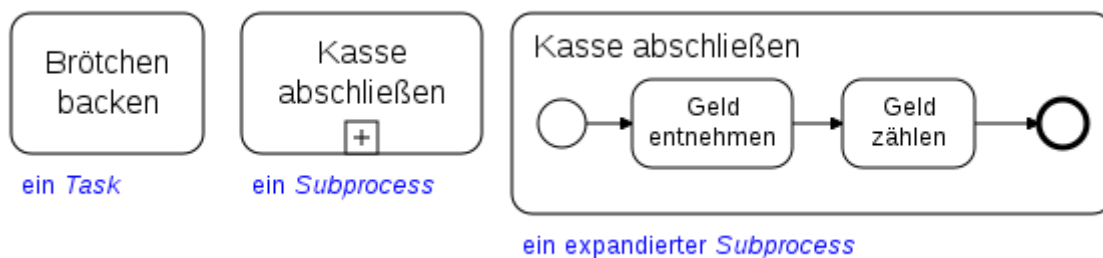
- Grafische Darstellung und Notation der Schritte im Geschäftsprozess
- Detaillierte Aktivitäten und Informationsflüsse
- Prozess-Orientiert (UML ist Objekt-Orientiert)
- BPMN und UML sind kompatibel miteinander
- Aktuelle Version: 2.0
- BPMN ist OMG Standard (Object Management Group)
- Standardisiertes, lizenzfreies Modell
- Mehrere Komplexitätslevel, für verschiedene Zielgruppen
- High level: Business-Leute, leicht zu verstehen
- Low level: IT-Leute, komplexer, detaillierter
 - Vereinfacht die Kommunikation zwischen Business und IT
- Low level BPMN kann zu ausführbarem Code übersetzt werden
 - Erlaubt Möglichkeiten zur Effizienzoptimierung von Geschäftsprozessen

Notation:

Flow Objects:

Activities:

- Beschreibt einzelne Aufgaben
- Einzelne Aktivitäten werden "Task" genannt, komplexe Aktivitäten "Sub-process"



Gateway:

- Sind Entscheidungspunkte
- Notation für Spaltungen oder Zusammenläufe
- Können abhängig von Events sein



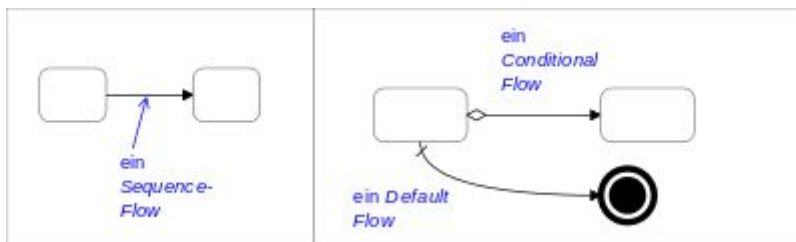
Events:

- Ereignisse in Geschäftsprozesse
- Eingeteilt in drei Klassen:
 - Position: Start, Intermediate, End-Event
 - Wirkung: Catching, Throwing
 - Art: Timer, Message, Exception etc.



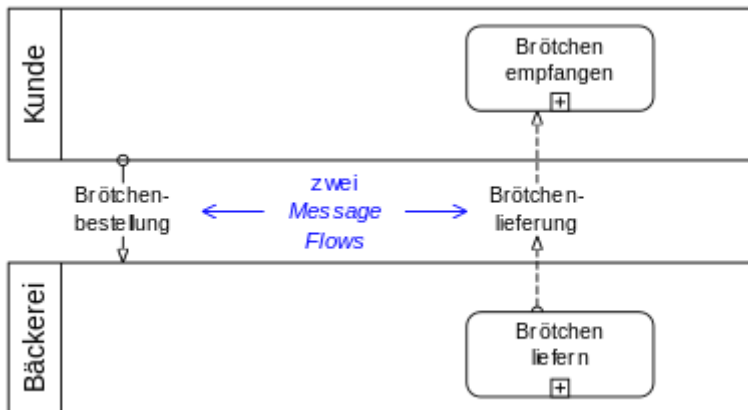
Connecting Objects:

- Sequence Flow: Verbinden Aktivitäten, Gateways und Events
 - Darstellung der Reihenfolge und Ablauf
- Conditional Flows gelten nur unter bestimmten Bedingungen
- Default Flows, wenn kein anderer Sequence Flow durchlaufen werden kann



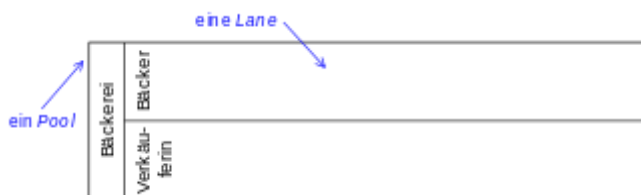
Message Flow:

- Austausch zwischen Lanes oder Pools
- Oder zwei Elemente Meldungen austauschen



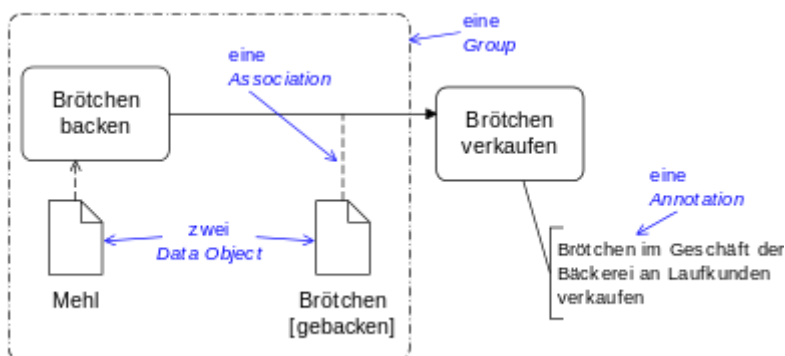
Pools und Swimlanes:

- Pool repräsentiert die Prozessbeteiligten (z.B.: eine Organisation)
- Lanes sind Unterteilungen des Pools über die gesamte Länge
- Lanes sind Teilnehmer (Benutzer, Benutzerrollen oder ein System)

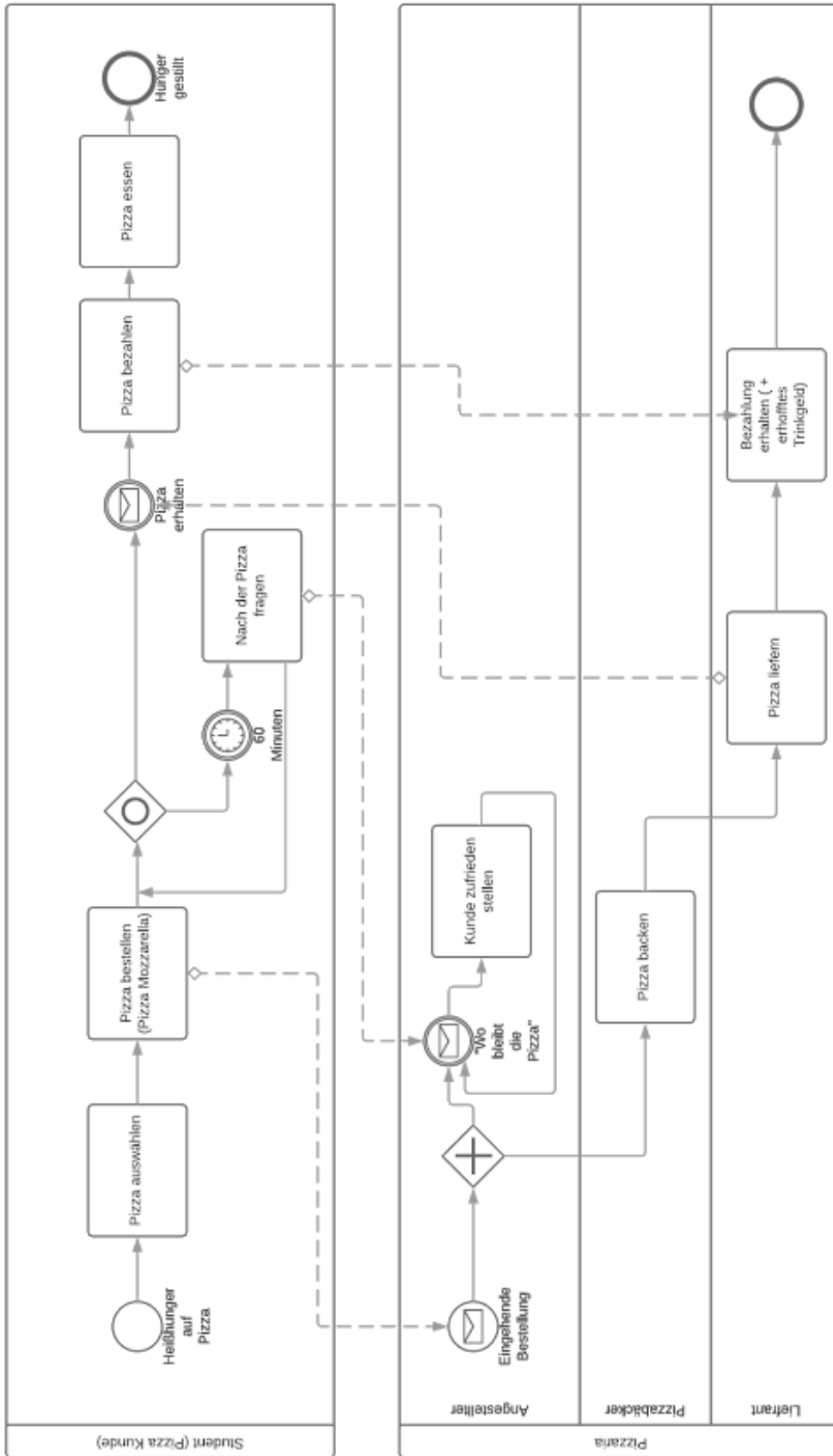


Artifacts:

- Annotation: Kommentar zu einem zugeordneten Element eines Geschäftsprozesses
- Data Object: repräsentiert ein Artefakt
 - z.B.: elektronische Objekte: Dokumente, Datensätze
 - z.B.: physische Objekte: Brötchen, Bücher
- Group: Hilfsmittel zur visuellen Zusammenfassung der Elemente



Pizza Bestellung Beispiel:



Activities

- Task**
A Task is a unit of work, the job to be performed. When marked with a symbol it indicates a Sub-Process, an activity that can be refined.
- Transaction**
A Transaction is a set of activities that logically belong together; it might follow a specified transaction protocol.
- Event Sub-Process**
An Event Sub-Process is placed into a Process or Sub-Process. It is activated when its start event gets triggered and can interrupt the higher level (interrupting) depending on the start event.
- Call Activity**
A Call Activity is a wrapper for a globally defined Task or Process raised in the current Process. A call to a Process is marked with a symbol.

Activity Markers

- Markers indicate execution behavior of activities:
- Sub-Process Marker**
 - Loop Marker**
 - Parallel (M) Marker**
 - Sequential (M) Marker**
 - Ad Hoc Marker**
 - Compensation Marker**

Task Types

- Types specify the nature of the action to be performed:
- Send Task**
 - Receive Task**
 - User Task**
 - Manual Task**
 - Business Rule Task**
 - Service Task**
 - Script Task**

Flow Types

- Sequence Flow**
defines the execution order of activities.
- Default Flow**
is the default branch to be chosen if all outgoing flows evaluate to false.
- Conditional Flow**
has a condition assigned that defines whether or not the flow is used.

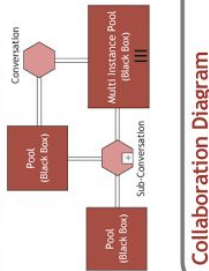
Gateways

- Exclusive Gateway**
When splitting, it routes the sequence flow to exactly one of the outgoing branches. When merging, it awaits one incoming branch to complete before triggering the outgoing flow.
- Event-based Gateway**
Sequence flow is routed to the subsequent event/task which happens first.
- Parallel Gateway**
When used to split the sequence flow, all outgoing parallel branches are activated simultaneously. When merging parallel branches it waits for all incoming branches to complete before triggering the outgoing flow.
- Inclusive Gateway**
When splitting, one or more branches are activated. All active incoming branches must complete before merging.
- Complex Gateway**
Complex merging and branching behavior that is not captured by other gateways.
- Exclusive Event-based Gateway (Instance)**
The occurrence of all subsequent events starts a new process instance.
- Parallel Event-based Gateway (Instance)**
The occurrence of all subsequent events starts a new process instance.

Conversations

- Conversation** defines a set of logically related message exchanges. When marked with a symbol it indicates a Sub-Conversation, a compound conversation element.
- Call Conversation** is a wrapper for a globally defined Conversation or Sub-Conversation. A call to a Sub-Conversation is marked with a symbol.
- Conversation Link** connects Conversations and Participants.

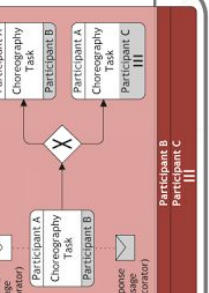
Conversation Diagram



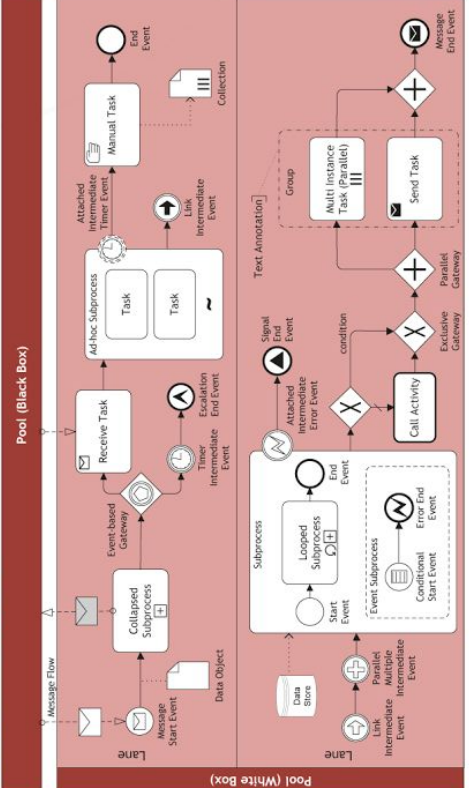
Choreographies

- Participant A Choreography Task**
- Participant B Choreography Task**
- Participant C Choreography Task**
- Sub-Choreography**
- Call Choreography**

Choreography Diagram



Collaboration Diagram



Events

	Start	Intermediate	End
None			
Message: Receiving and sending messages.			
Timer: Cyclic timer events, points in time, time spans or timeouts.			
Escalation: Escalating to an higher level of			
Conditional: Reacting to changed business conditions or integrating business rules.			
Link: Off-page connectors. Two corresponding link events equal a sequence flow.			
Error: Catching or throwing named errors.			
Cancel: Reacting to cancelled or triggering cancellation.			
Compensation: Handling or triggering compensation.			
Signal: Signalling across different processes. A signal thrown can be caught multiple times.			
Multiple: Catching one out of a set of events. Throwing all out of a set of parallel events.			
Parallel Multiple: Catching all out of a set of parallel events.			
Terminate: Triggering the immediate termination of a process.			

Gateways

- Exclusive Gateway**
When splitting, it routes the sequence flow to exactly one of the outgoing branches. When merging, it awaits one incoming branch to complete before triggering the outgoing flow.
- Event-based Gateway**
Sequence flow is routed to the subsequent event/task which happens first.
- Parallel Gateway**
When used to split the sequence flow, all outgoing parallel branches are activated simultaneously. When merging parallel branches it waits for all incoming branches to complete before triggering the outgoing flow.
- Inclusive Gateway**
When splitting, one or more branches are activated. All active incoming branches must complete before merging.
- Complex Gateway**
Complex merging and branching behavior that is not captured by other gateways.
- Exclusive Event-based Gateway**
Each occurrence of a subsequent event starts a new process instance.
- Parallel Event-based Gateway (Instance)**
The occurrence of all subsequent events starts a new process instance.

Data

- Data Object** represents information flowing through the process, such as business documents, e-mails, or letters.
- Data Association** is used to associate data elements to Activities, Processes and Global Tasks.
- Data Store** is a place where the process can read or write data, e.g., a database or a filing cabinet. It persists beyond the lifetime of the process instance.
- Data Input** is an external input for the entire process. A kind of input parameter.
- Data Output** is data result of the entire process. A kind of output parameter.
- Data Association** is used to associate data elements to Activities, Processes and Global Tasks.
- Data Store** is a place where the process can read or write data, e.g., a database or a filing cabinet. It persists beyond the lifetime of the process instance.