

Business Processes: Modeling, Simulation, Execution

Week 6: Process Execution

An openHPI Course

Prof. Dr. Mathias Weske
(Week 6 conducted by Camunda Teaching Team)

Hasso Plattner Institute, University of Potsdam
Business Process Technology

Week 6

Process Execution

- 6.1 Fundamentals of technical process models
- 6.2 Adding executable attributes (tasks)
- 6.3 Adding executable attributes (gateways and events)
- 6.4 Deploy and run a process
- 6.5 Demo 1: Add attributes to the model
- 6.6 Demo 2: Deploy the model to the engine

Video Clip 6.1

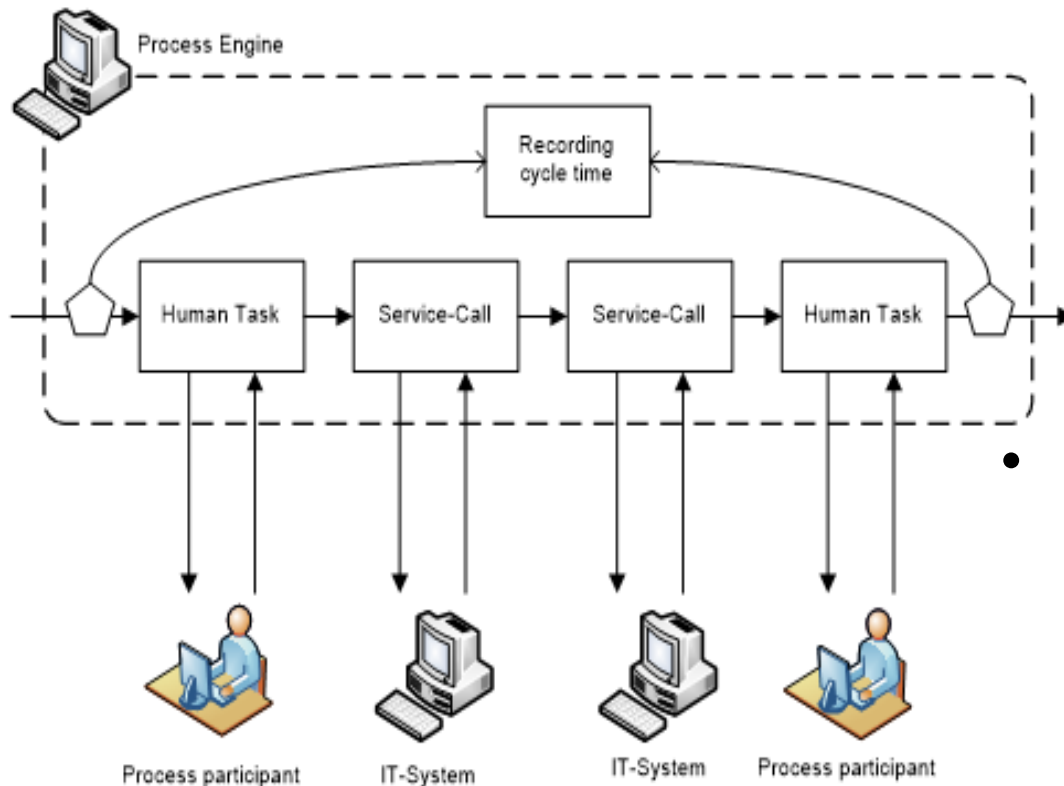
Fundamentals of technical process models

Business Process Modeling, Simulation and Execution:
An openHPI Course by Mathias Weske

Different application scenarios of BPMN

- Organizational Projects
 - Process Documentation
 - Achieve more transparency inside an organization
 - Process Design
 - Improve current business processes
 - IT Projects
 - Requirements Management
 - Process Execution
 - Orchestration between System and Human Workflow
 - Technical Implementation
- Achieve Business IT-Alignment

The process engine



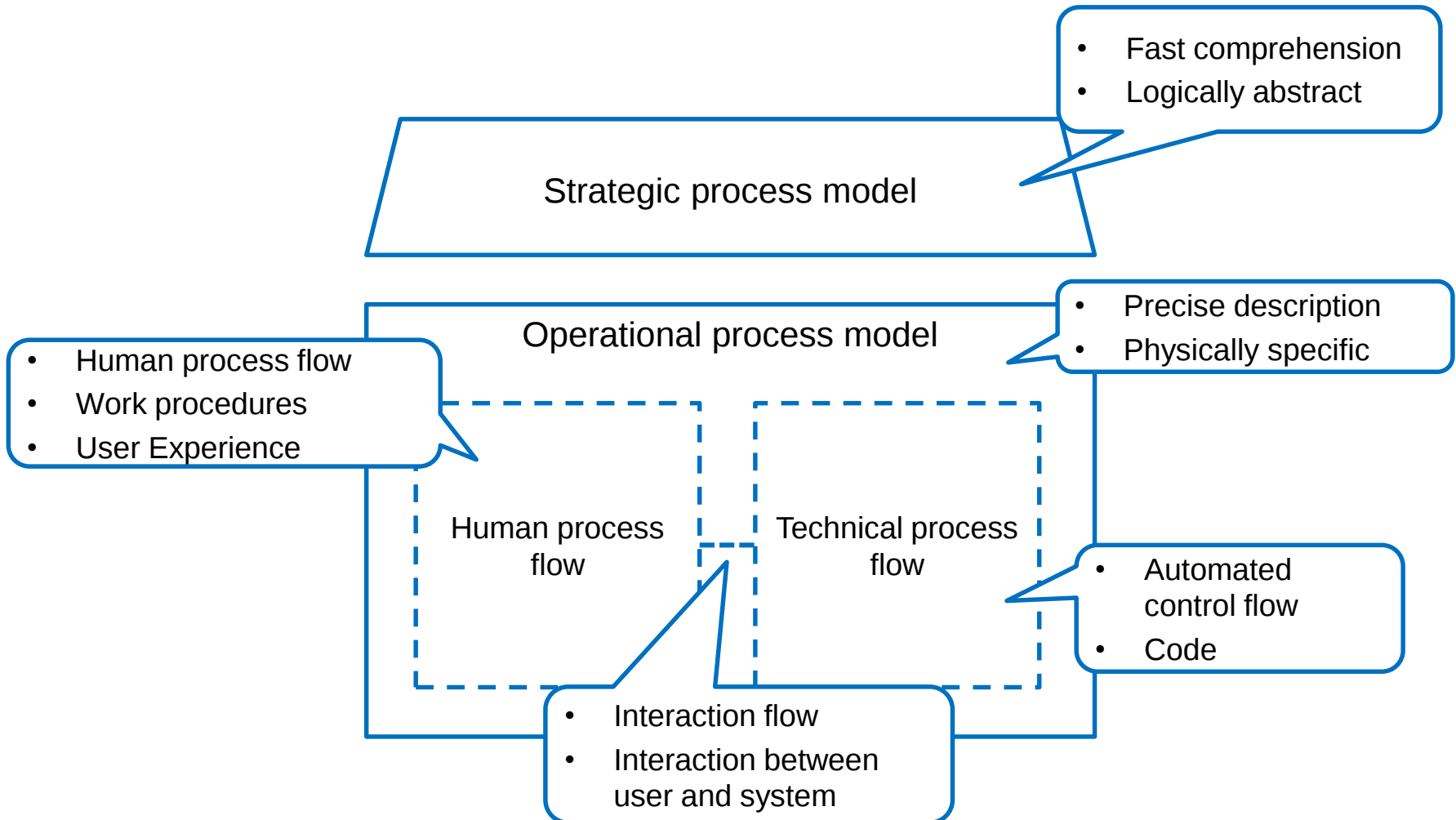
- Aspects
 - The execution of a technical process model
 - The measurement of technical indicators
 - Orchestration of human work and IT Systems
- In a broader sense
 - Monitoring the running instances
 - Human workflow management
 - Part of IT Architecture (SOA, microservices, integration,...)

The technical process model

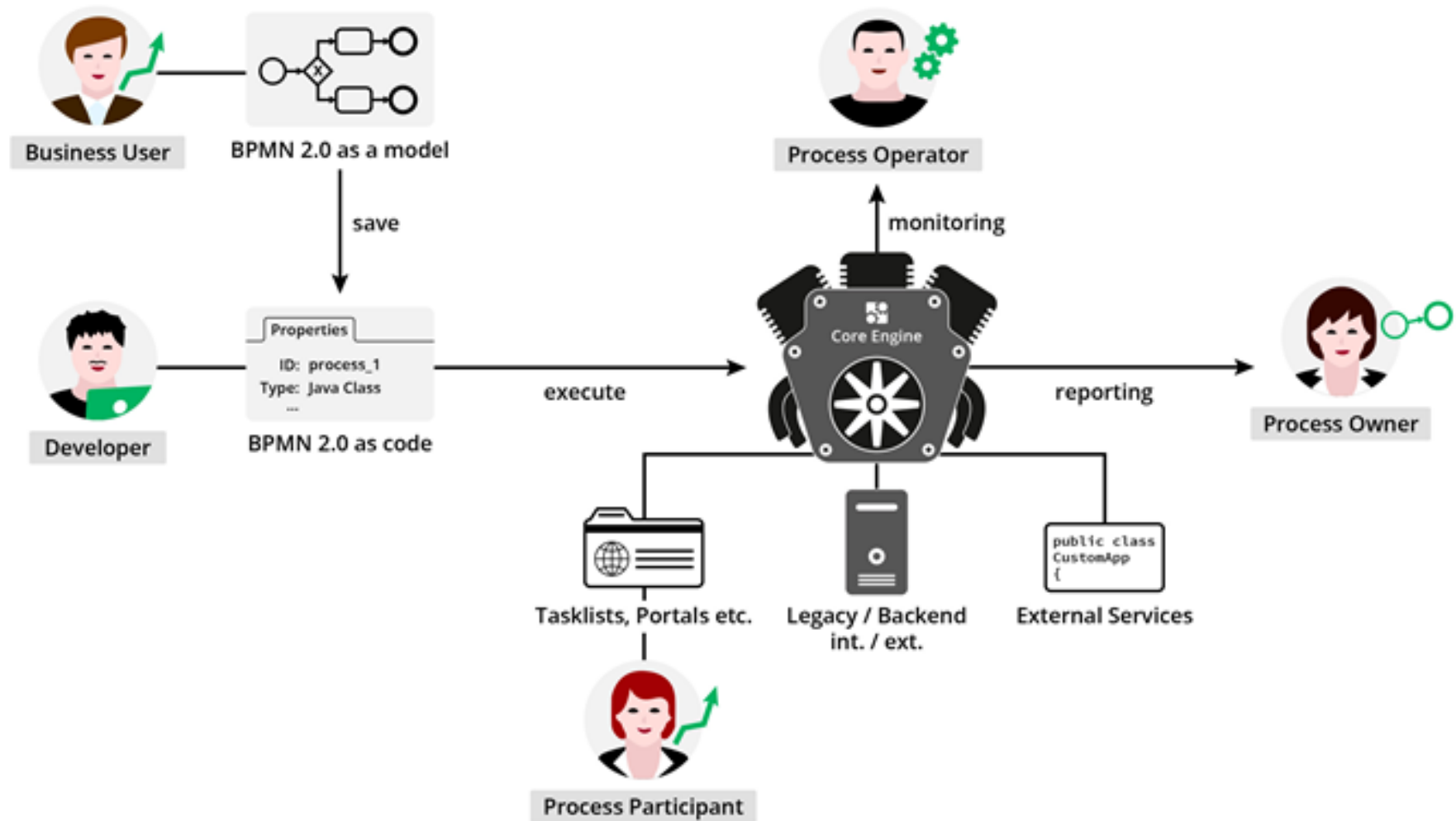
- The technical process model is the model the engine can execute
- Different application areas in BPMN lead to different stakeholders and participants

Role	Process Participant	Process Analyst	Process Engineer
Objective	How should I work?	How is the work done?	What is the engine doing exactly?
View	Own orchestration	Whole collaboration	Orchestration of the process engine

The Camunda House



Proposed architecture with BPMN 2.0



The XML of a BPMN 2.0

diagram_1.bpmn

```

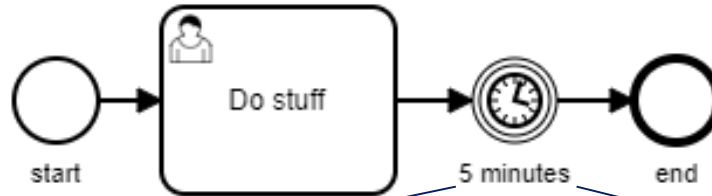
1 <?xml version="1.0" encoding="UTF-8"?>
2 <bpmn:definitions xmlns:bpmn="http://www.omg.org/spec/BPMN/20100524/MODEL" xmlns:bpmndi="http://www.omg.org/spec/BPMN/20100524/DI" xmlns:di="http://www.omg.org/spec/DD/20100524/DI" xmlns:dc="http://www.omg.org/spec/DD/20100524/DC"
3 id="Definitions_1epi2gj" targetNamespace="http://bpmn.io/schema/bpmn" exporter="Camunda Modeler" exporterVersion="1.16.2">
4   <bpmn:process id="Process_1" isExecutable="true">
5     <bpmn:startEvent id="StartEvent_1" name="start">
6       <bpmn:outgoing>SequenceFlow_1gyghmz</bpmn:outgoing>
7     </bpmn:startEvent>
8     <bpmn:sequenceFlow id="SequenceFlow_1gyghmz" sourceRef="StartEvent_1" targetRef="Task_1e57b9c" />
9     <bpmn:userTask id="Task_1e57b9c" name="Do stuff">
10      <bpmn:incoming>SequenceFlow_1gyghmz</bpmn:incoming>
11      <bpmn:outgoing>SequenceFlow_023vx9q</bpmn:outgoing>
12    </bpmn:userTask>
13    <bpmn:endEvent id="EndEvent_0g40ma2" name="end">
14      <bpmn:incoming>SequenceFlow_1qq5rnn</bpmn:incoming>
15    </bpmn:endEvent>
16    <bpmn:sequenceFlow id="SequenceFlow_023vx9q" sourceRef="Task_1e57b9c" targetRef="IntermediateThrowEvent_0zc7v61" />
17    <bpmn:sequenceFlow id="SequenceFlow_1qq5rnn" sourceRef="IntermediateThrowEvent_0zc7v61" targetRef="EndEvent_0g40ma2" />
18    <bpmn:intermediateCatchEvent id="IntermediateThrowEvent_0zc7v61" name="5 minutes">
19      <bpmn:incoming>SequenceFlow_023vx9q</bpmn:incoming>
20      <bpmn:outgoing>SequenceFlow_1qq5rnn</bpmn:outgoing>
21      <bpmn:timerEventDefinition />
22    </bpmn:intermediateCatchEvent>
23  </bpmn:process>
24  <bpmndi:BPMNDiagram id="BPMNDiagram_1">
25    <bpmndi:BPMNPlane id="BPMNPlane_1" bpmnElement="Process_1">
26      <bpmndi:BPMNShape id="BPMNShape_StartEvent_2" bpmnElement="StartEvent_1">
27        <dc:Bounds x="204" y="280" width="36" height="36" />
28      </bpmndi:BPMNShape>
29      <bpmndi:BPMNLabel>
30        <dc:Bounds x="211" y="323" width="23" height="14" />
31      </bpmndi:BPMNLabel>
32      <bpmndi:BPMNEdge id="SequenceFlow_1gyghmz_di" bpmnElement="SequenceFlow_1gyghmz">
33        <di:waypoint x="240" y="298" />
34        <di:waypoint x="267" y="298" />
35      </bpmndi:BPMNEdge>
36      <bpmndi:BPMNShape id="UserTask_05y5gvy_di" bpmnElement="Task_1e57b9c">
37        <dc:Bounds x="267" y="258" width="100" height="80" />
38      </bpmndi:BPMNShape>
39      <bpmndi:BPMNShape id="EndEvent_0g40ma2_di" bpmnElement="EndEvent_0g40ma2">
40        <dc:Bounds x="469" y="280" width="36" height="36" />
41      </bpmndi:BPMNShape>
42      <bpmndi:BPMNLabel>
43        <dc:Bounds x="478" y="323" width="19" height="14" />
44      </bpmndi:BPMNLabel>
45      <bpmndi:BPMNShape id="SequenceFlow_023vx9q_di" bpmnElement="SequenceFlow_023vx9q">
46        <di:waypoint x="367" y="298" />
47        <di:waypoint x="400" y="298" />
48      </bpmndi:BPMNShape>
49      <bpmndi:BPMNEdge id="SequenceFlow_1qq5rnn_di" bpmnElement="SequenceFlow_1qq5rnn">
50        <di:waypoint x="436" y="298" />
51        <di:waypoint x="469" y="298" />
52      </bpmndi:BPMNEdge>
53      <bpmndi:BPMNShape id="IntermediateCatchEvent_0wciagp_di" bpmnElement="IntermediateThrowEvent_0zc7v61">
54        <dc:Bounds x="400" y="280" width="36" height="36" />
55      </bpmndi:BPMNShape>
56      <bpmndi:BPMNLabel>
57        <dc:Bounds x="394" y="323" width="49" height="14" />
58      </bpmndi:BPMNLabel>
59    </bpmndi:BPMNPlane>
60  </bpmndi:BPMNDiagram>
61 </definitions>

```

Diagram XML

Diagram XML

Example: XML element of a timer event



```
<bpmn:intermediateCatchEvent id="IntermediateThrowEvent_0n5j7uh" name = "5 minutes">  
  <bpmn:incoming>SequenceFlow_0kj7pvb</bpmn:incoming>  
  <bpmn:outgoing>SequenceFlow_0nq3rhh</bpmn:outgoing>  
  <bpmn:timerEventDefinition />  
</bpmn:intermediateCatchEvent>
```

Duration?

Solution: Adding technical attributes to make the model executable

Technical Attributes

- Technical attributes make a BPMN file executable so an engine can interpret the model.
- Technical attributes will be represented as tags inside the XML structure.
- They can be added by using the properties panel of the Camunda Modeler.
- Technical attributes are essential for a technical process model

Fundamentals of technical process models

- Different application areas of BPMN
- Principals of a process engine
- The technical process model
- Purposed architecture with BPMN 2.0
- The XML meta model of BPMN 2.0

Video Clip 6.2

Adding executable attributes (tasks)

Business Process Modeling, Simulation and Execution:
An openHPI Course by Mathias Weske

Important task types



Service Task

Running Systems



User Task

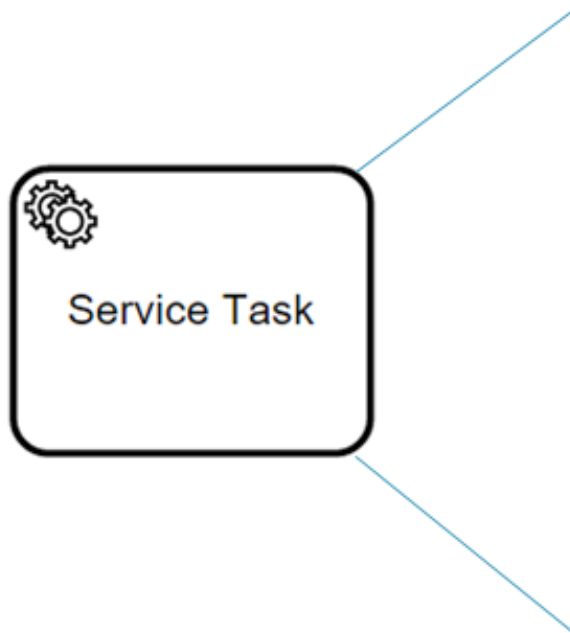
Getting User Input



Business Rule
Task

Making Business
Decisions

Service task



```
public class CalculateMutualisabilityDelegate implements JavaDelegate {

    @Override
    public void execute(DelegateExecution execution) throws Exception {
        boolean isItMutual = random();

        if(isItMutual)
        {
            execution.setVariable("isMutual", true);
        }else{
            execution.setVariable("isMutual", false);
        }
        String missionOrder = "This is the Default 'mission order' i suspect "
            + "there is more to it than this - but i don't actually "
            + "know for sure.";
        execution.setVariable("missionOrder", missionOrder);
    }
}
```

Attaching a java class

The image shows the Camunda BPMN Designer interface. On the left, a task named "Calculate Mutual" is highlighted with a dashed blue border. The task icon consists of two interlocking gears. To the right of the task, the Properties Panel is open, displaying the configuration for the "CalculateMutualTask".

The Properties Panel has four tabs: "General", "Listeners", "Input/Output", and "Field Injection". The "General" tab is selected.

General

- Id**: CalculateMutualTask
- Name**: Calculate Mutual

Details

- Implementation**: Java Class
- Java Class**: com.camunda.demo.CalculateMutual

Asynchronous Continuations

- ☐ Asynchronous Before
- ☐ Asynchronous After

Two blue arrows point from the "Implementation" and "Java Class" fields in the Properties Panel to the "Calculate Mutual" task in the BPMN diagram, indicating the attachment of the Java class to the task.

User Task



Form History Diagram Description

Please clarify: This invoice needs clarification and could not be approved due to missing data.

Invoice Document

Creditor

Papa Steve's all you can eat

Amount

10.99

Invoice Category

Travel Expenses

Invoice Number

PSACE-5342

Could you clarify the invoice?

☐

Save Complete

Creating form fields

The screenshot illustrates the process of creating form fields for a task in Camunda BPMN Designer. On the left, a task named 'Check Invoice' is shown with a person icon. A dashed blue box highlights the task. To the right, the 'Properties Panel' is open, displaying the 'UserTaskTask' configuration. The 'Forms' tab is selected, showing a list of form fields: 'invoiceNumber', 'totalAmount', 'approverName', and 'invoiceCategory'. The 'invoiceCategory' field is selected, and its properties are shown in the 'Form Field' section. The 'ID' is 'invoiceCategory', the 'Type' is 'string', the 'Label' is 'Invoice Category', and the 'Default Value' is an empty text box. The 'Validation' section has an 'Add Constraint' button, and the 'Properties' section has an 'Add Property' button. Four blue arrows point to specific elements: the first points to the 'Forms' tab, the second points to the 'invoiceCategory' field in the 'Form Fields' list, the third points to the 'string' type, and the fourth points to the 'Invoice Category' label.

UserTaskTask

< General Forms Listeners Input/Output Ext >

Forms

Form Key

Form Fields

invoiceNumber
totalAmount
approverName
invoiceCategory

Form Field

ID

invoiceCategory

Type

string

Label

Invoice Category

Default Value

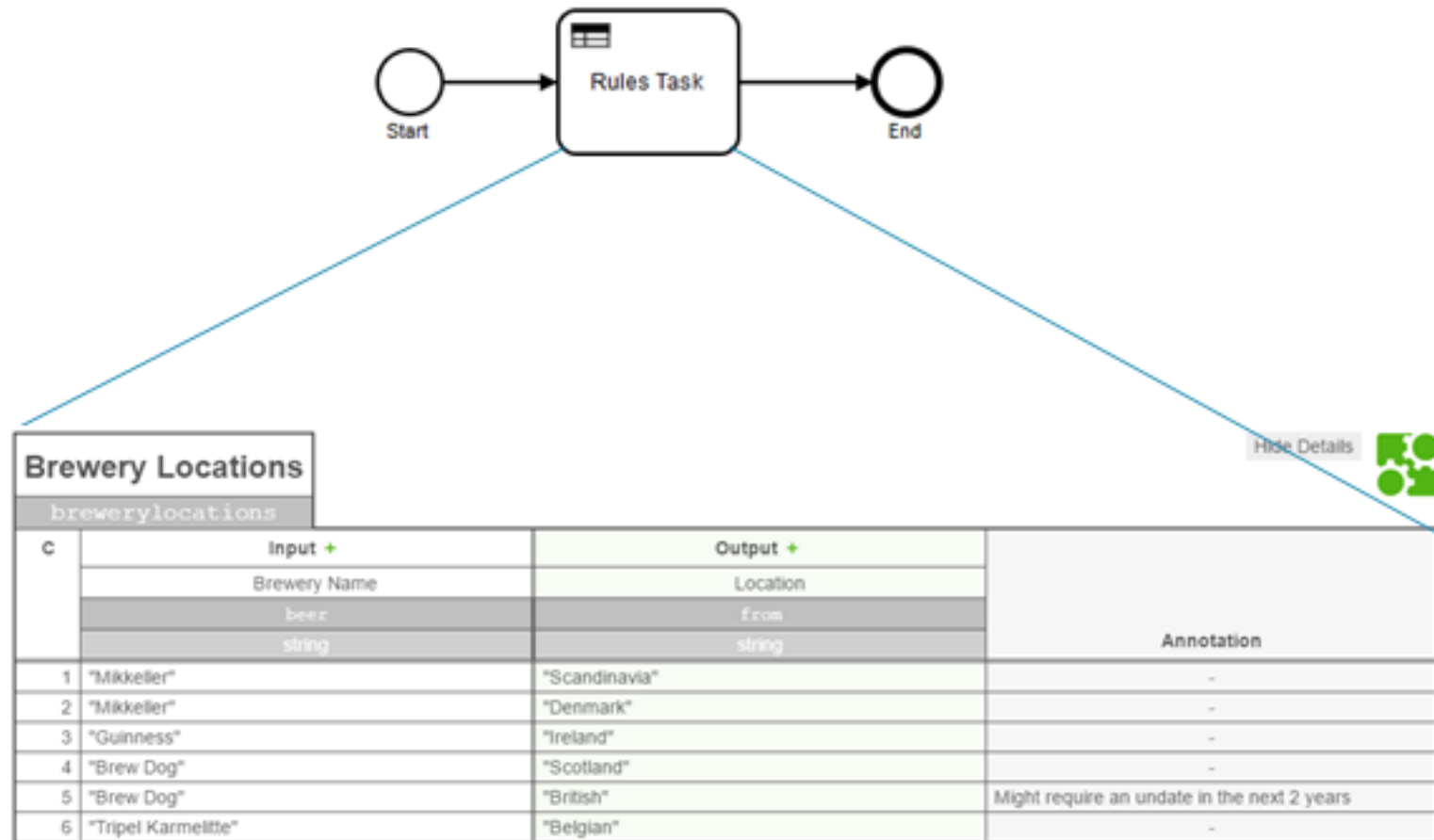
Validation

Add Constraint +

Properties

Add Property +

Business rules task



Calling a DMN table

The image shows a BPMN diagram element, a rounded rectangle labeled "Decide on Beverage" with a table icon, highlighted by a dashed blue border. To its right is the "Properties Panel" for this task. The panel has tabs for "General", "Listeners", "Input/Output", and "Field Injection". The "General" tab is active, showing the following configuration:

- Id:** BusinessRuleTaskTask
- Name:** Decide on Beverage
- Implementation:** DMN
- Decision Ref:** BeverageChoice
- Binding:** latest
- Tenant Id:** (empty)
- Result Variable:** beverage
- Map Decision Result:** singleEntry

Four blue arrows point from the BPMN task to the corresponding fields in the Properties Panel: from the task name to the Name field, from the task icon to the Implementation field, from the task to the Decision Ref field, and from the task to the Result Variable field.

Other Task Types



Script Task



Manual Task



Call Activity

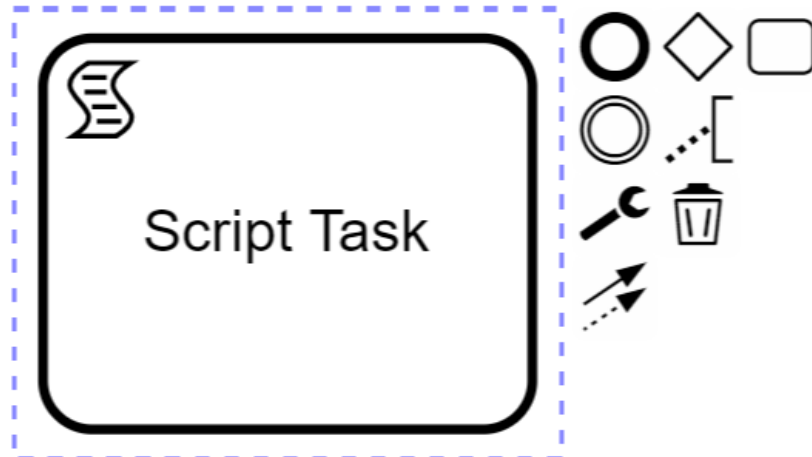


Send Task



Receive Task

Script Task



task_1n4pj8p

< General Listeners Input/Output Extens >

General

Id
Task_1n4pj8p x

Name
Script Task

Details

Script Format
javascript x

Script Type
Inline Script ▼

Script

```
var accountBalance =  
execution.getVariable('account  
Balance');  
var amount =  
execution.getVariable('amount'  
);  
execution.setVariable('account  
Differnce', accountBalance -  
amount);
```

Call Activity

The image shows a screenshot of the Camunda BPMN editor. On the left, a Call Activity is represented by a rounded rectangle with a thick black border, labeled "Call Activity" with a plus sign icon inside. It is surrounded by a dashed blue selection box. To its right is the "Properties Panel" which is open, showing the configuration for the selected element, "Task_02rp0av".

The Properties Panel has tabs for "General", "Variables", "Listeners", and "Input/Output". The "General" tab is active, showing the following fields:

- Id**: Task_02rp0av
- Name**: Call Activity
- Details**
 - CallActivityType**: BPMN
 - Called Element**: AnotherProcess
 - Binding**: latest
 - Tenant Id**: (empty field)
 - ☐ **Business Key**

Two blue arrows point from the "CallActivityType" and "Called Element" fields in the Properties Panel to the "Call Activity" element in the diagram.

Send & Receive

The image displays two screenshots of the Camunda BPMN editor interface, illustrating the configuration of a Receive Task and a Send Task.

Top Screenshot: Receive Task Properties

- The task icon is a rounded rectangle with an envelope symbol and the text "Receive Task".
- The **Properties** panel is open, showing the **Details** tab.
- In the **Details** section:
 - The **Message** dropdown is set to "MessageName (id=Message_0kqq42e)".
 - The **Message Name** field is set to "MessageName".
- The **Asynchronous Continuations** section shows the **Asynchronous Before** checkbox is unchecked.
- Blue arrows point to the **Message** dropdown and the **Message Name** field.

Bottom Screenshot: Send Task Properties

- The task icon is a rounded rectangle with an envelope symbol and the text "Send Task".
- The **Properties Panel** is open, showing the **General** tab.
- In the **General** section:
 - The **Id** field is set to "Task_1xywm6b".
 - The **Name** field is set to "Send Task".
- The **Details** section shows:
 - The **Implementation** dropdown is set to "Java Class".
 - The **Java Class** field is set to "some.java.class".
- Blue arrows point to the **Implementation** dropdown and the **Java Class** field.

Video Clip 6.2

Adding technical Attributes (task)

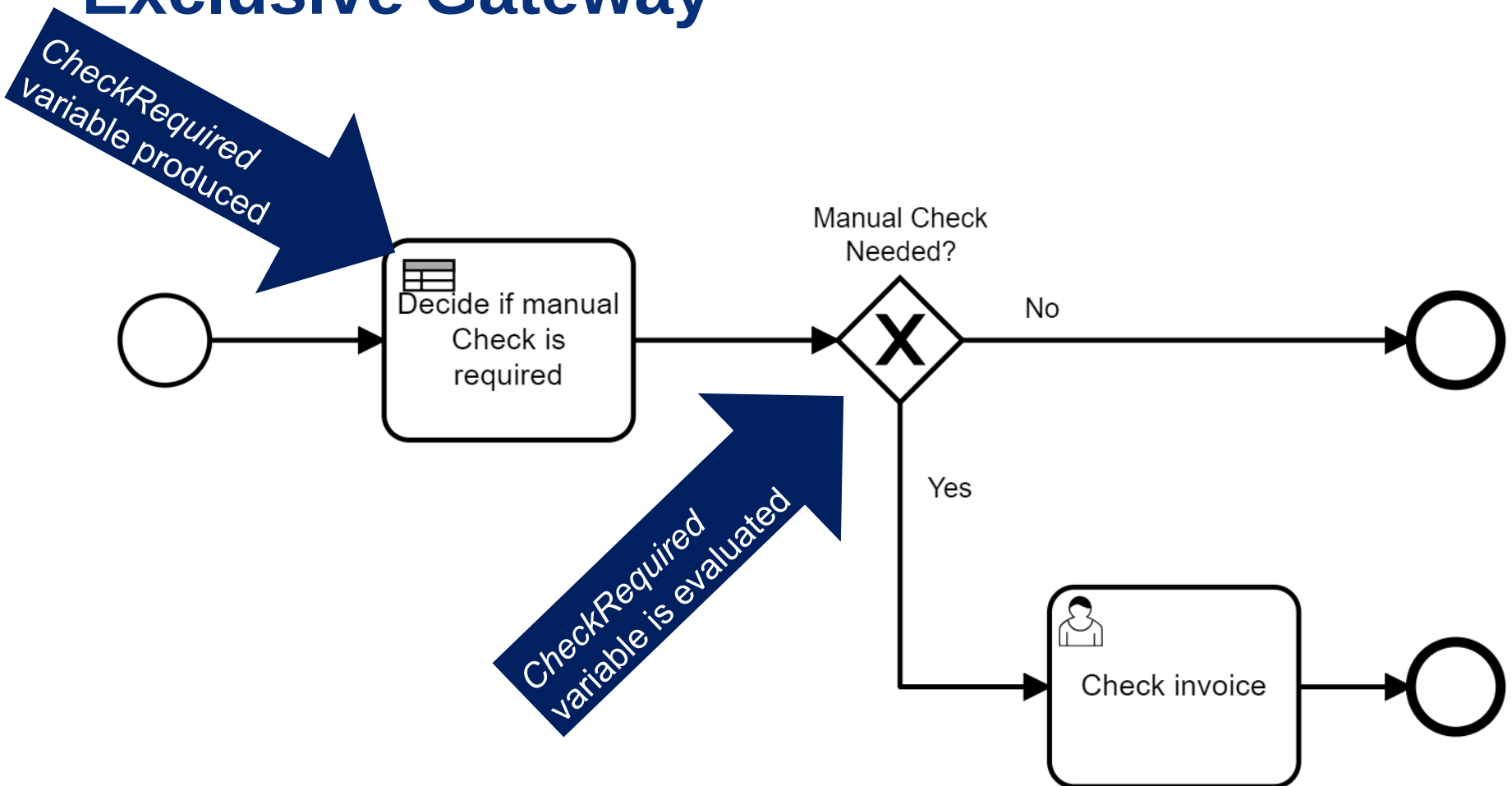
- Technical attributes of a service task
 - Implementation (Java Class, External)
 - Name/ Topic
- Technical attributes of a user task
 - User interface
 - Creating form fields for a user task
- Technical attributes of a business rule task
 - Implementation (DMN)
 - Result Variable

Video Clip 6.3

Adding executable attributes (gateways and events)

Business Process Modeling, Simulation and Execution:
An openHPI Course by Mathias Weske

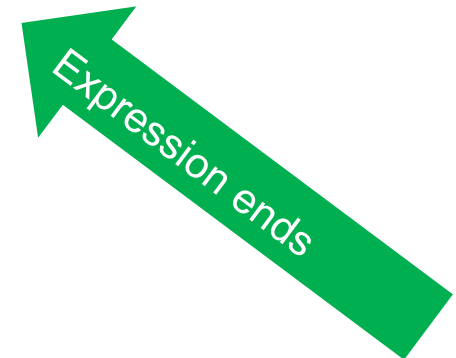
Exclusive Gateway



Expressions are used in evaluation

If the result is **true** it is the path we should take

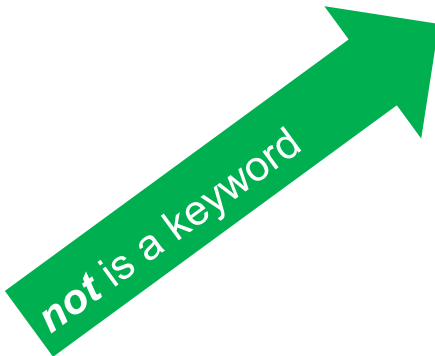
`${ approved }`



So if a variable called ***approved*** is ***true*** this expression returns ***true***

Expressions are used in evaluation

`${ not approved }`

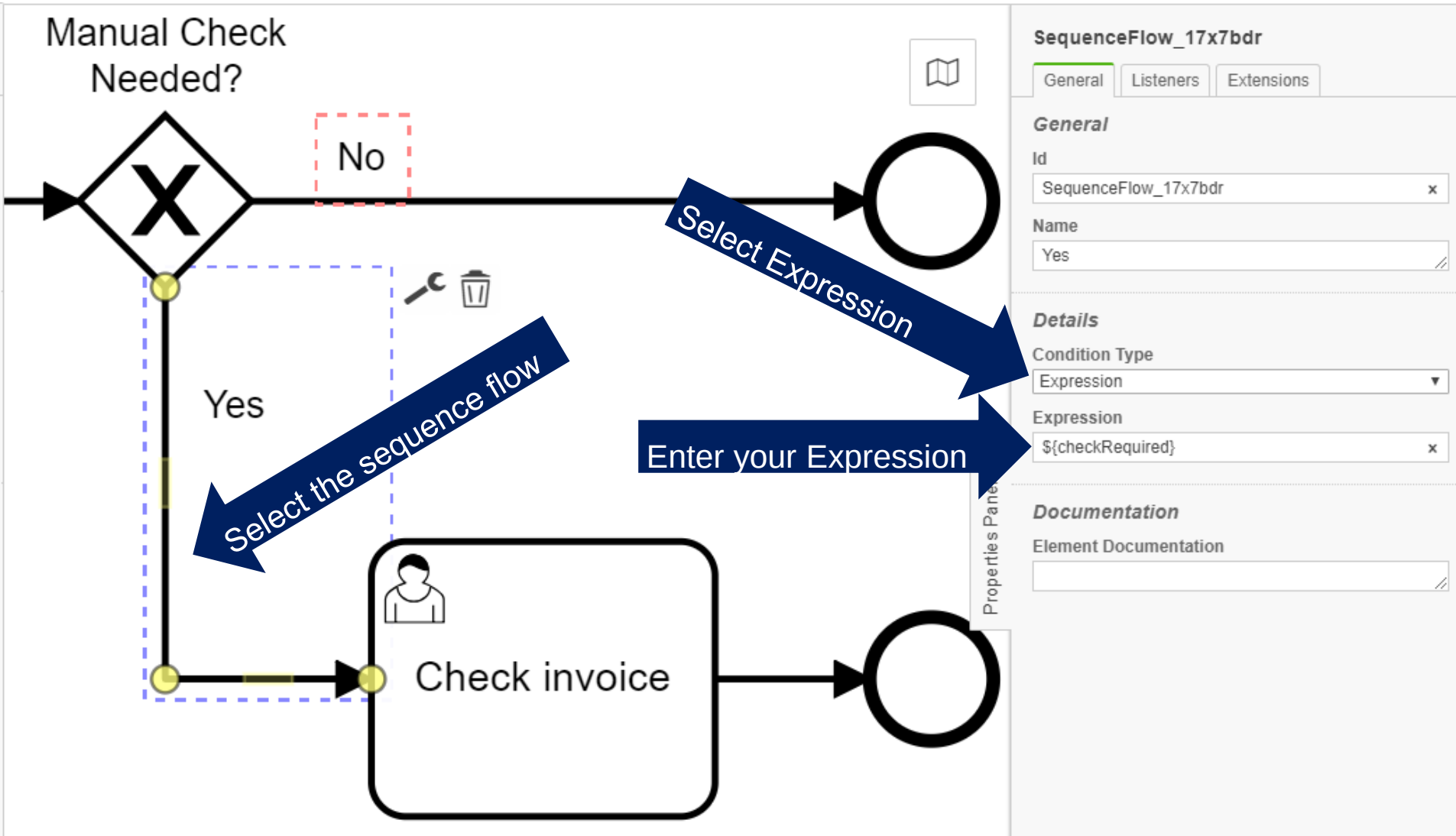


So if a variable called ***approved*** is ***false*** this expression returns ***true***

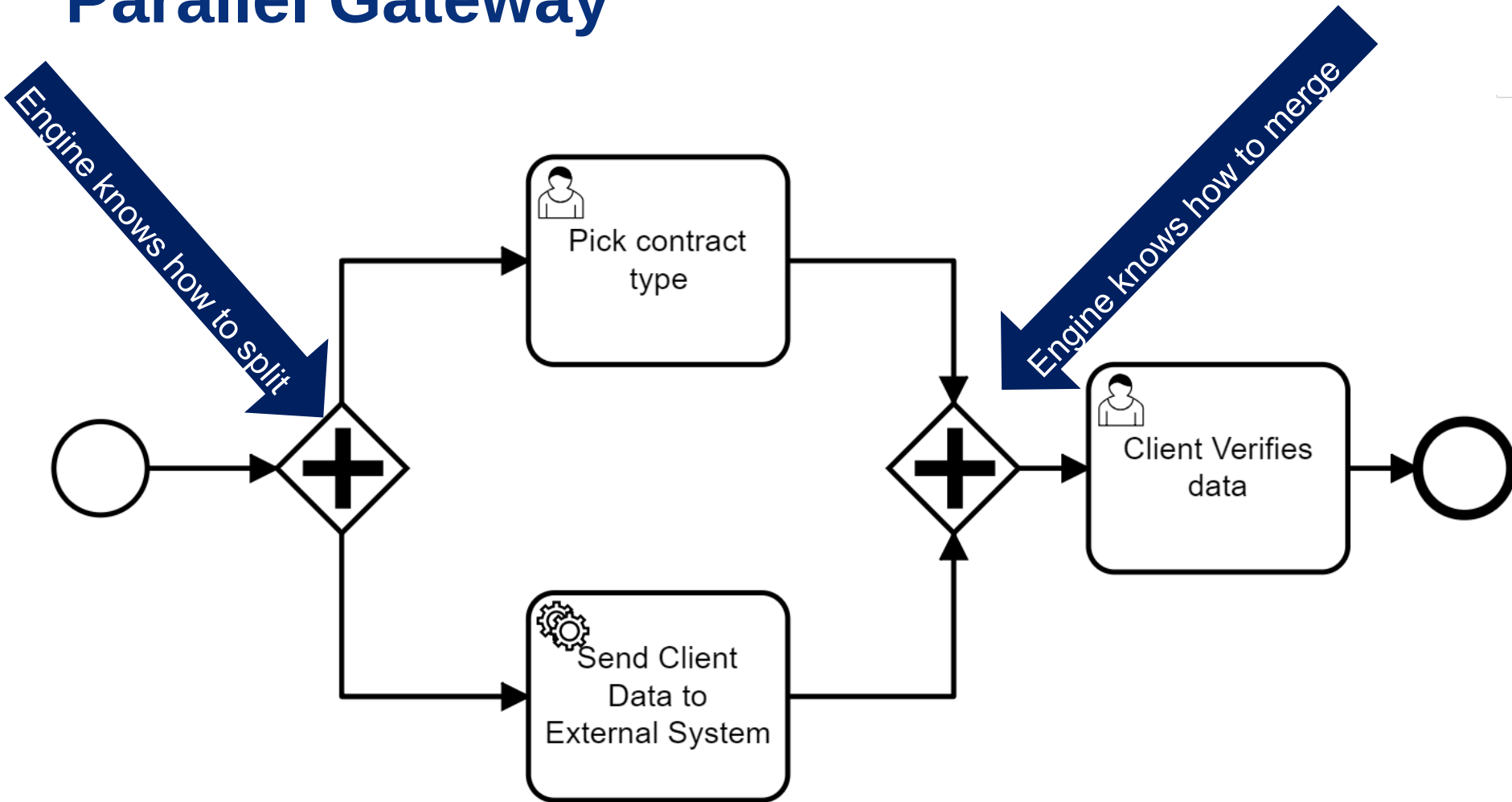
Slightly more complex expressions

`${ firstName == 'Niall' }`

So if a variable called ***firstName*** is ***'Niall'***
this expression returns ***true***



Parallel Gateway



Important events



Timer Event



Conditional Event

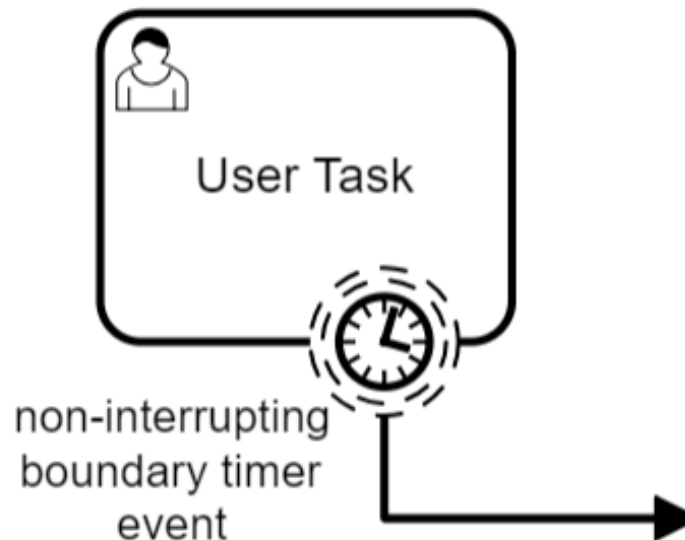


Message Event

Timer Event



Timer Event

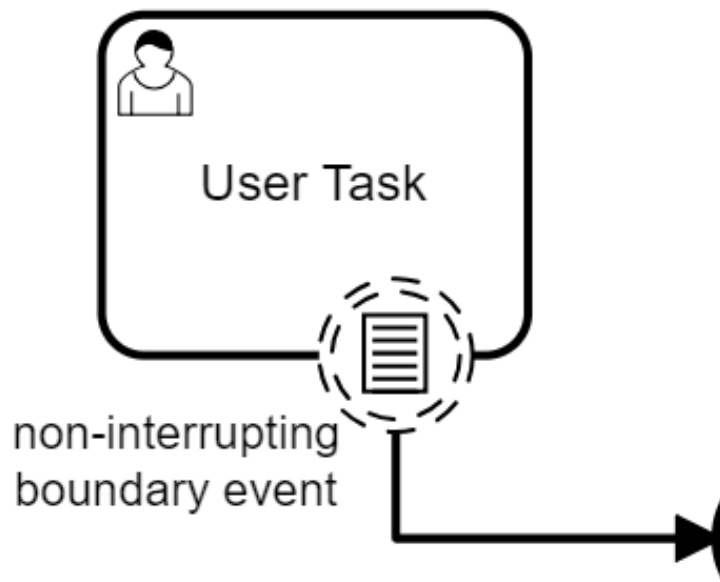


- Date
 - Wait until a specific date and time is in the past
 - e.g. **2020-02-20T12:30:44**
- Duration
 - Wait for a specific duration to expire
 - e.g. **'PT5M'** Wait for 5 minutes
 - The ISO 8601
 - e.g. **5 4 * * *** wait until 04:05 am
 - Cron Time
- Cycle
 - Repeated duration of waiting - often used with non interrupting timer events
 - e.g. **R/PT5M** repeat every 5 mins
 - e.g. **R3/PT5M** repeat 3 times, every 5 minutes

Conditional Event



Conditional Event



- **Expression**
 - Wait until this condition becomes true before it continues
 - e.g. `${someVariable}`
- Will be evaluated every time any variable has changed
- You can choose to only evaluate the condition if a specific variable is changed in a specific way

Properties Panel

Details

Variable Name

Variable Event

Specify more than one variable change event as a comma separated list.

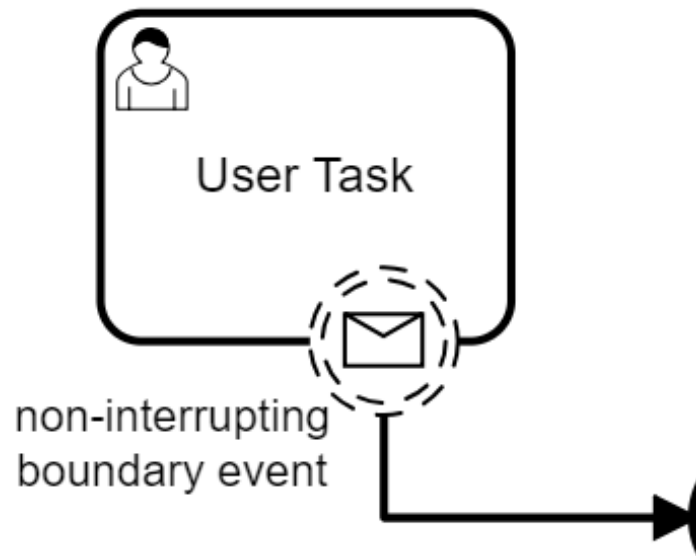
Condition Type

Expression

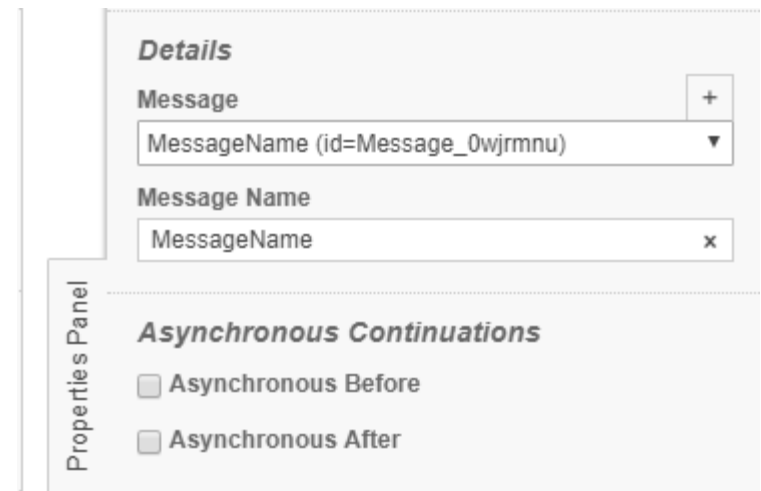
Message Event



Message Event



- Add a **Message Name** to the event.
- The message can be triggered when a message with the same name is sent to the engine
- Other criteria can be added to correlate the message
 - Business Key
 - Variable



Details

Message +

MessageName (id=Message_Owjrmnu) ▼

Message Name

MessageName x

Asynchronous Continuations

☐ Asynchronous Before

☐ Asynchronous After

Video Clip 6.3

Adding technical Attributes (gateways and events)

- Technical attributes on sequence flows
 - After an exclusive data based gateway
 - Using Expression language
- Technical attributes on timer events
 - Date, Duration, Cycle
- Technical attributes on conditional events
 - Expressions
 - Default check: Every time any variable changes, can be customize by certain variables
- Technical attributes on messages
 - Name of the message
 - Possibility to set other criteria (Business key,...)

Video Clip 6.4

Deploy and run a process

Business Process Modeling, Simulation and Execution:
An openHPI Course by Mathias Weske

Different ways of Deployment

- From the Modeler
 - (REST Call)
 - Contains BPMN File
- Making an own REST Call
 - Contains BPMN file and HTML forms
- Creating a WAR File
 - Contains BPMN File, Java Classes and HTML forms

Modeler Deployment

The image shows the Camunda Modeler interface. At the top, there is a menu bar with 'File', 'Edit', 'Window', 'Plugins', and 'Help'. Below the menu bar is a toolbar with various icons. The 'Deploy' button, represented by a person icon, is circled in blue. A large blue arrow points from the 'Deploy' button to the 'MyProcess' configuration panel on the right. Another blue arrow points from the text 'Make sure the box is ticked' to the 'Executable' checkbox in the 'MyProcess' configuration panel. The main workspace shows a BPMN diagram with a start event, a task labeled 'do some stuff', and an end event. The left sidebar contains a 'Token Simulation' button and a list of BPMN symbols. The bottom of the interface has tabs for 'Diagram' and 'XML'.

File Edit Window Plugins Help

process.bpmn x diagram_1.bpmn +

Token Simulation

start do some stuff end

Diagram XML

Make sure the box is ticked

MyProcess

General Listeners Extensions

General

Id
MyProcess x

This maps to the process key.

Name

Version Tag

☒ Executable

External Task Configuration

Task Priority

Job Configuration

Job Priority

Candidate Starter Configuration

Candidate Starter Groups

Specify more than one group as a comma separated list.

Candidate Starter Users

Specify more than one user as a comma separated list.

History Configuration

History Time To Live

Documentation

Element Documentation

ID: that name will show up

Modeler Deployment

Deploy Diagram

Specify deployment details and deploy this diagram to Camunda.

Deployment Details

Name

Tenant ID

Endpoint Configuration

URL

Should point to a running Camunda Engine REST API endpoint.

Authentication

HTTP Basic ▼

Username

Password

Deploy

Cancel

Address of the Engine

REST Call Deployment

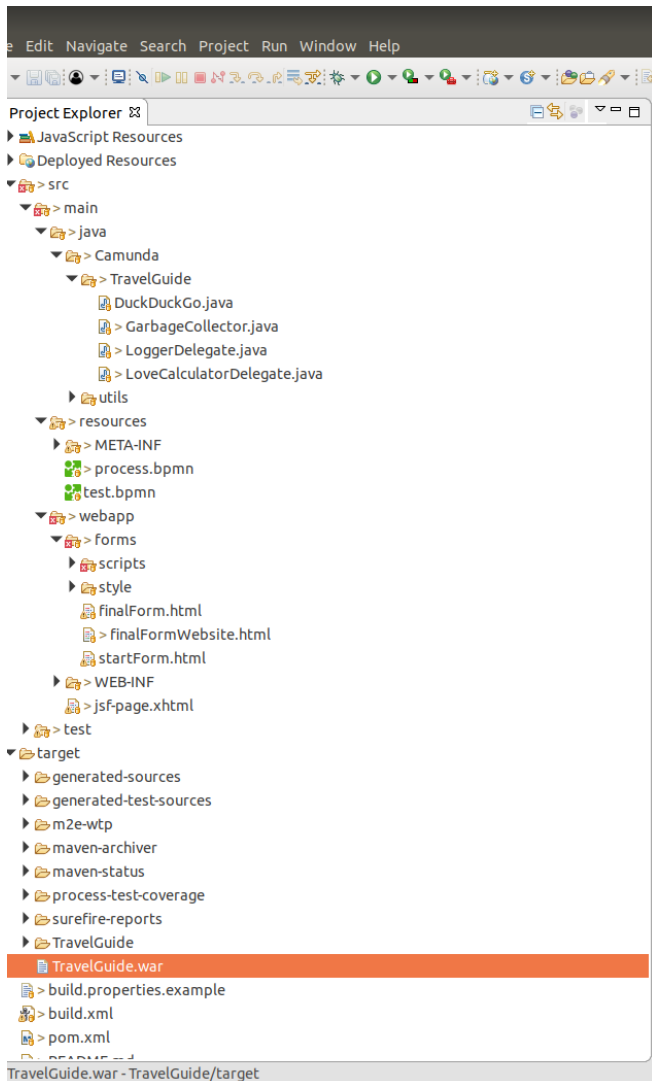
POST/deployment/create

Response

Status 200.

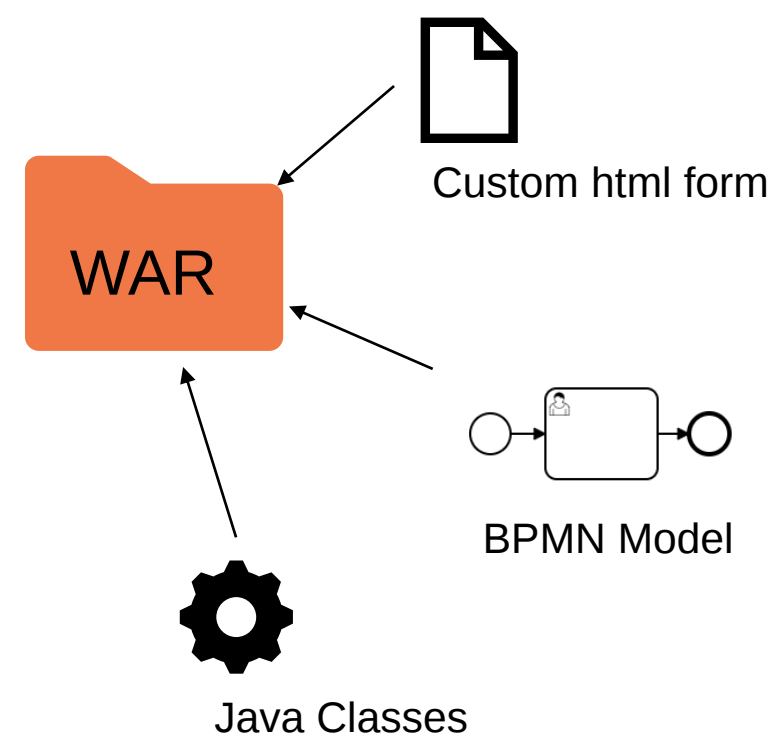
```
{
  "links": [
    {
      "method": "GET",
      "href": "http://localhost:38080/rest-test/deployment/aDeploymentId",
      "rel": "self"
    }
  ],
  "id": "aDeploymentId",
  "name": "aName",
  "source": "process application",
  "deploymentTime": "2013-01-23T13:59:43.000+0200",
  "tenantId": null,
  "deployedProcessDefinitions": {
    "aProcDefId": {
      "id": "aProcDefId",
      "key": "aKey",
      "category": "aCategory",
      "description": "aDescription",
      "name": "aName",
      "version": 42,
      "resource": "aResourceName",
      "deploymentId": "aDeploymentId",
      "diagram": "aResourceName.png",
      "suspended": true,
      "tenantId": null,
      "versionTag": null
    }
  },
  "deployedCaseDefinitions": null,
  "deployedDecisionDefinitions": null,
  "deployedDecisionRequirementsDefinitions": null
}
```

WAR file Deployment



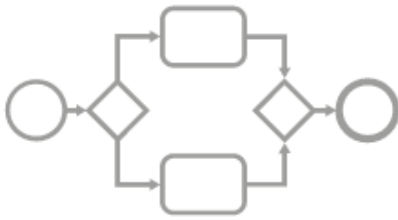
The screenshot shows the Eclipse IDE interface. The Project Explorer on the left displays the project structure for 'TravelGuide'. The 'webapp' folder is expanded, showing 'forms' and 'WEB-INF' subfolders. The 'TravelGuide.war' file is highlighted in the 'target' folder.

Web Application Resource



The diagram illustrates the components of a WAR file. An orange folder labeled 'WAR' is the central element. Three arrows point to it: one from a document icon labeled 'Custom html form', one from a BPMN diagram icon labeled 'BPMN Model', and one from a gear icon labeled 'Java Classes'.

Web Applications of the Camunda engine



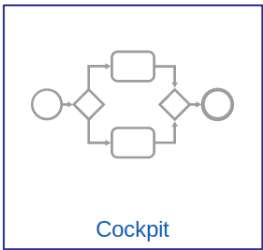
Cockpit

- Monitoring the instances of a process
- Manage deployments
- Handle incidents
- Variables of a process instance

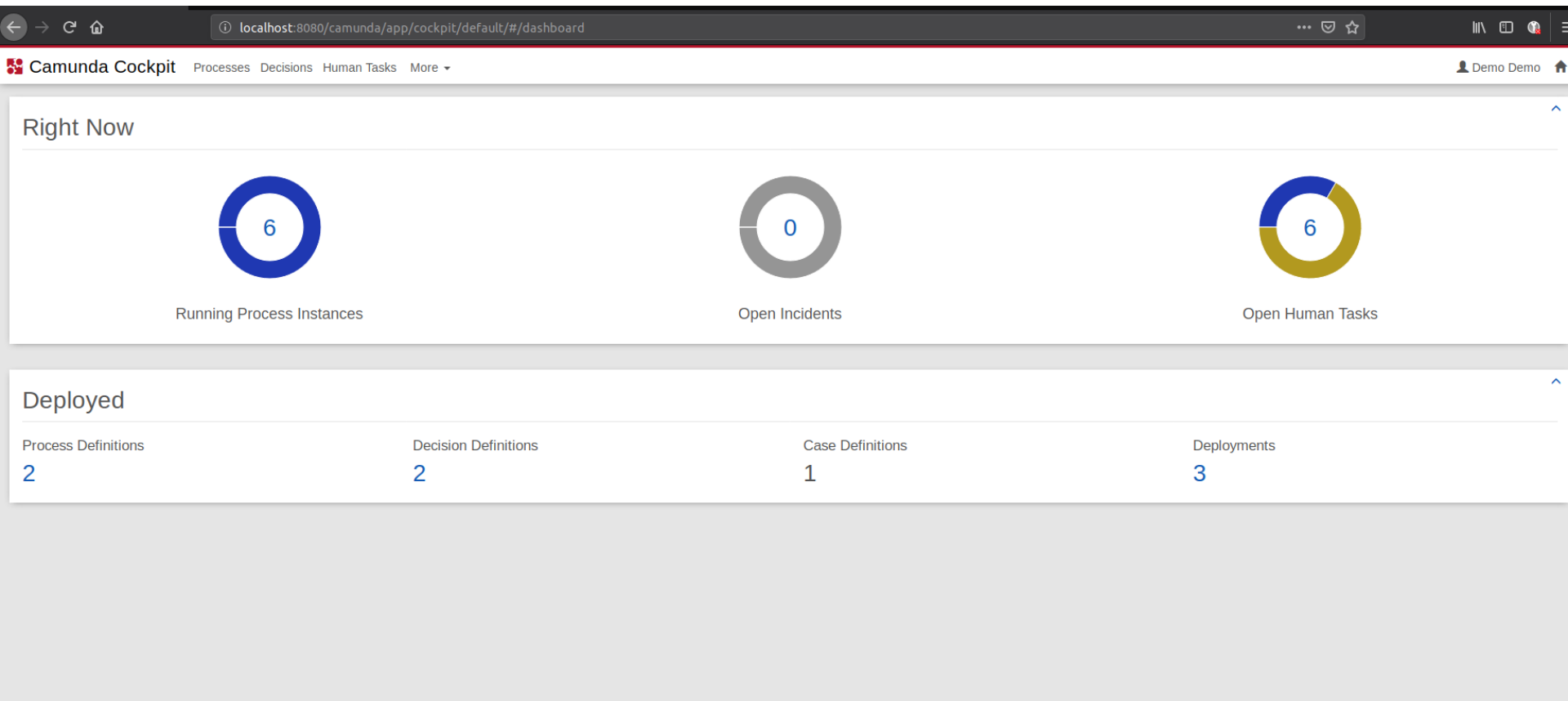


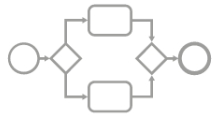
Tasklist

- User Tasks
- User Interface for human workflow
- See and work on tasks



Overview





Cockpit

Overview Deployments

Camunda Cockpit | Deployments

localhost:8080/camunda/app/cockpit/default/#/repository?page=1&deploymentsQuery=[]&deployment=9561c3db-3e87-11e9-af9e-f48c50c87387

Camunda Cockpit Processes Decisions Human Tasks More

Deployment Time

Add criteria 6

test.bpmn

Deployments
Batches

Select a resource in the list

helloProcess
Time: 2019-03-04T15:12:45
Source: null
Tenant ID: null

helloProcess
Time: 2019-03-04T15:12:39
Source: null
Tenant ID: null

helloProcess
Time: 2019-03-04T15:12:32
Source: null
Tenant ID: null

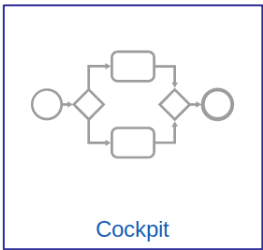
Reisekosten
Time: 2019-03-01T20:20:07
Source: process application
Tenant ID: null

camunda-invoice
Time: 2019-02-25T17:39:17
Source: process application
Tenant ID: null

e4a4652c-391b-11e9-ac71-f48c50c87387
Time: 2019-02-25T17:39:17
Source: process application
Tenant ID: null

localhost:8080/camunda/app/cockpit/default/#/repository

Powered by camunda BPM / v7.10.0



Overview Processes

Camunda Cockpit | Processes

localhost:8080/camunda/app/cockpit/default/#/processes

Camunda Cockpit Processes Decisions Human Tasks More

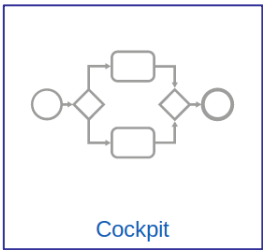
Dashboard » Processes

3 process definitions deployed

State	Incidents	Running Instances	Name	Tenant ID
✓	0	6	Invoice Receipt	
✓	0	1	MyProcess	
✓	0	0	Reisekosten	

localhost:8080/camunda/app/cockpit/default/#/processes

Powered by camunda BPM / v7.10



Overview Running Instances of a process

Camunda Cockpit | Invoic

localhost:8080/camunda/app/cockpit/default/#/process-definition/invoice:2:e4fbd40a-391b-11e9-ac71-f48c50c87387/runtime?searchQuery=[]&viewbox=["sid-0b0aaa25-3baf-487...

Camunda Cockpit

Processes Decisions Human Tasks More

Demo Demo

Dashboard » Processes » Invoice Receipt : Runtime

Definition Version: 2

Version Tag: V2.0

Definition ID: invoice:2:e4fbd40a-391b-11e9-ac71-...

Definition Key: invoice

Definition Name: Invoice Receipt

History Time To Live: 45 days

Tenant ID: null

Deployment ID: e4f4f636-391b-11e9-ac71-f48c50c8...

Instances Running: 3

Team Assistant

Invoice received

Assign Approver Group

Review Invoice

Review successful?

Invoice not processed

Invoice Receipt Approver

Approve Invoice

Invoice approved?

Accountant

Prepare Bank Transfer

Archive Invoice

Invoice processed

Financial Accounting System

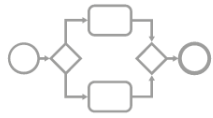
Activity Instance Statistics: on

Process Instances Incidents Called Process Definitions Job Definitions

Add criteria

3

State	ID	Start Time	Business Key
✓	e667991e-391b-11e9-ac71-f48c50c87387	2019-02-25T17:39:20	
✓	e6842244-391b-11e9-ac71-f48c50c87387	2019-02-20T17:39:20	
✓	e66ffdbe-391b-11e9-ac71-f48c50c87387	2019-02-11T17:39:20	



Cockpit

Overview One specific instance

Camunda Cockpit | Invoice X Naggisch Nad Morzer X Reservierungen (Teestübche X +

localhost:8080/camunda/app/cockpit/default/#/process-instance/e6842244-391b-11e9-ac71-f48c50c87387/runtime?viewbox={"sid-0b0aaa25-3baf-4875-9d7a-0907d599a9ef":{"x":...}}

Camunda Cockpit Processes Decisions Human Tasks More ▾ Demo Demo

Dashboard » Processes » Invoice Receipt : e6842244-391b-11e9-ac71-f48c50c87387 : Runtime

Information Filter

Instance ID:
e6842244-391b-11e9-ac71-f48c50c...

Business Key:
null

Definition Version:
2

Definition ID:
invoice:2:e4fbd40a-391b-11e9-ac71-...

Definition Key:
invoice

Definition Name:
Invoice Receipt

Tenant ID:
null

Deployment ID:
e4f4f636-391b-11e9-ac71-f48c50c...

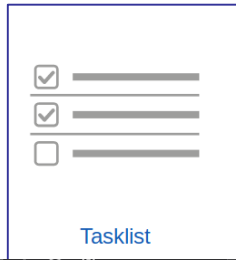
Super Process Instance ID:
null

Variables Incidents Called Process Instances User Tasks External Tasks

Add criteria

Name ▲	Type ▾	Value	Scope	Action...
amount	Double	10.99	Invoice Receipt	✗ 🗑
approved	Boolean	false	Invoice Receipt	✗ 🗑
approverGroups	Object	java.util.ArrayList	Invoice Receipt	✗ 🗑
creditor	String	Papa Steve's all you can eat	Invoice Receipt	✗ 🗑
invoiceCategory	String	Travel Expenses	Invoice Receipt	✗ 🗑
invoiceDocument	File	Download	Invoice Receipt	✗ 🗑

User Task Management



Camunda Tasklist

Keyboard Shortcuts Create task Start process Demo Demo

Create a filter + Created +

My Tasks (2) Filter Tasks 2

My Group Tasks

Accounting

John's Tasks

Mary's Tasks

Peter's Tasks

Employees

All Tasks

Assign Reviewer

Review Invoice Demo Demo

Created 7 days ago 50

Invoice Amount: 10.99 Invoice Number: PSACE-5342

Assign Reviewer

Review Invoice Demo Demo

Created 7 days ago 50

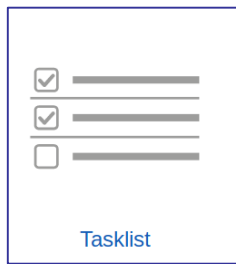
Invoice Amount: 10.99 Invoice Number: PSACE-5342

Select a task in the list.

Date and Time displayed in local timezone: Europe/Berlin

Powered by camunda BPM / v7.10.0

User Task: Form



Camunda Tasklist

localhost:8080/camunda/app/tasklist/default/#/?searchQuery=[]&filter=e524b90c-391b-11e9-ac71-f48c50c87387&sorting=[{"sortBy":"created","sortOrder":"desc"}]&task=41c8aa...

Camunda Tasklist

Keyboard Shortcuts Create task Start process Demo Demo

Create a filter +

My Tasks

My Group Tasks

Accounting

John's Tasks

Mary's Tasks

Peter's Tasks

Employees

All Tasks (8)

Filter Tasks 8

Approve Invoice

Invoice Receipt Demo Demo

Due in 7 days, Created a few seconds ago 50

Invoice Amount: 30 Invoice Number: 123445

do some stuff

MyProcess

Created 3 hours ago 50

Assign Reviewer

Review Invoice Demo Demo

Created 7 days ago 50

Invoice Amount: 10.99 Invoice Number: PSACE-5342

Prepare Bank Transfer

Invoice Receipt Demo Demo

Due 30 minutes ago, Created 7 days ago 50

Invoice Amount: 900 Invoice Number: BOS-43934

Approve Invoice

Invoice Receipt Demo Demo

Due 30 minutes ago, Created 7 days ago 50

Invoice Amount: 30 Invoice Number: GPFE-23232323

Assign Reviewer

Review Invoice Demo Demo

Created 7 days ago 50

Approve Invoice

Invoice Receipt

Set follow-up date in 7 days x

Buchhaltung, sales

Demo Demo

Form History Diagram Description

Do you approve this invoice?

Invoice Document C:\Invoice.pdf

Creditor HelloPizza

Amount 30

Invoice Category Travel Expenses

Invoice Number 123445

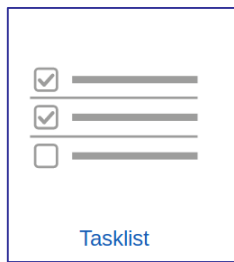
☒ Do you approve?

Save Complete

Date and Time displayed in local timezone: Europe/Berlin

Powered by camunda BPM / v7.10.

Start a process



Camunda Tasklist

Keyboard Shortcuts Create task Start process Demo Demo

Create a filter + Created +

My Tasks (2) Filter Tasks 2

My Group Tasks

Accounting

John's Tasks

Mary's Tasks

Peter's Tasks

Employees

All Tasks

Assign Reviewer

Review Invoice

Created 7 days ago

Invoice Amount: 10.99

Assign Reviewer

Review Invoice

Created 7 days ago

Invoice Amount: 10.99

Select a task in the list.

Start process

Upload your Invoice Document

Browse... C:\Invoice.pdf

This field is required

Creditor

HelloPizza

(e.g. "Great Pizza for Everyone Inc.")

Amount

30.00

(e.g. "30.00")

Invoice Category

Travel Expenses

Invoice Number

123445

(e.g. "I-12345")

Back Close Start

Video Clip 6.4

Deploy and Run a process

- Different ways of Deployment
 - Camunda Modeler, own REST Call or War-file
- Cockpit
 - Monitor running instances
 - Manage deployments
 - See and Change process variables
- Tasklist
 - Implementation of User tasks
 - Manage own tasks
 - Start a process

Video Clip 6.5

Demo 1: Import from Signavio the insurance application process and adding technical attributes

Business Process Modeling, Simulation and Execution:
An openHPI Course by Mathias Weske

Video Clip 6.6

Demo 2: Deploy the model to the engine and step through the process

Business Process Modeling, Simulation and Execution:
An openHPI Course by Mathias Weske