CAMUNDA COMMUNITY SUMMIT 2021

Making of bpmn-js Token Simulation

Understanding bpmn-js extensibility one token at a time.

Nico Rehwaldt

About Me

- Software developer at Camunda
- bpmn.io creator and project lead
- `@nikku` on GitHub

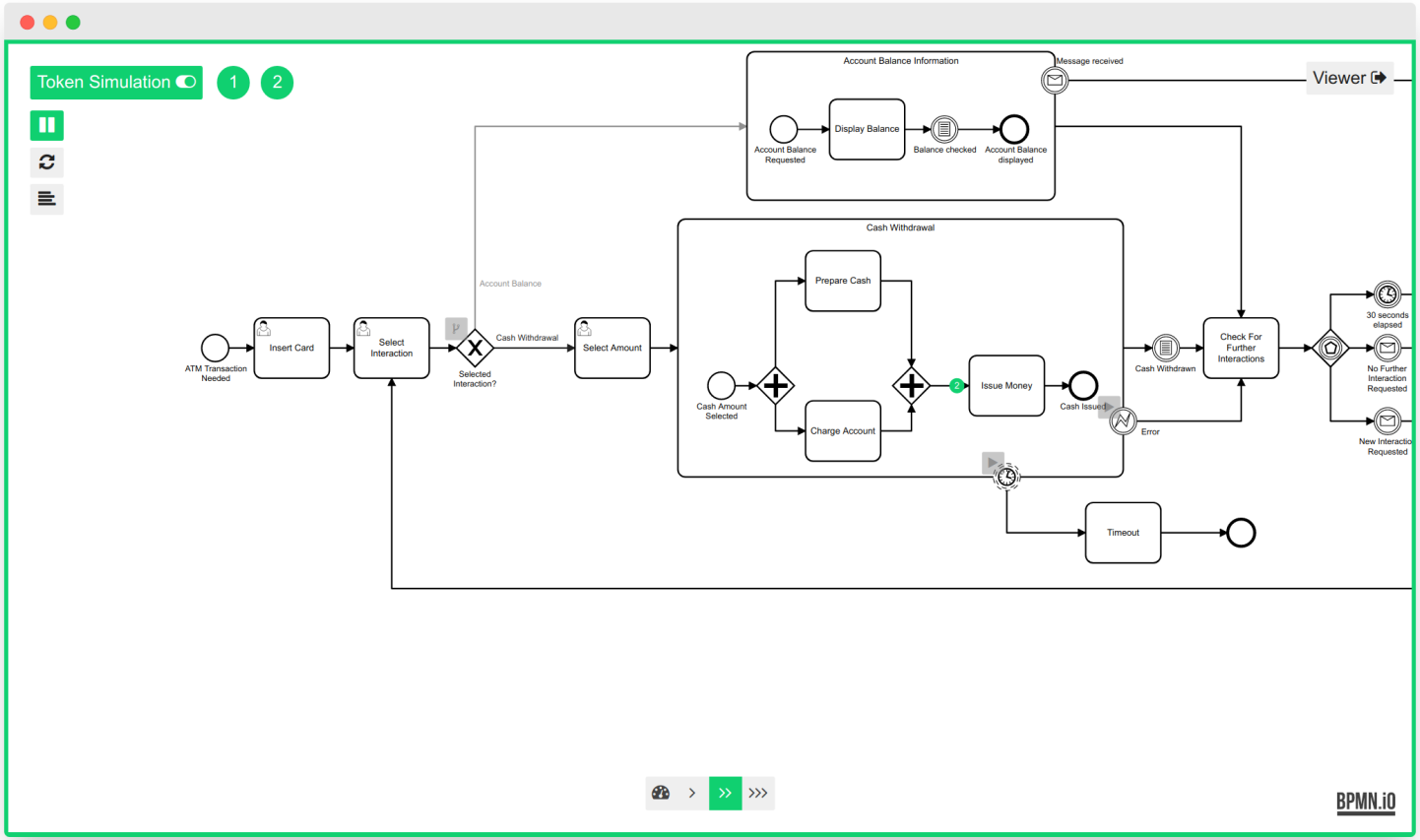
Agenda

1. Token simulation?
2. About bpmn-js
3. bpmn-js extensibility
4. How does token simulation plug in?

1 Token Simulation?

A picture is worth a thousand words.

A moving token is worth a whole bunch of static BPMN diagrams.


Token Simulation
1
2
Viewer
Account Balance Information
Message received
Account Balance Requested
Display Balance
Balance checked
Account Balance displayed
Cash Withdrawal
Account Balance
Prepare Cash
ATM Transaction Needed
Insert Card
Select Interaction
Selected Interaction?
Cash Withdrawal
Select Amount
Cash Amount Selected
Charge Account
Issue Money
Cash Issued
Error
Cash Withdrawn
Check For Further Interactions
30 seconds elapsed
No Further Interaction Requested
Timeout
BPMN.iO

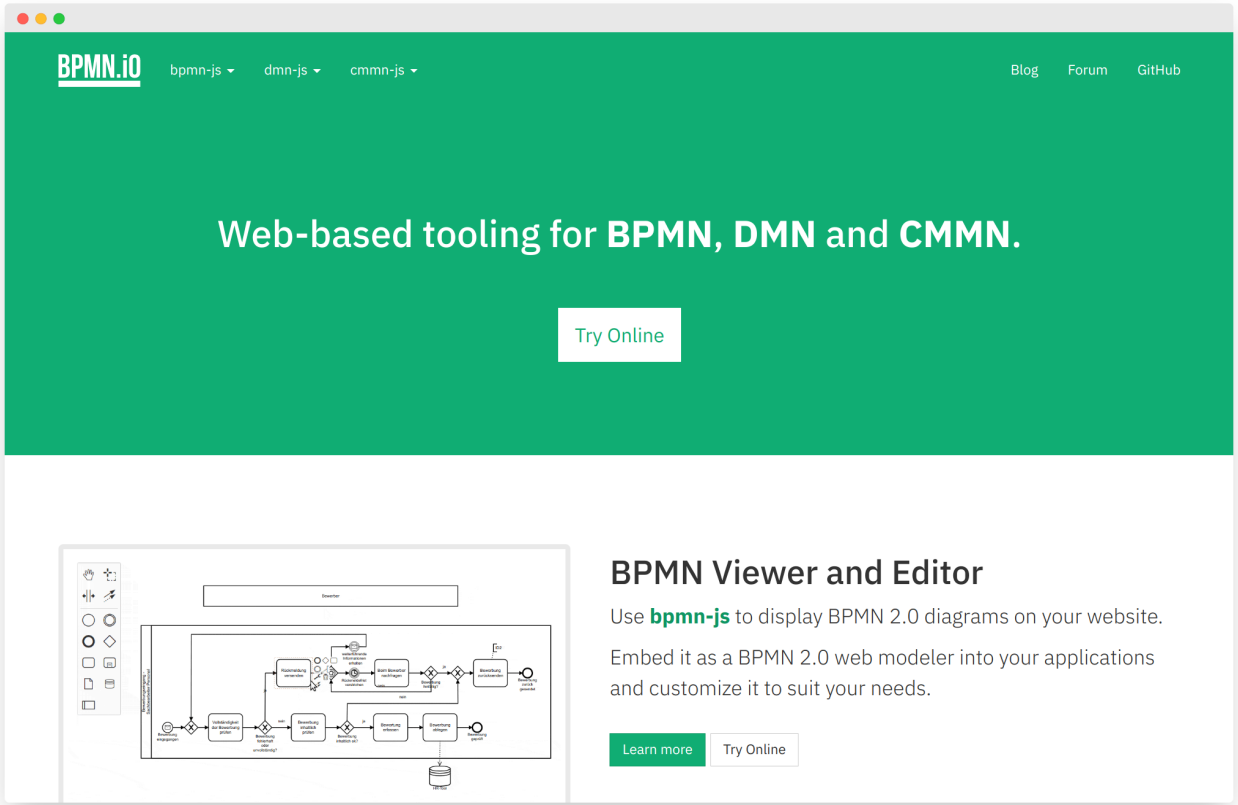
Core Idea: Token Flow = 💡

- Understand wait, join, and split semantics
- Learn BPMN execution in a playful manner
- Aid your understanding of a diagrams semantics

What it is not

- Batch processing simulator
- Business intelligence tool
- Verifier / dead lock / live lock analyzer

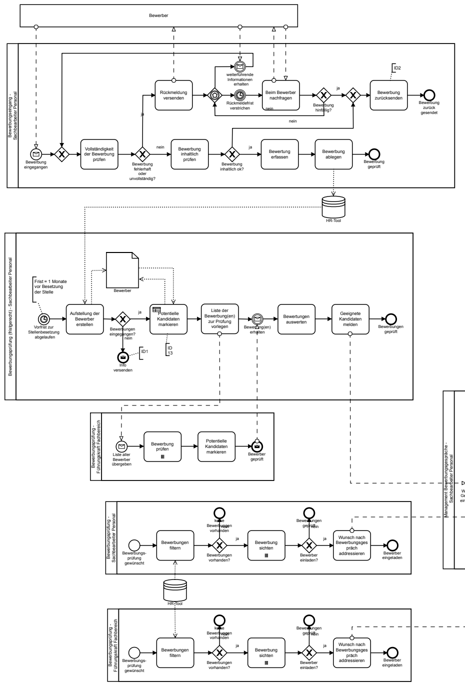
2 About bpmn-js


BPMN.iO
bpmn-js
dmn-js
cmmn-js
Blog
Forum
GitHub
Web-based tooling for BPMN, DMN and CMMN.
Try Online
BPMN Viewer and Editor
Use bpmn-js to display BPMN 2.0 diagrams on your website.
Embed it as a BPMN 2.0 web modeler into your applications and customize it to suit your needs.
Learn more
Try Online

bpmn-js

- A BPMN diagram renderer and editing toolkit
- Embeds into any web page
- Extensible by design

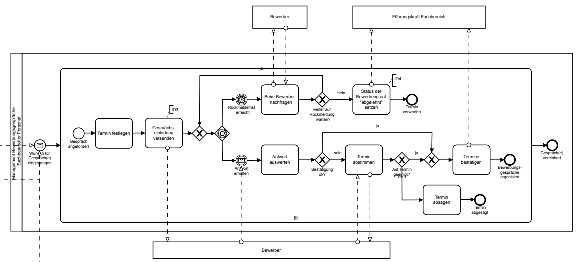
https://github.com/bpmn-io/bpmn-js

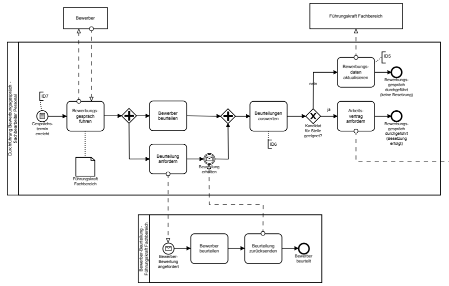


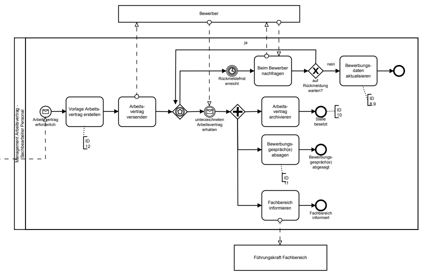
Annahmen:
- alle Bewerbungen gehen über Personal ein.
- unvollständige Bewerbungen werden nicht verfolgt bzw. nach einer Frist verworfen
- Es wurden nur Bewerber eingeladen, da aus Personalsicht inhaltlich geprüft sind. (z.B. müssen die Bewerbungen vollständig sein!)

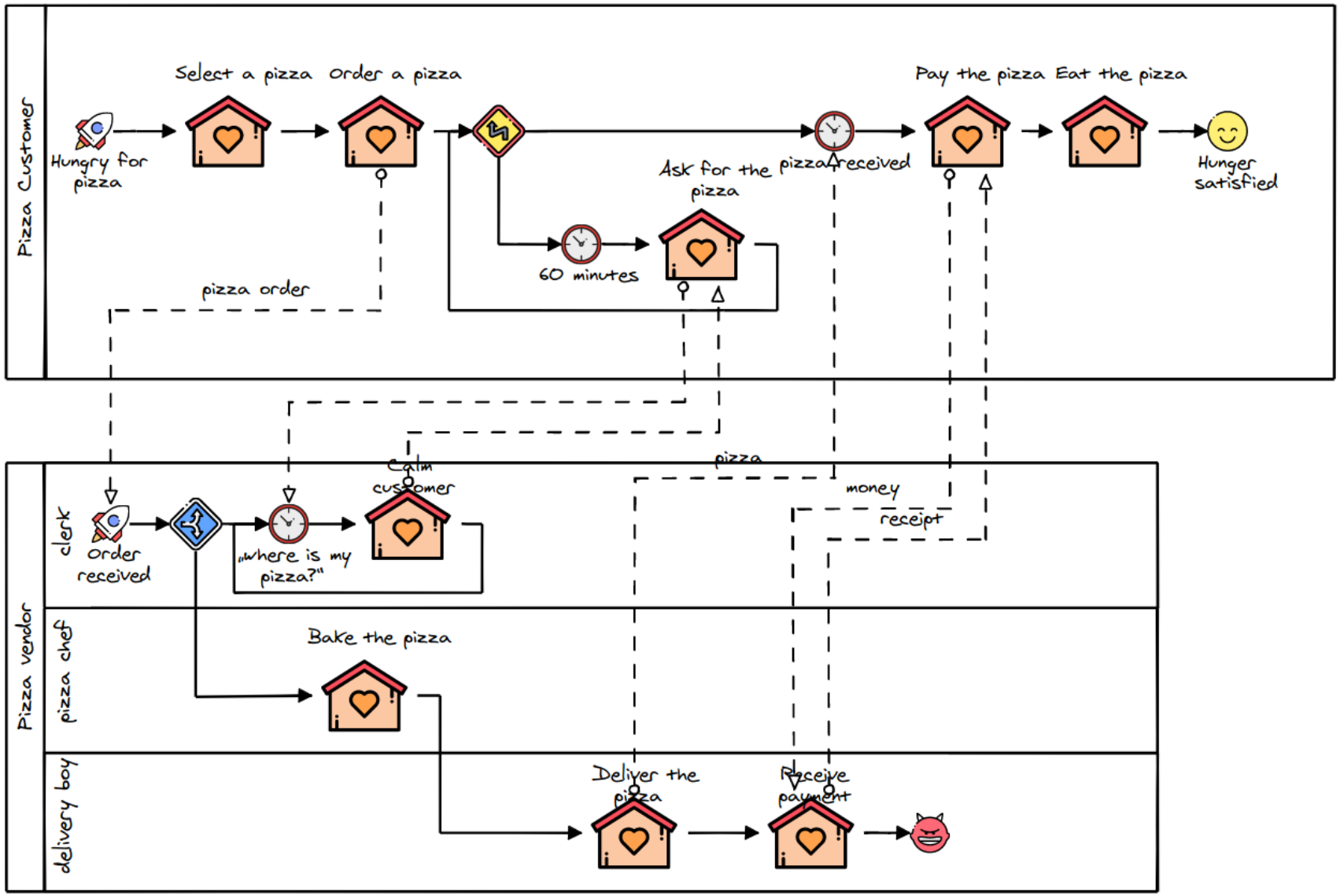
Fehlt noch:
- Teilnehmer kann rückmelden, dass er Bewerbung zurückzieht
- Teilnehmer kann rückmelden, dass er Bewerbungsgespräch nicht wahrnehmen kann (komplette Absage oder neuen Termin vereinbaren)
- Verwendung von Daten ist nicht konsistent im Modell eingezeichnet
- Klärung der offenen ToDos direkt im Diagramm

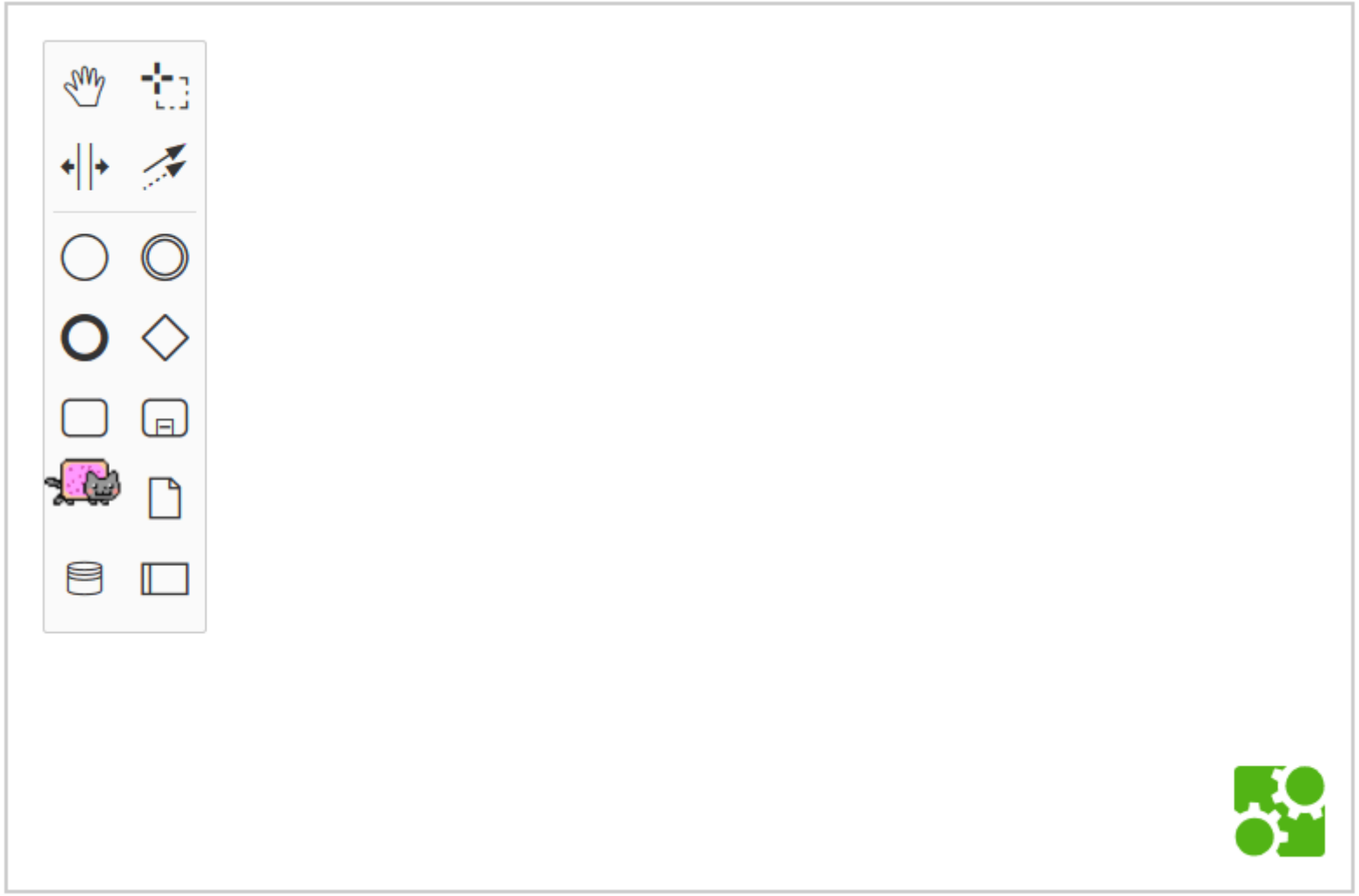
ID1: Was passiert wenn sich kein Bewerber gemeldet hat?
--> Anpassen der Stellenausschreibung, löschen dieser?
ID2: Die Bewerbung soll auch archiviert werden, jedoch mit Status "bei Eingang abgelehnt"
ID3: Die Informationen wann welcher Bewerber eingeladen wurde, soll bitte im Tool erfasst werden.
ID4: Müssen hier die Bewerbungs-unterlagen zurückgesendet werden?
ID5: Müssen hier die Bewerbungsunterlagen zurückgesendet werden?
ID6: kann der Bewerber auch "geparkt" werden, wenn weitere Gespräche folgen? wenn ja, wie soll das geschehen?
ID7: Was passiert wenn der Teilnehmer nicht erscheint?
ID8: Müssen hier die Bewerbungsunterlagen zurückgesendet werden?
ID9: Wie laufen jetzt alle anderen Bewerbungsgespräche? Werden die neu gestartet?
ID10: Muss jetzt die Stelle offline genommen werden?
ID11: Wie funktioniert das genau? Prozesskopplung zu Management Bewerbungsgespräche?
ID12: Soll IT-unterstützt laufen, gemäß vordefiniertem Template
ID13: Vorgabe wie das Filtern von statten gehen soll?! --> Business Rules








Pizza Customer
Select a pizza
Order a pizza
Pay the pizza
Eat the pizza
Hungry for pizza
Ask for the pizza
pizza received
Hunger satisfied
60 minutes
pizza order
pizza
money
receipt
Pizza vendor
clerk
Calm customer
Order received
„Where is my pizza?"
pizza chef
Bake the pizza
delivery boy
Deliver the pizza
Receive payment



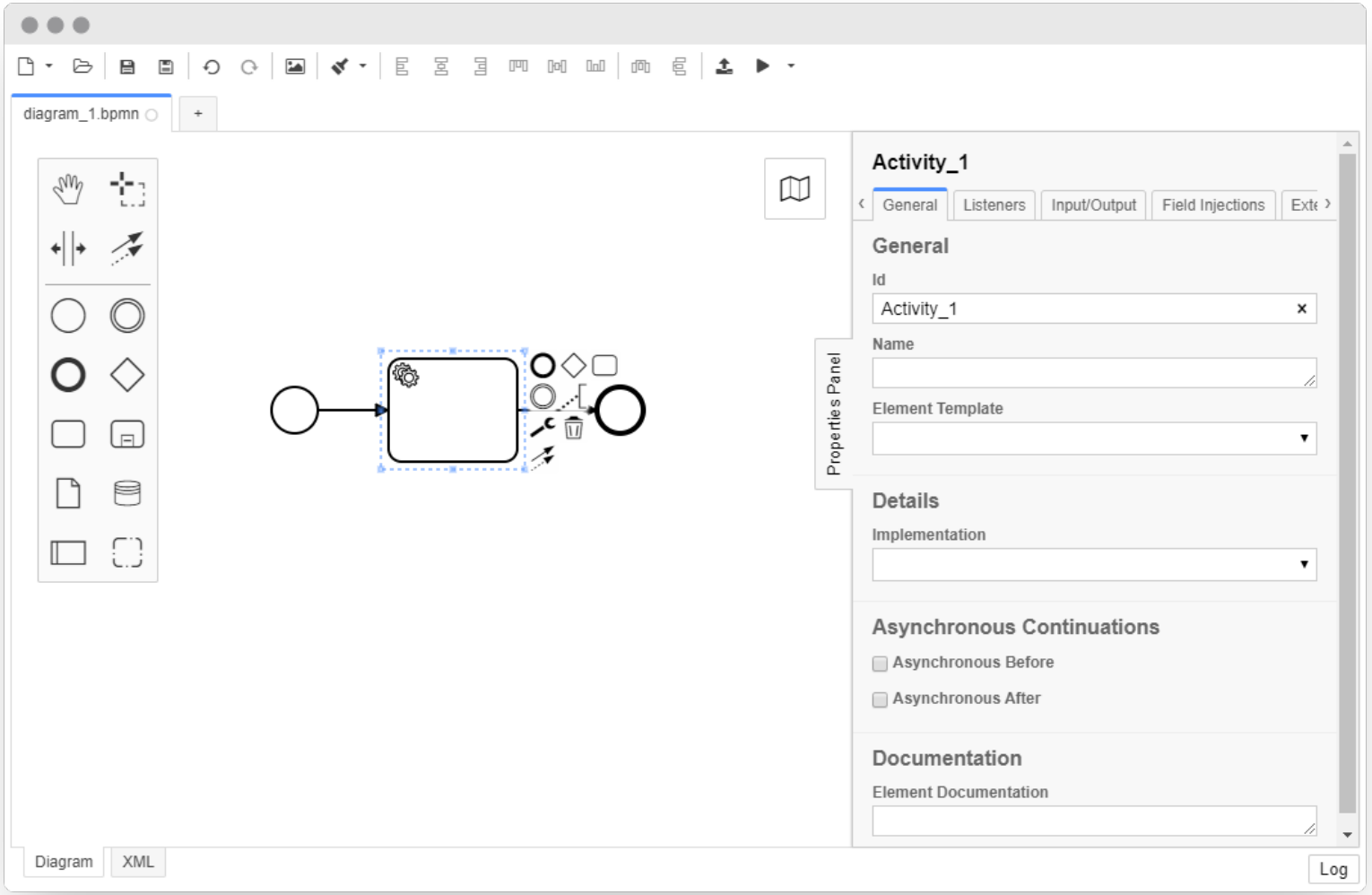


diagram_1.bpmn
Activity_1
General
Listeners
Input/Output
Field Injections
Exte
General
Id
Activity_1
Name
Element Template
Properties Panel
Details
Implementation
Asynchronous Continuations
Asynchronous Before
Asynchronous After
Documentation
Element Documentation
Diagram
XML
Log

3 bpmn-js Extensibility

A (BPMN) Diagram Toolbox

- Element discovery, rendering and interaction
- Selection, navigation, search
- Palette and context pad
- Modeling primitives and stacked behaviors
- Overlays
- …

An Extensible Architecture

- Named Services offer behavior
- Modules group services into logical units
- Instantiation and discovery via dependency injection

Extension Cases

- 🟧 Interface with bpmn-js via API
- 🟦 Build your own extensions
- 🟩 Replace an existing service / functionality

Select an Element

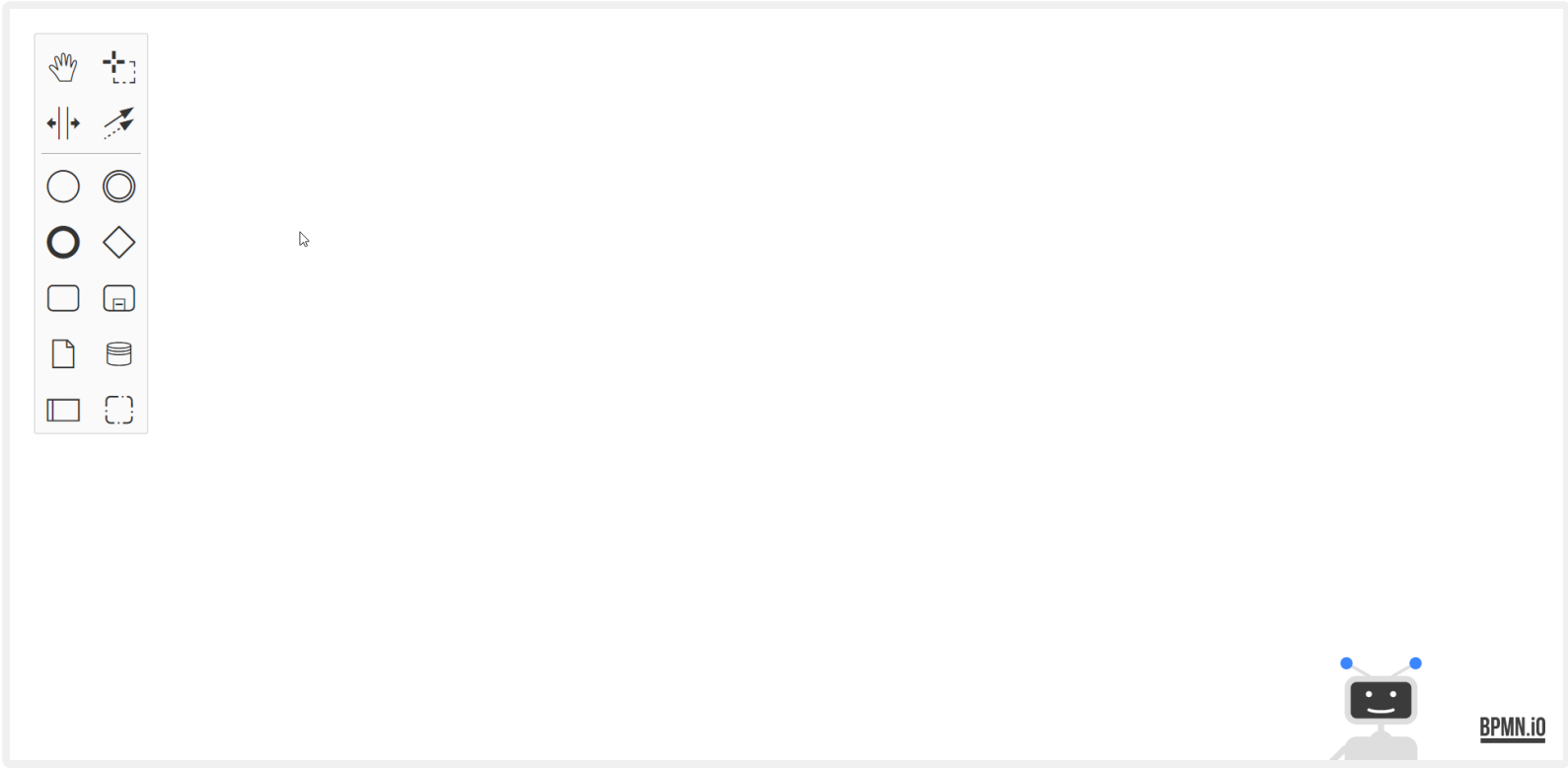
```
const bpmnModeler = new BpmnModeler();

const selection = bpmnModeler.get('selection');
const elementRegistry = bpmnModeler.get('elementRegistry');

selection.select([
  elementRegistry.get('Task_1')
]);
```

Hook into Events

```
const bpmnModeler = new BpmnModeler();

const eventBus = bpmnModeler.get('eventBus');

eventBus.on('element.changed', function(event) {
  console.log('Changed', event.element);
});
```

🟧 Model Programmatically

```
const modeling = bpmnModeler.get('modeling');

modeling.createShape(
  { type: 'bpmn:ServiceTask' },
  { x: 10, y: 20 }
);
```



https://bpmn.io/blog/posts/2021-wasdenn-ai-modeling-assistant.html

Implement a Service

```
// TaskSelection.js
export default function TaskSelection(selection, elementRegistry) {

  /**
   * Select this very special task
   */
  this.selectTask1 = function() {
    selection.select([
      elementRegistry.get('Task_1')
    ])
  };
}
```

🟦 Create a Module

```
// TaskSelectionModule.js
import TaskSelection from './TaskSelection';

export default {
  taskSelection: [ 'type', TaskSelection ]
};
```

Extend bpmn-js

```
import taskSelectionModule from './TaskSelectionModule';

const bpmnModeler = new BpmnModeler({
  additionalModules: [
    taskSelectionModule
  ]
});

const taskSelection = bpmnModeler.get('taskSelection');

taskSelection.selectTask1();
```

4️⃣ How does token simulation plug in?

Code Review!

In a Nutshell

- Visualizations on top of the actual BPMN diagram
- Accounts for editor or viewer
- Custom controls to interact with the simulator
- Disabled modeling interaction ☠️
- A (single instance) process engine

CAMUNDA
COMMUNITY
SUMMIT
2021

Thank you. Questions?

Resources

- https://github.com/bpmn-io/bpmn-js
- https://github.com/bpmn-io/bpmn-js-examples
- https://github.com/bpmn-io/bpmn-js-token-simulation
- https://forum.bpmn.io