CCS 2022

bpmn-js-token-simulation

Your next BPMN engine

Nico Rehwaldt

Introductions

Nico Rehwaldt

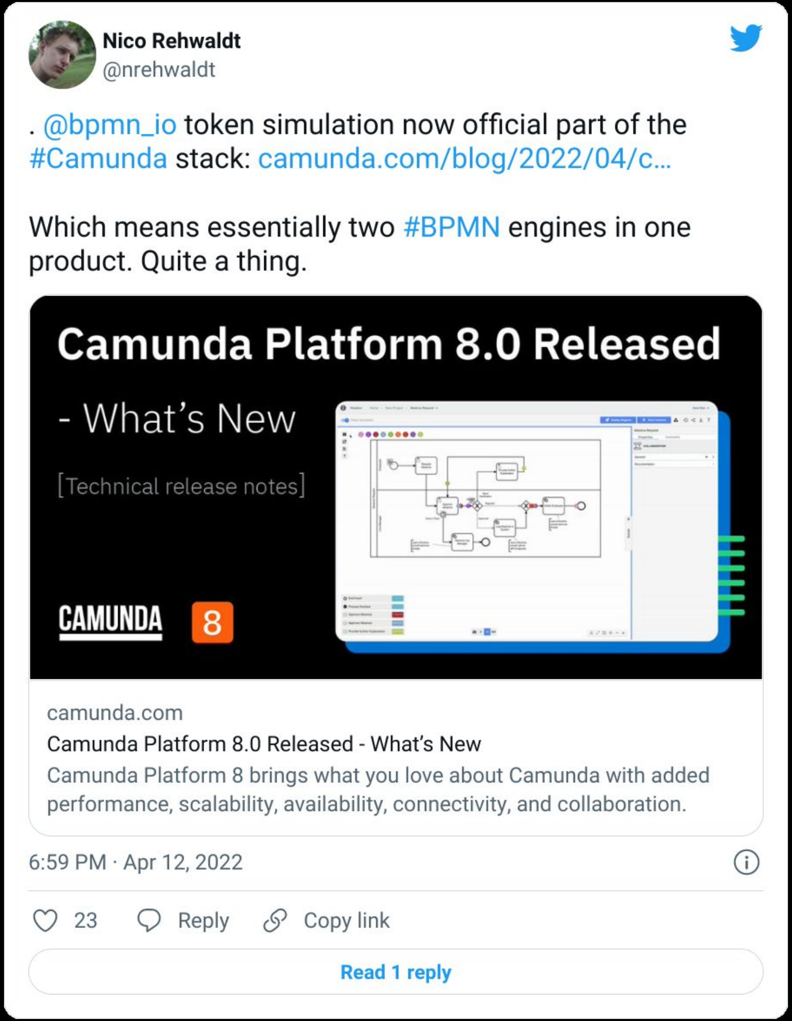
Modeling dude

Camunda

bpmn.io

Home of bpmn-js and friends

Camunda



Nico Rehwaldt
@nrehwaldt

. @bpmn_io token simulation now official part of the #Camunda stack: camunda.com/blog/2022/04/c…

Which means essentially two #BPMN engines in one product. Quite a thing.

camunda.com

Camunda Platform 8.0 Released - What's New

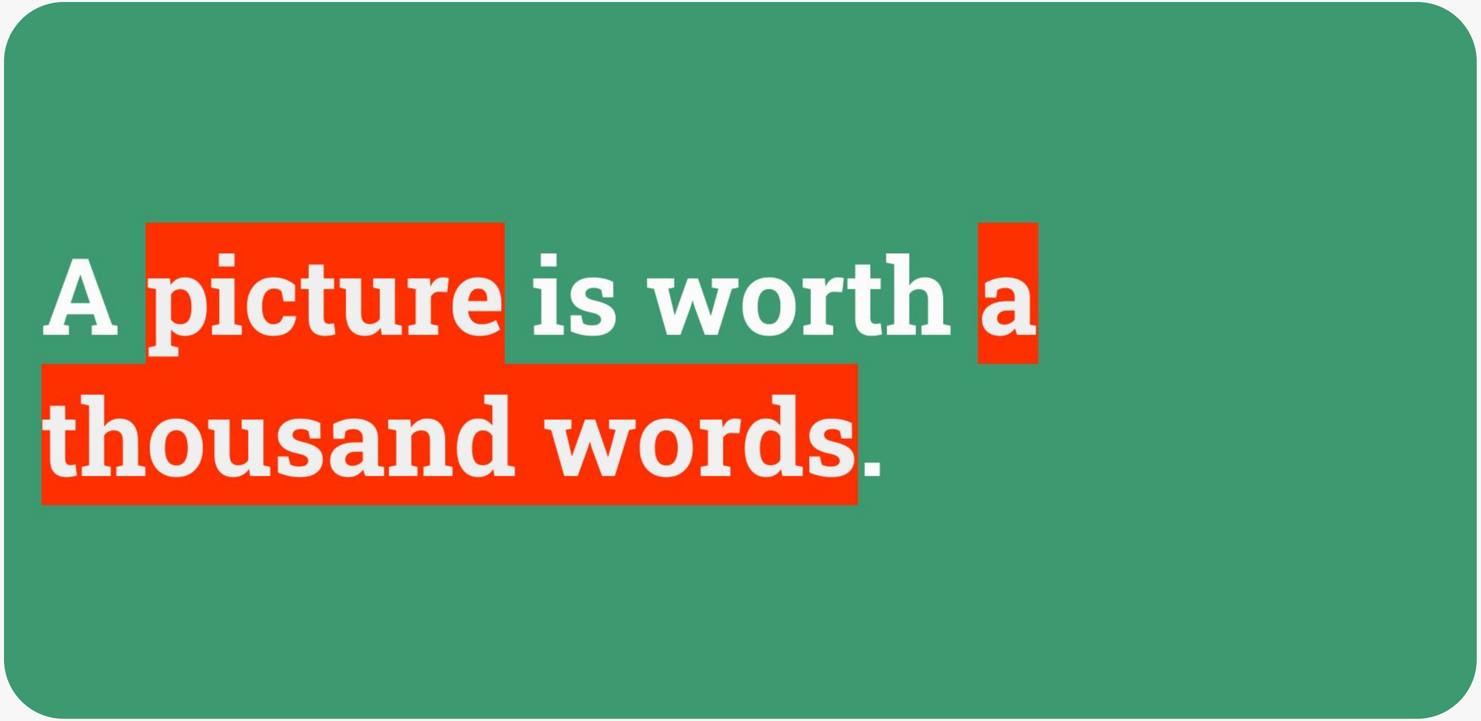
Camunda Platform 8 brings what you love about Camunda with added performance, scalability, availability, connectivity, and collaboration.

6:59 PM · Apr 12, 2022

Agenda

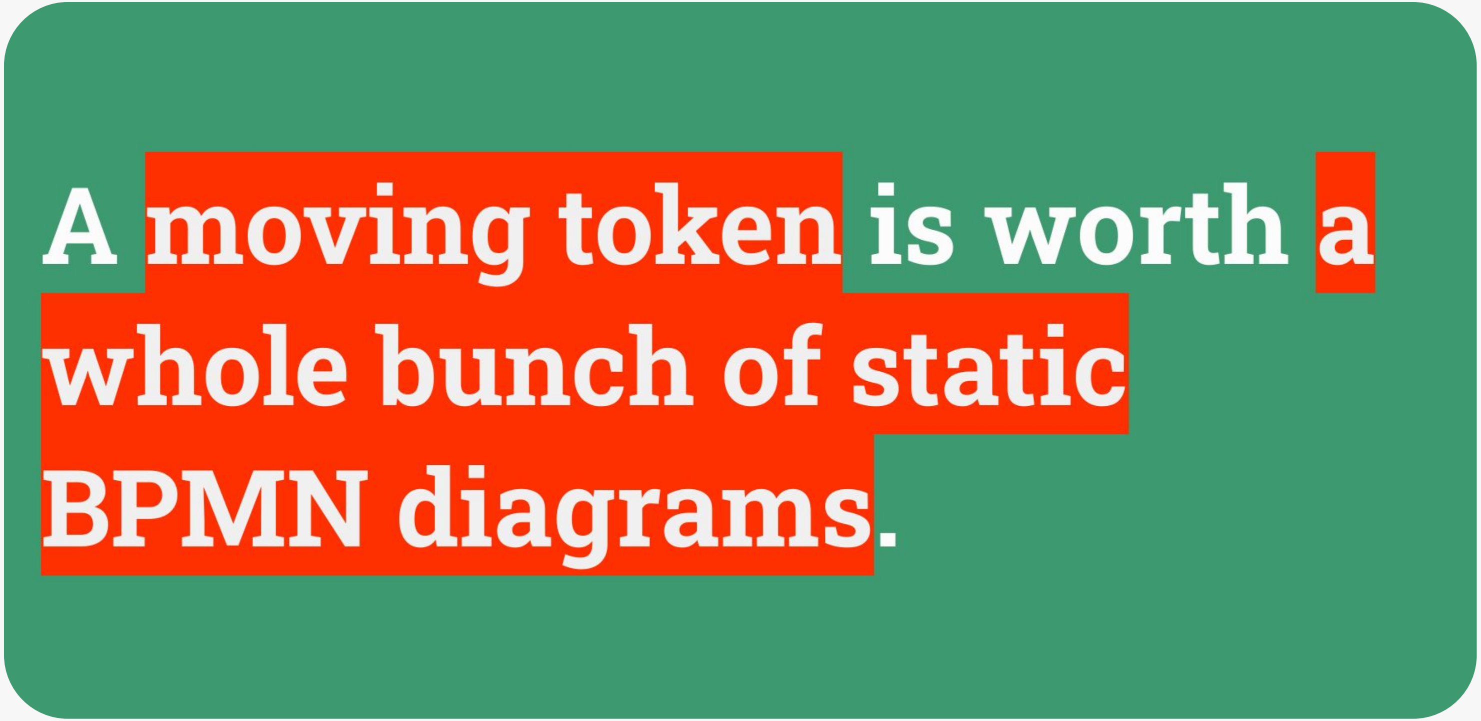
- The Case for Token Simulation
- Yet another BPMN engine?
- The missing pieces to replace Camunda

The Case for Token Simulation



A picture is worth a thousand words.

https://nikku.github.io/talks/2021-token-simulation/presentation.html#5

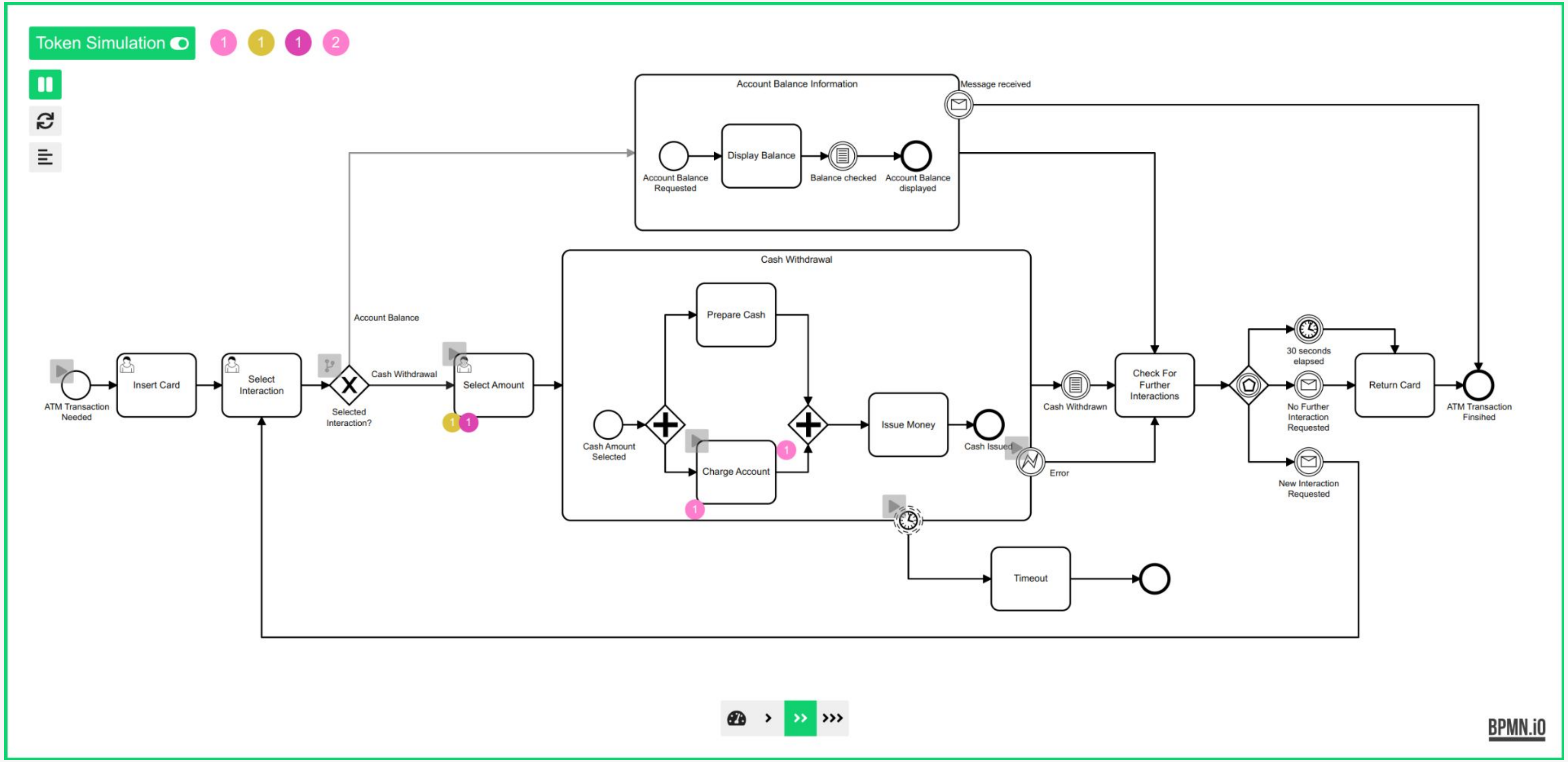


A moving token is worth a whole bunch of static BPMN diagrams.

https://nikku.github.io/talks/2021-token-simulation/presentation.html#6

> “Get these tokens moving and understand #BPMN core concepts one token at a time.

Marketing Dude
bpmn.io

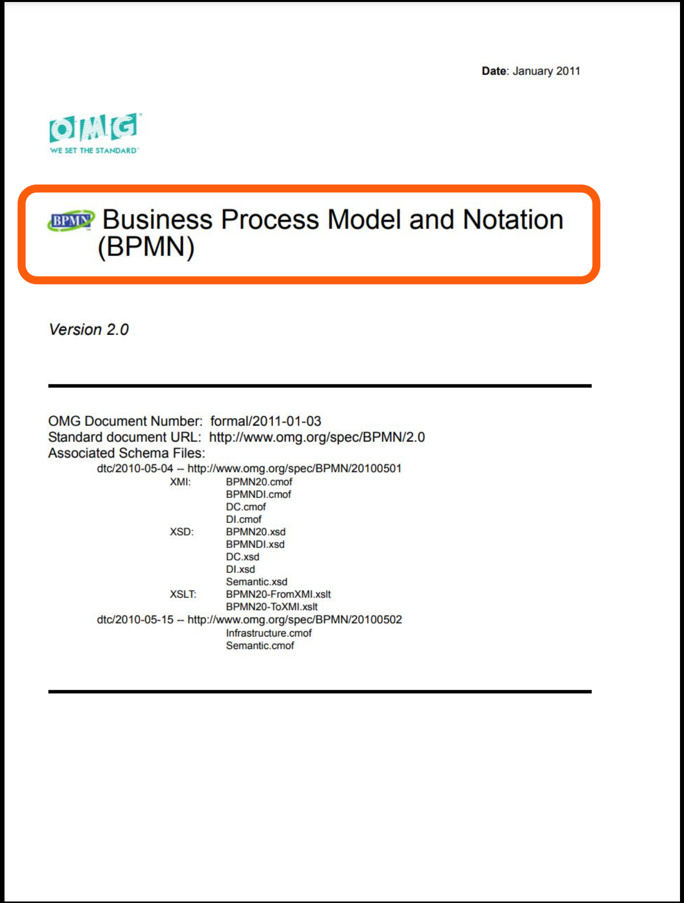


https://bpmn-io.github.io/bpmn-js-token-simulation/modeler.html

Takeaways

- It is a **BPMN learning tool**.
- It is **more and less** than a classic BPMN engine.
- It **builds on top of bpmn.io tooling** and is **open-source, on GitHub**.

Yet another BPMN engine?



Date: January 2011

Business Process Model and Notation (BPMN)

Version 2.0

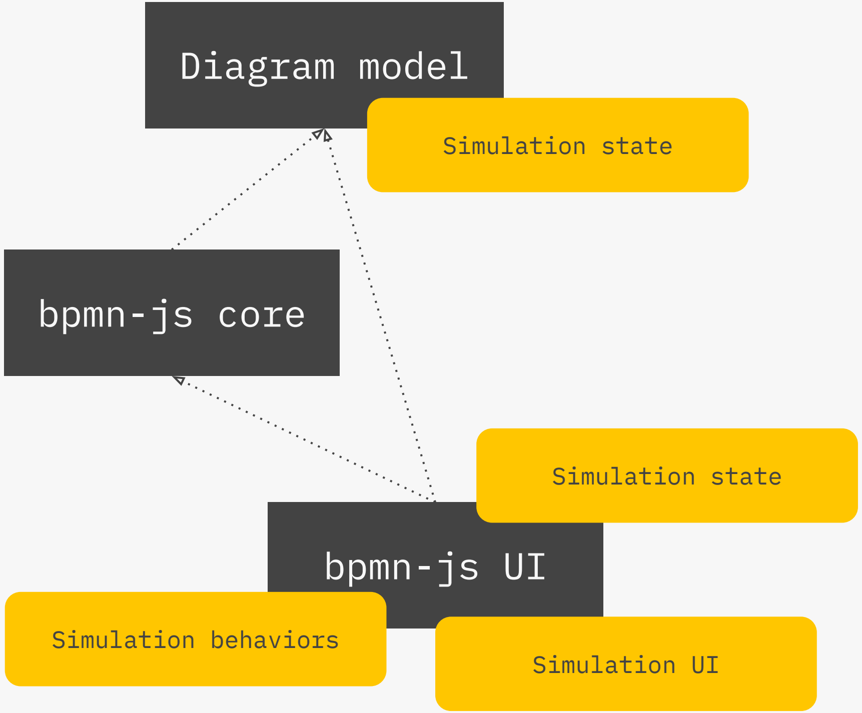
OMG Document Number: formal/2011-01-03
Standard document URL: http://www.omg.org/spec/BPMN/2.0
Associated Schema Files:

```
        dtc/2010-05-04 -- http://www.omg.org/spec/BPMN/20100501
                    XMI:        BPMN20.cmof
                                BPMNDI.cmof
                                DC.cmof
                                DI.cmof
                    XSD:        BPMN20.xsd
                                BPMNDI.xsd
                                DC.xsd
                                DI.xsd
                                Semantic.xsd
                    XSLT:       BPMN20-FromXMI.xslt
                                BPMN20-ToXMI.xslt
        dtc/2010-05-15 -- http://www.omg.org/spec/BPMN/20100502
                                Infrastructure.cmof
                                Semantic.cmof
```

508

Business Process Model and Notation, v2.0

A history in four chapters

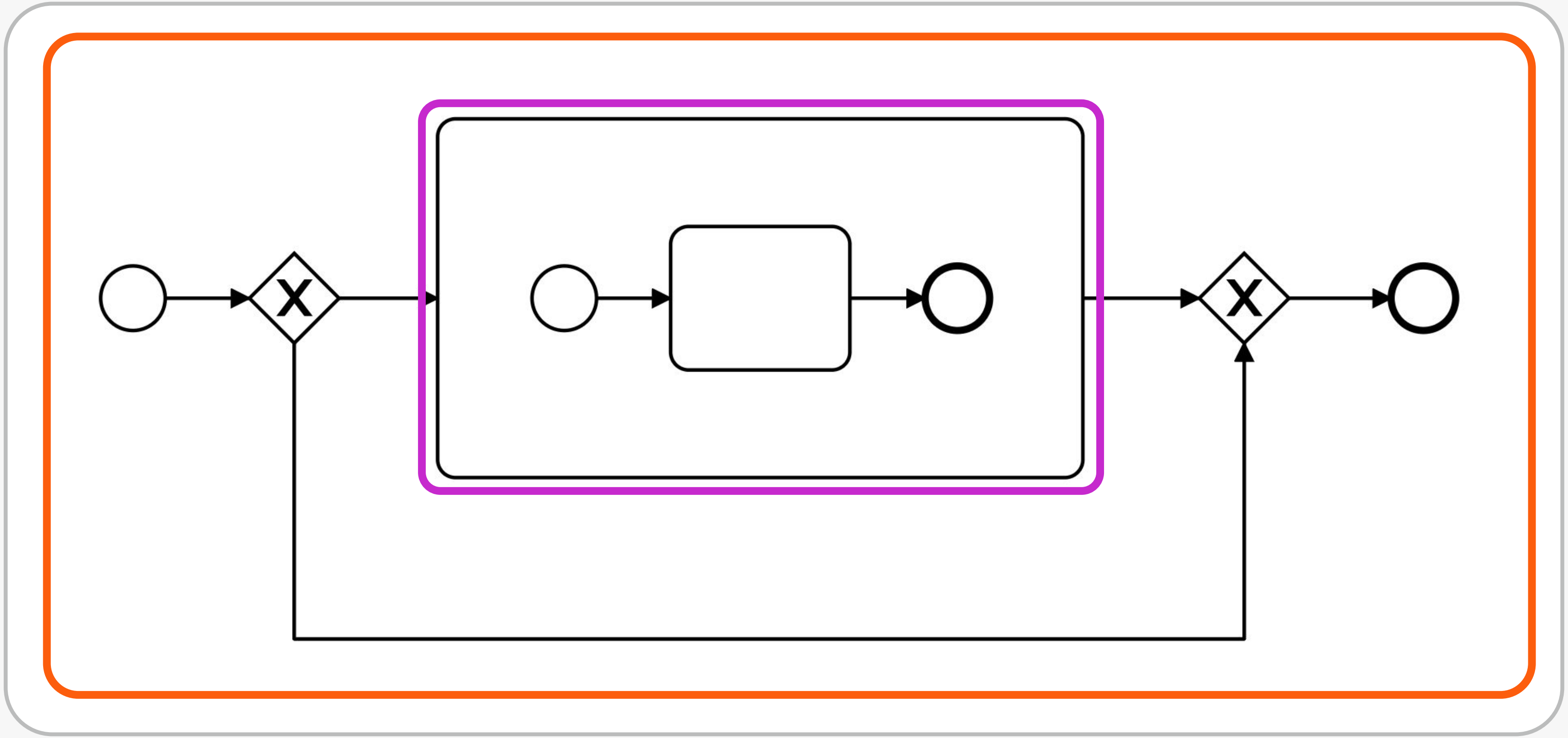


0.1.0

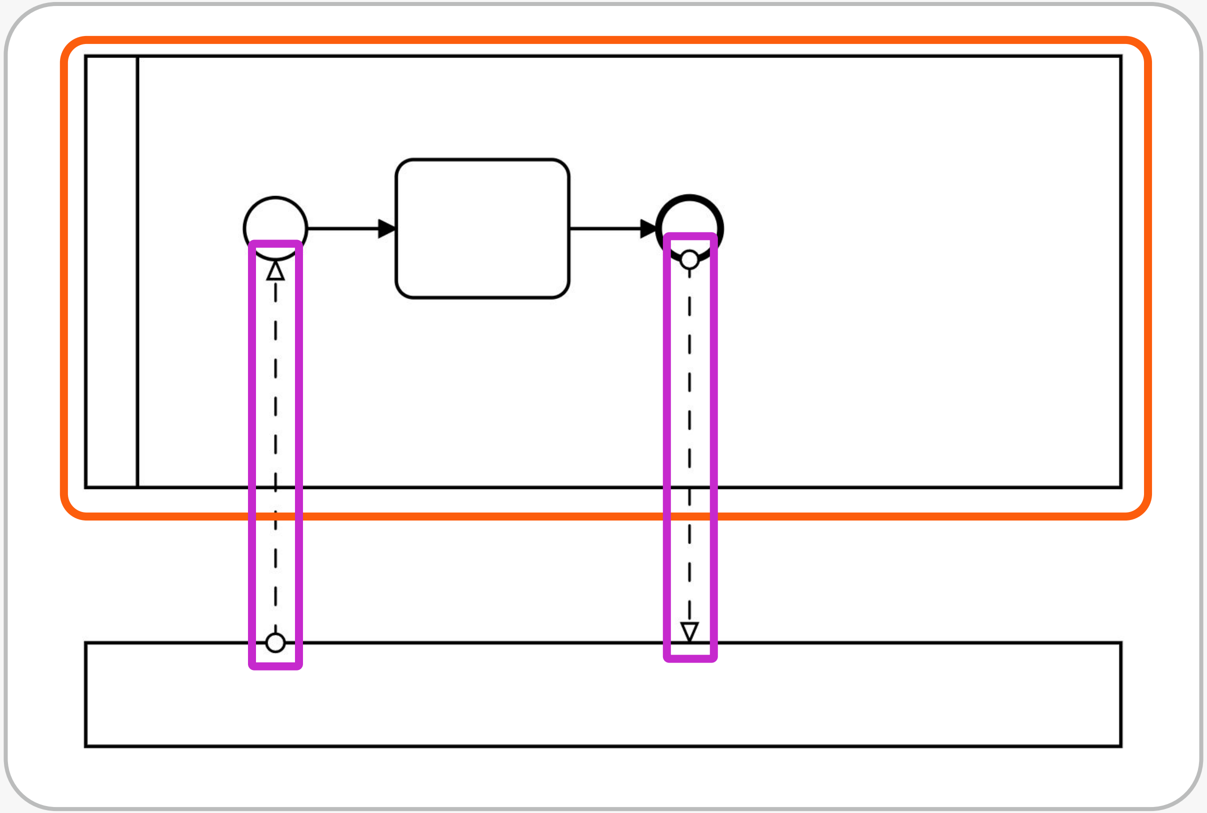
Aug 2017

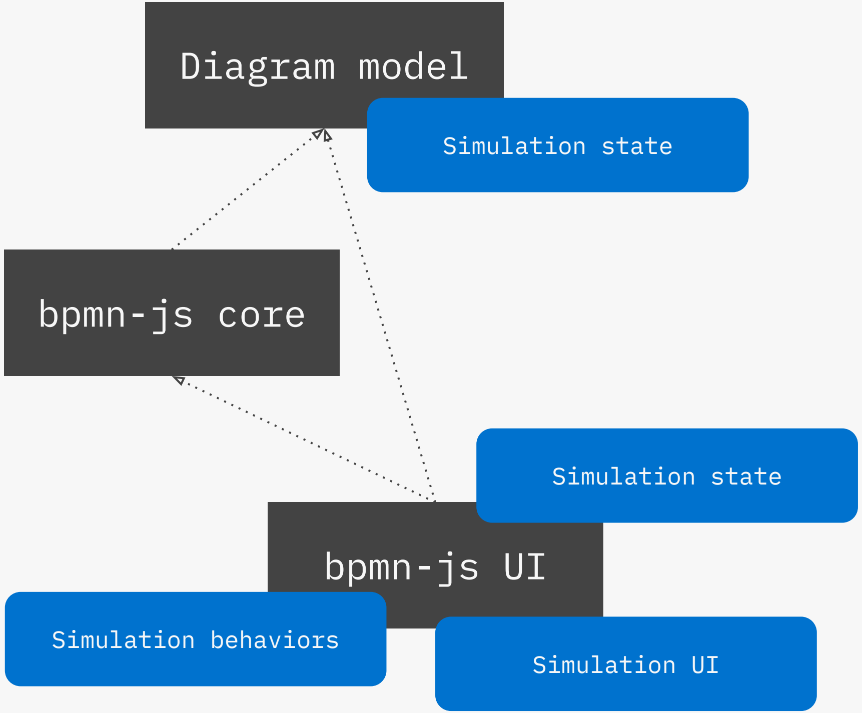
Early prototype / Camunda hackday project

Initial proof of concept, introducing UX foundations. Single process instance simulation.



X
X



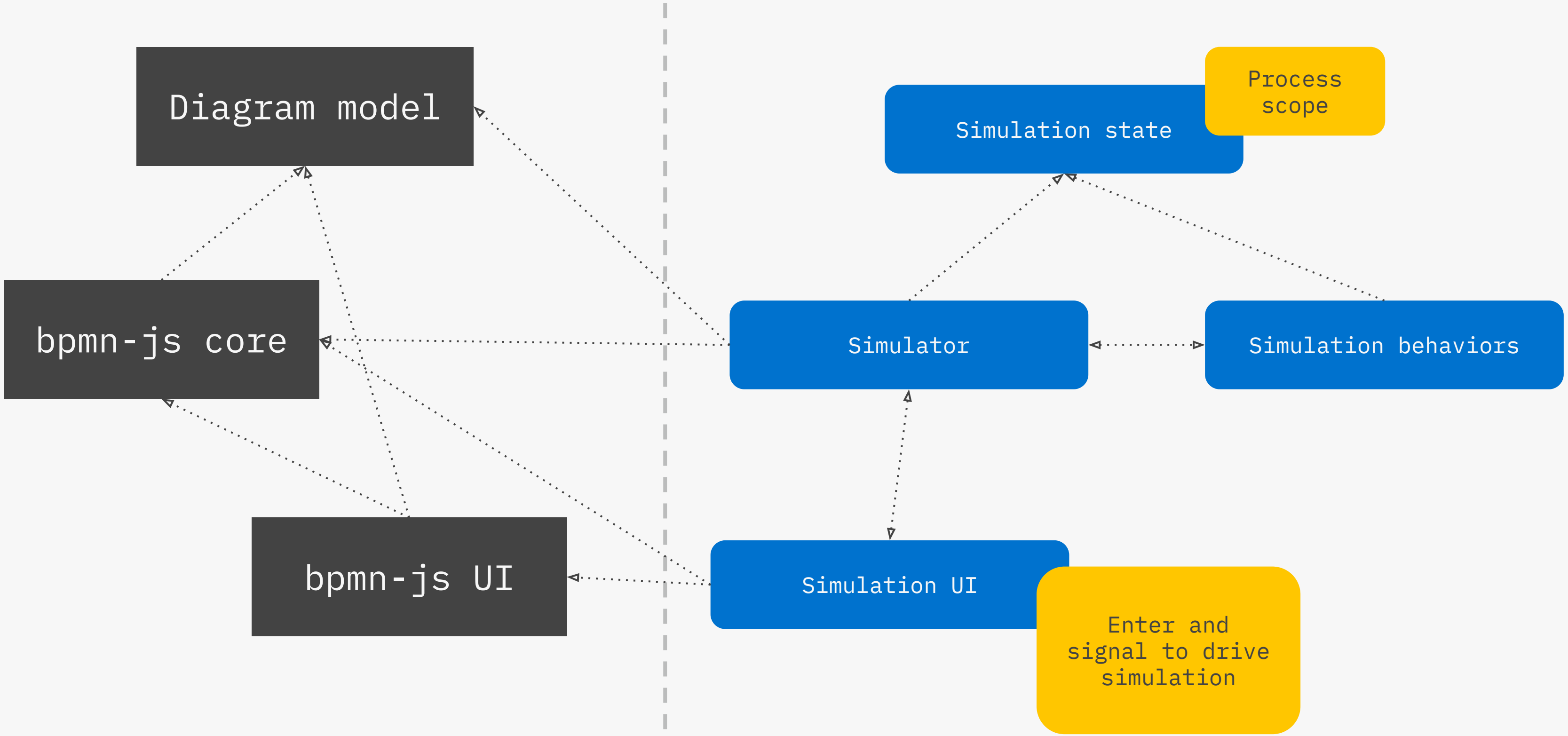


0.1.0

Aug 2017

Early prototype / Camunda hackday project

Initial proof of concept, introducing UX foundations. Single process instance simulation.

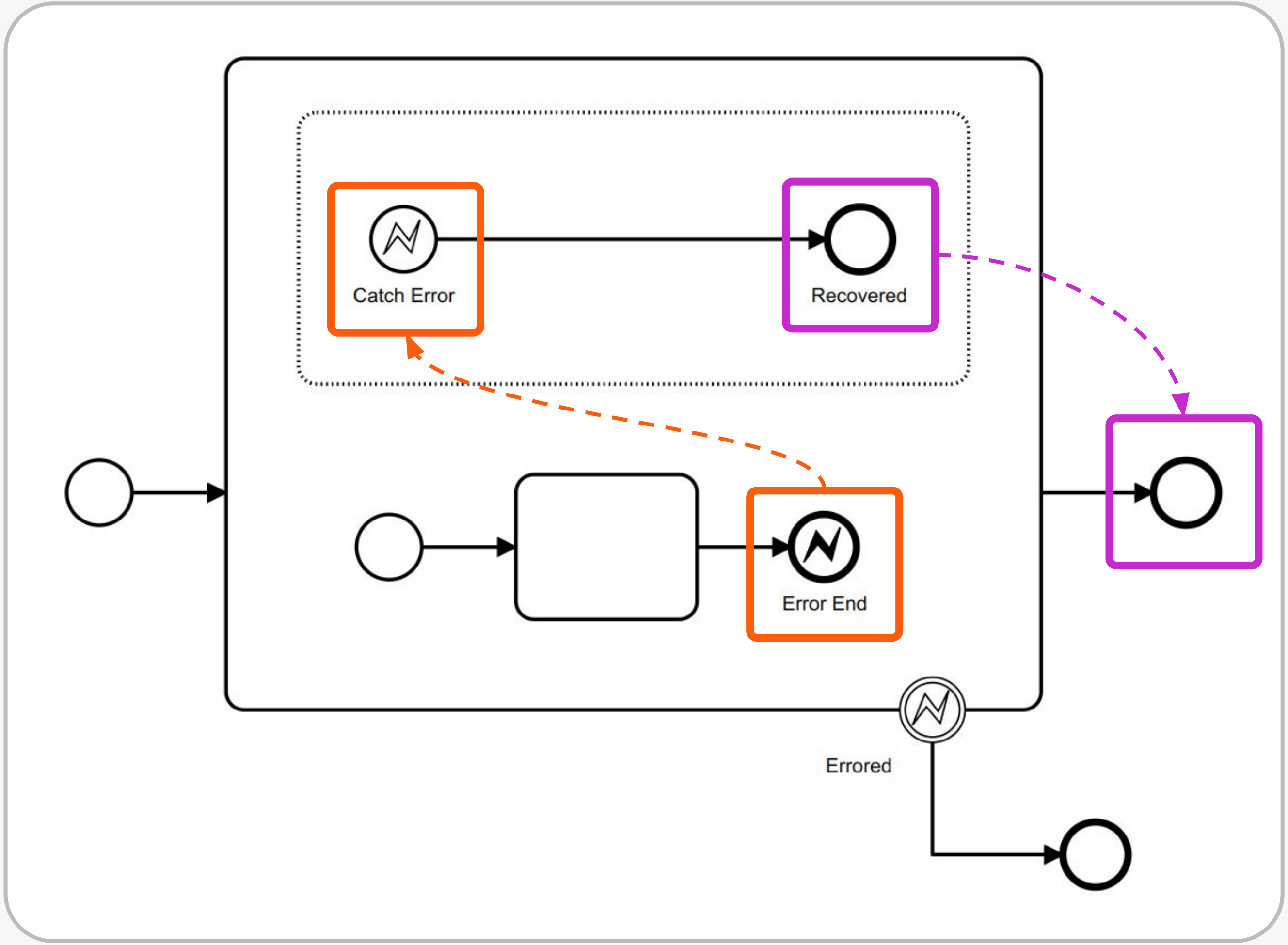


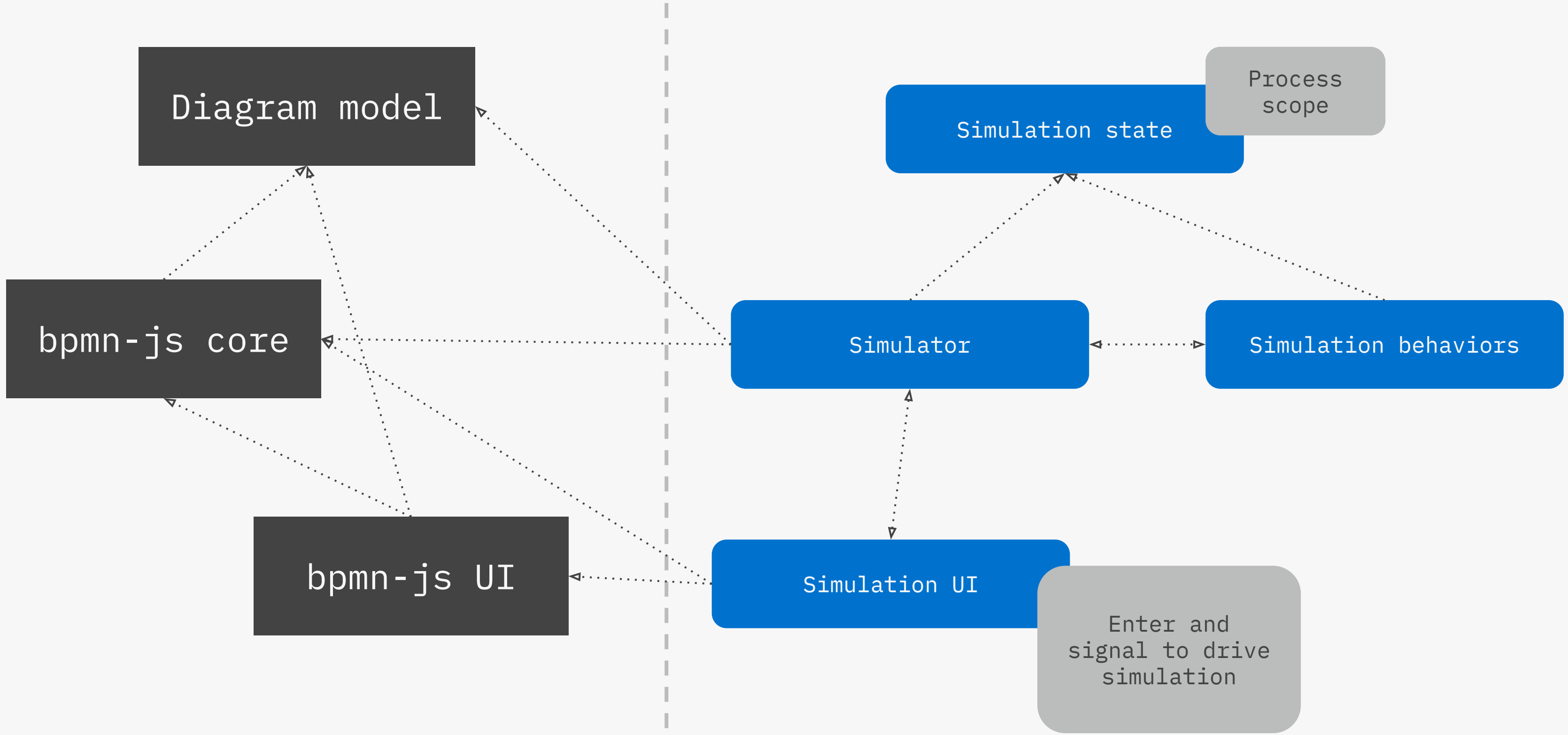
0.12.0

May 2021

Separate diagram visuals and simulation

Simulator is a standalone tool. Process scopes, multi-instance simulation.


Catch Error
Recovered
Error End
Errored

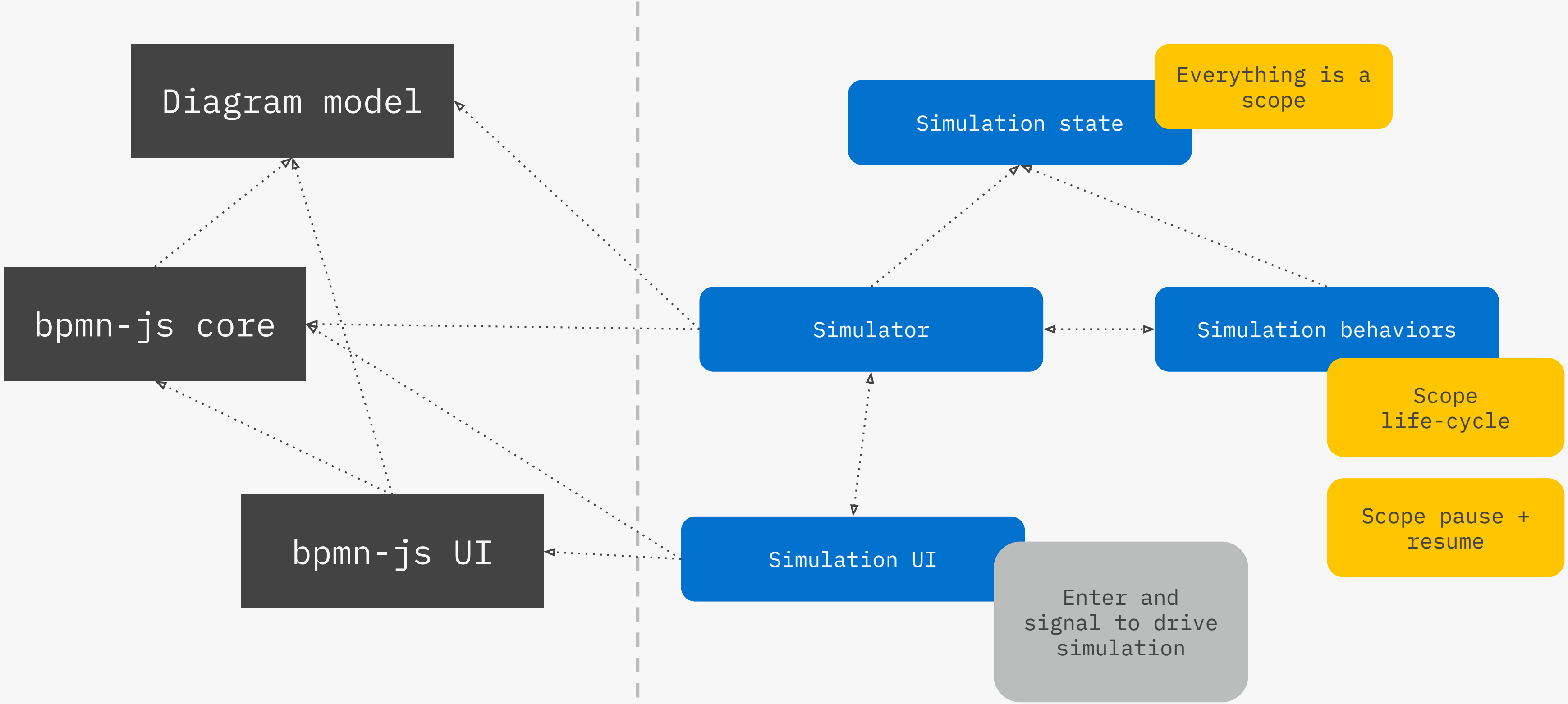


0.12.0

May 2021

Separate diagram visuals and simulation

Simulator is a standalone tool. Process scopes, multi-instance simulation.

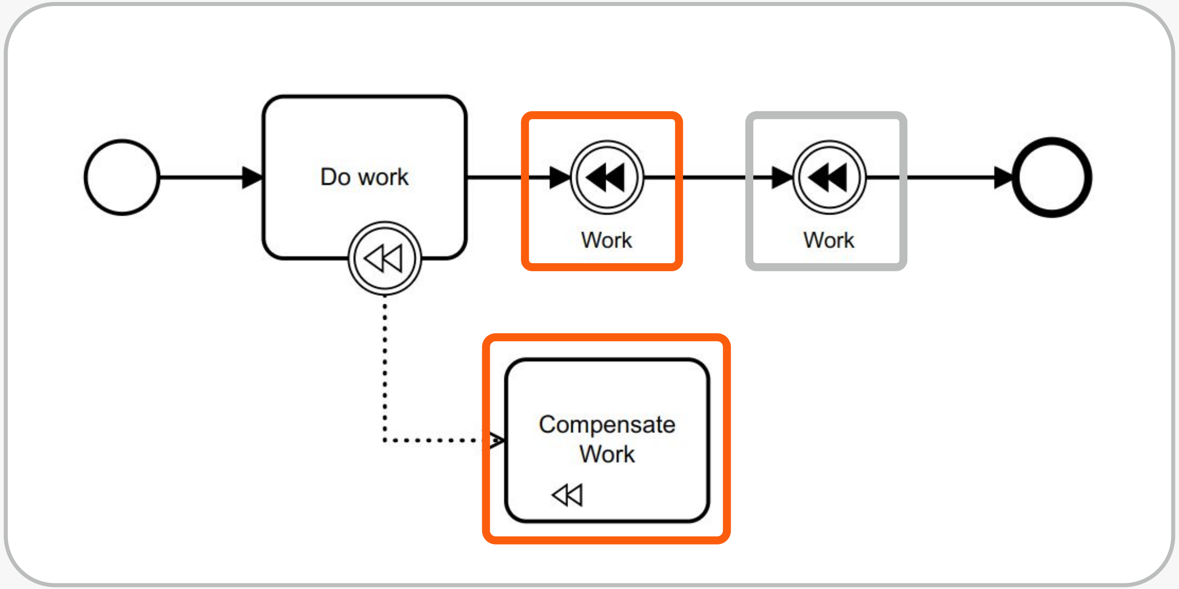


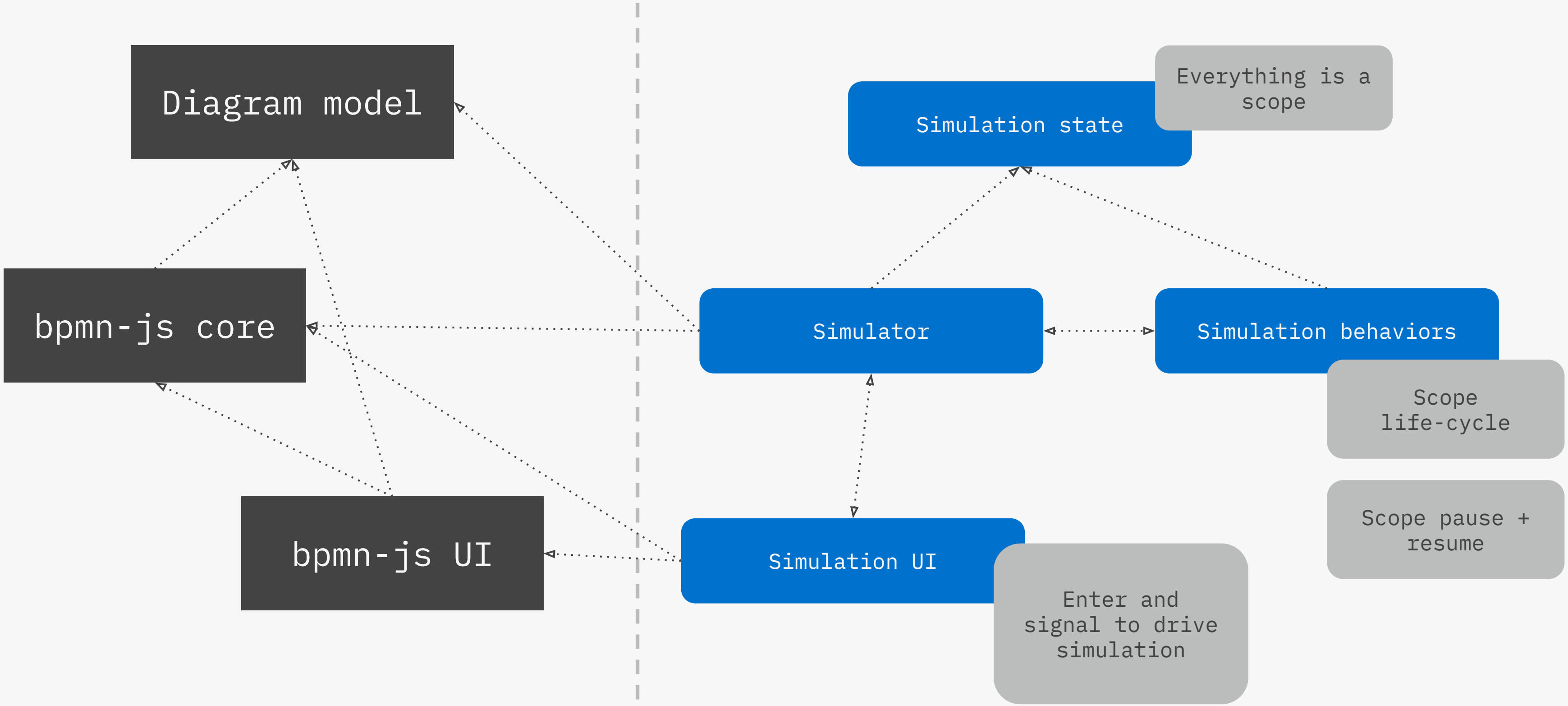
0.18.0

May 2021

Handle scope interruptions according to BPMN spec

Make everything a scope. Thinks work only if they work according to the BPMN spec.


Do work
Work
Work
Compensate Work

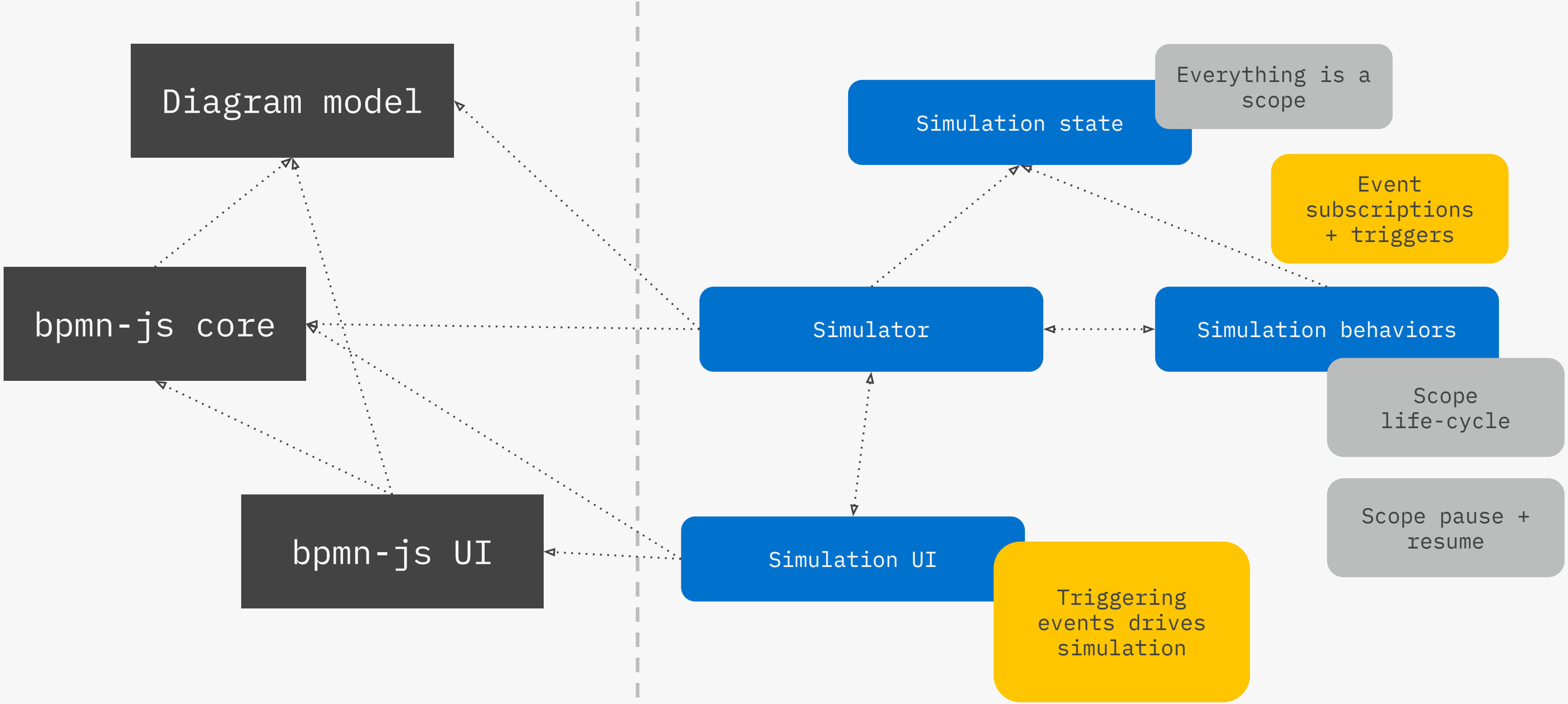


0.18.0

May 2021

Handle scope interruptions according to BPMN spec

Make everything a scope. Thinks work only if they work according to the BPMN spec.



0.25.0

Jan 2022

Rewrite simulator to use event subscriptions and triggers

Introducing transactions and compensation. Do we really have to re-read the BPMN 2.0 spec, *again?*

Project History

0.1.0
Aug 2017

Early prototype / Camunda hackday project

Initial proof of concept, introducing UX foundations. Single process instance simulation.

0.12.0
May 2021

Separate diagram visuals and simulation

Simulator is a standalone tool. Process scopes, multi-instance simulation.

0.18.0
May 2021

Handle scope interruptions according to BPMN spec

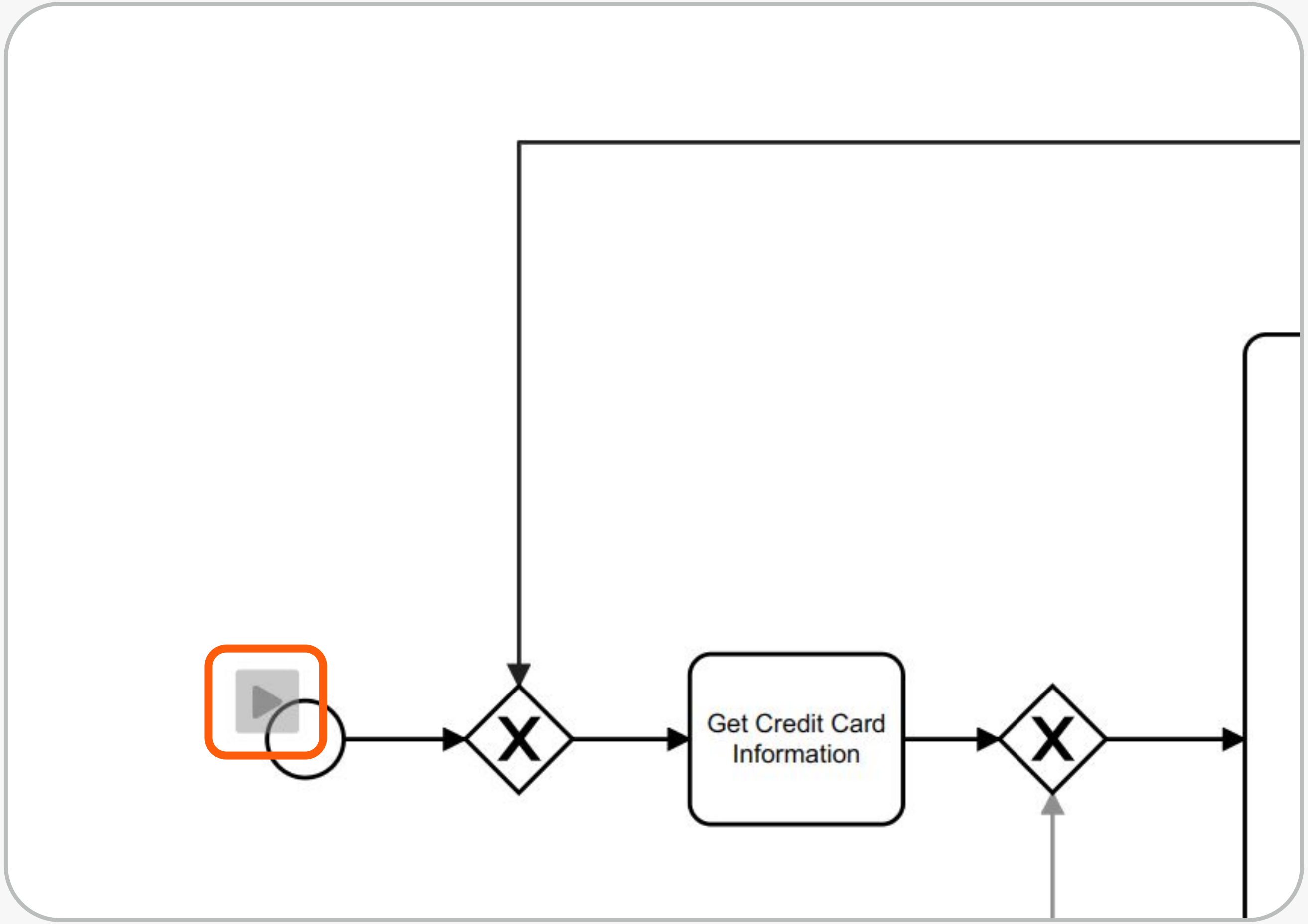
Make everything a scope. Thinks work only if they work according to the BPMN spec.

0.25.0
Jan 2022

Rewrite simulator to use event scopes and subscriptions

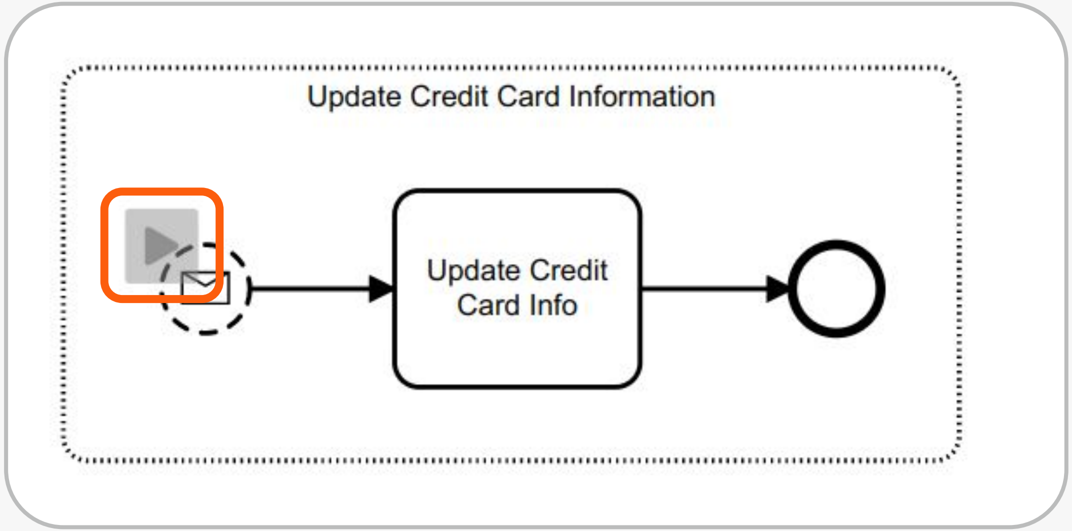
Introducing transactions and compensation. Do we really have to re-read the BPMN 2.0 spec, *again?*

▶ = Trigger any event



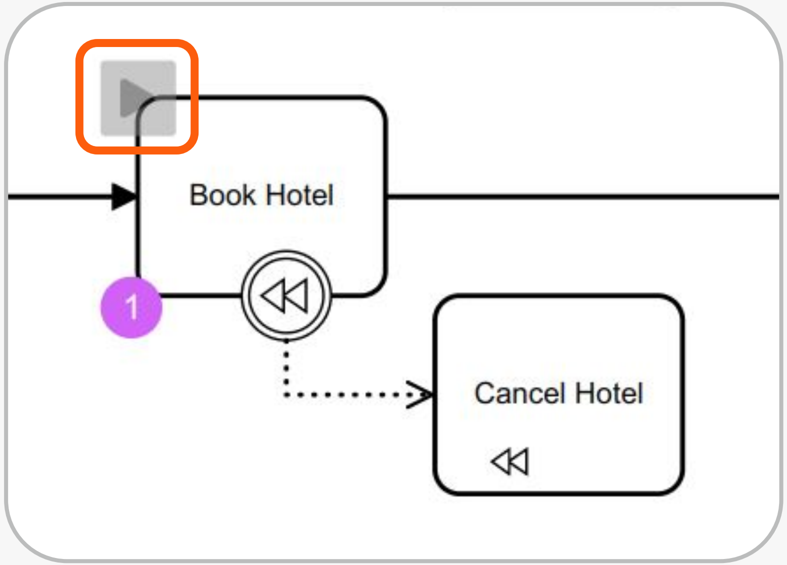
Process definitions are passive scopes.

They register global event subscriptions which trigger process instance start.



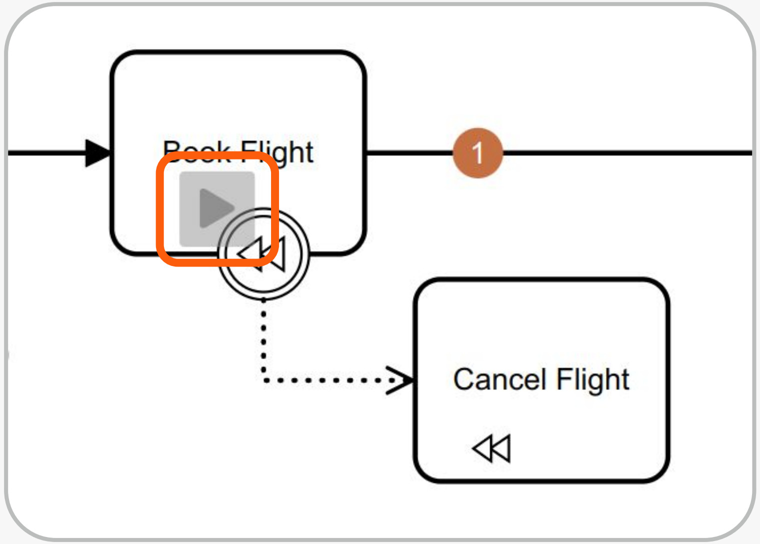
Event sub-process scopes instantiate on event trigger.

A subscription is created when the parent scope activates and removed once the parent scope interrupts or completes.

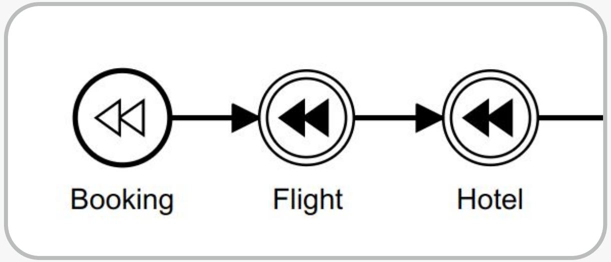


Activity instance scopes register an event subscription once waiting to allow resume.

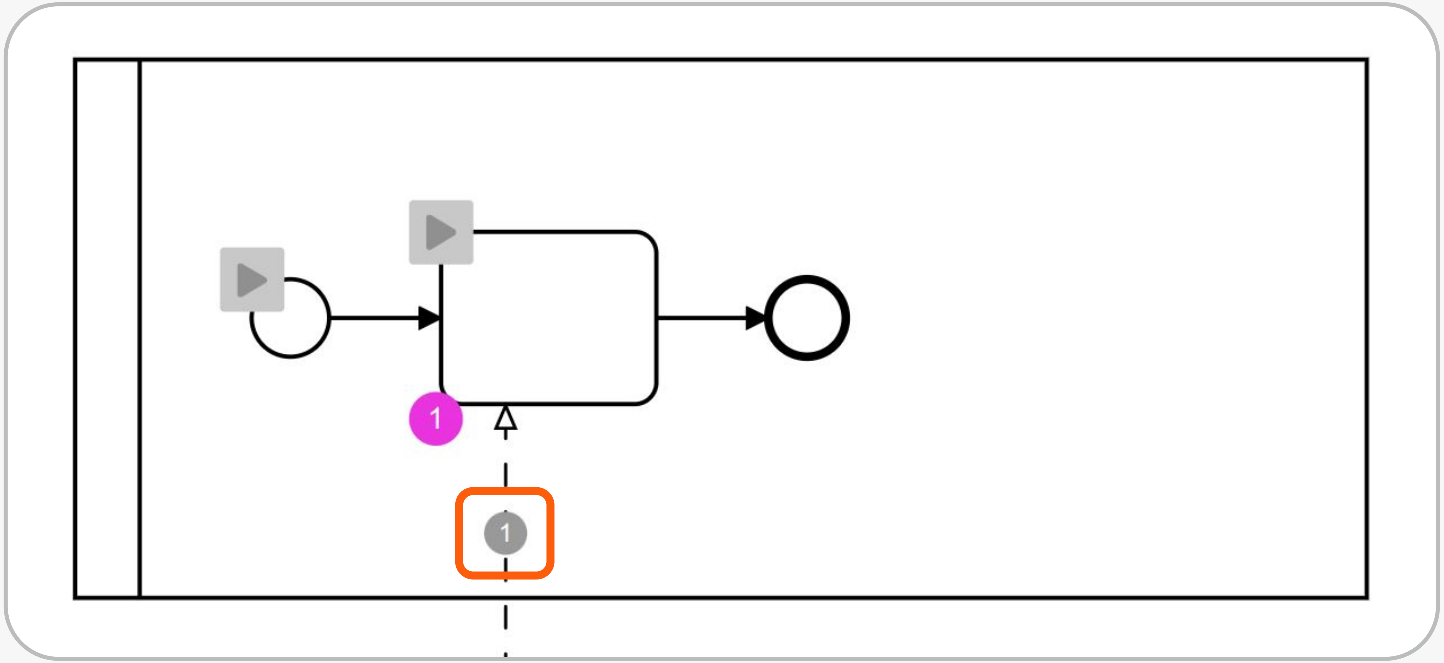
Subscription is canceled once the scope interrupts or completes.



Compensation is just another event subscription, registered upon scope completion...



...and triggered by the user or implicitly when “throwing” the related compensate event.



Any throwing event triggers actions of matching catching subscriptions.

We implemented scopes, life-cycle, behaviors and eventing in accordance with the BPMN spec.

We’ve built a BPMN engine.

The missing pieces to replace Camunda

A few things ;-)

Roadmap

- Headless (readonly) bpmn-js 💀💀
- Data (variable handling, expression evaluation) 💀
- Multi instance markers (parallel, sequential, loop) 💀
- Unbuild simulation only behaviors
- Call up real business logic
- Scalability + resilience 💀💀💀

Token simulation is awesome.

So get your tokens moving to better understand your diagrams.

Questions?

THANK YOU

nico.rehwaldt@camunda.com

github.com/nikku

@nrehwaldt