Towards the ontological characterisation of BPMN

Laboratory for Applied Ontology (ISTC-CNR)

January, 24th

Analysis of BPMN: Current approaches

(1) **Syntactic approaches**: formal representation of the BPMN language;

  - ▶ E.g., *Start Event* does not have any incoming *Sequence Flow*.
  - ▶ Application: to reason, validate structural constraints [**static validation**], annotate process models, etc.;

(2) **Behavioural approaches**: formal representation aimed at executing BPMN process models.

  - ▶ Application: simulate process models, e.g., to spot deadlocks or livelocks in process model execution [**dynamic validation**];

Analysis of BPMN

(3) **Ontological approaches**: ontological characterization of the BPMN meta-model

- ▶ SUMO-based analysis [2]; DOLCE-based analysis [4], WWB-based analysis [1], UFO-based analysis [3]
- ▶ E.g., what is the difference between BPMN tasks and events?
- ▶ Application: semantic enrichment of BPMN to foster interoperability, annotations, etc.

In [2], integration of syntactic and ontological (based on SUMO) approaches.

Goal-oriented process modelling

BPMN is goal-oriented

- An instance of a BPMN model is a DOLCE perdurant* aimed at achieving a (pre-defined) **goal**.
  - Perdurant p instantiates BPMN model M with goal g, if and only if p achieves g (among other constraints)
  - It seems to be a shared view;
- If the **goal** is NOT achieved? p is not an instance of M
- BPMN Activities are also goal-oriented: they always end by achieving (pre-defined) goals
  - The goal of a BPMN Activity A – within a BPMN process model M – contributes to M's overall goal

Tasks' atomicity

BPMN Task: <u>atomic</u> BPMN activity

- ▶ A task does not comprise further tasks

Two interpretations:

- ▶ A task is represented in a BPMN model as atomic and its occurrence in time is atomic, too.
- ▶ A task is represented in a BPMN model as atomic, BUT its occurrence in time comprise further temporal parts.
  [**atomicity at the level of the representation**]
  - ▶ E.g., *write email* (task) comprises *opening a browser*, *go to email provider*, etc.

Tasks vs Events

OVERLAP between (some) tasks and (throw/catch) events:

- ▶ Send Task
- ▶ Throw Event
    - ▶ Both can be used to send, e.g., a message
- ▶ Receive Task
- ▶ Catch event
    - ▶ Both can be used to wait for something to happen

In [1], **avoid** send/receive tasks, "because their semantics, as described in the BPMN standard, does not significantly differ from the semantics of intermediate throwing and catching message events" [1, p.49] (also [2])

Receive Tasks vs Catch Events

Receive Tasks:

- "It is designed to wait for a Message to arrive from an external participant" (BPMN p.161)

Catch Event:

- "catch[es] a trigger" (BPMN p.233)
- "[S]omething that influences processes **beyond the design of their control flow**. In many if not most cases, events occur at a point in time which is **not predictable** within a process" [1, p.32].

Receive tasks are under the control of the agent participating in a perdurant. (The agent knows when it will receive something (?))

Catch events are not under the control of the agent. (The agent does not know when it will be triggered (?))

Send Tasks vs Throw Events

In both cases,

- ▶ They are under the control of the agent participating in a perdurant
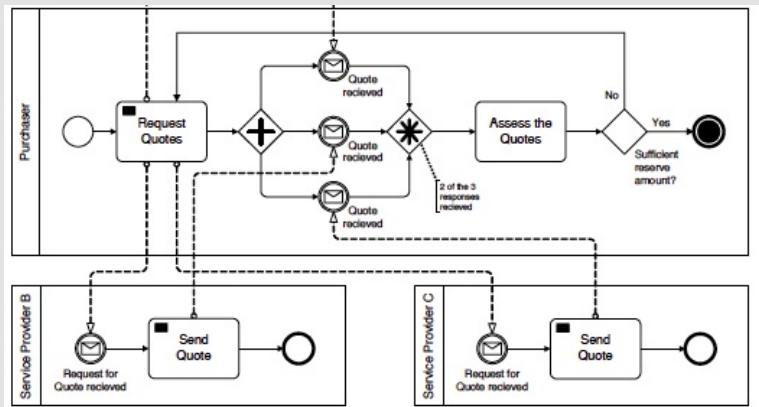
Any relevant distinction?

Rule of thumb to prefer the use of one over the other?

Participants

Participants in perdurants (represented by pools/lanes)

- Participants (e.g., Company, John, etc.)
- Participants' roles (e.g., designer, writer, etc.)

Pool: a specific participant? A type of participant?

References

📄 Kossak et al., A Rigorous Semantics for BPMN 2.0 Process Diagrams, Springer 2016

📄 Fellmann et al., A query-driven approach for checking the semantic correctness of ontology-based process representations, International Conference on Business Information Systems, Springer, 2011

📄 Guizzardi et al., Can BPMN Be Used for Making Simulation Models?, Workshop on Enterprise and Organizational Modeling and Simulation, Springer 2011

📄 Sanfilippo et al., Events and Activities: Is there any Ontology behind BPMN?, FOIS 2014

References

📄 Recker et al., Do process modelling techniques get better? A comparative ontological analysis of BPMN, Australasian Chapter of the Association for Information Systems, 2005

📄 Ferreira, Enterprise Systems Integration: A Process-Oriented Approach, Springer 2013