State-based event modeling

Alessander Botti Benevides[1,2,3] and Claudio Masolo[1]

[1]Laboratory for Applied Ontology, ISTC-CNR, Trento, Italy
[2]Fondazione Bruno Kessler, Trento, Italy [3]ICT Doctoral School, University of Trento, Italy

Abstract. The lack of formal semantics of traditional approaches to business process modeling negatively affects the integration and sharing of models. The approaches that provide a semantics often do not discuss the *ontological foundation* of the assumed primitives hindering the intelligibility of the models. We propose an ontological founded modeling framework in which synchronic properties and relations among *objects* are reified into *states*. States represent the basic blocks on which *events* are built according to unity criteria. We discuss how our framework accommodates different philosophical standpoints on events and we illustrate how it can be used to ground *event calculus*, CLIMB, PSL, and BPMN.

1 Introduction

The ontological analysis of conceptual models in general, and business process models in particular, is still at the early stages. The lack of semantic transparency of the used modeling languages often prevents models developed in different frameworks to be compared and integrated (and understood). The representation of processes involve general notions—*e.g.*, object, state, event, participation—that have been analyzed in philosophy and artificial intelligence (see, for instance, [4] and [8] for the concepts of state and event) but, with few exceptions [20], relatively neglected in BPM formalisms.

This work illustrates a preliminary effort towards the development of an ontologically well founded theory that, trying to accommodate different philosophical standpoints about the nature and structure of events, can be used to ground different BPM formalisms. The proposed framework in based on the reification into *states* of the (partial) descriptions of world's snapshots provided in terms of propositions that involve objects (and their properties/relations). Complex *events* are built by collecting states according to some unity criteria and types of events can be syntactically defined avoiding the "philosophically unsound reification of types" [9]. Events and event-types serve a compact and cognitively-oriented perspective on the world's dynamic. Note that, both the propositions and their reifications are present in our framework. This increases the generality of our theory, an important feature for providing a common foundation to different modeling languages. To keep the framework simple, we consider a discrete and linear time. This hypothesis can be relaxed without compromising the general framework, even though we think that, in practical terms, it is not too limiting. We conclude the paper sketching how our theory could be used to found CLIMB [15], Process Specification Language (PSL) [1] [2], *event calculus* (EC) [16], and Business Process Model and Notation (BPMN) [2] [17].

---

[1] `http://www.mel.nist.gov/psl/ontology.html`.

[2] `http://www.bpmn.org`.

2 The basic framework

We consider 3 disjoint and non-empty *basic categories*: *time* (TM), *object* (OB), and *state of affairs*, or simply *state*, (ST). Everything else will be built starting from these categories. Time is considered here as linear, discrete, and atomic. Atoms are called *times*, they are not composed by means of a mereological relation, and the *precedence* relation defined on them is noted $\leq$. We leave open if times are punctual or extended entities. Objects—also called *substances*, *endurants*, or *continuants*—exist in time and they are wholly present at every time they exist. *E.g.*, tables, persons, bits of stuff. We introduce the primitive of *existence* for objects: $\varepsilon_t x$ stands for "the object x *exists* at time t". Events are usually understood as changes, in particular changes in objects, *i.e.*, intuitively, events occur when objects acquire or loose some properties. Conversely, states are static, nothing changes during them. States correspond to—using the terminology of Kim [11]—*exemplifications* by objects of properties at a time, *i.e.*, a state corresponds to the fact that an object (several objects) has a given property (are in a given relation) at a given time. *E.g.*, sun's being hot in 2013, Luca's being 180cm high now, Luca's being enrolled in the University of Trento in 2010. More specifically, according to Armstrong [1], states are linked to *contingent* properties, *i.e.*, in temporal terms, properties that objects do not necessarily have during their whole life.

More technically, our idea is to *reify into states*, ST-instances, *temporally qualified* atomic propositions (on objects) of the FOL theory under development. In knowledge representation, this technique in not new. It is explicitly addressed in [3], it is used in conceptual modeling, both in ER and UML (see [10]), and it has been taken into account in reified temporal logics as, for instance, situation and event calculi (see [16]).

Following these approaches, let $\mathcal{V}$ be the extra-logical vocabulary of the FOL-theory under consideration—we assume only *closed* formulas in this theory—and $\mathcal{P} \subset \mathcal{V}$ be the set of *temporally contingent* predicates with one argument of type TM while all the other arguments are *existing* entities of type OB (as better represented in (a1)).[3] A univocally temporally qualified predicate $\mathtt{P}$ is temporally contingent if and only if $\mathtt{P}_t x \land \varepsilon_{t'} x \to \mathtt{P}_{t'} x$ does not hold. This excludes *kinds*, *e.g.*, 'being a person' or 'being an electron', as well as ε, from $\mathcal{P}$.[4] In addition we assume $\mathcal{P}$ to be *finite*.

$\mathcal{P}$ does not include *attributes*—to be understood as relations between objects and concrete values, *e.g.*, $\mathtt{HAS\_COLOR}_t(x, \mathtt{red})$. Our framework can be extended to account for attributes, however here, to have a simpler framework, we just consider for each attribute, *e.g.* color, a set of color-properties, *e.g.*, $\mathtt{RED}_t$, $\mathtt{BLUE}_t$, *etc.*. We also exclude from $\mathcal{P}$ diachronic predicates, *e.g.*, $\mathtt{HIGHER}_{tt'} x$ standing for 'x grew from t to t''. Using attributes, $\mathtt{HIGHER}_{tt'} x$ is reducible only to synchronic and atemporal predicates, *e.g.*, $\exists hh'(\mathtt{HEIGHT}_t(h, x) \land \mathtt{HEIGHT}_{t'}(h', x) \land h < h')$. Without attributes the reduction must enumerate all the possible (finite, because $\mathcal{P}$ is finite) combinations of height-properties. Assuming $\mathtt{1M}_t$, $\mathtt{2M}_t$, $\mathtt{3M}_t$ are the only heigh-properties, $\mathtt{HIGHER}_{tt'} x$ can be reduced to $(\mathtt{1M}_t x \land \mathtt{2M}_{t'} x) \lor (\mathtt{1M}_t x \land \mathtt{3M}_{t'} x) \lor (\mathtt{2M}_t x \land \mathtt{3M}_{t'} x)$. It is not clear to us if this reduction holds for all all the diachronic predicates but at least it is viable for *attributes* of objects.

---

[3] We indicate the temporal argument as a subscript. Furthermore, x^n stands for $x_1, \ldots, x_n$, $\mathrm{OB}x^n$ stands for $\mathrm{OB}x_1 \land \cdots \land \mathrm{OB}x_n$, $\varepsilon_t x^n$ stands for $\varepsilon_t x_1 \land \cdots \land \varepsilon_t x_n$.

[4] Existence is trivially rigid: $\varepsilon_t x \land \varepsilon_{t'} x \to \varepsilon_{t'} x$.

Finally note that the requirement on the existence of objects in (a1) is quite problematic for properties like $\mathtt{FAMOUS}_t x$ because, intuitively, somebody can still be, or become, famous after his/her death. At the same time, a sort of social recognition, at t, that what x has done in the past is important seems necessary in order to be famous. These predicates seem to have a double, both historical and actual, nature (see [13] for more details). Leaving this delicate discussion apart, we adopt (a1).

$\mathcal{V}^e$ is the extension of $\mathcal{V}$ by a set $\bar{\mathcal{P}}$ of unary predicates defined on states that are in a 1-1 relation with the predicates in $\mathcal{P}$. We indicate with $\bar{\mathsf{P}} \in \bar{\mathcal{P}}$ the predicate associated with $\mathsf{P} \in \mathcal{P}$. The primitives $\multimap_{\mathsf{i}}$ identify the ith object involved in the state where, by convention, $\multimap_{\mathsf{0}}$ identifies the time, *i.e.*, existence can be extended to states as in (d1).

a1 $\bigwedge_{\mathsf{P}\in\mathcal{P}} \mathsf{P}_t x^n \rightarrow \mathrm{TM}t \wedge \mathrm{OB}x^n \wedge \varepsilon_t x^n$
a2 $\mathrm{ST}s \leftrightarrow \bigvee_{\bar{\mathsf{P}}\in\bar{\mathcal{P}}} \bar{\mathsf{P}}s$
a3 $\bar{\mathsf{P}}s \wedge \bar{\mathsf{P}}s' \wedge \varepsilon_t s \wedge \varepsilon_t s' \bigwedge_{\mathsf{i}\leq\alpha} \forall x(x \multimap_{\mathsf{i}} s \leftrightarrow x \multimap_{\mathsf{i}} s') \rightarrow s = s'$
a4 $\mathsf{P}_t x^n \leftrightarrow \exists s(\bar{\mathsf{P}}s \wedge \varepsilon_t s \wedge x^n \multimap s)$ ($\bar{\mathsf{P}} \in \bar{\mathcal{P}}$ is the predicate associated to P)
a5 $x \multimap_{\mathsf{i}} s \wedge y \multimap_{\mathsf{i}} s \rightarrow x = y$
d1 $\varepsilon_t s \triangleq t \multimap_{\mathsf{0}} s$
d2 $\mathbf{p}_t x^n \triangleq \iota s(\bar{\mathsf{P}}s \wedge \varepsilon_t s \wedge x^n \multimap s)$
d3 $x \multimap s \triangleq x \multimap_{\mathsf{1}} s \vee \cdots \vee x \multimap_{\alpha} s$ (α is the largest arity of the predicates in $\mathcal{P}$)

(a2) guarantees that all states are covered, but not necessarily partitionated, by the predicates in $\bar{\mathcal{P}}$ while (a3)[5] enforces sufficient identity conditions for states. (a4)[6] assures that the reification of $\mathsf{P}_t x^n$ exists only when $\mathsf{P}_t x^n$ holds, *i.e.*, states reify only *true* propositions. (a5) enforces $\multimap_{\mathsf{i}}$ to be injective. (a3) supports the definition (d2)—where ι is a description operator *à la Russell* [7]—*i.e.*, it is possible to introduce a set $\mathcal{D}$ of *descriptions* $\mathbf{p}$ that are in a 1-1 relation with both the predicates $\bar{\mathsf{P}} \in \bar{\mathcal{P}}$ and $\mathsf{P} \in \mathcal{P}$. Finally, (d3) defines a participation relation $\multimap$ in which the ith position is irrelevant.

Our approach is in line with Kim's theory; (a2)-(a4) closely correspond to Kim's *existence condition*— "the state $[x, P, t]$ exists if and only if substance x has property P at time t" [11] (where $[x, P, t]$ corresponds to our $\mathbf{p}_t x$)—even though properties P are here in the domain of quantification while we have a 1-1 meta-relation between $\mathcal{P}$ and $\bar{\mathcal{P}}$. Differently from Kim's *identity condition*—$[x, P, t] = [y, Q, t']$ if and only if $x = y$, $P = Q$, and $t = t'$—our framework allows for $\mathbf{p}_t x^n = \mathbf{q}_t x^n$ with $\mathbf{p}$ different from $\mathbf{q}$. In addition, while *facts* are usually considered to be in a 1-1 correspondence, *up to logical equivalence*, with true propositions, this is not the case of ST-instances. Given two logically equivalent predicates, say $\mathsf{P}_t x^n \leftrightarrow \mathsf{Q}_t x^n$, our theory guarantees $\bar{\mathsf{P}}$-states and $\bar{\mathsf{Q}}$-states to be *mutually existentially dependent* but not identical. More specifically both $\mathbf{p}_t x^n \neq \mathbf{q}_t x^n$ and $\mathbf{p}_t x^n = \mathbf{q}_t x^n$ are consistent statements. We can then partially account for the *intension* of properties. Let us suppose to organize the $\mathcal{P}$-predicates into a taxonomy by considering the isa relation. All the *necessarily* disjoint taxonomical leaves can be safely reified into disjoint classes of states. Consider now the leaves that can have common instances. One can distinguish a purely

---

[5] Where α is the largest arity of the predicates in $\mathcal{P}$.

[6] $x^n \multimap s$ stands for $x_1 \multimap_{\mathsf{1}} s \wedge \cdots \wedge x_n \multimap_{\mathsf{n}} s \wedge \neg\exists y(y \multimap_{\mathsf{n+1}} s \vee \cdots \vee y \multimap_{\alpha} s)$, while $x^n = y^n$ stands for $x_1 = y_1 \wedge \cdots \wedge x_n = y_n$.

[7] $\psi(\iota x(\phi x))$ is equivalent to $\exists x(\phi x \wedge \forall y(\phi y \rightarrow y = x) \wedge \psi(x))$.

extensional overlap from an *intensional* one. For instance, 'being tri-lateral' ($\mathtt{3L}$) and 'being tri-angular' ($\mathtt{3A}$) are a typical example of extensionally coincident but intensionally different properties. Vice versa, 'being red' and 'being orange' are often considered as non-intensionally disjoint because, for instance, vermilion objects are both red and orange.[8] We can represent this difference by imposing the disjointness of $\mathtt{3L}$- and $\mathtt{3A}$-states or the $\mathtt{RED}_t x \wedge \mathtt{ORANGE}_t x \rightarrow \mathbf{red}_t x = \mathbf{orange}_t x$ constraint. Similarly, given $\mathtt{ELECTRON}_t x \rightarrow [\mathtt{9.11{\times}10^{-31}KG}]_t x$ and $\mathtt{SCARLET}_t x \rightarrow \mathtt{RED}_t x$,[9] we can impose that $\mathtt{ELECTRON}$-states are disjoint from $[\mathtt{9.11{\times}10^{-31}KG}]$-states while $\mathtt{SCARLET}$-states are $\mathtt{RED}$-states.[10] These distinctions are not logical, they require an ontological analysis in charge of the user that, however, can now at least partially be represented. Our theory is also compatible with Armstrong's view on states of affairs intended as *truth-makers* of propositions [1], even though we do not commit to the ontological primacy of states with respect to (true) propositions.[11]

States are not reducible to tuples because it is possible to have $\mathbf{p}_t x^n \neq \mathbf{q}_t x^n$ (with $\mathbf{p}, \mathbf{q} \in \mathcal{D}$). Furthermore, states are completely determined, *i.e.*, they reify atomic propositions with no variables; closed formulas with existentially or universally quantified variables do not say anything of the actual configuration of the world, they do not introduce states but only existential constraints on them. For instance, formulas like $\forall t x^n (\mathtt{P}_t x^n \rightarrow \mathtt{Q}_t x^n)$ do not state the existence of any state, they just introduce a dependence of $\mathtt{P}$-states[12] on $\mathtt{Q}$-states. Formulas like $\exists t x^n (\mathtt{P}_t x^n)$—or disjunctions and negations of atomic propositions—introduce a sort of *indeterminism* because they are compatible with different configurations of the world.

Unless explicitly forced, our framework is quite conservative: the isa relation just implies an existential dependence (see (a4)). This holds in general for implications with form $\mathtt{P}_1 \wedge \cdots \wedge \mathtt{P}_n \rightarrow \mathtt{Q}_1 \vee \cdots \vee \mathtt{Q}_n$.[13] The only exceptions are the equivalences with form $\mathtt{P}_1 \wedge \cdots \wedge \mathtt{P}_n \leftrightarrow \mathtt{Q}$ that are taken into account in Section 3.

3 Events

We extend our framework by introducing a new kind of entities called *event* (EV)[14] and an atomic classical extensional mereology (see [5] for the details) defined on them: $x \sqsubseteq y$ stands for 'the event x is part of the event y'. The usual mereological notions are defined in (d4)-(d8). In this theory, events can always be decomposed into atoms in an unique way, see (t1) and (t2). (a6) enforces atoms to be states, *i.e.*, EV subsumes ST and events are sums of states. Thus, an event corresponds to the conjunction of the atomic $\mathcal{P}$-propositions corresponding to its atomic parts (states).

---

[8] We assume here that vermillion is not in $\mathcal{P}$.

[9] The last isa relation is called determinate-determinable and has been deeply studied in philosophy (see [22]).

[10] One could also introduce a *causation* relation defined on states (or, more generally, on events) that does not reduce to material implication.

[11] In Armstrong's theory, properties are in the domain and states of affairs are *composed* by substances and properties. His theory is close to Kim's one but time is very marginally treated.

[12] A P-state is a state s such that $\exists t x^n (s = \mathbf{p}_t x^n)$.

[13] Actually this is the *implicative normal form* every FOL-sentence can be converted in.

[14] Our events correspond to what Galton calls *eventualities* [7].

d4 $x \sqsubset y \triangleq x \sqsubseteq y \wedge x \neq y$ (*x* is a proper part of *y*)
d5 $x \between y \triangleq \exists z(z \sqsubseteq x \wedge z \sqsubseteq y)$ (*x* and *y* overlap)
d6 $x \Sigma y^n \triangleq \forall w(w \between x \leftrightarrow (w \between y_1 \vee \cdots \vee w \between y_n))$ (*x* is the sum of y^ns)
d7 $\Lambda x \triangleq \textsc{ev}x \wedge \neg\exists y(y \sqsubset x)$ (*x* is an atom)
d8 $x \,{}_{\Lambda}{\sqsubseteq}\, y \triangleq \Lambda x \wedge x \sqsubseteq y$ (*x* is an atomic part of *y*)
t1 $x \sqsubseteq y \leftrightarrow \forall z(z \,{}_{\Lambda}{\sqsubseteq}\, x \rightarrow z \,{}_{\Lambda}{\sqsubseteq}\, y)$
t2 $x \Sigma y^n \wedge x' \Sigma y^n \rightarrow x = x'$
a6 $\textsc{st}s \leftrightarrow \Lambda s$

Let us go back to the form $\mathtt{P}_1 \wedge \cdots \wedge \mathtt{P}_n \leftrightarrow \mathtt{Q}$, *i.e.*, $\mathtt{Q}$ is logically reducible to $\mathtt{P}_1, \ldots, \mathtt{P}_n$. By (a2) and (a4) there exist a *state* corresponding to $\mathtt{Q}$ and *n*-states corresponding to the $\mathtt{P}_\mathtt{i}$s. According to the discussion at the end of the Section 2, the difference between the $\mathtt{Q}$-state and the sum of the *n* $\mathtt{P}_\mathtt{i}$-states is justified only in case $\mathtt{Q}$ and $\mathtt{P}_1 \wedge \cdots \wedge \mathtt{P}_n$ *intensionally* differ. Vice versa, $\mathtt{Q}$ is redundant and should be removed from $\mathcal{P}$.

With a slight abuse of notation, (d9) and (d10) extend ε and $\multimap$, respectively, to events. (d11) defines a temporally qualified version of $\multimap$ while (d12) introduces the usual notion of *temporal slice*.

d9 $\varepsilon_t e \triangleq \exists s(s \,{}_{\Lambda}{\sqsubseteq}\, e \wedge \varepsilon_t s)$
d10 $x \multimap e \triangleq \exists s(s \,{}_{\Lambda}{\sqsubseteq}\, e \wedge x \multimap s)$
d11 $x \multimap_t e \triangleq \exists s(s \,{}_{\Lambda}{\sqsubseteq}\, e \wedge x \multimap s \wedge \varepsilon_t s)$
d12 $x \Vdash_t y \triangleq \forall z(z \sqsubseteq x \leftrightarrow \varepsilon_t z \wedge z \sqsubseteq y)$

From a *structuralist* perspective, states can be seen as ‘sensory atoms’, as ‘temporal dependent data’, as ‘pointlike observations’ on which complex entities can be built. In a movie metaphor, states represent *dynamic* factual knowledge, *i.e.*, factual knowledge that concerns a single snapshot, that is ‘acquired’ at a given time. Vice versa, *static* factual knowledge, once acquired, does not need to be re-considered. For instance, one checks (at a given time) for the existence or some contingent properties of a person but not for her personhood. *Terminological* knowledge can concern static-predicates, *e.g.*, $\mathtt{PERSON}x \rightarrow \mathtt{MORTAL}x$ as well as dynamic ones. Both *synchronic*- and *diachronic*-terminological knowledge, *e.g.*, $\mathtt{SCARLET}_t x \rightarrow \mathtt{RED}_t x$ and $\mathtt{2M}_t x \wedge \varepsilon_{t'} x \rightarrow (\mathtt{2M}_{t'} x \vee \mathtt{3M}_{t'} x)$, introduce existential dependences among states that can be further specialized by means of static-predicates, *e.g.*, $\mathtt{FERRARI}x \wedge \mathtt{RED}_t x \wedge \varepsilon_{t'} x \rightarrow \mathtt{RED}_{t'} x$.

In our movie metaphor, a movie—a *narrative*—can be seen as a sequence of snapshots (described in terms of our vocabulary) containing states that satisfy the existential dependences introduced by the laws. Events offer an abstract and dynamically-oriented point of view on narratives. Perception organizes stimuli by grouping them in unitary objects that allow us to interact with the world in a quick and fruitful way (see [18] for an introduction). Similarly, states can be organized by, synchronically or diachronically, grouping them into events, entities that have a cognitive and/or practical relevance for understanding the dynamic of the world. Furthermore, *types* of events can be used to represent the laws that regulate the world in a cognitive-friendly fashion.

It would be noted that nothing prevents the user to introduce predicates like $\mathtt{STAB}_t xy$ in $\mathcal{P}$. Differently from the usual conceptualisation, $\mathbf{stab}_{44\mathrm{bc}}(\mathtt{brutus}, \mathtt{caesar})$ is here an atomic state. This counter-intuitive classification can be explained in terms of *granularity*. In temporal terms, $\mathtt{STAB}$ cannot be further analyzed. The user decided to consider stabbing-events as atomic, *i.e.*, as 'observable' in a single snapshot, but no changes can be observed in a snapshot. This does not contradict the foundations of our framework. To intuitively explain this fact, one can think that the user considered a coarse temporal granularity, *i.e.*, a time corresponds to an interval in the actual movie. The FOL-theory can then be seen as an (abstract) annotation of what happens during an interval. Assume now that $\mathtt{STAB\_WITH}_t xyz$ is also in $\mathcal{P}$. $\mathbf{stab\_with}_{\mathtt{44bc}}(\mathtt{brutus}, \mathtt{caesar}, \mathtt{k\#1})$ and $\mathbf{stab}_{\mathtt{44bc}}(\mathtt{brutus}, \mathtt{caesar})$ can be dependent but they differ because they have different participants. We loose the most relevant aspect of the approach of Davidson [6], *i.e.*, there is only one event, the stabbing, that is performed with a knife. By assuming that STAB_WITH is definable as, for instance, $\mathtt{STAB\_WITH}_t xyz \triangleq \mathtt{STAB}_t xy \wedge \mathtt{USE}_t xz$, STAB_WITH-states could be considered as sums of STAB- and USE-states.[15]

3.1 Changes

Changes *in objects* and changes in general are often distinguished, compare (OC) and (GC) below taken from [12]. (OC) commits to the survival of the object x: if something does not exist it cannot lack a property ([12], p.82).[16] (GC) commits to events (entities that occur), it does not refer to objects, and even assuming that all the proposition S concern only objects, still (GC) does not commit to the *persistence* of any object.

(**OC**) An object changes if and only if
1. there is a property, P,
2. there is an object, x,
3. there are distinct times, t and t', and
4. x has P at t and fails to have P at t'.

(**GC**) A change occurs if and only if
1. there are distinct times, t and t',
2. there is a proposition, S, and
3. S is true at t and false at t'.

In our framework, (d13) and (d14) simulate, respectively, (GC) and the event version of (OC). Note that (d14) just adds $\varepsilon_{t'} x$ to (d13), in both cases x could not participate to e at t'. One could strength (d14) as in (d15), however the clause $x \multimap_{t'} e$ is problematic because it requires the existence of a state $\mathbf{q}_{t'} x$ with $\mathbf{q} \in \mathcal{D}$ (and different from $\mathbf{p}$) but nothing guarantees that $\mathtt{P}_t x \wedge \neg \mathtt{P}_{t'} x \wedge \varepsilon_{t'} x \rightarrow \bigvee_{\mathtt{Q} \in \mathcal{P}} \mathtt{Q}_{t'} x$ (more generally, nothing guarantees that $\varepsilon_t x \rightarrow \bigvee_{\mathtt{Q} \in \mathcal{P}} \mathtt{Q}_t x$). Lombard addresses this problem through the notion of *quality space*—a set S of mutually exclusive (non-relational) properties such that if $\mathtt{P}^{\mathtt{i}}_t x$, with $\mathtt{P}^{\mathtt{i}} \in S$, at every time t' at which the object x exists there is a $\mathtt{P}^{\mathtt{j}} \in S$ such that $\mathtt{P}^{\mathtt{j}}_{t'} x$—and assuming that (basic) changes are *movements* of objects through quality spaces. Assume $\mathcal{P}$ is partitioned in n-quality spaces that induce a partition $\mathcal{S}_1, \ldots, \mathcal{S}_n$ in $\mathcal{D}$. Basic changes can then be defined as in (d16) (where $\mathbf{p}$ is different from $\mathbf{q}$).

d13 $\mathtt{CNG}e \triangleq \bigvee_{\mathbf{p} \in \mathcal{D}} \exists x t t' (\mathbf{p}_t x \sqsubseteq e \wedge \neg \varepsilon_{t'} (\mathbf{p}_{t'} x) \wedge \varepsilon_{t'} e)$

d14 $\mathtt{oCNG}e \triangleq \bigvee_{\mathbf{p} \in \mathcal{D}} \exists x t t' (\mathbf{p}_t x \sqsubseteq e \wedge \neg \varepsilon_{t'} (\mathbf{p}_{t'} x) \wedge \varepsilon_{t'} e \wedge \varepsilon_{t'} x)$

d15 $\mathtt{sCNG}e \triangleq \bigvee_{\mathbf{p} \in \mathcal{D}} \exists x t t' (\mathbf{p}_t x \Vdash_t e \wedge \neg \varepsilon_{t'} (\mathbf{p}_{t'} x) \wedge x \multimap_{t'} e \wedge \neg \exists y (y \neq x \wedge y \multimap_{t'} e))$

d16 $\mathtt{bCNG}e \triangleq \bigvee_{\mathbf{p},\mathbf{q} \in \mathcal{S}_i} \exists x t t' (\mathbf{p}_t x \Vdash_t e \wedge \mathbf{q}_{t'} x \Vdash_{t'} e)$ (for some $\mathcal{S}_i$)

d17 $\mathtt{gCNG}e \triangleq \bigvee_{\mathbf{p} \in \mathcal{D}} \exists x^n t t' (\mathbf{p}_t x^n \sqsubseteq e \wedge \neg \varepsilon_{t'} (\mathbf{p}_{t'} x^n) \wedge \varepsilon_{t'} e \wedge \varepsilon_{t'} x^n)$

---

[15] However the 'using' and the 'stabbing' must be linked, x could do simultaneous actions.

[16] As observed by Kim, "[w]hether coming into being and passing away can be construed as changes in substances" ([11], p.310) is a question to be addressed.

(d14) (similarly for (d13)) can be generalized by allowing propositions that involve several objects (d17). This generalization matches (GC) because there is no commitment to the number (and nature) of entities involved in the proposition S. Still $\mathsf{P}_t x^n \wedge \neg\mathsf{P}_{t'} x^n$ is an evidence of a change in the world delimitating this change to the objects x^n, *i.e.*, it points out the part of the world involved in the change. However, $\mathsf{P}_t x^n \wedge \neg\mathsf{P}_{t'} x^n \wedge \varepsilon_{t'} x^n$ does not say what objects change. For reason of space we cannot discuss in more details this problem and the one (discussed in [12]) related to the cases where a parthood relation between objects is included in $\mathcal{P}$. Here we assume a liberal approach that accepts also 'events' where no object changes (*e.g.*, homeomeric events like the sum of $\mathbf{p}_t x$ and $\mathbf{p}_{t'} x$) and leaves to the user the possibility to filter the sums of states according to her needs and the primitives of the FOL-theory under construction.

Finally, Galton [7] defines *instantaneous transitions* as transitions from a state that holds at t to a state of different kind that holds at t' (the successor of t), *e.g.*, the transition from $\mathbf{red}_t x$ to $\mathbf{blue}_{t'} x$. He claims that this transition does not occur at any time, it occurs *between* times, between t and t' in the example. Consequently he includes these 'interfaces' among the temporal entities. From a cognitive perspective, to be observed, a transition requires the observation of two distinct states. According to (GC), we tend then to see these transitions as non-instantaneous, *i.e.*, as (specific) changes.

4 Comparison and discussion

We compare our framework with four approaches developed for representing and reasoning on events and processes: the *event calculus* (EC) [16], Computational Logic for the verIfication and Modeling of Business constraints (CLIMB) [15], the PSL [2], and BPMN [17]. Given the limited space and the preliminary nature of this work, we cannot fully introduce these approaches nor provide a complete formal comparison. We focus on some differences and similarities relevant from the ontological and representational perspective outlining some strategies one could follow for a full comparison.

The EC considers three sorts of entities: *event*, *fluent*, and *timepoint*.[17] Events occur in the world at times, $Happens(e, t)$, while fluents are time-varying propositions that hold, are true, at times, $HoldsAt(f, t)$. Both events and fluents are terms individuated by total functions. The user decides which functions identify events, *e.g.* $\mathbf{stab}(x, y)$, and which functions identify fluents, *e.g.*, $\mathbf{on}(x, y)$. While fluents can exist and hold at different times, our states exist at single times. Consider the situation where x is on y both at t_1 and t_2. In the EC there is a single fluent, $\mathbf{on}(x, y)$, that holds both at t_1 and t_2 while, in our framework, there is a complex event composed by two distinct states: $\mathbf{on}_{t_1}(x, y)$ and $\mathbf{on}_{t_2}(x, y)$. In our framework, a fluent could be defined as the sum of all the states of the same type (identified by the same description) that involve the same objects (in the same order), a notion quite similar to the one of homeomerous-perdurants in the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) [14].[18] However, note that in EC the representation of fluents through total functions $\mathbf{f}$ forces

---

[17] Differently from *situation calculus* [21], in EC time is usually considered as linear even though a branching version of EC exists.

[18] We would need to introduce *infinite* sums, see [5].

the existence of $\mathbf{f}x^n$ (n is the arity of $\mathbf{f}$) whatever x^n one considers. The fluent $\mathbf{f}x^n$ exists in the domain of quantification even though it never holds. Fluents have then a propositional nature.[19] Vive versa, adopting the proposed reduction of fluents to sums of states, fluents necessarily hold, they are sums of exemplifications of properties by objects at a time. To avoid total functions and propositions, we pay the cost of introducing definite descriptions and $\multimap_{\mathsf{i}}$ primitives in our framework. As said, we see changes and actions as intrinsically non atomic; it is not possible to observe changes or actions in a single snapshot. The situation is different for EV-instances that persist in time. As done for fluents, we can characterize a specific notion of event—disjoint from the one of fluent—*e.g.*, the one of change discussed in Section 3.1. $\mathit{Happens}(x, t)$ and $\mathit{HoldsAt}(x, t)$ collapse then to $\varepsilon_t x$ while all the EC's primitives—$\mathit{Initiates}(e, f, t)$, $\mathit{Terminates}(e, f, t)$, $\mathit{Releases}(e, f, t)$, $\mathit{ReleasedAt}(f, t)$—are now defined on sums of states. Finally, while EC embrace the unique name assumption—*i.e.*, given two different descriptions $\mathbf{p}$ and $\mathbf{q}$, necessarily $\mathbf{p}x^n \neq \mathbf{q}y^m$—we can force $\mathbf{p}_t x^n = \mathbf{q}_t x^n$ as in the case of scarlet- and red-states. This difference reminds us the one between *identity* and *coincidence* [19].

Similarly to our approach, CLIMB does not make a distinction between events and fluents. In addition to times, only *events*—represented as terms (usually identified by functions)—are present. Both $\mathit{send}(x, y, \mathit{msg})$ and $\mathit{status}(\mathit{cable}, \mathit{off})$ are reported as examples of events in [15].[20] Similarly to EC, in CLIMB, the same event can happen at different times; $\mathbf{H}(E, T)$ stands for "event E happens at timepoint T", where E and T are terms. A big difference concerns the fact that free variables are admitted in E and T. In these cases events are *sets* of event-tokens (*traces*) and $\mathbf{H}(E, T)$ means that some of these event-tokens occur at one time in T. The variables can also be constrained in scope; for instance one could consider only the messages sent by a given group of persons. We prefer to clearly distinguish event-tokens from event-types. This is why we excluded free variables from state-terms, associating to existentially and universally quantified $\mathcal{P}$-formulas only existential constraints on states. Thus, we can reduce $\mathbf{H}(E, T)$ to $\varepsilon_T E$ only in absence of free variables in E and T. In the other cases, $\mathbf{H}(E, T)$ corresponds to the existential constraint $\exists et(\mathtt{E}_t e \wedge \mathtt{T}t)$ where $\mathtt{T}$ is a temporal constraint and $\mathtt{E}$ has the form $\mathtt{E}_t e \triangleq \exists x^n(e = \mathbf{p}_t x^n \wedge \mathtt{C}x^n)$, where $\mathbf{p} \in \mathcal{D}$ is a given description (*i.e.*, all the $\mathtt{E}$-instances are linked to an unique description $\mathbf{p}$) and $\mathtt{C}$ represents the constraints on the objects x^n. The previous reduction can be generalized to allow $\mathbf{H}$ to apply also to complex events. *Integrity constraints* are central in CLIMB. Roughly speaking, integrity constraints represent the expected 'outcomes' of some events that happened at given times, *i.e.*, the *possible* events that would satisfy the system requirements. They are represented by rules $\mathit{Body} \rightarrow \mathit{Head}$ where, in addition to temporal constraints, the body is a conjunction of $\mathbf{H}(E, T)$ clauses and the head is a disjunction of conjunctions of $\mathbf{E}(E, T)$ and $\mathbf{EN}(E, T)$ clauses, where $\mathbf{E}(E, T)$ ($\mathbf{EN}(E, T)$) represents the positive (negative) *expectation* that E happens at T. Since integrity constraints are seen as requirements, they can be fulfilled or violated. Given a specific sequence s of (actual) events, CLIMB is able to check if s satisfies (is compliant with) the integrity constraints. Our framework (as well as the EC) does not contemplate *possible* evolutions of the world, it is narrative-based, *hypothetical* states are not represented. Our synchronic

---

[19] This maybe explains why EC uses the predicate *HoldsAt* instead of existence in time.

[20] One could discuss what is the ontological nature of *off*.

and diachronic (existential) constraints among (types of) states are not requirements; like *natural laws*, they necessary hold, they cannot be violated, they represent how the 'world works'. In our framework, integrity constraints can be (partially) represented by introducing *types* of events that collect all the non-compliant sequences of states. *E.g.*, given the requirement ($\mathtt{t}$ and $\mathtt{t}'$ are constants) $\mathbf{H}(E, \mathtt{t}) \rightarrow \mathbf{E}(E', \mathtt{t}')$, one can introduce $\mathtt{R}_1 e \triangleq \exists s(s \sqsubseteq e \wedge \mathtt{E}_{\mathtt{t}} s \wedge \varepsilon_{\mathtt{t}'} e \wedge \neg\exists s'(\mathtt{E}'_{\mathtt{t}'} s'))$, where $\mathtt{E}$ and $\mathtt{E}'$ are defined as before, while for $\mathbf{H}(E, \mathtt{t}) \rightarrow \mathbf{EN}(E', \mathtt{t}')$ one can introduce $\mathtt{R}_2 e \triangleq \exists s s'(\mathtt{E}_{\mathtt{t}} s \wedge \mathtt{E}'_{\mathtt{t}'} s' \wedge s \sqsubseteq e \wedge s' \sqsubseteq e)$.[21] The compliance-check becomes then a sort of *classification* problem, we need to check that there are no $\mathtt{R}_1$- or $\mathtt{R}_2$-instances.

PSL theory encompasses a core theory (PSL-Core) and a number of extensions. PSL-Core considers four kinds of entities: *activity*, *activity occurrence*, *timepoint* and *object*. Every activity occurrence is an *occurrence of*—a primitive of PSL—a unique activity and has a begin and an end timepoint. For instance, the activity (*paint House#1 Paintcan#1*) can have different occurrences: the *House#1* (a specific object) can be (partially) paint several times, during disjoint time intervals, using the same *Paintcan#1*. Timepoints form a discrete infinite linear ordering with endpoints at infinity, while "[a]n object is anything that is not a timepoint, nor an activity nor an activity-occurrence". An object can participate at a timepoint in an activity occurrence (*participates_in* is a ternary primitive of PSL) only when the object exists and the activity is occurring.[22]

In our framework, an occurrence corresponds to a (non atomic) event[23] while activities can be introduced as maximally specified event-types, *i.e.*, leafs of the taxonomy of (non-atomic) events. Classes of activities can then be reduced to isa-generalizations of the leafs in the taxonomy of events. The PSL-primitive *occurrence_of* becomes instantiation while *participates_in* can be mapped to $\multimap_t$. Note that occurrences are always occurrences of an *unique* activity. We can introduce this constraint by assuming that the leafs associated to activities partitionate the domain of (non atomic) events.

From a general perspective, we can see BPMN-*models* as definitions of (our) event-types. The core of BPMN provides a set of modeling constructs to specify how a process (an event in our terminology) is structured in sub-activities. *Activities*, *events*, and *gateways* (called Flow Objects) are used to specify this structure. Activities seem to correspond to event-types—*i.e.*, the whole process and its (sub-)activities have the same ontological nature—while gateways, as well as *sequence flows*, introduce temporal constraints on (sub-)activities and add some indeterminism that, as said, can be represented

---

[21] Note that $\mathtt{R}_1$ and $\mathtt{R}_2$ are quite close to changes as discussed on Section 3.1.

[22] It is not clear whether and how the constants in the activity-term (*paint House#1 Paintcan#1*), *i.e.*, *House#1* and *Paintcan#1*, are linked by *participates_in* to the occurrences of the activity. In addition, even though there are no axioms that guarantee that all the occurrences of an activity have the same participants, the examples reported in the PSL documentation consider activity terms with specific objects (constants) and no free variables.

[23] In PSL, activities are entities that *can* have occurrences happening at different intervals of time. The Outer Core extension called 'Theory of Occurrence Trees' defines *occurrence tree* as a poset representing all *possible* sequences of occurrences of all activities. It is not clear to us whether activity occurrences are *actual* or *possible* individuals (whether there is prescriptive or descriptive perspective on occurrences). In our framework we consider only actual events and we assume a descriptive attitude. Prescriptive laws can be enforced by means of axioms on types of events.

by existential abstractions in our framework. BPMN-events seem to correspond to very general types of states, however types of events cannot be introduced in BPMN preventing, for instance, the representation of pre- or post-conditions of activities.

Pools are intended to capture the notion of participant—*the* participant of a given sub-process. *Message Flows* characterize the interchange of messages between participants. The exchanges of messages can be seen as (sending/receiving) events that involve a document, a *data object*, or simply as a synchronization mechanism across pools that can be reduced to some temporal constraints in our framework.

The lack of semantics of BPMN prevents us to provide a *safe* semantics in terms of our framework. Given the preliminary nature of this work, to stress our own view on BPMN-constructs seems not appropriate.

References

1. Armstrong, D.M.: A World of States of Affairs. Cambridge University Press (1997)
2. Bock, C., Grüninger, M.: PSL: A semantic domain for flow models. Software and Systems Modeling 4(2), 209–231 (2005)
3. Brachman, R.J., Levesque, H.J.: Knowledge representation and reasoning. Elsevier (2004)
4. Casati, R., Varzi, A.: Events. Dartmund, Aldershot (1996)
5. Casati, R., Varzi, A.: Parts and Places. MIT Press, Cambridge, MA (1999)
6. Davidson, D.: Essays on Actions and Events. Clarendon Press, Oxford (1980)
7. Galton, A.: Eventualities. In: Fisher, M., Gabbay, D., L., V. (eds.) Handbook of Temporal Reasoning in Artificial Intelligence, pp. 25–58. Elsevier (2005)
8. Galton, A.: Operators vs. arguments: the ins and outs of reification. Synthese 150, 415–441 (2006)
9. Galton, A.: Reified temporal theories and how to unreify them. In: International Joint Conference on Artificial Intelligence (IJCAI91). pp. 1177–1182 (1991)
10. Halpin, T., Morgan, T.: Information Modeling and Relational Databases. Morgan Kaufmann, second edn. (2008)
11. Kim, J.: Events as property exemplifications. In: Brand, M., Walton, D. (eds.) Action Theory, pp. 159–177. Reidel, Dordrecht (1976)
12. Lombard, L.B.: Events. A Metaphysical Study. Routledge and Keegan (1986)
13. Masolo, C., Vieu, L., Kitamura, Y., Kozaki, K., Mizoguchi, R.: The counting problem in the light of role kinds. In: Davis, E., Doherty, P., Erdem, E. (eds.) AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning. AAAI Press (2011)
14. Masolo, C., Borgo, S., Gangemi, A., Guarino, N., Oltramari, A.: Wonderweb deliverable d18. Tech. rep., CNR (2003)
15. Montali, M.: Specification and Verification of Declarative Open Interaction Models. A Logic-Based Approach. Lecture Notes in Business Information Processing, Springer (2010)
16. Mueller, E.T.: Commonsense Reasoning. Morgan Kaufmann (2006)
17. OMG: Business Process Model and Notation (BPMN) version 2.0. Tech. Rep. formal/2011-01-03, Object Management Group (January 2011)
18. Palmer, S.E.: Vision Science : Photons to Phenomenology. MIT Press (1999)
19. Rea, M. (ed.): Material Constitution. Rowman and Littlefield Publishers (1996)
20. Recker, J.C., Indulska, M., Rosemann, M., Green, P.F.: How good is BPMN really? Insights from theory and practice. In: Ljungberg, J., Andersson, M. (eds.) Proceedings of the Fourteenth European Conference on Information Systems, ECIS 2006. pp. 1582–1593 (2006)
21. Reiter, R.: Knowledge In Action. MIT Press, Cambridge, Massachussetts (2001)
22. Sanford, D.H.: Determinates vs. determinables. In: Zalta, E.N. (ed.) The Stanford Encyclopedia of Philosophy (Spring 2013 Edition)