Bridging Representations of Laws, of Implementations and of Behaviours

Sileno, G.; Boer, A.; van Engers, T.

DOI
10.3233/978-1-61499-609-5-169

Publication date
2015

Document Version
Accepted author manuscript

Published in
Legal knowledge and information systems

Citation for published version (APA):

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.
Introduction

While law is often expressed in general terms, addressing classes of persons and abstract normative notions, courts and public administrations are required to translate those provisions in actual, contextualized behavioural terms. To facilitate this contextualization process, one of the objectives of our current research (cf. [3,2]) is to support alignment between representations of legal-institutional mechanisms, representations of implementations in business processes in administrative organizations, and representations of social behaviour as in case law. Pursuant to our work in e.g. [3,2], our present objective is to work on an operational definition of subsumption – covering instantiation, generalization, and, to some degree, nonmonotonic abstraction – for positional models belonging to different views, to aid the organization of these models in a component library.

The core problem – of ontological, epistemic and normative frictions (cf. Schlag [8] on arguments in courts) – is not specific to law; The semi-formal process of law eases, and is plausibly meant to reduce, such frictions. This contextualization process exists in any agency (individual or organization). From an internal perspective to agency, social norms provide reasons for the agent to promote or demote certain action-selections, or even create the possibility of certain action-selections (via institutional power). In contrast, behavioural representations that take an external perspective to agency have a focus on observable interactions between agent identities, without relying on internal mechanisms. Internal and external perspectives are separated by degree of granularity, and by making reference or not to the mechanisms which are supposed to produce the events. Similar dichotomies can be found in internal and external views in modeling, cf. UML; or in software engineering, with the notions of orchestration and choreography. The two perspectives can be seen opposite poles in a representational spectrum to describe, explain and prescribe behaviour.

1 Corresponding author: g.sileno@uva.nl.
they have behaved they usually behave they should behave
how occurrence description pattern description normative definition
why occurrence explanation behavioural mechanism norm-creating mechanism

Table 1. What agents may tell with narratives: a simple taxonomy. They targets both self and/or others.

1. Narratives and Positional modeling

Narratives – cases in law – are the most expressive medium humans use to share knowledge about how and why they have behaved in a certain way, or usually behave, or should behave or have behaved. Table 1 shows a simple taxonomy that summarizes the possible contents that may be acquired from narrative, revealing the presence of ontological stratification [1, p. 38], with at least six strata whose positions we need to align. A position in our terms is a local state of the system that can be (locally) related to other positions in dimensional terms. For instance, in a classic logic system, each proposition can be put in relation at least with its negation by definition, but analytic philosophy (internal vs external negation) and logic programming (classic vs default negation/negation as failure) recognize a practical multi-dimensional problem. We recently presented two contributions about the use of positions in institutional and intentional modeling. First, we remapped Hohfeld’s obligatory and potestative squares into the two Hohfeldian prisms in [10], discovering the neglected positions of negative power and negative liability, in an attempt to express the motives of the prevent-acquire-cure-keep (PACK) framework [6] in Hohfeld’s terms. In [9], we explored the different pragmatic meanings of agentic negative counterparts of commitment, affordance, expectation and susceptibility.

For evaluation of our work on subsumption we considered a sale transaction from different views, modeled in an extension of Petri Nets\(^2\). The first, simplest representation is that of a simple description of occurrence (Fig. 1). This corresponds in practice to an event log. Events are represented with grounded predicates, and the sequence is captured by the structure of the net. Note that the representation does not (globally) reify time. In the second we generalize the model of sale (based on the Dutch civil code) to a general pattern (Fig. 2). In the third we progress to the normative definition of a sale as defined in the Dutch civil code as an agreement between two parties, each promising the other to perform a certain action (Fig. 3). In the fourth, the agent script, institutional positions play the role of reasons weighted before taking a decision, with economic ones, heuristics, etc.

2. Subsumption for positional models expressed as Petri Nets

How to recognize whether, given two models of the types shown above, they are referencing to the same component? In theoretical terms, the transformations of physical or abstract entities preserving (part of) the original structure are called morphisms: homomorphism, isomorphism, and intermediate notions between homomorphism and isomorphism, amongst which bisimilarity [7]. A complementary approach is log-based analysis, highly tolerant of incomplete knowledge and visibility on the environment, e.g. re-

\(^2\)Logic Programming Petri Nets, see https://github.com/s1l3n0/lppneu
Figure 1. Example of occurrence: a sale instance.

Figure 2. Example of pattern: a generic sale.

Figure 3. Example of normative specification: a sale transaction.

**play fitness** [4] quantifies the extent to which the model reproduces logs. Recent works compute fitness in linear time (e.g. [5]), based on a hierarchy of single-entry-single-exit (SESE) components. This is particularly relevant for a library of components.

For reasons of space, the hybrid approach we propose is presented briefly and informally. Subsumption is checked at different levels. In preparation, we ground both models, i.e. we compute all possible execution paths, obtaining two sets of stories specific $\Sigma_S$ and general $\Sigma_G$. Specific stories should be subsumed by at least one general story:

$$\forall \text{story}_S \in \Sigma_S, \exists \text{story}_G \in \Sigma_G/\text{subsumes}(\text{story}_G; \text{story}_S)$$

Iterating over the steps in the general story, the algorithm looks first to settle a state in the specific story which is subsumed by the current general state. Then it goes further to find a specific event which is subsumed by the general one. If it misses one of the two, it returns false. At this point, to check whether a state subsumes the other, we have to check if each local state (i.e. place/marking coupling) of the general state subsumes a **distinct** local state in the specific state.
procedure SUBSUMES(general story, specific story)
for general step in general story do
  found = false; pos = 0
for specific step after pos in specific story do
  gap = 0
  if subsumes(general step.state, specific step.state) then
    for step after pos + gap in the specific story do
      if subsumes(generic step.event, specific step.event) then
        pos = current pos in specific story
        found = true; break
      end if
    end for
    if found then break
  else gap++
end if
if not found then return false
else return true
end procedure

3. Discussion and conclusion

Models and results running the subsumption algorithm appear satisfactory so far, and are available to the reader online at http://justinian.leibnizcenter.org/lnpetri. The evaluation models have been constructed using Yasper\(^3\), as basic Petri nets, and then mapped to our notation. Ongoing efforts are towards strengthening the connection between institutional and intentional positions (should behave versus why), i.e. the module concerning pli ance, cf. [10], which is not yet fully integrated at the moment.

References


\(^3\)http://www.yasper.org/