2008

# Declarative Process Modeling with Business Vocabulary and Business Rules

Stijn Goedertier
*Katholieke Universiteit Leuven, Belgium*

Jan Vanthienen
*Katholieke Universiteit Leuven, Belgium*

Follow this and additional works at: http://aisel.aisnet.org/eis2008

# Declarative Process Modeling with Business Vocabulary and Business Rules

Stijn Goedertier and Jan Vanthienen

Katholieke Universiteit Leuven, Belgium

**Abstract.** In the literature, there exist already many languages for declarative process modeling. Each language addresses only one specific business concern. In our work, we define a unified framework for declarative process modeling, consisting of a unified vocabulary, execution model, and business rule types [1]. It can be used both as an expressive informal language for documenting business concerns, and as an ontological foundation to compare and develop declarative languages.

## 1  Introduction

In general, one can think of business concerns such as regulations, time constraints, resource constraints, and information prerequisites, to play a governing role in the organization of work. Organizations often only implicitly think about these business concerns and immediately determine task control flows, information flows and work allocation schemes. In other words, the business concerns remain implicit but their effects are *hard-coded* directly in **procedural languages** such as the Business Process Modeling Notation (BPMN).

**Declarative** process models, in contrast, explicitly take into account the business concerns that govern business processes and leave as much freedom as is permissible at execution time for determining a valid and suitable execution scenario. In the literature, there exist already many declarative process modeling languages such as the case handling paradigm [2], the constraint specification framework [3], the Web Service Modeling Ontology (WSMO) [4], and the ConDec language [5]. Each language addresses only one specific business concern.

## 2  The EM-BrA$^2$CE Framework

We take a different approach to declarative process modeling. Rather than defining another language, we define the EM-BrA$^2$CE Framework, a unified framework, within which the aforementioned languages can be positioned. EM-BrA$^2$CE stands for 'Enterprise Modeling using Business Rules, Agents, Activities, Concepts and Events'. It consists of a formal, unified vocabulary, and execution model. The differences between declarative process languages can in part be brought back to a different perception of *state space*, *transition types*, and *transition constraints*. The remainder of this text briefly discusses the unified state space, transition types, and transition constraints within the framework.

Within the framework processes are 'activities' that can consisting of other activities. At any particular moment, an activity is in a particular state. In particular, its **state** is composed of facts (propositions) about activities, agents, business concepts, and events. **Activities** (or services) are individual concepts that represent a unit of (coordination) work to be performed by an agent. **Agents** (or service providers) are individual concepts that represent an actor or a group of actors who can perform activities. **Business concepts** are individual concepts of which the facts can be manipulated in the context of performing activities. **Events** are individual concepts that correspond to an instantaneous, discrete state change of a concept in the world. The EM-BrA$^2$CE Vocabulary provides a detailed definition of these concepts, their concept types, and fact types. It is defined in terms of the SBVR specification [6].

The unified **execution model** considers a business process instance to be a *trajectory* in a *state space* that consists of the possible sub-activities, events and business concepts that occur in the business process. Each activity in a process instance can undergo a number of distinct **state transitions**. The occurrence of a state transition is logged as an activity event. **Business rules** determine whether or not a particular state transition can occur. In our work a total of twelve **state transition types** have been identified and a generic execution model is defined in terms of Colored Petri Nets (CPN) [1]. The formal CPN execution model allows defining declarative simulation models.

## 3  Conclusion

In our work, we define a unified framework for declarative process modeling, consisting of a unified vocabulary, execution model, and business rule types [1]. We have validated the framework by constructing declarative process simulations models, and by using it as a language spec for declarative process mining [7].

## References

1. Goedertier, S., Vanthienen, J.: Declarative process modeling with business vocabulary and business rules. Volume 4805 of LNCS., Springer (2007) 603–612
2. van der Aalst, W.M.P., Weske, M., Grünbauer, D.: Case handling: a new paradigm for business process support. Data & Knowledge Engineering **53**(2) (2005) 129–162
3. Sadiq, S.W., Orlowska, M.E., Sadiq, W.: Specification and validation of process constraints for flexible workflows. Information Systems **30**(5) (2005) 349–378
4. Roman, D., Keller, U., Lausen, H., de Bruijn, J., Lara, R., Stollberg, M., Polleres, A., Feier, C., Bussler, C., Fensel, D.: Web service modeling ontology. Applied Ontology **1**(1) (2005) 77–106
5. Pesic, M., van der Aalst, W.M.P.: A declarative approach for flexible business processes management. Volume 4103 of LNCS, Springer (2006) 169–180
6. Object Management Group: Semantics of Business Vocabulary and Business Rules (SBVR), v1.0. OMG Document – dtc/08-01-02 (2008)
7. Goedertier, S., Martens, D., Baesens, B., Haesen, R., and Vanthienen, J. Process Mining as First-Order Classification Learning on Logs with Negative Events. Volume 4928 of LNCS. Springer (2008).