Validating Process Refinement with Ontologies

Yuan Ren[1], Gerd Groener[2], Jens Lemcke[3], Tirdad Rahmani[3], Andreas Friesen[3], Yuting Zhao[1], Jeff Z. Pan[1] and Steffen Staab[2]

[1]University of Aberdeen, [2]University of Koblenz-Landau, [3]SAP AG

Abstract. A crucial task in process management is the validation of process refinements. In this work we define process refinement based on the execution set semantics, represent process models with ontologies and validate refinements by reasoning.

1 Introduction

As a widely adopted graphical process syntax in industry, we use the business process modelling notation (BPMN) to describe process. Fig. 1a is an example of a BPMN diagram consisting of two activities, and Fig. 1b is a refinement of it, where (⊗) and (⊕) are exclusive and paralle gateways, respectively. In Fig. 1b the originators of a_1, a_2, a_3 and b are A, A, A and B, respectively. The execution sets [6] of Fig. 1a and Fig.1b are $\{[\mathsf{AB}]\}$ and $\{[\mathsf{a}_1\mathsf{a}_2\mathsf{b}], [\mathsf{a}_1\mathsf{b}\mathsf{a}_2], [\mathsf{a}_1\mathsf{a}_3]\}$, respectively.

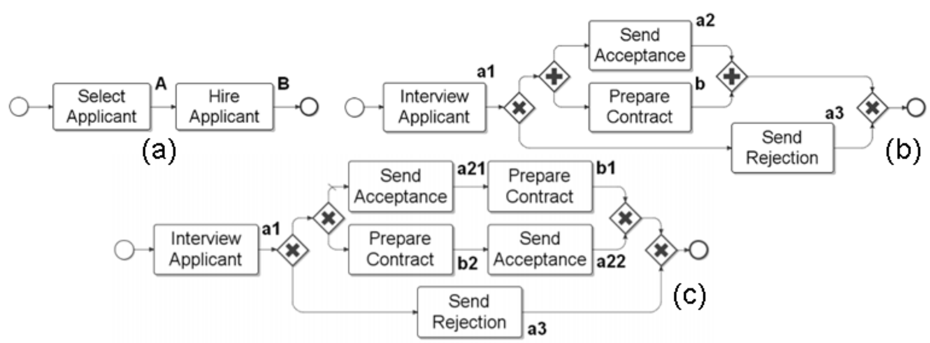

Fig. 1. Process refinement

We say that a process model m is a correct refinement of a process model n if the execution set of m subsumes the execution set of n after 1.**Renaming all activities in each execution of** m **by their originators.**; 2. **Replacing all sequences of equal activities by a single activity in each execution of** m**.** For our example, the execution set of Fig.1b becomes $\{[\mathsf{AB}], [\mathsf{ABA}], [\mathsf{A}]\}$.

As $\{[\mathsf{AB}]\} \not\supseteq \{[\mathsf{AB}], [\mathsf{ABA}], [\mathsf{A}]\}$, Fig. 1b is a wrong refinement of Fig. 1a. As the execution sets may contain infinite loops, our solution works with descriptions in ontologies rather than with the execution sets themselves.

2 Validation with Ontologies

We first **reduce process diagrams into execution diagrams** to make the execution order between branches of a parallel gateway explicit and **rename the duplicated activities** to differ their executions(see Fig. 1c). All the paths in Fig.1c make up the execution set of Fig.1a.

Then we **represent the execution flows with ontology.** We describe Fig.1a by universal restrictions, Fig.1c by existential restrictions, the originator of activities by the subsumptions and also the disjointness among activities:

$\mathcal{O} = \{ A \sqsubseteq (\forall from.(Start \sqcup A)) \sqcap (\forall to.(B \sqcup A)),\ B \sqsubseteq (\forall from.(A \sqcup B)) \sqcap (\forall to.(End \sqcup B)),\ a1 \sqsubseteq \exists from.Start \sqcap (\exists to.a2_1 \sqcup \exists to.b_2 \sqcup \exists to.a3),\ a2_1 \sqsubseteq \exists from.a1 \sqcap \exists to.b_1,\ a2_2 \sqsubseteq \exists from.b_2 \sqcap \exists to.End,\ b_1 \sqsubseteq \exists from.a2_1 \sqcap \exists to.End,\ b_2 \sqsubseteq \exists from.a1 \sqcap \exists to.a2_2,\ a3 \sqsubseteq \exists from.a1 \sqcap \exists to.End,\ a1, a2_1, a2_2, a3 \sqsubseteq A,\ b_1, b_2 \sqsubseteq B,\ disjoint(A, B, Start, End),\ disjoint(a1, a2_1, a2_2, a3),\ disjoint(b_1, b_2)\}$

Concept satisfiability checking shows that concepts $a2_2$, b_2 and $a3$ are unsatisfiable in above ontology, which implies that the routes constituted by $a2$, b and $a3$ are invalid in the original process diagram.

3 Related works & Conclusion

In [2] an OWL-DL model for service behavior and interaction is described. In [1] two formal semantics for a process algebra are described. Semantics for BPMN models in process algebra and process refinements are outlined in [5]. In [3] the bisimulation is defined for the pi-calculus. Process bisimulation with higher-order process calculi is analyzed in [4].

Our work represents process executions with ontologies and validate the refinement of process diagrams with reasoning. We restricted our solution to a commonly used subset of BPMN [7], which we may extend in the future.

References

1. A.J. Cowie. *The Modelling of Temporal Properties in a Process Algebra Framework*. PhD thesis, University of South Australia, 1999.
2. Markus Fronk and Jens Lemcke. Expressing semantic Web service behavior with description logics. In *Semantics for Business Process Management Workshop at ESWC*, 2006.
3. R. Milner, J. Parrow, and D. Walker. A Calculus of Mobile Processes. *Information and Computation*, pages 41–77, 1992.
4. Davide Sangiorgi. Bisimulation for Higher-Order Process Calculi. *Information and Computation*, 131:141–178, 1996.
5. P.Y.H. Wong and J. Gibbons. A process-algebraic approach to workflow specification and refinement. *Lecture Notes in Computer Science*, 4829:51, 2007.
6. George M. Wyner and Jintae Lee. Defining specialization for process models. In *Organizing Business Knowledge: The MIT Process Handbook*, chapter 5, pages 131–174. MIT Press, 2003.
7. Michael zur Muehlen. How much BPMN do you need? Online, March 2008. `http://www.bpm-research.com/2008/03/03/how-much-bpmn-do-you-need/`.