

A Proposal for Checking the Conformance of ebBP-ST Choreographies and WS-BPEL Orchestrations

Matthias Geiger, Andreas Schönberger and Guido Wirtz

Distributed and Mobile Systems Group,
University of Bamberg
Bamberg, Germany

{matthias.geiger, andreas.schoenberger, guido.wirtz}@uni-bamberg.de

Abstract. A common problem in applying choreographies and orchestrations is ensuring and enforcing the consistency of the models which is often referred to as “conformance checking”. In this position paper, we introduce a concept for checking the conformance of WS-BPEL based orchestrations to ebBP-ST choreographies: First, the ebBP-ST and WS-BPEL models will be transformed into the process algebra CCS. Afterwards, the actual conformance check is performed by checking these CCS models for bisimulation equivalence.

Keywords: choreography, orchestration, conformance checking, WS-BPEL, ebBP

Today, the concepts of choreography and orchestration are well-known and widely accepted in the services community. While choreographies describe a scenario from a global point of view, orchestrations concentrate on the local implementations of each involved party. While the de-facto standard on the orchestration level is the *Web Services Business Process Execution Language* (WS-BPEL; [3]), there exist various choreography languages.

Especially in the business-to-business integration (B2Bi) domain, the *ebXML Business Process Specification Schema* (BPSS or ebBP; [4]) is a suitable choreography language as it provides B2Bi specific features. In [5], the authors present ebBP-ST as an ebBP modeling dialect that captures collaborations between exactly two partners as state-machine based choreographies. ebBP-ST choreographies define an interaction protocol between partners that defines admissible business document exchanges and the effect of those exchanges.

At the implementation level, this protocol has to be implemented for the involved partners. Figure 1 shows the basic architecture of these implementations: For each involved partner, the (existing) systems containing application logic are encapsulated as so-called backend systems. The actual orchestrations are realized by so-called control processes (one for each partner) which are realized in WS-BPEL. Within these WS-BPEL processes the automata structure defined in the choreography is reused in order to govern the control flow of the choreography.

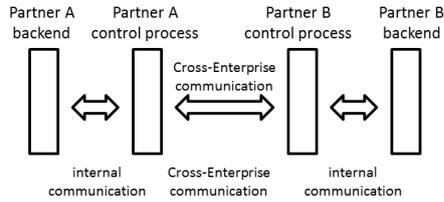


Fig. 1. Basic Architecture

Nevertheless, the actual decision which activity should be performed next is not initiated by the control process but by the backend systems resp. the interaction partner.

As control processes do not contain application logic, it is clear that a rather strict notion of conformance has to be used: A control process is not able to know how the backend systems or the partner control process will continue in the protocol execution and therefore all foreseen possibilities defined by the choreography have to be implemented. (Weak) bisimulation ([2]) is a suitable equivalence notion for this purpose.

To actually check the conformance between ebBP-ST choreographies and corresponding WS-BPEL orchestrations, the process definitions have to be transformed into a common formal basis. We propose the process algebra *Calculus of Communicating Systems* (CCS; [1]) as a suitable process representation because of its straightforward support for state machine like structures.

After specifying the conformance notion and the process algebra to use, the overall approach to check the conformance is as follows: As ebBP-ST choreographies describe binary collaborations, the ebBP-ST choreography has to be implemented by two WS-BPEL orchestrations - one for each involved partner. Therefore, the ebBP-ST model as well as the two WS-BPEL orchestrations have to be transformed to CCS, resulting in three CCS representations. Afterwards each of the CCS orchestration models has to be checked against the CCS choreography model regarding bisimulation equivalence using a model checking tool like the Edinburgh Concurrency Workbench (CWB).

References

1. R. Milner. *A Calculus of Communicating Systems*, volume 92 of *Lecture Notes in Computer Science*. Springer, 1980.
2. R. Milner. Calculi for Synchrony and Asynchrony. *Theoretical Computer Science*, 25:267–310, 1983.
3. OASIS. *Web Services Business Process Execution Language (WSBPEL) Version 2.0*, April 2007.
4. OASIS. *ebXML Business Process Specification Schema Technical Specification Version 2.0.4*, Oktober 2006.
5. A. Schönberger, C. Pflügler, and G. Wirtz. Translating Shared State Based ebXML BPSS models to WS-BPEL. *International Journal of Business Intelligence and Data Mining - Special Issue: 11th International Conference on Information Integration and Web-Based Applications and Services in December 2009*, 5(4), 2010.