Reflections on a Decade of Interorganizational Workflow Research

Wil M.P. van der Aalst and Mathias Weske

Abstract The Public-To-Private (P2P) approach presented at CAiSE in 2001 provides a correctness-by-construction approach to realize interorganizational workflows. A behavioral inheritance notion is used to ensure correctness: organizations can alter their private workflows as long as these remain subclasses of the agreed-upon public workflow. The CAiSE'01 paper illustrates the strong relationship between business process management and service-orientation. Since 2001, there is a trend from the investigation of individual process orchestrations to interacting processes, i.e., process choreographies. In this paper, we reflect on the original problem statement and discuss related work.

1 Introduction

In a Service Oriented Architecture (SOA) services are interacting by exchanging messages and by combining services more complex services are created. Choreography is concerned with the composition of such services seen from a global viewpoint focusing on the common and complementary observable behavior. Choreography is particularly relevant in a setting where there is not a single coordinator. Orchestration is concerned with the composition of such services seen from the viewpoint of single service. Independent of the viewpoint (choreography or orchestration) there

Wil M.P. van der Aalst
Architecture of Information Systems, Eindhoven University of Technology, P.O. Box 513, NL-5600 MB, Eindhoven, The Netherlands; Business Process Management Discipline, Queensland University of Technology, Australia; and International Laboratory of Process-Aware Information Systems, National Research University Higher School of Economics, Moscow, Russia, e-mail: w.m.p.v.d.aalst@tue.nl.

Mathias Weske
Hasso Plattner Institute at the University of Potsdam, Prof.-Dr.-Helmert-Strasse 2-3, 14482 Potsdam, Germany, e-mail: Mathias.Weske@hpi.uni-potsdam.de.

is a need to make sure that the services work together properly to ensure the correct execution of business processes. The resulting system should be free of deadlocks, livelocks, and other anomalies.

The *Public-To-Private* (P2P) approach presented at CAiSE'01 [7] addressed such correctness concerns using a notion of *inheritance* defined for Workflow nets (WF-nets) [3, 4, 9]. The P2P approach consists of three steps: (1) create a common understanding of the interorganizational workflow by specifying a shared public workflow, (2) partition the public workflow over the organizations involved, and (3) for each organization, create a private workflow which is a *subclass* of the respective part of the public workflow. Subsequently, projection inheritance ensures that the resulting interorganizational workflow realizes the behavior specified in the public workflow.

In the remainder, we reflect on a decade of interorganizational workflow research.[1] In Section 2 we study the trend in business process management research from process orchestrations to process choreographies, which started about ten years ago. Two streams of research are highlighted. Formal investigations on how interacting business processes can be analyzed and results related to the modeling of process choreographies and the impact of this research stream on today's standards in business process modeling. In Section 3, we challenge the correctness-by-construction approach of [7] and advocate the more active use of event data at run-time.

2 From Process Orchestrations to Process Choreographies

Until about 2001, research in business process management or — at that time — workflow management, centered around individual processes that are enacted within a single organization, i.e., process orchestrations. Process orchestrations consist of activities that are executed in coordination in a technical and organizational environment and are performed to achieve a business goal [22]. Workflow research looked at formal aspects related to process behavior but also at conceptual aspects like the flexibility of processes. In all of these research areas, individual processes were in the center of attention.

After 2001, the scope of research broadened from individual processes performed by single organizations to interactions between several processes performed by different organizations. From today's perspective, this step was quite obvious, since process orchestrations tend to talk to process orchestrations performed by other organizations.

For instance, when ordering a new laptop computer, we ask several hardware suppliers for quotes. The receipt of such a quote by a supplier spawns a new process orchestration at the supplier's side. Depending on, e.g., the specification of the laptop, the dealer might decide to issue a quote. On receiving a sufficiently large set of quotes, we collect and compare them, and send an purchase order to one of

[1] Due to space restrictions, we can only list a tiny fraction of the work on process orchestrations and choreographies and do not suggest being complete in any way.

them. In real-world scenarios, the interactions of processes can be much more complex than in this example. However, it shows that process orchestrations are actually interconnected with each other. Studying these types of connections is worthwhile and challenging, both from an academic and from a practical perspective.

2.1 Formal Investigations

A major stream of work relates to the formal investigation of interacting processes. At the beginning of the Millennium, Service Oriented Architectures (SOA) were "en vogue", so many academics started to formalize service notions. After abstracting reality to formal models, such as Petri nets, services and process orchestrations can no longer be distinguished from one another.

One of the earliest results were presented in [16, 15], where the interactions of services were defined by a specific type of Petri nets, called workflow modules, and correctness criteria for interacting services were proposed. Based on this work, [17] looked at the service selection problem, which so far had mostly been discussed from a either a software technology or from a semantics perspective. Operating guidelines for services have been introduced as a powerful behavioral specification of all services that can successfully cooperate with the specific service under consideration. At the same time papers such as [14] related concrete execution languages like the Business Process Execution Language for Web Services (BPEL [8]) to formalisms like Petri nets. The main results of this stream of research are surveyed and partly extended in [23], where controllability of services is in the center of attention by answering the question "Does my service have partners?". Based on this work, a question very similar to that of the original P2P paper was addressed in [6], where multiparty contracts are proposed. These define the overall intended process interactions and the roles of the parties involved. Based on a contract, each party implements its own process orchestration, guided by an accordance criterion.

There is a specific aspect that separates process orchestrations from choreographies; while the former have a static structure, the latter have a dynamic structure. During run-time, a participating organization might select a new partner, so that the structure of the system evolves over time. These aspects can be captured using the pi calculus which provides a mobility notion allowing for communication structures to be changed while the system runs. [10] formally specifies a set of service interaction patterns based on the pi calculus. With interaction soundness, a new criterion for interacting processes was defined in [20]. These results are surveyed and partly extended in [21].

2.2 Modeling and Impact

In addition to the investigation of formal aspects, considerable work on the modeling of process choreographies has been conducted. As of version Version 1 released in 2003, BPMN can be used to model interacting business processes (by drawing a pool for each participant and specifying the interactions between pools by message flow). There were two options to do so. Either the internal processes were hidden or only communication activities were drawn with their local control flow constraints. This modeling technique proves error prone, since the distribution of responsibilities among the participants could not be described properly, which could lead to undesired interaction behavior, such as deadlocks.

In [25, 24], a new modeling technique called Let's Dance was introduced, together with a set of desirable properties of interacting processes, such as local enforceability. The basic idea of this approach is avoiding to connect the communication interfaces of the participants, but to concentrate on the actual interactions and define control flow between them. The term *interaction-based choreography* modeling was coined for this modeling style. In a follow-up paper on interaction BPMN [11], the basic concepts of Let's Dance were maintained, while taking advantage of the BPMN notation. Behavioral consistency of interacting processes was addressed in [12]; the results of this stream of research was surveyed and partly extended in [13].

Based on these insights, BPMN provides dedicated diagram types for modeling process choreographies as of Version 2 [19]. For example, choreography diagrams are directly based on the concepts introduced in the research papers mentioned.

3 Correctness-By-Construction Versus Service Mining

The P2P approach provides a *correctness-by-construction* approach, i.e., parties do not need to know each others' private workflows. However, one needs to assume that the private workflow of another organization is indeed a subclass of the respective part of the public workflow. This assumption seems to be too strong:

- Organizations may implement a non-compliant private workflow (i.e., a workflow that is not a subclass under projection inheritance).
- Private workflows may change over time without an explicit notification and possibly violating earlier agreements.
- There are private workflows that are not a subclass under projection inheritance, but that can never lead to problems. For example, two parallel sending transitions can be made sequential without causing any problems. However, the resulting workflow is not a subclass.

As suggested in [2, 5, 18], it may be better to observe the messages exchanged and use conformance checking instead. Consider for example the public view shown in Figure 1. Sending payments *sp* before receiving goods *rg* (i.e., effectively removing

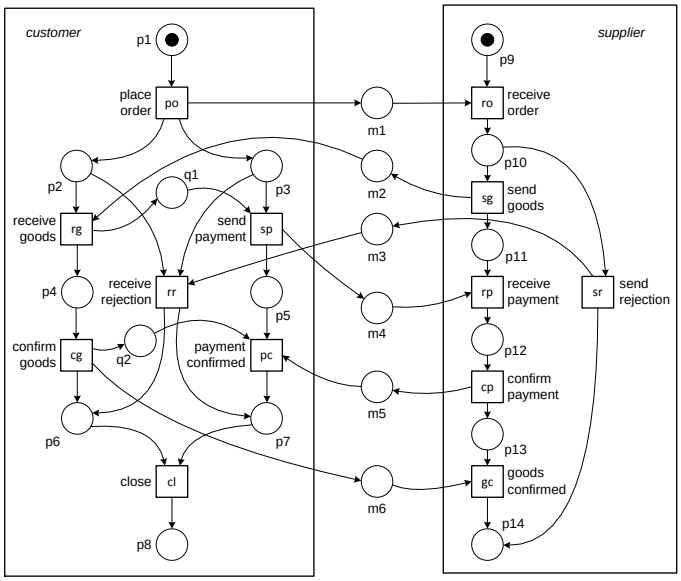

Fig. 1 An interorganizational workflow modeling two interacting processes. Removing place $q1$ may result in deadlock, cf. trace $\langle po, ro, sr, sp \rangle$. Removing place $q2$ changes behavior (payment confirmation may occur before confirming goods) but this cannot jeopardize correctness.

place $q1$) may cause deadlocks. This can be observed when message $m4$ precedes message $m2$. Receiving a payment conformation *pc* before confirming goods *cg* (i.e., effectively removing place $q2$) is harmless. All suppliers that can cooperate well with the *customer* workflow shown in Figure 1, can also cooperate with the alternative workflow without place $q2$ (which is not a subclass). This illustrates that the P2P approach may be too strict. Moreover, one needs to monitor the message exchanges to detect violations, because, often, private workflows of other parties cannot be controlled. Therefore, we suggest putting more effort in *service mining* [2], i.e., the application of process mining techniques [1] as a tool for discovering, checking, and improving interorganizational workflows.

References

1. W.M.P. van der Aalst. *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer-Verlag, 2011.
2. W.M.P. van der Aalst. Service Mining: Using Process Mining to Discover, Check, and Improve Service Behavior. *IEEE Transactions on Services Computing*, 2013.
3. W.M.P. van der Aalst and T. Basten. Life-cycle Inheritance: A Petri-net-based Approach. In P. Azéma and G. Balbo, editors, *Application and Theory of Petri Nets 1997*, volume 1248 of *Lecture Notes in Computer Science*, pages 62–81. Springer-Verlag, 1997.
4. W.M.P. van der Aalst and T. Basten. Inheritance of Workflows: An Approach to Tackling Problems Related to Change. *Theoretical Computer Science*, 270(1-2):125–203, 2002.

5. W.M.P. van der Aalst, M. Dumas, C. Ouyang, A. Rozinat, and H.M.W. Verbeek. Conformance Checking of Service Behavior. *ACM Transactions on Internet Technology*, 8(3):29–59, 2008.
6. W.M.P. van der Aalst, N. Lohmann, P. Massuthe, C. Stahl, and K. Wolf. Multiparty Contracts: Agreeing and Implementing Interorganizational Processes. *The Computer Journal*, 53(1):90–106, 2010.
7. W.M.P. van der Aalst and M. Weske. The P2P approach to Interorganizational Workflows. In *International Conference on Advanced Information Systems Engineering (CAiSE'01)*, volume 2068 of *Lecture Notes in Computer Science*, pages 140–156. Springer-Verlag, 2001.
8. A. Alves et al. Web Services Business Process Execution Language Version 2.0 (OASIS Standard). WS-BPEL TC OASIS, http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.html, 2007.
9. T. Basten and W.M.P. van der Aalst. Inheritance of Behavior. *Journal of Logic and Algebraic Programming*, 47(2):47–145, 2001.
10. G. Decker, F. Puhlmann, and M. Weske. Formalizing Service Interactions. In *International Conference on Business Process Management (BPM 2006)*, volume 4102 of *Lecture Notes in Computer Science*, pages 414–419. Springer-Verlag, 2006.
11. G. Decker and A. Barros. Interaction Modeling Using BPMN. In Arthur H. M. ter Hofstede, Boualem Benatallah, and Hye-Young Paik, editors, *Business Process Management Workshops*, volume 4928 of *Lecture Notes in Computer Science*, pages 208–219. Springer, 2007.
12. G. Decker and M. Weske. Behavioral Consistency for B2B Process Integration. In John Krogstie, Andreas L. Opdahl, and Guttorm Sindre, editors, *CAiSE*, volume 4495 of *Lecture Notes in Computer Science*, pages 81–95. Springer, 2007.
13. G. Decker and M. Weske. Interaction-centric Modeling of Process Choreographies. *Information Systems*, 36(2):292–312, 2011.
14. N. Lohmann, P. Massuthe, C. Stahl, and D. Weinberg. Analyzing Interacting BPEL Processes. In *International Conference on Business Process Management (BPM 2006)*, volume 4102 of *Lecture Notes in Computer Science*, pages 17–32. Springer-Verlag, 2006.
15. A. Martens. On Compatibility of Web Services. In *10th Workshop on Algorithms and Tools for Petri Nets (AWPN 2003), Eichstätt, Germany*, 2003.
16. A. Martens. On Usability of Web Services. In *Fourth International Conference on Web Information Systems Engineering Workshops*, IEEE, 2003.
17. P. Massuthe, W. Reisig, and K. Schmidt. An Operating Guideline Approach to the SOA. In *South-East European Workshop on Formal Methods (SEEFM'05)*, Ohrid, 2005.
18. R. Müller, W.M.P. van der Aalst, and C. Stahl. Conformance Checking of Services Using the Best Matching Private View. In N. Lohmann and M. ter Beek, editors, *WS-FM 2012*, Lecture Notes in Computer Science. Springer-Verlag, 2012.
19. Object Management Group. *Business Process Model and Notation (BPMN) Version 2.0*, formal/2011-01-03 edition, 2011.
20. F. Puhlmann and M. Weske. Interaction Soundness for Service Orchestrations. In *Service-Oriented Computing (ICSOC 2006)*, volume 4294 of *Lecture Notes in Computer Science*, pages 302–313. Springer-Verlag, 2006.
21. F. Puhlmann and M. Weske. A Look Around the Corner: The Pi-Calculus. In *Transactions on Petri Nets and Other Models of Concurrency II*, pages 64–78. Springer-Verlag, 2009.
22. M. Weske. *Business Process Management: Concepts, Languages, Architectures*. Springer-Verlag, second edition, 2012.
23. K. Wolf. Does my Service Have Partners? In *Transactions on Petri Nets and Other Models of Concurrency II*, pages 152–171. Springer-Verlag, 2009.
24. J.M. Zaha, A. Barros, M. Dumas, and A.H.M. ter Hofstede. Lets Dance: A Language for Service Behavior Modeling. In *International Conference on Cooperative Information Systems*, volume 4275 of *Lecture Notes in Computer Science*, pages 145–162. Springer-Verlag, 2006.
25. J.M. Zaha, M. Dumas, A. ter Hofstede, A. Barros, and G. Decker. Service Interaction Modeling: Bridging Global and Local Views. In *International Enterprise Distributed Object Computing Conference (EDOC 2006)*, pages 45–55. IEEE Computer Society, 2006.