

# Challenges in Service Mining: Record, Check, Discover

Wil M.P. van der Aalst<sup>1,2</sup>

<sup>1</sup> Architecture of Information Systems, Eindhoven University of Technology,  
P.O. Box 513, NL-5600 MB, Eindhoven, The Netherlands.

`w.m.p.v.d.aalst@tue.nl`

<sup>2</sup> International Laboratory of Process-Aware Information Systems, National  
Research University Higher School of Economics (HSE),  
33 Kirpichnaya Str., Moscow, Russia.

**Abstract.** Process mining aims to discover, monitor and improve real processes by extracting knowledge from event logs abundantly available in today's information systems. Although process mining has been applied in hundreds of organizations and process mining techniques have been embedded in a variety of commercial tools, to date these techniques have rarely been used for analyzing web services. One of the obvious reasons is that cross-organizational event data cannot be shared easily. However, (1) messages exchanged between services tend to be structured, (2) service-orientation continues to be the predominant implementation paradigm, and (3) the most substantial efficiency gains can often only be achieved across different organizations. Hence, there are many possible applications for *service mining*, i.e., applying process mining techniques to services. If messages are recorded, then one can discover a process describing interactions between services. If, in addition, descriptive or normative models are available, one can use process mining to check conformance and highlight performance problems. This extended abstract aims to provide pointers to ongoing work on service mining and lists some of the main challenges in this emerging field.

## 1 From Process Mining to Service Mining

*Process mining* is an enabling technology for service mining. Process mining can be used to *discover* processes from raw event data, *check* the conformance of observed and modeled behavior, *enhance* models by improving or extending them with knowledge extracted from event logs [2]. The uptake of process mining is not only illustrated by the growing number of papers, but also by commercial analysis tools providing process mining capabilities, cf. Disco (Fluxicon), Perceptive Process Mining (Perceptive Software, before Futura Reflect and BPMone by Pallas Athena), ARIS Process Performance Manager (Software AG), ProcessAnalyzer (QPR), Interstage Process Discovery (Fujitsu), Discovery Analyst (StereoLOGIC), and XMAalyzer (XMPro).

Web services have become one of the main paradigms for architecting and implementing business collaborations within and across organizational boundaries

[10, 20]. The functionality provided by many of today’s business applications is encapsulated within web services, i.e., software components described at a semantic level, which can be invoked by application programs or by other services through a stack of Internet standards including HTTP, XML, SOAP, WSDL and UDDI [10, 20]. Once deployed, web services provided by various organizations can be inter-connected in order to implement business collaborations, leading to composite web services.

In the context of web services, typically all kinds of *events* are being recorded. It is possible to record events related to *activities inside services* or *interactions between services* (e.g., messages) [6, 8, 9]. The autonomous nature of services and the fact that they are loosely coupled makes it important to monitor and analyze their behavior. In this paper, we will refer to this as *service mining*.

Starting point for process mining is an *event log*. Each event in such a log refers to an *activity* (i.e., a well-defined step in some process) and is related to a particular *case* (i.e., a process instance). The events belonging to a case are ordered and describe one “run” of the process. Event logs may store additional information about events. In fact, whenever possible, process mining techniques use supplementary information such as the *resource* (i.e., person, device, or software component) executing or initiating the activity, the *timestamp* of the event, and other *data attributes* (e.g., the size of an order). As mentioned before, three types of process mining can be distinguished: (1) *process discovery*, (2) *conformance checking*, and (3) *model enhancement*. See [2] for an introduction to the corresponding techniques.

The *correlation of messages* is a particular challenge for service mining [3]. Process models always describe the behavior of cases, also referred to as process instances. Without correlating messages, it is impossible to discover causalities. Another challenge is to *use additional information provided by such messages*. In case of asynchronous messages with sender and receiver information we can exploit knowledge about distributed processes, e.g., choices need to be communicated. For example, service *x* cannot expect the service *y* to take action because *x* did *not* send a message to *y*. Thus far, these insights are not used in process discovery [16].

## 2 Related Work on Service Mining

In this section, we provide some pointers to papers on services mining and related topics. Given space restrictions, we do not aim to be complete. For additional references we refer the interested reader to [3].

In [9] a concrete application of process mining to web services is described. IBM’s WebSphere product is used as a reference system and its CEI (Common Event Infrastructure) logs are analyzed using ProM.

An approach to check the conformance of web services was described in [6]. The paper includes a description of various experiments using Oracle BPEL. The token-based replay techniques presented in [18] were used to measure conformance.

In [8] an LTL-based approach to check conformance was proposed. This approach uses a graphical declarative language to describe the normative behavior of services. Rather than modeling a detailed process, this approach allows for checking graphically specified constraints such as “a payment should always be confirmed”.

The topic of event correlation has been investigated in the context of system specification, system development, and services analysis. In [7] and [11] various interaction and correlation patterns are described. In [17] a technique is presented for correlating messages with the goal to visualize the execution of web services. In [16] so-called operating guidelines are exploited for conformance checking.

Dustdar et al. [12, 14] proposed techniques for services interaction mining, i.e., applying process mining techniques to the analysis of service interactions.

Nezhad et al. [15] developed techniques for event correlation and process discovery from web service interaction logs. The authors introduce the notion of a “process view” which is the result of a particular event correlation. However, they argue that correlation is subjective and that multiple views are possible. A collection of process views is called the “process space”.

In [19], Simmonds et al. propose a technique for the run-time monitoring of web service conversations. The authors monitor conversations between partners at runtime as a means of checking behavioral correctness of the entire web service system. This is related to the earlier work on conformance checking [4, 6, 18] mentioned before.

Within the ACSI project the focus is on many-to-many relationships between instances. So-called “proclets” [5] are used to model artifact centric models. A conformance checking approach for such models is presented in [13] and implemented in ProM.

In [1] the topic of “cross-organizational mining” was introduced. Here the goal is not to analyze interacting services but to compare services that are variants of one another. Cross-organizational mining can be used for benchmarking and reference modeling.

## Acknowledgements

This work was supported by the Basic Research Program of the National Research University Higher School of Economics (HSE).

## References

1. W.M.P. van der Aalst. Configurable Services in the Cloud: Supporting Variability While Enabling Cross-Organizational Process Mining. In *CoopIS 2010*, volume 6426 of *Lecture Notes in Computer Science*, pages 8–25. Springer-Verlag, 2010.
2. W.M.P. van der Aalst. *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer-Verlag, Berlin, 2011.
3. W.M.P. van der Aalst. Service Mining: Using Process Mining to Discover, Check, and Improve Service Behavior. *IEEE Transactions on Services Computing*, 2013 (in print, <http://doi.ieeecomputersociety.org/10.1109/TSC.2012.25>).

4. W.M.P. van der Aalst, A. Adriansyah, and B. van Dongen. Replaying History on Process Models for Conformance Checking and Performance Analysis. *WIREs Data Mining and Knowledge Discovery*, 2(2):182–192, 2012.
5. W.M.P. van der Aalst, P. Barthelmess, C.A. Ellis, and J. Wainer. Proclets: A Framework for Lightweight Interacting Workflow Processes. *International Journal of Cooperative Information Systems*, 10(4):443–482, 2001.
6. 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.
7. W.M.P. van der Aalst, A.J. Mooij, C. Stahl, and K. Wolf. Service Interaction: Patterns, Formalization, and Analysis. In *Formal Methods for Web Services*, volume 5569 of *Lecture Notes in Computer Science*, pages 42–88. Springer-Verlag, 2009.
8. W.M.P. van der Aalst and M. Pesic. Chapter 2: Specifying and Monitoring Service Flows: Making Web Services Process-Aware. In *Test and Analysis of Web Services*, pages 11–56. Springer-Verlag, Berlin, 2007.
9. W.M.P. van der Aalst and H.M.W. Verbeek. Process Mining in Web Services: The WebSphere Case. *IEEE Bulletin of the Technical Committee on Data Engineering*, 31(3):45–48, 2008.
10. G. Alonso, F. Casati, H. Kuno, and V. Machiraju. *Web Services Concepts, Architectures and Applications*. Springer-Verlag, Berlin, 2004.
11. A. Barros, G. Decker, M. Dumas, and F. Weber. Correlation Patterns in Service-Oriented Architectures. In *Fundamental Approaches to Software Engineering (FASE 2007)*, volume 4422 of *Lecture Notes in Computer Science*, pages 245–259. Springer-Verlag, 2007.
12. S. Dustdar and R. Gombotz. Discovering Web Service Workflows Using Web Services Interaction Mining. *International Journal of Business Process Integration and Management*, 1(4):256–266, 2006.
13. D. Fahland, M. de Leoni, B.F. van Dongen, and W.M.P. van der Aalst. Conformance Checking of Interacting Processes with Overlapping Instances. In *Business Process Management (BPM 2011)*, volume 6896 of *Lecture Notes in Computer Science*, pages 345–361. Springer-Verlag, Berlin, 2011.
14. R. Gombotz and S. Dustdar. On Web Services Mining. In *BPM 2005 Workshops (Workshop on Business Process Intelligence)*, volume 3812 of *Lecture Notes in Computer Science*, pages 216–228. Springer-Verlag, Berlin, 2005.
15. H.R. Montahari-Nezhad, R. Saint-Paul, F. Casati, and B. Benatallah. Event Correlation for Process Discovery from Web Service Interaction Logs. *VLBD Journal*, 20(3):417–444, 2011.
16. R. Müller, W.M.P. van der Aalst, and C. Stahl. Conformance Checking of Services Using the Best Matching Private View. In *WS-FM 2012*, volume 7843 of *Lecture Notes in Computer Science*, pages 49–68. Springer-Verlag, Berlin, 2013.
17. W. De Pauw, M. Lei, E. Pring, L. Villard, M. Arnold, and J.F. Morar. Web Services Navigator: Visualizing the Execution of Web Services. *IBM Systems Journal*, 44(4):821–845, 2005.
18. A. Rozinat and W.M.P. van der Aalst. Conformance Checking of Processes Based on Monitoring Real Behavior. *Information Systems*, 33(1):64–95, 2008.
19. J. Simmonds, Y. Gan, M. Chechik, S. Nejati, B. Farrell, E. Litani, and J. Waterhouse. Runtime Monitoring of Web Service Conversations. *IEEE Transactions on Services Computing*, 2(3):223–244, 2009.
20. L.J. Zhang, J. Zhang, and H. Cai. *Services Computing, Core Enabling Technology of the Modern Services Industry*. Springer-Verlag, Berlin, 2007.