

# Analysis of Service Level Agreements using Process Mining techniques

CHRISTIAN MAGER

University of Applied Sciences Wuerzburg-Schweinfurt

Process Mining offers powerful methods to extract knowledge from event logs. These event logs can be recorded by any process aware information system and must not conform to a particular format. The main contribution of this paper is to show how Process Mining can be used to analyze the time perspective of processes determined in so called Service Level Agreements (SLA). In terms of processes determined by SLAs, the processing time is extremely crucial. Frequently, delays in processes lead to SLA violations. Therefore, the Process Mining based approach in this paper goes beyond SLA monitoring techniques which simply display the correct execution of a process. In contrast to SLA monitoring, already executed processes can be analyzed from a service provider perspective. The paper describes the creation of an integrated process model which supports a fine-grained investigation of the time perspective. Moreover, this model helps to identify time related bottle-necks in these processes. All Process Mining techniques necessary to create the integrated process model are explained based on a prototypical implementation. The main benefit of this approach is to support a continuous improvement of the quality of service as claimed in ITIL.

Categories and Subject Descriptors: Process Mining

Additional Key Words and Phrases: Process Model Discovery,  $\alpha$ -Algorithm, Time perspective, ITIL, Service Level Agreement, Service Level Controlling, SLA Monitoring

## 1 INTRODUCTION

In the last decade real-time communication via internet enabled a close collaboration between multiple organizations. Flexibility and an improved competitiveness are the major benefits of these collaborations. Yet, if two parties work together, duties and responsibilities need to be defined to enable a fruitful collaboration for all parties. In terms of IT supported processes, the ITIL framework (IT Infrastructure Library) offers a powerful set of concepts and approaches to root a service culture in an organization [Van Bon 2008]. One major objective in every effort regarding ITIL is a sustainable management of processes in a service aware environment. Thereby, the quality of service (QoS) is determined in so called Service Level Agreements. To maintain the quality of these services, an ongoing analysis and controlling of an organization's processes is inevitable. However, it is not addressed in ITIL how to analyze the quality of services in Service Level Agreements (SLA). Most efforts that can be found in the literature are dealing with simple SLA monitoring. These approaches focus on monitoring only and are not suitable for detecting issues in the process. Nevertheless, putting more effort on improving the quality of processes is extremely crucial. Studies like the one presented in [Mending et al. 2007] prove that organizations tend to overestimate the knowledge of their own processes. Particularly large process models tend to be of poor quality in practice, containing formal flaws. Process Mining can be helpful with providing analysis techniques to improve process quality.

Process Mining is a relatively young research discipline offering powerful methods to extract knowledge from event logs of already executed business processes. Thereby, the major advantage of Process Mining is its' platform and format independence. Event logs of every kind of workflow-engine-like information system can be used and transformed into a Process Mining

---

Author's address: Christian Mager, University of Applied Sciences Wuerzburg-Schweinfurt, Germany.  
www.fhws.de.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than FHWS must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission.

conform semantic [Van der Aalst et al. 2012]; [Van der Aalst 2011]. In Process Mining the main focus is on extracting the ordering of events in a process which is commonly called the control-flow perspective. Yet, typically event logs contain richer information such as resource, cost or time information. This makes Process Mining an extremely powerful tool, in particular for organizations aiming to improve their processes continuously.

In this paper an integrated process model on the basis of Process Mining techniques is introduced that allows the analysis of the processes determined by SLAs. In particular this approach allows the analysis of executed process and a fine-grained analysis of the time perspective. First of all, Process Model Discovery techniques are used to extract information of the actual process model from event logs. Afterwards, the model is extended by the time perspective to an integrated process model which allows the detection of time-related bottlenecks in the process.

To specify the details of the approach, some requirements must be complied to successfully implement an analysis instrument for SLA aware processes. First of all, for this kind of effort, the process that is analyzed has to be constant. In other words, processes that are dynamically changing over time are not appropriate for using the techniques presented in this paper. Moreover, a process needs to be executed frequently to allow a Process Model Discovery algorithm to recreate the corresponding process model reliably. Every possible firing sequence of activities has to be recorded in the so called event log. A solid number of process executions increase the chance of getting a complete event log. A complete log is the basis of quality mining results [Van der Aalst 2011]. For instance, the processing of a classical IT-service-helpdesk which is repeated frequently and recorded by a ticketing-system is a good example for Process Mining implementation. One further requirement on a process is the causal dependency between a delay in the execution of the process and a violation of the SLA. This means a delayed delivery of the service provider directly prompts an SLA violation. Control-flow information is the least information necessary to allow Process Model Discovery techniques to work with the event log data [Van der Aalst et al. 2004]. However, for the approach presented in this paper, time information needs to be available in the event log as well. In general, due to particular objectives and questions, it has to be decided individually, what kind of information needs to be available in the event log. When using Process Mining techniques, noise data is a common issue that may distort the results fundamentally and lead to wrong interpretation. Multiple approaches exist to prohibit noise data and ensure quality event logs [Weijters et al. 2006]; [De Medeiros and Weijters 2005]. However, this topic is not addressed in this work.

The following section 2 starts with an introduction on Process Mining as well as on Service Management and SLA monitoring. Afterwards, in section 3 an approach is presented that allows the analysis of the time perspective of processes, determined by an SLA. In section 4 the approach is expounded based on an example process. Finally, the outline in section 5 summarizes all results.

## **2 OVERVIEW**

### *a) Process Mining techniques*

Process Mining is a relatively young research discipline which uses concepts of computational intelligence and data mining on the one hand and process modeling and analysis techniques on the other hand [Van der Aalst et al. 2010]; [Van der Aalst et al. 2012]. Further work dealing with this topic was published under the name "Business Process Intelligence" [Grigori et al. 2004]. Discovering, monitoring and improving real processes are the key objectives of Process Mining. This is accomplished by extracting knowledge from so called event logs. Process Mining provides techniques and tools for discovering process, organizational, social, and performance information from event logs. Today's information systems record an enormous amount of data including

information about all activities that have been executed. However, this data might be unstructured, e.g. scattered over many tables or spread over more than one system. In this case, some effort needs to be done to extract and link data and merge it into a unified event log format. Due to this, specific event log formats like the XES-Standard have been proposed in the past which are suitable for Process Mining [Günther 2009]. The ProM framework is a state of the art open source framework provided by the Technical University of Eindhoven. It supports a variety of Process Mining techniques and uses the XES-Standard [TU/e 2012]; [Van Dongen et al. 2005].

Process Model Discovery is one of the major disciplines of Process Mining. In model discovery, algorithms are provided that try to explain the behavior of a process which is recorded in the event log. Thereby, most of the algorithms do not consider any a-priori information. One simple but very famous algorithm for model discovery is the  $\alpha$ -Algorithm [Van der Aalst et al. 2004]. Some more advanced mining algorithms were proposed meanwhile which are able to deal with noise, take heuristics into account and so on. The  $\alpha$ -Algorithm suits well to the approach described in this paper because it is simple to apply and still very powerful. The  $\alpha$ -Algorithm takes a structured event log as input and computes a Petri net model of the process as output. Therefore, the algorithm scans the event log for patterns in order to find causal dependencies between activities. These causal dependencies are used to identify simple sequences, choices, parallelism or loops between activities in the process. Taking these so called ordering relations into account, a Petri net model is computed. Petri net based process models are commonly used in Process Mining due to their high expressiveness, as well as their well-founded mathematical and graphical modeling notation. The well-defined semantic of Petri nets allows a transformation into other modeling notations like BPMN or YAWL. For this reason, Petri nets suit very well into the context of Process Mining [Van der Aalst 1998]; [Van der Aalst 2011].

Process model enhancement is another discipline in Process Mining. Enhancement aims to enrich a process model with additional information of the event log. Ideally the process model is discovered beforehand using a Process Model Discovery algorithm. The output of any enhancement effort is an improved or extended process model. In [Song and Van der Aalst 2008] the following perspectives are proposed that can be used to enrich a process model by using enhancement:

- Control-flow perspective: focus on ordering of events.
- Organizational perspective: focus on information about resources.
- Time perspective: focus on timing and frequency of events.

#### *b) IT-Service Management and SLA Controlling*

IT-Service Management refers to the implementation and management of quality IT services that meet the needs and requirements of a service consumer. The ITIL framework (IT Infrastructure Library) offers a collection of best-practices for the implementation of an IT Service Management [Van Bon 2008]. Duties and responsibilities between service provider and service customer are defined by Service Level Agreements specified in a service contract. Thereby, SLAs define the Quality of Service (QoS) with the help of quality attributes [Ferguson and Huston 1998]; [Molina-Jimenez et al. 2004]. One fundamental aspect of defining Service Level Agreements is that all agreed details in the contract must be measurable. Therefore, it is important to define statistical metrics about the performance of services. This enables an evaluation of the performance of a service and shows if it meets the agreed level of QoS. However, the quality of services in service level agreements and its metrification are very individual for every organization. Availability, response time or delivery time are attributes that are obviously measurable, for instance. These kinds of attributes can be grouped as performance attributes [Trienekens et al. 2004]. Usually, in case of violations of these SLAs, the accountable party has to pay a compensation agreed on in the contract. In order to avoid violations from the perspective of the provider, the continuous measurement and analysis of its' service delivery processes is crucial. Continual Service Improvement (CSI) is one phase in ITIL dealing with reviewing and

maintaining the processes [Van Bon 2008]. CSI recommends a permanent performance measurement and analysis of service delivery processes. Due to this, the measurement of SLA and the comparison to defined requirements are extremely important. To discover violations and measure the quality of service, SLA monitoring techniques can be used. Multiple technical approaches have been developed in the past to enable SLA monitoring [Ameller and Franch 2008]; [Molina-Jimenez et al. 2004]; [Keller and Ludwig 2003]. However, all of these approaches focus on monitoring only and are not suitable for detecting issues in the processes. In [Grigori et al. 2004] an approach is introduced which considers the idea to measure and control business processes in an integrated framework. This idea comes close to the one addressed in this paper, yet it doesn't allow the dynamical adjustment of process models. State-of-the art Process Mining techniques are able to adjust a model considering additional event logs.

### 3 PROCESS MINING FOR ANALYZING SERVICE LEVEL AGREEMENTS

One of the most important quality attributes processes determined by SLAs is the time perspective. For this reason, proper management and monitoring of time-related process performance is of utmost importance. Van der Aalst introduced an approach called: "Mining additional Perspectives" in [Van der Aalst 2011]. This approach can be used to extend a process model generated by a Process Model Discovery algorithm with additional perspectives. His idea is generic and can be adopted with changes to the problem addressed in this paper.

To create an integrated process model that contains not only the control-flow perspective but also the time perspective, the following information has to be available. Each process instance in the log needs to be identifiable on the basis of a unique case identifier. Additionally, all events that took place within one instance have to be ordered due to their occurrence. These two pieces of information are enough for the  $\alpha$ -Algorithm for model discovery to extract the control-flow of a process. The control-flow of a process illustrates the actual process model as it is documented in the process log. Yet, for enriching the model with the time perspective, a timestamp of start and end time of each activity needs to be recorded in the log. These timestamps are used to calculate the waiting time and the service time of all activities in the model. In this context, waiting time corresponds to the time that passes by between the complete event of an activity and the start event of the following activity. During the waiting time, the process is simply waiting for the next execution step and no progress is made. Waiting time is particularly destructive from the point of view of the service provider as time passes by although the contractually agreed processing time is running. The so called service time refers to the time between the start event and the complete event of an activity. Having an event log with at least this data, the way of creating the integrated process model is explained shortly in the following three steps:

(1) Derive event log suitable for Process Model Discovery. In other words, gather and merge event log data and transform it into a format, e.g. the XES-format, which is suitable for Process Mining.

(2) Use the  $\alpha$ -Algorithm for model discovery to compute an appropriate process model. Thereby, the algorithm scans the event log for causal dependencies of activities and puts them together to a Petri net process model. The Petri net model describes the control-flow perspective of the process.

(3) Extend the Petri net model by an additional perspective including time information. The dotted chart analysis, for instance, suits well for comparing and analyzing processing time. The dotted chart analysis shows the occurrence of events over time. This time information for each activity is based on its' start and complete event in the event log. By using start and end time of each activity, the waiting and the service time can be derived. Afterwards, this time information is displayed in the integrated model and compared to the target time.

#### 4 PROCESS MINING APPROACH BASED ON AN EXAMPLE

In this section, a three step approach to generate an integrated process model is described based on the event log in table I.

Case ID	Event name	Ressource	State [lifecycle:transition]	Timestamp
100	Register ticket	system	complete	01.06.2011 06:02
100	Analyze ticket	solver4	start	01.06.2011 06:06
100	Analyze ticket	solver4	complete	01.06.2011 07:12
100	Repair complex	solver2	start	01.06.2011 08:01
100	Repair complex	solver2	complete	01.06.2011 10:32
100	Test repair	tester3	start	01.06.2011 11:05
100	Test repair	tester3	complete	01.06.2011 11:32
100	Submit result	system	complete	01.06.2011 11:40
101	Register	system	complete	01.06.2011 06:08
101	Analyze ticket	solver2	start	01.06.2011 06:34
101	Analyze ticket	solver2	complete	01.06.2011 07:22
101	Repair simple	solver6	start	01.06.2011 08:03
101	Repair simple	solver6	complete	01.06.2011 09:41
101	Test repair	tester3	start	01.06.2011 12:01
101	Test repair	tester3	complete	01.06.2011 12:22
101	Submit result	system	complete	01.06.2011 12:39
102	Register	system	start	01.06.2011 06:10
102	Register	system	complete	01.06.2011 06:10
102	Analyze ticket	solver2	start	01.06.2011 06:45
102	Analyze ticket	solver2	complete	01.06.2011 07:19
102	Repair complex	solver3	start	01.06.2011 08:01
102	Repair complex	solver3	complete	01.06.2011 10:01
102	Test repair	tester5	start	01.06.2011 11:35
102	Test repair	tester5	complete	01.06.2011 12:24
102	Submit result	system	complete	01.06.2011 12:26

Table 1. Event log of help desk process

The approach is implemented in this section by using existing Process Mining techniques available in the ProM framework. Yet, multiple plug-ins have to be used to extract all information necessary for generating the integrated view on a process. So far, there is no plug-in in ProM allowing the creation of integrated process models. The following approach can be seen as an idea for a plug-in unifying the generation of integrated views.

First of all, event log data from a process aware information system needs to be extracted and transformed into a format suitable for Process Mining. The event log used for this example is shown in table I. This log contains all necessary information and is suitable for Process Mining. Only an extract of the full event log is used to generate the process model, as the whole log would be too long to fit in this paper. However, the extract gives a good idea on what the event log data looks like. The first column of the table named "Case ID" describes the case perspective. Each qualifier in this column corresponds to one unique process instance of the process model which has been executed before. In the second column "Event name", the ordering of events within one case is given. These two columns include enough information for computing the control-flow perspective applying the  $\alpha$ -Algorithm. Column three "Resource" shows the name of the resource that was executing the event. This column could be added as further perspective to the integrated model. Yet, the resource perspective is not considered in this work. Column four and five show

the "State" and the "Timestamp" of the event. The combined information of these two columns is used later to calculate service and waiting time.

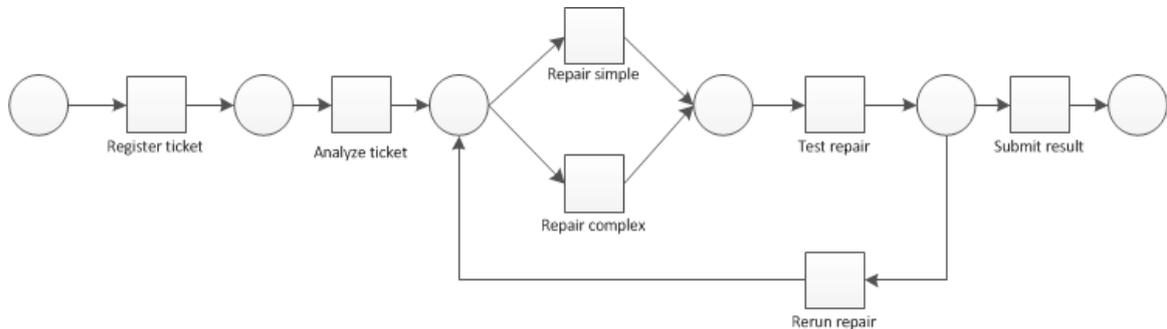


Fig. 1. Petri net model of event log in table I

Assuming this event log is provided as stated in step (1) of the concept, the next task is to compute the process model using the  $\alpha$ -Algorithm. The algorithm is provided in ProM by a plug-in. First, the event log as stated in table I is loaded into the mining framework. Therefore, the log needs to be transformed into the XES-format. The  $\alpha$ -Algorithm doesn't require any a-priori information provided by the user for computing the Petri net model. The algorithm uses only the case quantifier and the ordering of activities to construct a Petri net model which describes the model of the event log. The output of the algorithm is the Petri net shown in figure 1. Receiving the actual process model alone is already extremely valuable for analyzing processes determined by an SLA. The service provider gets the chance to monitor the actual execution of process instances and not only the process model as it was planned beforehand. In other words, the reengineering approach of Process Mining can be extremely valuable for auditing existing processes. This analysis mechanism is particularly helpful for processes specified by Service Level Agreements, as unforeseen occurrences in these processes may have an undesirable impact for the service provider. Not only the provider is directly affected by contractual penalties, moreover customer dissatisfaction can be prevented. For this reason, information provided by Process Mining can be supportive when aiming to improve services steadily as stated in ITIL.

Figure 2 shows a dotted chart that corresponds to the event log in table I. This chart can be generated using the ProM plug-in called "Dotted Chart Analysis". In this chart the time perspective of an event log is illustrated in a flexible diagram. The dots correspond to the starting point or to the point of completion of an activity in the event log. Thereby, the axis of the diagram can be customized individually to get a static or a relative view.



Fig. 2. Dotted chart analyzer plug-in in ProM6

Finally, in the third step the control-flow and the time perspective are assembled together to an integrated model. This model is illustrated in figure 3. The Petri net based model explains the behavior of the process corresponding to the event log in table I. Thereby, the time perspective is presented in form of a bar chart above places in the Petri net. The waiting time is projected as a red bar and the service time as a blue bar. The bar chart shows waiting and service time for the three process instances 100, 101 and 102.

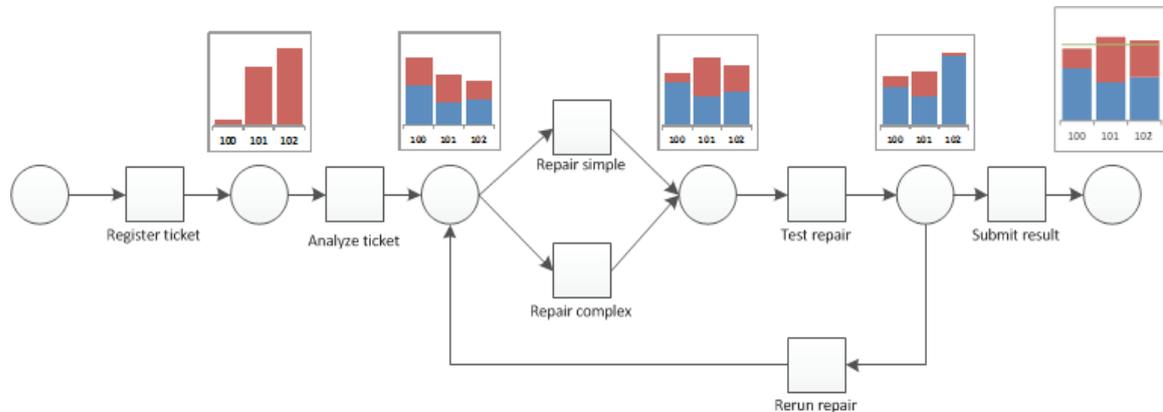


Fig. 3. Integrated process model including control-flow and time perspective

The process in the integrated model in figure 3 operates as follows. First, the activity "Register Ticket" is executed by the system. This activity is executed immediately and doesn't need any service time. However, before the next activity "Analyze Ticket" is processed, waiting time has passed by, as one can see in the bar chart above the place before the activity. The bar chart shows the waiting time for all three instances. The place after the following activity "Analyze Ticket" has two outgoing arcs that correspond to an exclusive or-split. This means either the activity "Repair complex" or the activity "Repair simple" is conducted. The bar chart above this place shows the service time of the previous activity and the waiting time till either one of the repair activities starts. The next place that contains two incoming arcs from both repair activities corresponds to an or-join. Subsequently, the repair is tested by the activity "Test repair". The next

place has again two outgoing arcs. Either the activity "Submit result" is executed and the process terminates or a rerun of the repair activity is performed.

What is particularly interesting about this figure, besides the process model as it is actually executed, is the possibility to analyze the time perspective of the process. The assumption that the processing time is six hours for each ticket is determined in the SLA. This agreement is illustrated in a green line in the bar chart above the final place in the Petri net. At this final place, the delivery of the service to the customer takes place. The bar chart above this final place shows the waiting and service time in total for all instances. One can see that the processing time of instance 100 is below the six hour line and instances 101, as well as 102 are above this line. This monitoring of service processing time is also possible with simple SLA monitoring approaches. However, these approaches don't offer any further details on the violations of the SLA. For improving the processes structure one might be interested in what caused the violations of instance 101 and 102. Therefore, a more fine-grained analysis is necessary. Having a closer look at the integrated model can be helpful here and give some hints on what might have caused the violations. Obviously, when looking at the bar chart above the final node, one can see that not necessarily the service time but the high waiting time of instance 101 and 102 have been problematic. Reviewing the detailed time chart for each processing step, lead to the conclusion that the waiting time before the activities "Analyze Ticket" and "Test repair" has been relatively high in case of instance 101 and 102. The next step would be to use these hints for a deeper analysis within the organization, why the waiting time for these two instances was so high.

## 5 CONCLUSION

This paper introduced an approach of generating an integrated process model that is suitable for analyzing the control-flow and the time perspective of processes specified by SLAs. This effort can be clearly delimited from classical SLA monitoring techniques, which simply provide an alerting in case of SLA violations. The analysis of SLAs based on Process Mining techniques allows a more precise search for unknown issues in the process composition like deadlocks. The approach presented in this paper enables a fine-grained analysis of the time perspective and the detection of bottle-necks in the process. Delays in the control-flow that may lead to SLA violations can be detected soon after the process execution. Certainly, all information detected by analyzing the integrated model can be extremely valuable for the continuous improvement of a process. The very simple example shows the power of the Process Mining based analysis approach. In case of larger processes with hundreds or even thousands of activities and a much more complex structure these hints derived from an integrated view can be extremely supportive. They might be a valuable first step to improve structure and organization of business processes.

All necessary functionality for this work is available in ProM. Yet, Process Model Discovery techniques and the Enhancement methods are separated over a couple of plug-in. In regard to future work, the approach described in this work could be bound together to one unit in a new plug-in. Furthermore, the next step could be to extend the integrated process model by additional perspectives like organizational or cost perspective. Calculating the approximate waiting and service time or a target time for every activity would enable further time related analysis. A scenario where time information is used to provide operational support in real time to control business processes is imaginable as well.

## REFERENCES

- AMELLER, D. AND FRANCH, X. 2008. Service level agreement monitor (salmon). *Composition-Based Software Systems, ICCBSS 2008*, 224-227.
- DE MEDEIROS, A. AND WEIJTERS, A. 2005. Genetic process mining. *Applications and Theory of Petri Nets 2005*, volume 3536 of *Lecture Notes in Computer Science*.
- FERGUSON, P. AND HUSTON, G. 1998. *Quality of service: delivering QoS on the Internet and in corporate networks*. Wiley New York.
- GÜNTHER, C. 2009. Xes standard definition. [http://www.xes-standard.org/\\_media/xes/xes\\_standard\\_proposal.pdf](http://www.xes-standard.org/_media/xes/xes_standard_proposal.pdf) (visited on 11/30/2012).
- GRIGORI, D., CASATI, F., CASTELLANOS, M., DAYAL, U., SAYAL, M., AND SHAN, M. 2004. Business process intelligence. *Computers in Industry* 53, 3, 32-343.
- KELLER, A. AND LUDWIG, H. 2003. The wsla framework: Specifying and monitoring service level agreements for web services. *Journal of Network and Systems Management* 11, 1, 57-81.
- MENDLING, J. , NEUMANN, G., AND VAN DER AALST, W. 2007. Understanding the occurrence of errors in process models based on metrics. *On the Move to Meaningful Internet Systems 2007: CoopIS, DOA, ODBASE, GADA, and IS*, 113-130.
- MOLINA-JIMENEZ, C., SHRIVASTAVA, S., CROWCROFT, J., AND GEVROS, P. 2004. On the monitoring of contractual service level agreements. *Proceedings of the First IEEE International Workshop on Electronic Contracting*, 1-8.
- SONG, M. AND VAN DER AALST, W. 2008. Towards comprehensive support for organizational mining. *Decision Support Systems* 46, 1, 300-317.
- TRIENEKENS, J., BOUMAN, J., AND VAN DER ZWAN, M. 2004. Specification of service level agreements: Problems, principles and practices. *Software Quality Journal* 12, 1, 43-57.
- TU/E. 2012. Prom 6. Published by: Process Mining Group, CS department, Eindhoven University of Technology - <http://www.promtools.org/prom6/> (visited on 11/30/2012).
- VAN BON, J. 2008. *Foundations in IT service management: Basierend auf ITIL*. Vol. 3. Van Haren Publishing.
- VAN DER AALST, W. 1998. The application of petri nets to workflow management. *Journal of circuits, systems, and computers* 8, 01.
- VAN DER AALST, W. 2011. *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer Berlin and Heidelberg.
- VAN DER AALST, W., ANDRIANSYAH, A., ALVES DE MEDEIROS, A., ARCIERI, F., BAIER, T., ET AL. 2012. Process mining manifesto. *BPM 2011 Workshops Proceedings*, 169-194.
- VAN DER AALST, W., VAN HEE, K., VAN WERF, J., AND VERDONK, M. 2010. Auditing 2.0: Using process mining to support tomorrow's auditor. *Computer* 43, 3, 90-93.
- VAN DER AALST, W., WEIJTERS, T., AND MARUSTER, L. 2004. Workflow mining: discovering process models from event logs. *IEEE Transactions on Knowledge and Data Engineering* 16, 9, 1128-1142.
- VAN DONGEN, B., DE MEDEIROS, A., VERBEEK, H. , WEIJTERS, A., AND VAN DER AALST, W. 2005. The prom framework: A new era in process mining tool support. *Applications and Theory of Petri Nets 2005*, 1105-1116.
- WEIJTERS, A., VAN DER AALST, W., AND DE MEDEIROS, A. 2006. Process mining with the heuristics miner algorithm. Technische Universiteit Eindhoven, Tech. Rep. WP 166.

