Towards Cross-Organizational Process Mining in Collections of Process Models and their Executions

J.C.A.M. Buijs, B.F. van Dongen, W.M.P. van der Aalst

Department of Mathematics and Computer Science,
Eindhoven University of Technology
P.O. Box 513, 5600 MB Eindhoven, The Netherlands
{j.c.a.m.buijs,b.f.v.dongen,w.m.p.v.d.aalst}@tue.nl

Abstract. Variants of the same process may be encountered in different organizations, e.g., any municipality will have a process to handle building permits. New paradigms such as Software-as-a-Service (SaaS) and Cloud Computing stimulate organizations to share a BPM infrastructure. The shared infrastructure has to support many processes and their variants. Dealing with such large collections of similar process models for multiple organizations is challenging. However, a shared BPM infrastructure also enables *cross-organizational process mining*. Since events are recorded in a unified way, it is possible to cross-correlate process models and the actual observed behavior in different organizations. This paper presents a novel approach to compare collections of process models and their events logs. The approach is used to compare processes in different Dutch municipalities.

Key words: cross-organizational process mining, software-as-a-service, process model collections, configurable process models

1 Introduction

More and more organizations will use a *Shared Business Process Management Infrastructure (SBPMI)*. The interest in Software-as-a-Service (SaaS) and Cloud Computing demonstrate that organizations want to share development and maintenance costs. Examples such as salesforce.com, Google Apps, NetSuite and Microsoft Online Services illustrate this. At the same time, organizations need to continuously improve their processes. Moreover, there is the need to support local variations of the same process. Often there are good reasons for differentiation between processes in different organizations, e.g., size of a municipality or local priorities may influence the way building permits are handled.

Configurable process models [2, 6] provide a way to model variability in the processes supported by an SBPMI. Given a shared configurable model, organizations can use different configurations to adapt to local needs. Current infrastructures such as salesforce.com hide these configurable models. Nevertheless, the processes supported by salesforce.com can be configured within predefined boundaries.

Existing research on process model collections, such as the Apromore [8] project, tends to focus on informal process models and does *not* consider the event logs of the corresponding processes. However, SBPMIs allow for the recording of event

Table 1: Metrics Example

	PM 1	PM 2	PM 3	PM 4	Average Throughput Time
Log 1	1.0	0.6	0.8	0.4	10 days
Log 2	0.8	0.9	0.7	0.4	40 days
Log 3	0.9	0.4	0.9	0.5	22 days
Log 4	0.9	0.5	0.8	0.8	16 days
Complexity	5	20	10	26	

logs in a unified manner across different organizations. Moreover, the process variants/configurations can be compared among one another and can be related to the actual behavior observed in event logs.

Process mining is an emerging discipline providing comprehensive sets of tools to provide fact-based insights and to support process improvements [1]. This new discipline builds on process model-driven approaches and data mining. Thus far the focus of process mining has been on process discovery and conformance checking *within one organization*. SBPMIs, however, enable *cross-organizational process mining*.

The availability of (a) process model collections, (b) organization specific variants, and (c) observed behavior recorded in event logs, generates interesting questions from the organizations‘ point of view:

1. Which organizations support my “behavior” with better process models?
2. Which organizations have better “behavior” which my process model supports?
3. Which set of organizations can I support with my process model?

Consider for instance Table 1, where the behavior of four organizations, recorded in event logs, is compared to the process models of these organizations. Furthermore, an example quality metric is depicted for each event log ($Log1 - Log4$) and process model ($PM1 - PM4$). This quality metric allows us to reason about “better” models and “better” behavior. Note that the approach is independent of the quality metrics selected. The ‘complexity’ metric shown in Table 1 indicates how ‘complex’ a certain process model is. For each recording of a process execution, or event log, the average time required to handle a single case is shown. A third viewpoint that can be taken is that of comparing a process model with recordings of process executions. In Table 1 we show the ‘fitness’ of an event log on a certain process model. The higher the fitness, the better the process model describes the behavior recorded in the event log. Besides the comparison between event logs and process models as shown in Table 1, other comparisons are also possible. Event logs can also be compared to the behavior of different organizations. In a similar way, the process models of organizations could also be compared. The metrics in Table 1 are only examples. Any metric that measures the quality of process models or event logs can be used. In a similar way, any metric that provides comparisons between business processes and/or event logs can be used.

Table 1 provides insights into the business processes, and their executions, of four organizations. For instance, organization 1 has the simplest process model (‘complexity’5) and handles a case in only 10 days. Furthermore, organization 1 never deviates from the modeled process, as is indicated by a fitness of 1 for event log 1.

Organizations 1 and 3 have the simplest process models, while the fitness of these models compared to the logs of organizations 2 and 4 is relatively high. The more complex process models of organizations 2 and 4 however have a low fitness for all organizations other than themselves. We might be tempted to suggest organization 2 to switch to a simpler process model to reduce the average case handling time. However, we do have to keep in mind that other factors might play a role here. It could be the case that organization 2 implements many checks to ensure a high-quality product while organization 1 performs less rigorous check on the products they deliver. This indicates that we need more than a single metric to be able to correctly advise organizations how they could improve their processes.

In this paper, we propose an approach for cross-organizational process mining. As discussed, this is highly relevant for emerging SBPMIs. Section 2 discusses metrics related to process models, process behavior and comparisons of these. In Section 3, we then show that with only a few metrics one can already provide valuable insights and we conclude the paper in Section 4.

2 Analyzing Process Models and Event Logs

In this section we discuss examples for three different types of metrics. We first briefly discuss *process model quality metrics* in Section 2.1, such as process model complexity. Next we mention *behavioral quality metrics* in Section 2.2 which are similar to the 'average throughput time' metric used as an example in Table 1. Finally, we discuss *comparison metrics* that can be of interest when comparing process models, process executions or combinations of these in Section 2.3.

2.1 Process Model Quality Metrics

Recently, the topic of process model complexity has attracted the attention of many BPM researchers. Many structural process model complexity metrics exist, ranging from simply counting the elements in the process model to more or less complex formulas to indicate process model complexity [9]. Besides structural metrics there are also quality metrics for behavior allowed by the process model. These metrics include soundness, density, separability, sequentiality, connector mismatch, cyclicity and concurrency [9, Chapter 4]. Furthermore, not all metrics are related to the structure or allowed behavior of the process model. Operational metrics such as resource cost or process model maintenance costs are also used.

In this paper, we use simple metrics which have proven to be good predictors of errors [9]. The general approach however does not depend on the selected metrics.

2.2 Performance Indicators (Log Metrics)

An event log records events that are relevant for a particular process. Each event corresponds to an execution of a certain activity for a certain case by a resource (e.g. employee or system) at a certain point in time. By using this information, many different metrics can be calculated. As was illustrated in Table 1, we can calculate the average time required for a case to be processed. This is visualized in Figure 1 using a dotted chart representation of the event log. In a dotted chart each dot represents a single event where the color of the dot indicates which activity was executed. Each row in the chart is a case and the horizontal axis is the time. In this case the dotted chart is sorted on case duration and the time is relative, e.g. x days after the case started. These settings clearly show the distribution of the case durations.

Other good performance indicators include:

1. Arrival rate of new cases over time;
2. Time a case spend waiting versus the time it was actually worked upon;
3. Average number of different activities per case.

Actually, most (key) performance indicators used in business process intelligence suites, for instance related to time, cost or quality, can be used. Think for instance of reaching service level agreements (SLAs), resource utilization or the number of failures.

For the purpose of this paper we simply focus on the average time required for a case to be processed in Section 3. Again, the approach does not depend on the chosen metrics.

2.3 Comparison metrics

Besides metrics related to a single process model or a single event log, there are also comparison metrics. One could for instance do a *Model-Model* comparison to detect whether the process models describe similar behavior [4, 5, 10].

Another comparison that can be done is a *Log-Log* comparison. The behavior can for instance be used to discover a process model. The resulting process model can then be compared with other records of behavior or with the prescribed process model.

Another comparison that can be done is a *Log-Model* comparison. This type of comparison is often used to test the level of conformance of the process execution with respect to the process model. Most algorithms can also visualize where the process execution deviated from the prescribed process model. An example of such fitness metric is the cost-based fitness metric [3]. This metric calculates the fitness of the process execution with respect to the process model. It does so by assigning costs for skipping or inserting activities in the process model in order to be able to correctly replay the recorded behavior. Part of such a comparison is shown in Figure 2 which shows the different traces found in the event log and where they deviate from the process model.

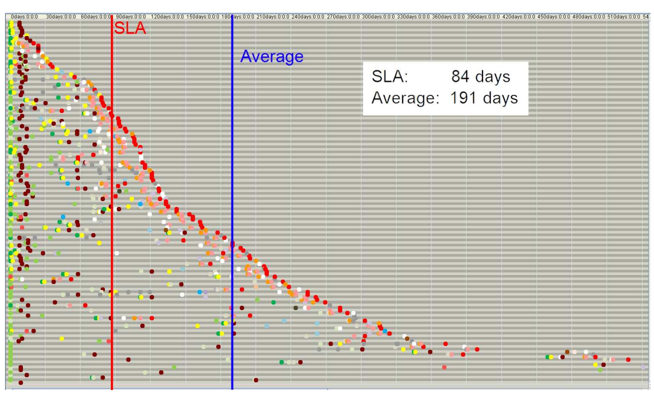


Fig. 1: Example of time metrics visualized on a dotted chart.

Other metrics that calculate fitness are the token-based fitness metric [1, 13], the hidden Markov models' event, trace and model fitness metrics [14], the completeness metric [7] and the continuous parsing measure [15].

A metric related to the fitness metric is behavioral precision [13]. This metric indicates how precisely a process model describes the recorded process executions. A high behavioral precision indicates that the process model does not allow for more behavior than seen in the event log. The 'behavioral appropriateness' metrics [13] keep track of the transitions that are enabled during the replay of the event log on the process model. The more transitions that are enabled at once, the more behavior is allowed and therefore the behavioral appropriateness is reduced. Other behavioral precision metrics are the precision metric of [12] and the ETC$_{precision}$ metric discussed in [11].

When comparing process models and/or behavior, it is very important to take the *vocabulary* into account. For instance, in the Apromore process repository [8] different process models can describe a similar process while using completely different vocabularies. Even though some techniques exist to (automatically) map activities between process models with different vocabularies [4], this remains a difficult task which is error-prone. Since in a SBPMI environment the process models are configurations, they share a common vocabulary.

Even in a SBPMI environment the variants of a given process model may use different sets of activities. Note that different configurations may result in processes of very different sizes. Because the overlap of vocabulary influences the comparison results of most metrics, the overlap should always be taken into account when interpreting the comparison metrics.

Fig. 2: Fitness analysis for the process model of Municipality 1 ($PM1$) and an event log of the same municipality ($Log1$).

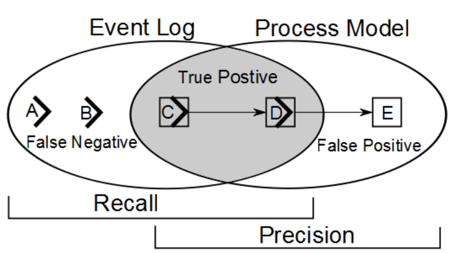


Fig. 3: Precision and recall measures for process models describing behavior in event logs

To calculate the overlap of activities we use the *precision* metric. Precision indicates the fraction of correct results in the result set. We define precision as the number of activities in both the process model and the event log divided by the total number of activities in the process model as is formally defined in Equation 1.

$$Precision = \frac{\#True\ Positive}{\#True\ Positive + \#False\ Positive} \tag{1}$$

Figure 3 shows the application of precision in the context of process models and event logs. In this example the precision is $\frac{2}{3}$ since there are 2 activities in both the process model and the event log while the process model contains 3 activities in total. Intuitively, precision indicates the extent to which the activities of the process model occur in the event log. A high precision therefore indicates that many of the activities in the process model are also present in the event log. A metric related to precision, recall, indicates which fraction of the events in the event log are also covered by the process model. This is however less important when replaying event logs on process models. If the precision is low, this means that many activities in the process model have no corresponding event in the event log. During fitness calculation these 'free' activities cause a higher fitness than if they were bound to an event in the event log.

3 Cross-Organizational Process Mining in Dutch Municipalities

In the previous section we described which metrics can be used to compare process models and their executions between multiple organizations in a SBPMI environment. In this section we illustrate how to apply a selection of these metrics to several real life examples. To measure the quality of a process model we count the number of tasks and routing elements in the process model. As a quality measure for the behavior we calculate the average flow time of a case. Furthermore, we compare the process model with the recorded behavior using three metrics: precision, cost-based fitness and behavioral appropriateness. These simple metrics allow us to provide answers to questions such as the ones listed in Section 1.

The real life examples come from the CoSeLoG research project[1]. In the CoSeLoG project we investigate how 10 Dutch municipalities execute their processes. Note that municipalities need to support a common set of processes, e.g. requests for passports, handling of taxes and citizen registration. Therefore, different groups of Dutch municipalities are working towards a common SBPMI. For two of the three largest processes in the CoSeLoG project we selected four municipalities that use the same type of information system. This allows us to compare the process executions between these municipalities. Since each municipality starts from the same default process model, the implementation of activities with equal names is the same. In the following we discuss the comparison of these two processes across the municipalities.

3.1 Process 1: Building Permits

The first process we investigate is a process that handles building permits. The four municipalities from which we got the data actually collaborated during the definition of the process model and the implementation of the supporting information system. At

[1] See `http://www.win.tue.nl/coselog`

Table 2: Process model complexity metrics for process 1

	Activities	AND		XOR	
		splits	joins	splits	joins
PM 1	28	2	3	5	4
PM 2	26	1	1	4	4
PM 3	24	2	2	4	4
PM 4	26	2	2	3	4

Table 3: Throughput time metrics for process 1

	Average Throughput Time	C.V.	SLA
Log 1	190d 20h	0.9489	0.2624
Log 2	112d 17h	0.9900	0.4470
Log 3	267d 04h	1.6423	0.2787
Log 4	73d 23h	0.7215	0.8191

a certain point in time they continued individually. Each municipality uses a separate instance of the information system installation. Despite this common set-up and the fact that the process boundaries are given by legal requirements, we can clearly see that the system is used in different ways by different municipalities.

The process models of the four municipalities are shown globally in Figure 4. Larger versions of the process models are attached as appendices. Table 2 displays structural process model quality metrics. First, the number of different activities in the process model is listed. The last four columns show the number of AND and XOR splits and joins. Verification using the Woflan plug-in in ProM shows that each process model is structurally correct. Looking at the metrics in Table 2 we can see that the process models are similar in complexity.

Table 3 shows the average throughput time as a performance indicator for the event logs. The coefficient of variation indicates the variability, i.e. the deviation from the mean. All coefficients of variation are rather large, e.g $M3$ (municipality 3) has a coefficient of variation of more than 1.5. This indicates that all municipalities have cases that take exceptionally long. The process time of municipality 4 is significantly less than for the other municipalities. More detailed analysis of the event log revealed that a lot of the cases where only recorded in the system but no further actions were recorded. The third performance indicator shown in Table 3 is the percentage of cases that is handled within 12 weeks which is a service level requirement set by law. Note that cases can be put on hold when not enough data is present. Furthermore, the municipality can extend the deadline once for each case. This is not taken into account when calculating the metric.

Finally, Table 4 shows the *Log-Model* comparison metrics results. Specifically, Table 4a shows the calculated precision, indicating the amount of overlap in the vocabularies. Table 4b shows the cost-based replay fitness and Table 4c shows the behavioral appropriateness values. Looking at the precision metrics in Table 4a we see a precision of 1.000 on the diagonal. This can easily be explained since the vocabularies of a process model and its event log are equal. From the precision values we can also conclude that $Model2$ and $Model3$ contain only activities that are also present in $Log1$. This is indicated by the precision values of 1.000 for $Log1$ compared with $Model2$ and $Model3$. Given that the vocabulary of $Model1$ is equal to that of $Log1$, the same observation holds for $Model1$ compared to $Model2$ and $Model3$. However, $Model1$ does contain activities that are not present in $Log2$ and $Log3$. This can be observed by the precision values of 0.9286 and 0.8571 when comparing $Log2$ and $Log3$ with $Model1$.

This indicates that *M2* and *M3* execute a subset of the activities of *M1*. Given the fact that all precision values are rather high this indicates that there is a large overlap of activities between municipalities. Therefore we can also take the fitness and behaviorial appropriateness values into account.

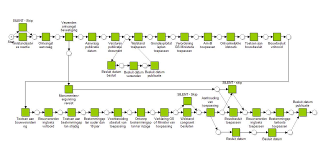
If we look at the cost-based replay fitness values in Table 4b, we see that *Model3* has a high fitness for all event logs. We see that the cost-based fitness for *Model3* is highest for *Log1*, with a fitness value of 0.9021. The fitness value when replaying *Log3* on *Model3* is the lowest fitness for *Model3* with 0.8202. The cause for this low fitness can be twofold: first, if some activities in the process model are not mapped to events in the event log, the fitness will go up. Since all activities in *Model3* have a corresponding event in *Log3*, the fitness value will be lower since more activities are taken into account. A second explanation is that the behavior contained in *Log3* is not very structured. This is supported by the low fitness values of *Log3* on the other process models.

Table 4c shows the behavioral appropriateness. Recall that a low behavioral appropriateness indicates that the process model allows for more behavior than what was seen in the event log. We see that *Model1* and *Model2* have a high behavioral appropriateness value of at least 0.9467 for all event logs. When we take a closer look at the process models, as shown in Figure4, we see that *Model1* and *Model2* are very sequential, they don't allow much variation. *Model3* contains three parallel paths and therefore allows for more behavior. The behavioral appropriateness values for *Model3* are still rather high, especially for *Log1* and *Log3*. *Model4* seems to allow even more behavior as is indicated by behavioral appropriateness values as low as 0.7748.

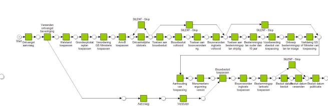
Table 2, Table 3 and Table 4 can be combined into Table 5 to create a table similar to Table 1. The three comparison metrics are combined into a single cell in Table 5. The value in the middle, aligned to the left, is the precision. The value in the top of each cell is the cost-based fitness and the bottom value is the behavioral appropriateness.

Using Table 5 we can answer the following questions from Section 1:

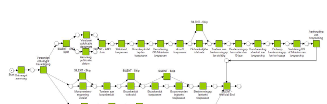
1. *Which organizations support my behavior with better process models?*
   For municipalities 1 and 2 the process model of municipality 3 describes their process behavior rather well while still covering most of the activities. The pro-



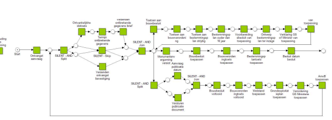
(a) Process Model 1



(b) Process Model 2



(c) Process Model 3



(d) Process Model 4

Fig. 4: Process models for process 1 (for a larger version please refer to the Appendix)

Table 4: Process 1 Comparison metrics

	PM 1	PM 2	PM 3	PM 4
Log 1	1.0000	1.0000	1.0000	0.9231
Log 2	0.9286	1.0000	1.0000	0.9231
Log 3	0.8571	0.9231	1.0000	0.8462
Log 4	0.8571	0.9231	0.9167	1.0000

(a) Precision

	PM 1	PM 2	PM 3	PM 4
Log 1	0.8268	0.7788	0.9021	0.7232
Log 2	0.7611	0.8404	0.8300	0.7398
Log 3	0.7048	0.7045	0.8202	0.6920
Log 4	0.8288	0.7892	0.8642	0.8636

(b) Cost-Based replay fitness

	PM 1	PM 2	PM 3	PM 4
Log 1	0.9487	0.9915	0.9740	0.8735
Log 2	0.9662	0.9943	0.8990	0.7968
Log 3	0.9799	0.9929	0.9415	0.8882
Log 4	0.9718	0.9467	0.9047	0.7748

(c) Behavioral Appropriateness

Table 5: Combined Metrics for Process 1

	PM 1	**PM 2**	**PM 3**	**PM 4**	**Average Throughput Time**	**C.V.**	**SLA**
Log 1	0.8268 1.0000 0.9487	0.7788 1.0000 0.9915	0.9021 1.0000 0.9740	0.7232 0.9231 0.8735	190d 20h	0.9489	0.2624
Log 2	0.7611 0.9286 0.9662	0.8404 1.0000 0.9943	0.8300 1.0000 0.8990	0.7398 0.9231 0.7968	112d 17h	0.9900	0.4470
Log 3	0.7048 0.8571 0.9799	0.7045 0.9231 0.9929	0.8202 1.0000 0.9415	0.6920 0.8462 0.8882	267d 04h	1.6423	0.2787
Log 4	0.8288 0.8571 0.9718	0.7892 0.9231 0.9467	0.8642 0.9167 0.9047	0.8636 1.0000 0.7748	73d 23h	0.7215	0.8191
Activities	28	26	24	26			
AND split/join	2/3	1/1	2/2	2/2			
XOR split/join	5/4	4/4	4/4	3/4			

cess model of municipality 3 is equally complex as that of municipalities 1 and 2. Therefore, these municipalities might want to investigate the process model of municipality 3.

2. *Which organizations have better behavior which my process model supports?* When we take the viewpoint of municipality 3 then municipalities 1 and 2 show behavior supported by their process model. If we look at the average throughput time of a case then municipalities 1 and 2 perform much better. So, municipality 3 might want to look at how municipalities 1 and 2 execute their process.

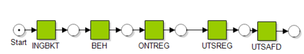
3. *Which set of organizations can I support with my process model?*
   When the process model of municipality 3 is extended with a couple of activities then the processes of municipalities 1 and 2 can also be supported. The process of municipality 4 could also be supported by this process model but that would require more changes.

3.2 Process 2: Housing Tax

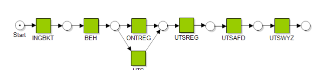
Another process investigated in the CoSeLoG project is that of handling citizen complaints on housing tax. Since these complaints arrive in a six week period every year, this is an interesting process to investigate. The four process models are shown globally in Figure 7. Table 6 shows the same metrics as we used for process 1. The three columns on the right provide quality metrics on the event logs. The bottom three rows show quality metrics for the process models. In the center of the table the comparison metrics are shown, on the top of each cell the fitness between the process model and the event log is shown. On the bottom of each cell the behavioral appropriateness is shown. The value in the middle, slightly aligned to the left, indicates the precision of the process model with respect to the event log.

Using the combined metrics of Table 6 we can now again answer a selection of the questions as proposed in Section 1:

1. *Which organizations support my behavior with better process models?*
   The municipalities can be divided in two groups, according to the comparison values. Municipalities 1 and 2 execute similar activities, as can be observed by the high precision values. Municipalities 3 and 4 also form a group, even though the precision values between these municipalities are 0.5000 and 0.4667. The fitness value of replaying event log 4 on process model 3 is rather high. So the process of municipality 4 can be supported by the process model of municipality 3, after adding the missing activities. However, the process model of municipality 3 is more complex than that of municipality 4.
2. *Which organizations have better behavior which my process model supports?*
   The process model of municipality 3 supports the behavior of municipality 4. If we look at the average throughput time of a case then we see that municipality 4 handles a case quicker than municipality 3. Municipality 3 therefore might want to look at the process of municipality 4 to improve the throughput times.



(a) Process Model 1



(b) Process Model 2


(c) Process Model 3

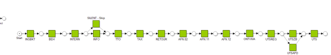

(d) Process Model 4

Fig. 5: Process models for process 2 (for a larger version please refer to the Appendix)

Table 6: Combined Metrics for Process 2

	PM 1	PM 2	PM 3	PM 4	Average Throughput Time	C.V.	SLA
Log 1	1.0000 1.0000 1.0000	1.0000 0.7143 0.6667	1.0000 0.2857 0.2500	1.0000 0.2667 1.0000	22d 20h	3.6007	0.9697
Log 2	0.9705 1.0000 1.0000	0.8850 1.0000 0.8750	0.8963 0.3571 0.3333	0.8210 0.3333 1.0000	110d 09h	1.0206	0.9522
Log 3	0.4853 0.8000 1.0000	0.4034 0.7143 0.8750	0.9155 1.0000 0.9167	0.5253 0.4667 1.0000	227d 17h	0.3813	0.7014
Log 4	0.9918 0.8000 1.0000	0.8124 0.7143 0.6667	0.9145 0.5000 0.9167	0.9373 1.0000 1.0000	120d 10h	0.6614	0.9861
Activities	5	7	14	15			
AND split/join	0/0	0/0	0/0	0/0			
XOR split/join	0/0	1/1	3/3	2/2			

3. *Which set of organizations can I support with my process model?*
   The set of municipalities 1 and 2 can best be supported by the process model of municipality 1. The process model of municipality 1 does need to be extended with 2 activities to fully support the process.
   For municipalities 3 and 4 the process model of municipality 3 seems the best candidate. Given the precision of only 5.000 several activities need to be added to this process model to fully support the process of municipality 4.

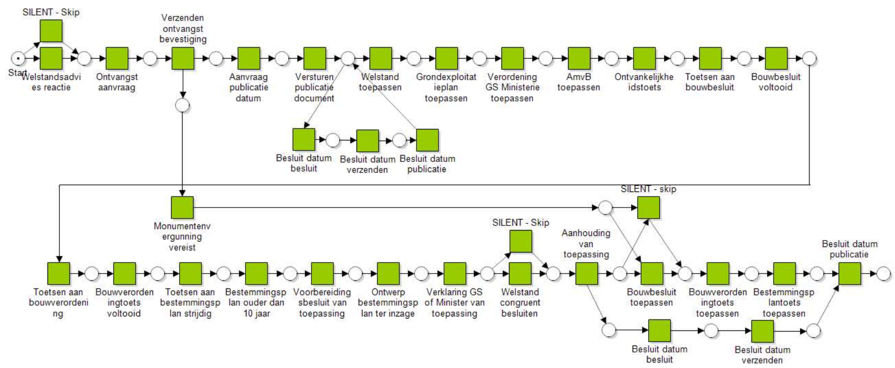
4 Conclusion

Until now process mining efforts focussed on analyzing a process within a single organization. In this paper, we propose an approach for the comparison of process models and process executions between organizations. Emerging SaaS and Cloud infrastructures stimulate organizations to share a common BPM infrastructure (SBPMI). As a result large collections of process model variants and their execution histories become readily available. One of the challenges for SBPMIs is that they should be able to support different process variations through configuration. By utilizing the possibilities of configurable process models, different variations of a process model can be supported. At the same time this ensures a consistent set of activities in the process model and their executions. This allows for easy comparison of the process models and their executions between organizations. By comparing organizations we can suggest improvements.

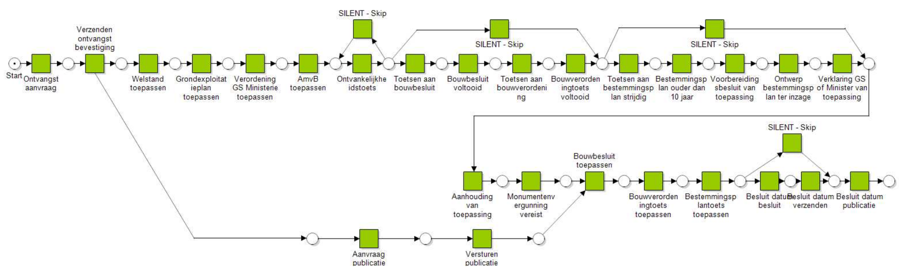
Process mining is typically used to gain insights into processes in a single organization. The SBPMI setting allows for cross-organizational process mining, i.e., suggesting improvements for different organizations based on facts/comparisons of process models and event logs across organizations. Three types of metrics can be used: metrics related to process models, metrics related to process executions, and metrics for comparing process models and/or process executions. We presented specific examples for each type of metric. However, the approach is generic and allows the use of any metric. As an example we used a small set of simple metrics to analyse two sets of process executions across municipalities. We showed that even simple metrics provide valuable insights on how to improve processes.

References

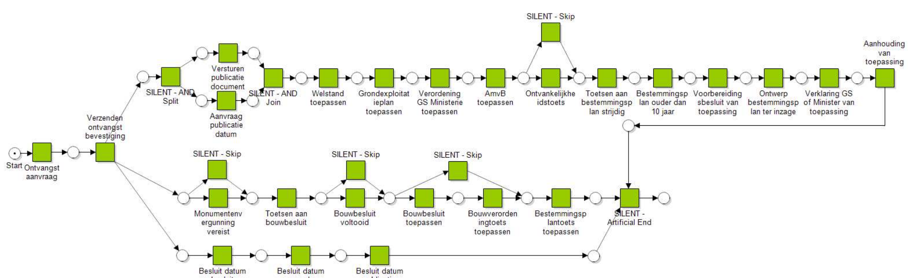
1. W.M.P. van der Aalst. *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer-Verlag, Berlin-Heidelberg, 1st edition, 2011.
2. W.M.P. van der Aalst, M. Dumas, F. Gottschalk, A.H.M. ter Hofstede, M. La Rosa, and J. Mendling. Preserving Correctness during Business Model Configuration. *Formal Aspects of Computing*, 22(3-4):459 – 482, May 2010.
3. A. Adriansyah, N. Sidorova, and B.F. van Dongen. Cost-based Fitness in Conformance Checking. *11th International Conference on Application of Concurrency to System Design (ACSD 2011), IEEE*, 2011.
4. R.M. Dijkman, M. Dumas, B.F. van Dongen, R. Käärik, and J. Mendling. Similarity of Business Process Models: Metrics and Evaluation. *Information Systems*, 36(2):498 – 516, 2011. Special Issue: Semantic Integration of Data, Multimedia, and Services.
5. B.F. van Dongen, R.M. Dijkman, and J. Mendling. Measuring Similarity between Business Process Models. In Zohra Bellahsne and Michel Lonard, editors, *Advanced Information Systems Engineering*, volume 5074 of *Lecture Notes in Computer Science*, pages 450–464. Springer Berlin / Heidelberg, 2008.
6. F. Gottschalk, T. Wagemakers, M.H. Jansen-Vullers, W.M.P. van der Aalst, and M. La Rosa. Configurable Process Models: Experiences from a Municipality Case Study. *Advanced Information Systems Engineering*, pages 486–500, 2009.
7. G. Greco, A. Guzzo, L. Pontieri, and D. Sacca. Discovering Expressive Process Models by Clustering Log Traces. *IEEE Transactions on Knowledge and Data Engineering*, 18:1010–1027, August 2006.
8. M. La Rosa, H.A. Reijers, W.M.P. van der Aalst, R.M. Dijkman, J. Mendling, M. Dumas, and L. García-Bañuelos. APROMORE: An Advanced Process Model Repository. *Expert Systems with Applications*, 38:7029–7040, June 2011.
9. J. Mendling. *Metrics for Process Models: Empirical Foundations of Verification, Error Prediction, and Guidelines for Correctness*, volume 6 of *Lecture Notes in Business Information Processing*. Springer, 2008.
10. J. Mendling and W.M. P. Van Der Dongen, B.F. Aalst. On the Degree of Behavioral Similarity between Business Process Models. In *Geschäftsprozessmanagement mit Ereignisgesteuerten Prozessketten*, pages 39–58, 2007.
11. J. Muñoz Gama and J. Carmona. A fresh look at precision in process conformance. In *Proceedings of the 8th international conference on Business process management*, BPM'10, pages 211–226, Berlin, Heidelberg, 2010. Springer-Verlag.
12. S.S. Pinter and M. Golani. Discovering workflow models from activities' lifespans. *Computers in Industry*, 53(3):283 – 296, 2004. Process / Workflow Mining.
13. 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.
14. A. Rozinat, M. Veloso, and W.M.P. van der Aalst. Using hidden Markov models to evaluate the quality of discovered process models. Technical Report BPM Center Report No. BPM-08-10, Eindhoven: BPMcenter.org, 2008.
15. A.J.M.M. Weijters, W.M.P. Van Der Aalst, and A.K. Alves De Medeiros. Process Mining with the Heuristics Miner-Algorithm. Technical report, Eindhoven University of Technology, 2006.



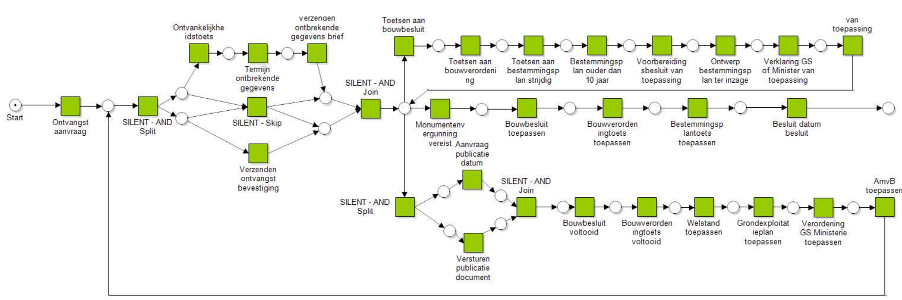
(a) Process Model 1



(b) Process Model 2

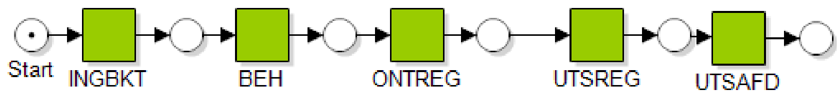


(c) Process Model 3

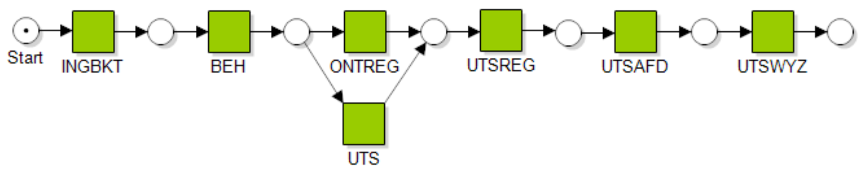


(d) Process Model 4

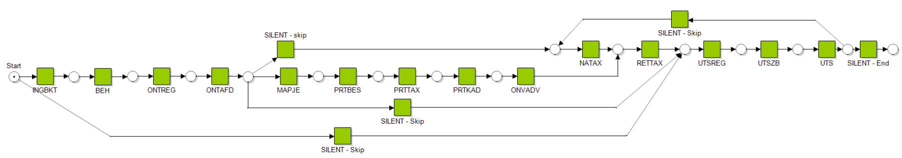
Fig. 6: Larger versions of the process models for process 1



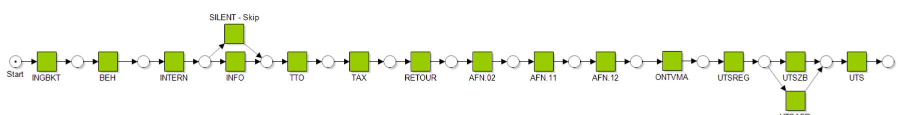
(a) Process Model 1



(b) Process Model 2



(c) Process Model 3



(d) Process Model 4

Fig. 7: Larger versions of the process models for process 2