

Name Organization

Experiment merging event logs

In this experiment we ask you to merge two event logs using a ProM plug-in and to afterwards answer the questions in attachment.

We would appreciate if you work accurately and provide an honest answer to the questions.

Case description

The even log *Sales Order* was extracted from the accounting software and contains activities at the level of a sales order. The possible activities in this process are displayed below.

- 'creation': contains data about the creation of the order (date, customer, delivery address and currency)
- 'shipment': contains extra data about delivery (date and logistical partner)
- 'sent invoice': contains data about sent invoices (date and invoice address)
- 'invoice paid': contains the date the invoice was paid

In principle, these activities can only occur once per process execution, but we cannot guarantee that the file does not contain errors.

The event log *Sales Order Lines* was extracted from the software in the warehouse and contains information about ordered products. The information that the log contains is displayed below.

- 'creation': contains data about the creation of a specific article order (article, price, amount, order number, date)
- 'modification': contains data about the modification of article orders (new price and /or amount, date of change)
- 'price calculation': contains the date the total price was calculated

In principle, each product order should first be created; afterwards it can be changed, but this is not mandatory; and finally the total price is calculated. There should be no changes after the total price is calculated. We cannot guarantee that the file does not contain errors.

Exercise description

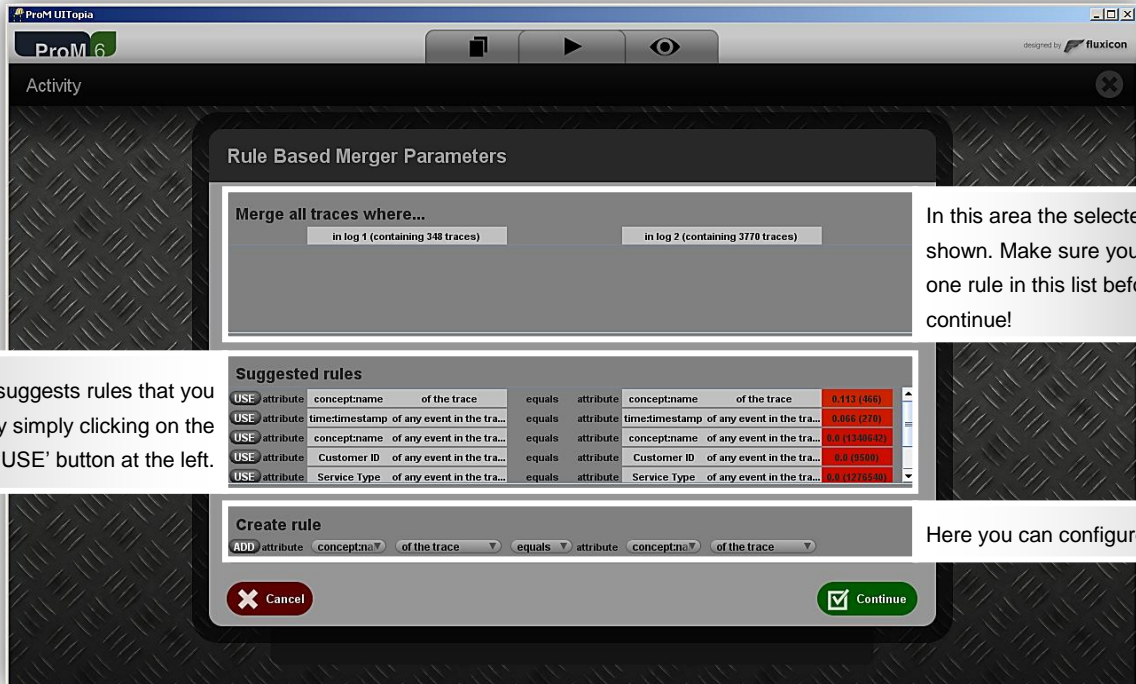
The assignment is to merge both event logs based on the description of a ProM plug-in below and to generate a process model using Heuristics Miner. Examine the process model to be able to critically analyze the merge. If you have any concerns about the merge, write it down in the appropriate box on the question sheet.

Thanks for your cooperation!

Jan Claes
Room 120.029
Tweekerkenstraat 2
9000 Gent
Belgium
jan.claes@ugent.be

Plug-in description

The plug-in is called **Merge two Event Logs using a rule based algorithm** and requires two event logs as input. When you start the plug-in, ProM will ask you to configure the following information.



Rule Based Merger Parameters

Merge all traces where...

in log 1 (containing 348 traces) in log 2 (containing 3770 traces)

Suggested rules

USE	attribute	conceptname	of the trace	equals	attribute	conceptname	of the trace	8.113 (466)
USE	attribute	timestamp	of any event in the tra...	equals	attribute	timestamp	of any event in the tra...	0.666 (270)
USE	attribute	conceptname	of any event in the tra...	equals	attribute	conceptname	of any event in the tra...	6.0 (1349942)
USE	attribute	Customer ID	of any event in the tra...	equals	attribute	Customer ID	of any event in the tra...	6.0 (9580)
USE	attribute	Service Type	of any event in the tra...	equals	attribute	Service Type	of any event in the tra...	6.0 (1276596)

Create rule

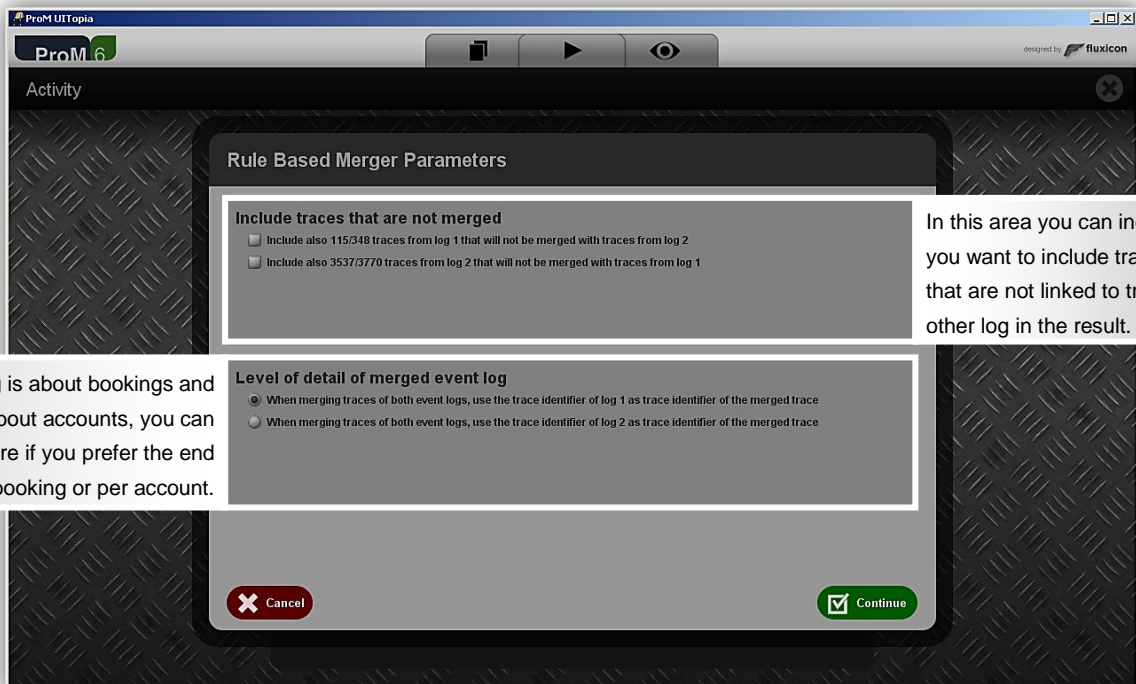
ADD attribute conceptna of the trace equals attribute conceptna of the trace

Cancel Continue

In this area the selected rules are shown. Make sure you have at least one rule in this list before you press continue!

This area suggests rules that you can use by simply clicking on the 'USE' button at the left.

Here you can configure rules manually



Rule Based Merger Parameters

Include traces that are not merged

Include also 115/348 traces from log 1 that will not be merged with traces from log 2

Include also 3537/3770 traces from log 2 that will not be merged with traces from log 1

Level of detail of merged event log

When merging traces of both event logs, use the trace identifier of log 1 as trace identifier of the merged trace

When merging traces of both event logs, use the trace identifier of log 2 as trace identifier of the merged trace

Cancel Continue

In this area you can indicate whether you want to include traces of one log that are not linked to traces of the other log in the result.

When one log is about bookings and the other is about accounts, you can determine here if you prefer the end result to be per booking or per account.

Questions

I experienced the following problems while trying to use the plug-in:

	Strongly disagree	Disagree	Neutral	Agree	Strongly agree
Q1. I found the procedure for applying the method complex and difficult to follow	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q2. I believe that this method would reduce the effort required to merge event logs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q3. Merging event logs using this method would be more difficult for users to understand	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q4. Overall, I found the method difficult to use	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q5. This method would make it easier for users to bring together process mining data of different sources	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q6. I found the method easy to learn	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q7. Overall, I found the method to be useful	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q8. Using this method would make it more difficult to decide which data should be used to determine how both event logs should be merged	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q9. I found it difficult to apply the method to the example data	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q10. I would definitely not use this method to merge event logs for process mining	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q11. I found the rules of the method clear and easy to understand	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q12. Overall, I think this method does not provide an effective solution to the problem of merging event logs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q13. Using this method would make it easier to merge event logs quickly	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q14. I am not confident that I am now competent to apply this method in practice	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q15. Overall, I think this method is an improvement to the process mining tool set	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q16. I intend to use this plug-in in preference to other techniques I know for merging event logs (e.g., using Excel functions)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>