PROOF-OF-CONCEPT APPLICATION

1.1 Tools

We implemented a Java plug-in within ProM framework as a proof of concept of our approach. We utilized the following tools for our implementation:

- **ProM (Process Mining framework)**: ProM is a framwork that supports many different process mining techniques in the form of plug-ins. It is developed in Java, therefore it is platform independent.
- **Eclipse IDE for Java Developers**: Since ProM is a framework developed in Java, therefore we implemented a plug-in for ProM framework using Java.
- **MySQL database**: Data related to process event logs and matching values of activities are stored in MySQL database.
- **mysql-connector-java**: Handle java connectivity with MySQL database.

1.2 Environment installation

First of all, the following applications and tools need to be install:

1. **Java SDK**
2. **Eclipse IDE**
3. **MySQL database**

1.3 Database schema

Create new databases based on the following schema from:

1. http://www-inf.it-sudparis.eu/SIMBAD/tools/LogRec/src/process_event_logs_empty.sql
   (This schema contains no data)
2. http://www-inf.it-sudparis.eu/SIMBAD/tools/LogRec/src/process_event_logs_dataset.sql
   (This schema contains data from the experimentation)

1.4 XES dataset

Download the folowing xes dataset from the given URLs, then extract them into the local disk:

1. http://www-inf.it-sudparis.eu/SIMBAD/tools/LogRec/src/xes-simple.zip
   (This dataset contains event logs for only 2 processes, suitable for testing)
2. http://www-inf.it-sudparis.eu/SIMBAD/tools/LogRec/src/xes.zip
   (This dataset contains event logs for 719 processes from the experiment)

1.5 Eclipse projects

Download the folowing Eclipse projects from the given URLs, then import them into Eclipse workspace:

1. **Recommender**: An application developed as a plug-in within ProM. Since this project contain ProM framework already, therefore ProM is not required to install seperately.
   http://www-inf.it-sudparis.eu/SIMBAD/tools/LogRec/src/Recommender.zip
2. **EventLogDatabase**: This is an interface to handle MySQL database requests from **Recommender** application.
   http://www-inf.it-sudparis.eu/SIMBAD/tools/LogRec/src /EventLogDatabase.zip

1.6 Configuration

Before running the application, **Recommender\config.ini** need to be configured. Multiple database sources can be define as the following properties, where N is the identifier number of a data source.

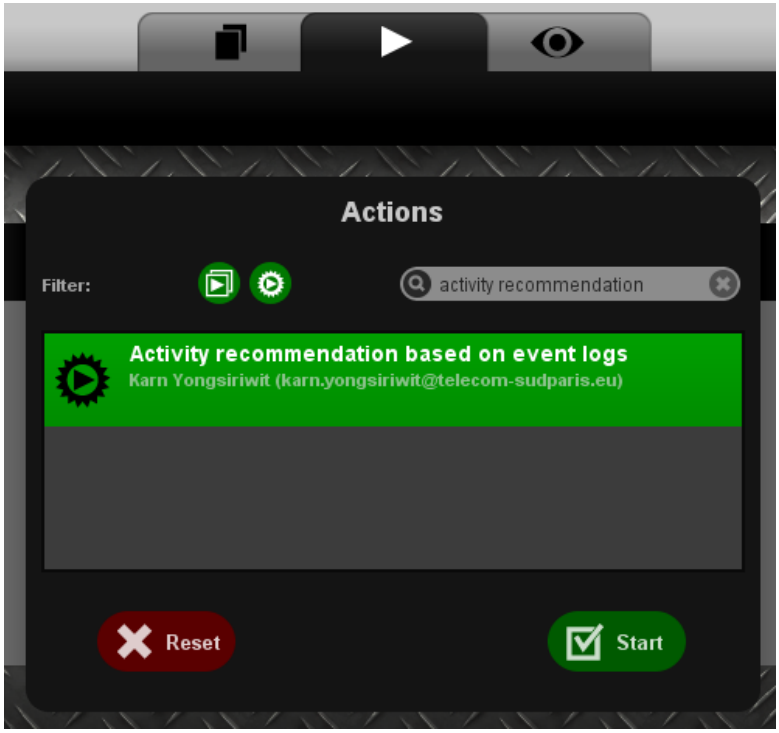
```
DataSourceHostname_1=localhost
DataSourcePort_1=3306
DataSourceUserName_1=root
DataSourcePassword_1=123456
DataSourceDbName_1=process_event_logs

DataSourceHostname_2=localhost
DataSourcePort_2=3306
DataSourceUserName_2=root
DataSourcePassword_2=123456
DataSourceDbName_2=process_event_logs_dataset

...
...
...

DataSourceHostname_N=157.159.110.224
DataSourcePort_N=3306
DataSourceUserName_N=karn
DataSourcePassword_N=karn
DataSourceDbName_N=karn_db
```

Moreover, a default path for XES dataset that will be use in the application shall be defined with **EventLogsDir** property in the ini file.

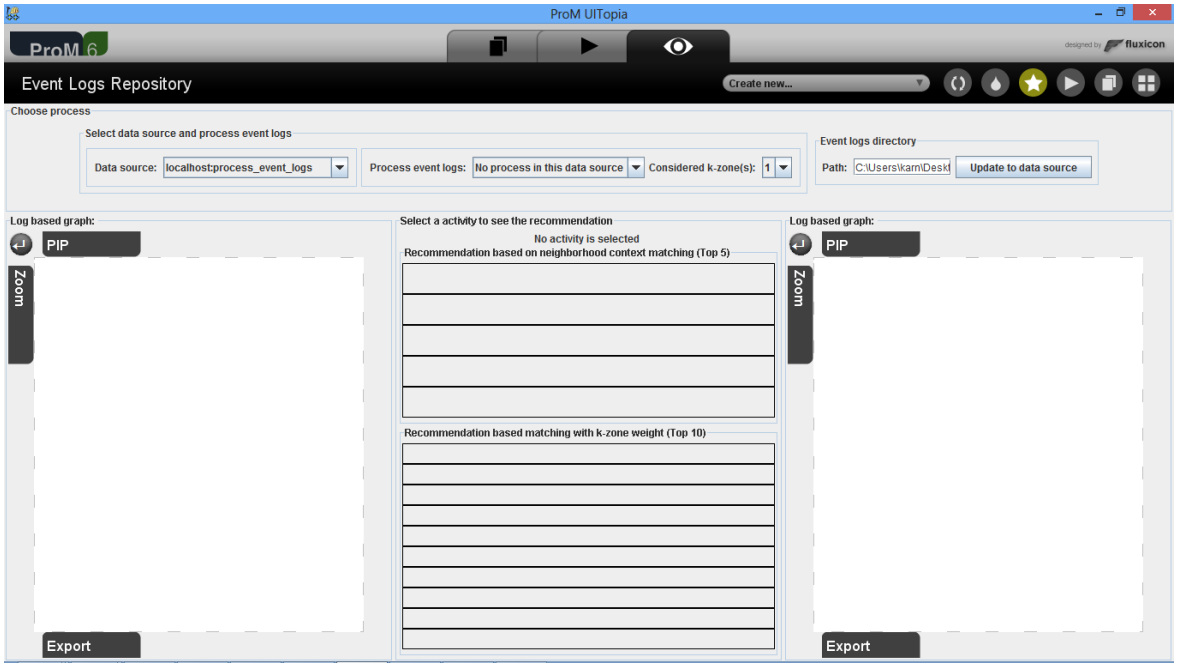
1.7 Demonstrating an application

The application can be demonstrated by the following steps:

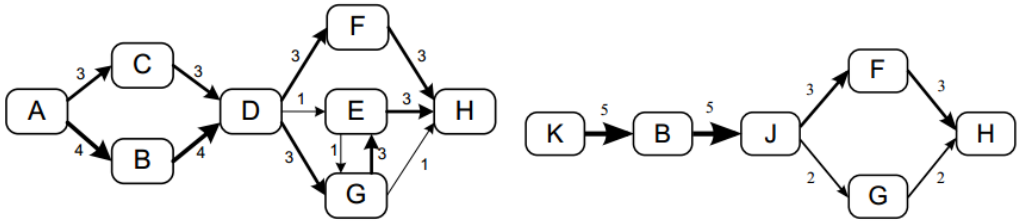
1. Within Eclipse IDE, navigate to **Recommender** project.
2. Right-click on **ProM with UITopia (Recommender).launch** within the project, select **Run As**, then choose **ProM with UITopia (Recommender).launch**.
3. Finally, ProM will be started. Navigate to **Action** pane then select ProM plug-in called **Activity recommendation based on event logs** as shown below:



4. After that, the GUI will be shown up as below:

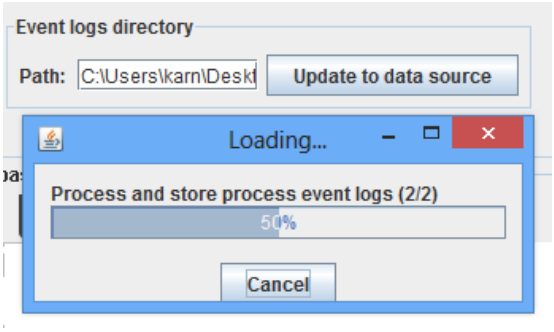


5. Assume that there is no data in the data source. This instuction will demonstrate with the **xes-simple** dataset that contains event logs for only 2 processes as shown below:

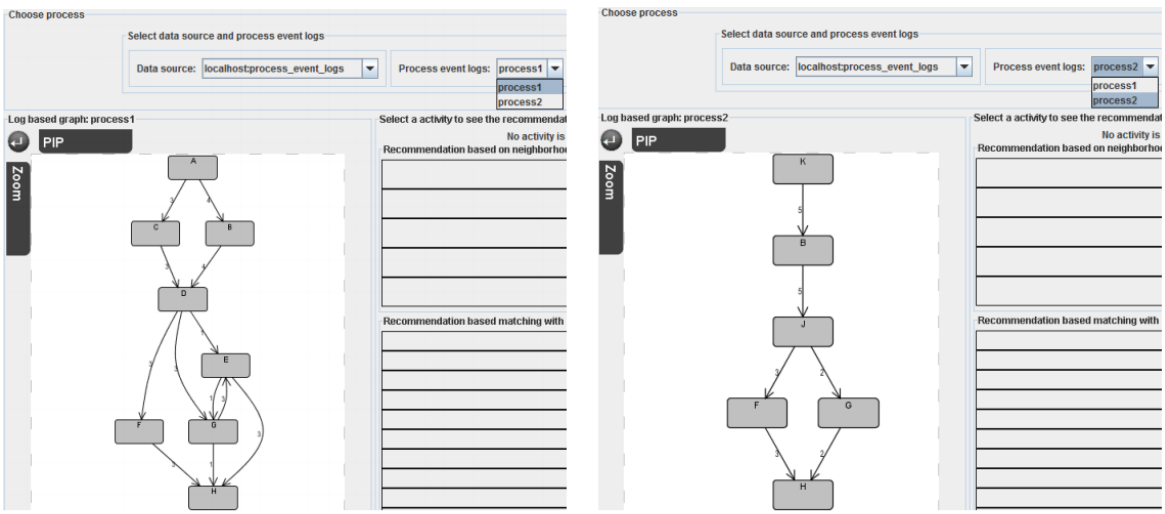


(a) (b)

6. Suppose that the path in the text field for **Event logs directory** panel in thoses contains process event logs. We can compute the neighborhood matching values and store them into data source by clicking the **Update to data source** button, the **Loading** frame will be shown up. After finishing the computation with the data related to matching values are stored, the **Loading** frame will be disappeared.

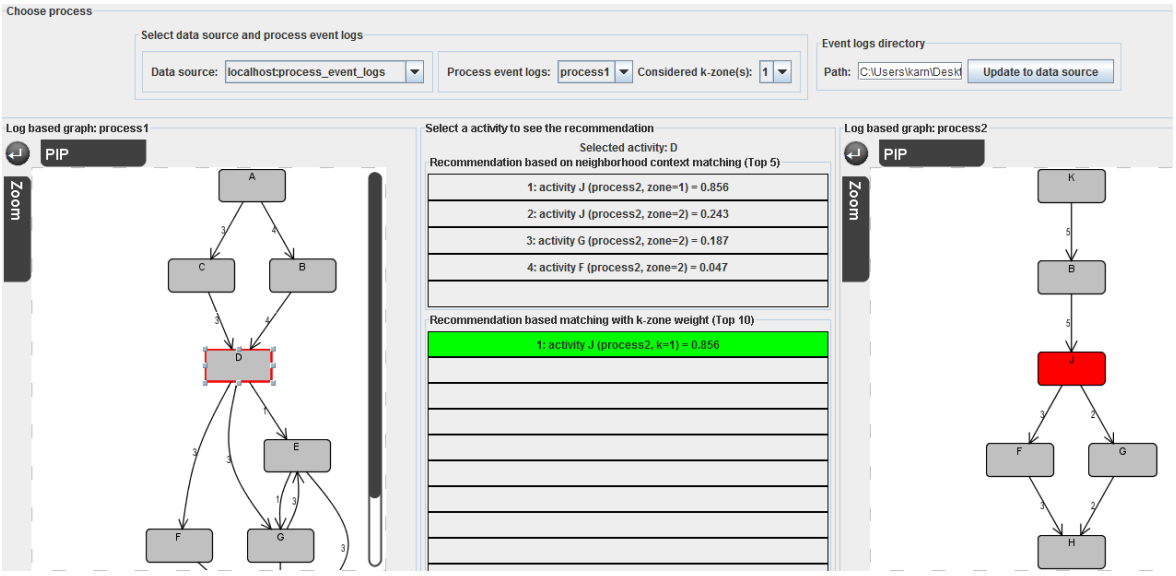


7. Currently, two process event logs with the matching values for each activity are stored in the data source. With the Process event logs combo box, the names of two processes are listed. We can select a process in this combo box to see the visualization of process event log 1 and 2 in log-based process graph as shown below:

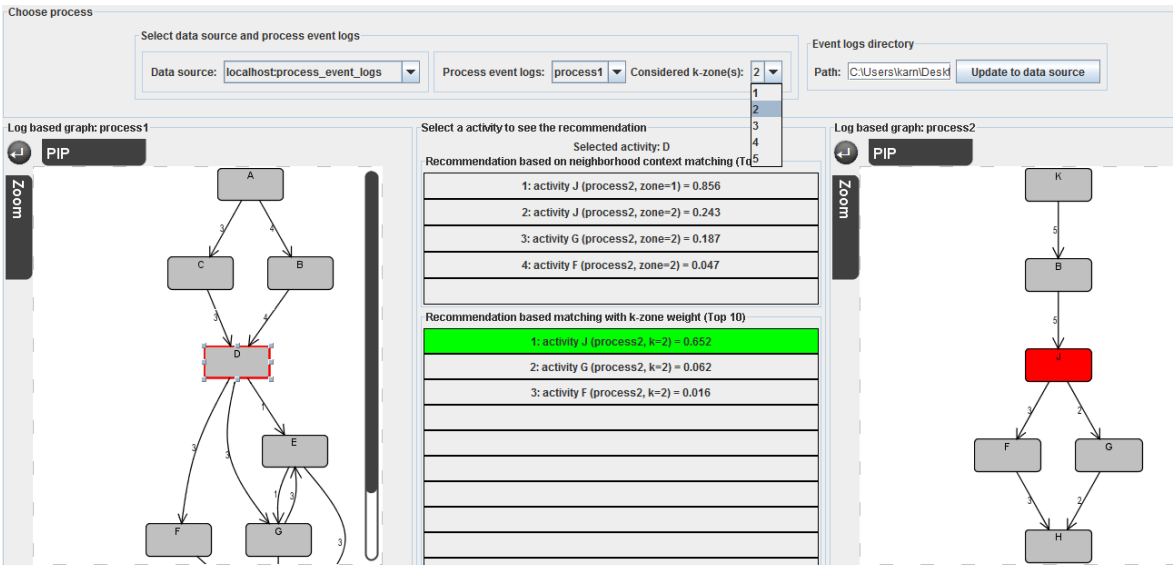


(a) (b)

8. By selecting a activity in log-based process graph, the recommendation of activities based on the neighborhood context matching values stored in the data source will be shown at the **Select a activity to see the recommendation** panel in the middle of th GUI. The first **top-5 recommendation** does not consider the k-zone weight while the **top-10 recommendation** concern the k-zone weight. Recommended activities can be selected in order to see the visualization of the process that the activity belongs to on the right log-based process graph as shown below:



9. In addition, k-zone number can be change from 1 to 5 by selecting the **Considered k-zone(s)** combo box on top of the GUI as shown below:



10. Furthermore, data source can also be changed. The recommendation for a activity in one of a process from data source generated from dataset that we used for our experiments is shown below: