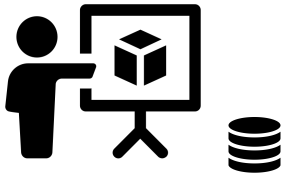
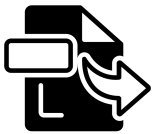
Multiple Case Notion in Log Extraction

Rachel Brabender and Oliver Clasen

Current State of the Art

- XES notation as the current standard to visualize a process flow
- object-centric information systems are builded on top of relational database technology
- There are already approaches to extract XES log files from object-centric data





http://xes-standard.org/ and icons from thenounproject.com

Event Log Extraction from Databases

How to retrieve events from database?

Redo Logs

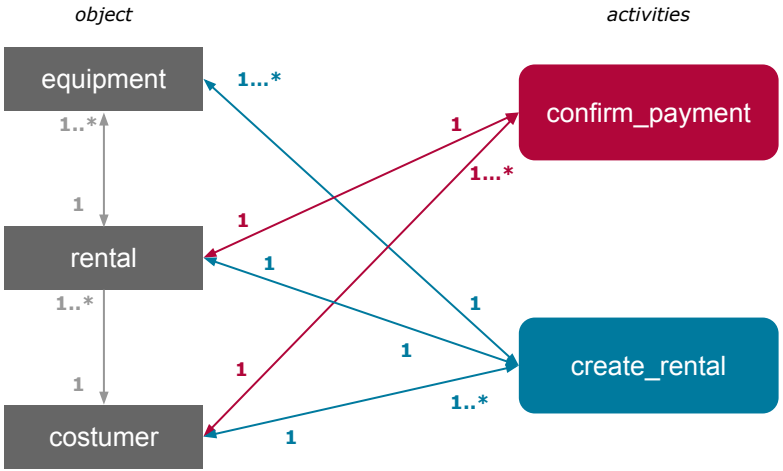
In-table versioning

It requires …

Domain Knowledge

Time

Discover a Relational Database of a Rental Shop

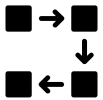


Event Log

event identifier	activity	timestamp	objects involved		
			costumer	rental	equipment
1	*create_rental*	2020-02-20 22:21	{c1}	{r1}	{e1,e2}
2	*confirm_payment*	2020-02-21 08:22	{c1}	{r1}	{}
3	*create_rental*	2020-02-21 10:10	{c2}	{r2}	{e2}

From object-centric data to the XES format

Challenges [1]



"It is required to identify the case id for the whole process."

"The quality of the input data gets compromised."

"Interactions between process instances get lost."

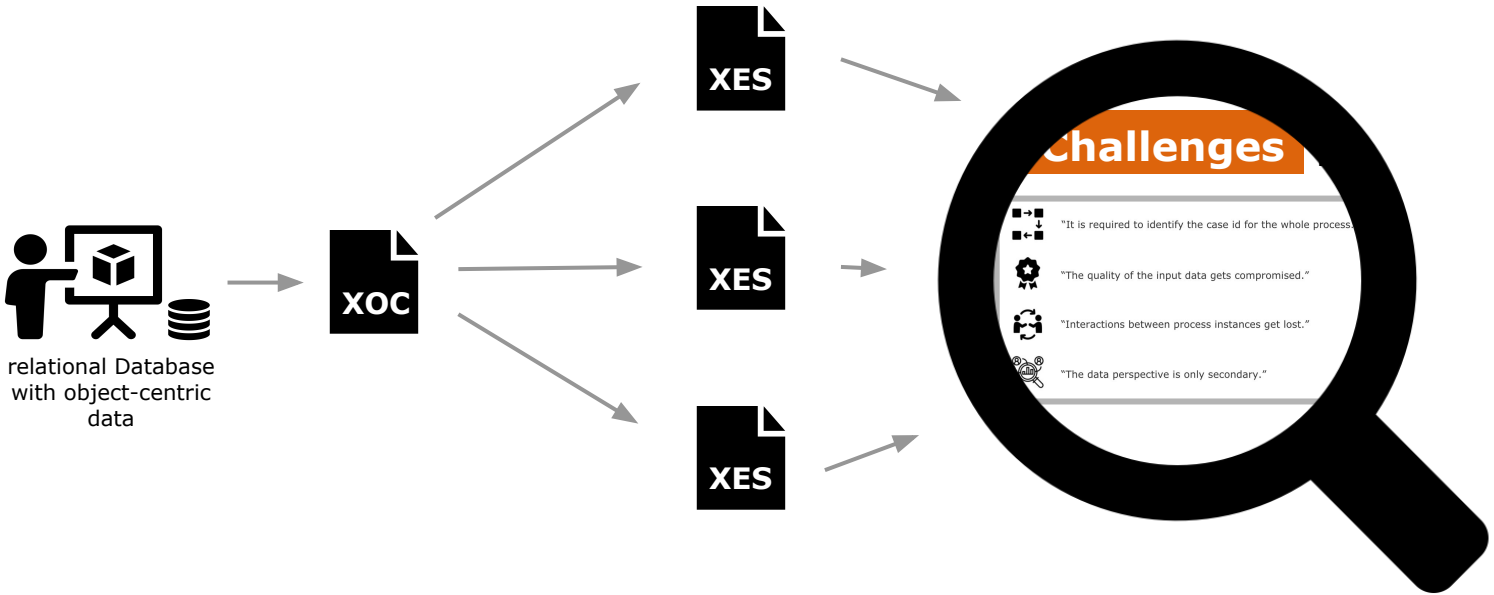
"The data perspective is only secondary."

Modeling Notations

	eXtensible Event Stream (XES)	eXtensible Object-Centric (XOC)
Perspective	Inheritance Hierarchy Structure (Top-Down Approach)	Object-Centric Data (e.g. database tables)
Log entries	Single Case Notion: events belonging to cases	Multiple Case Notions: events describing interaction between objects
Relations	1 case to 1...* events and 1 event to 1 case	1 object to 1…* events and 1 event to 1..* objects
Drawback	the decision for a perspective discards information	tend to become complex and difficult to understand

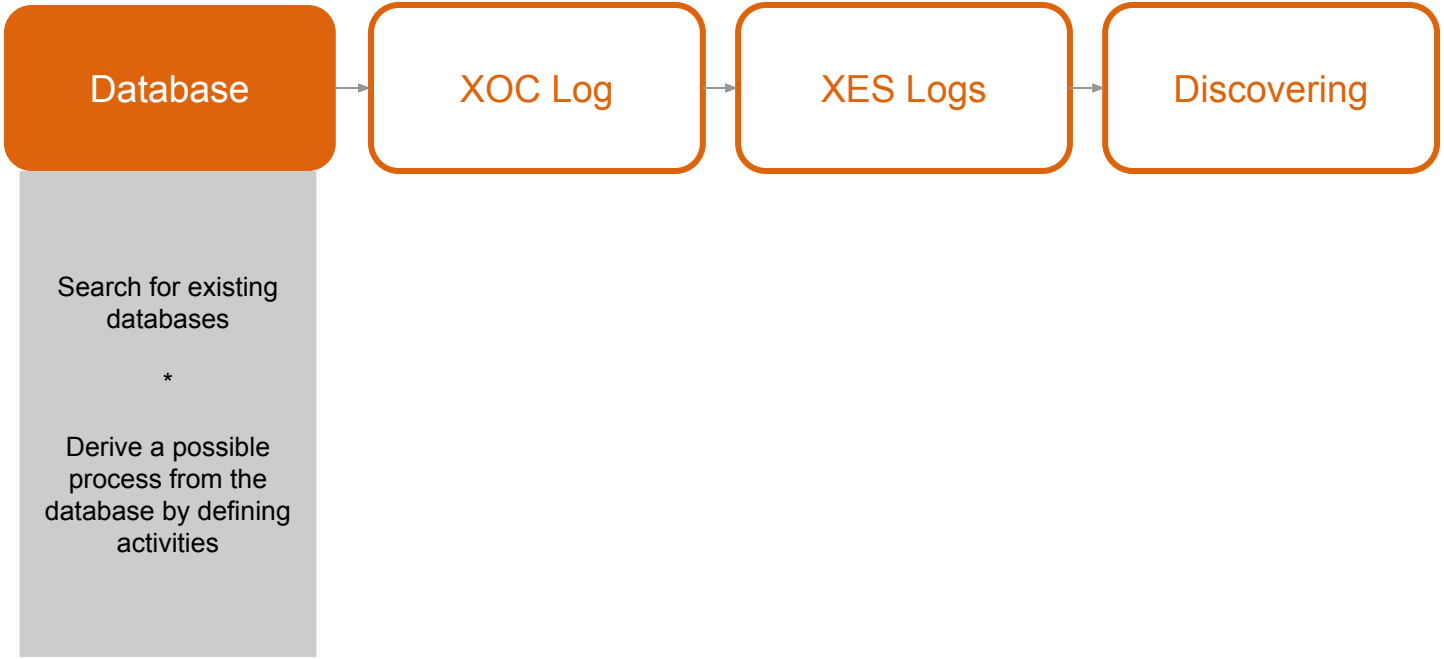
can be used to describe a certain perspective

Goal

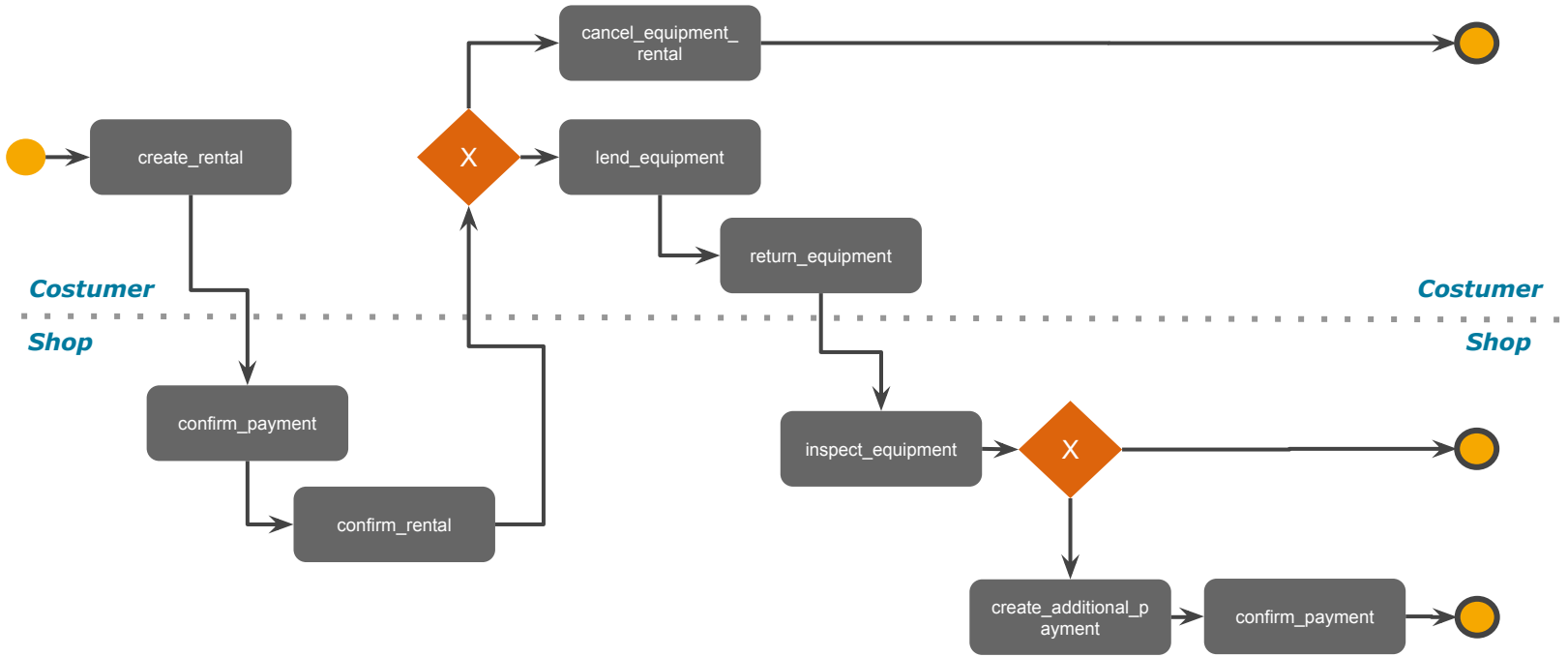


icons from thenounproject.com

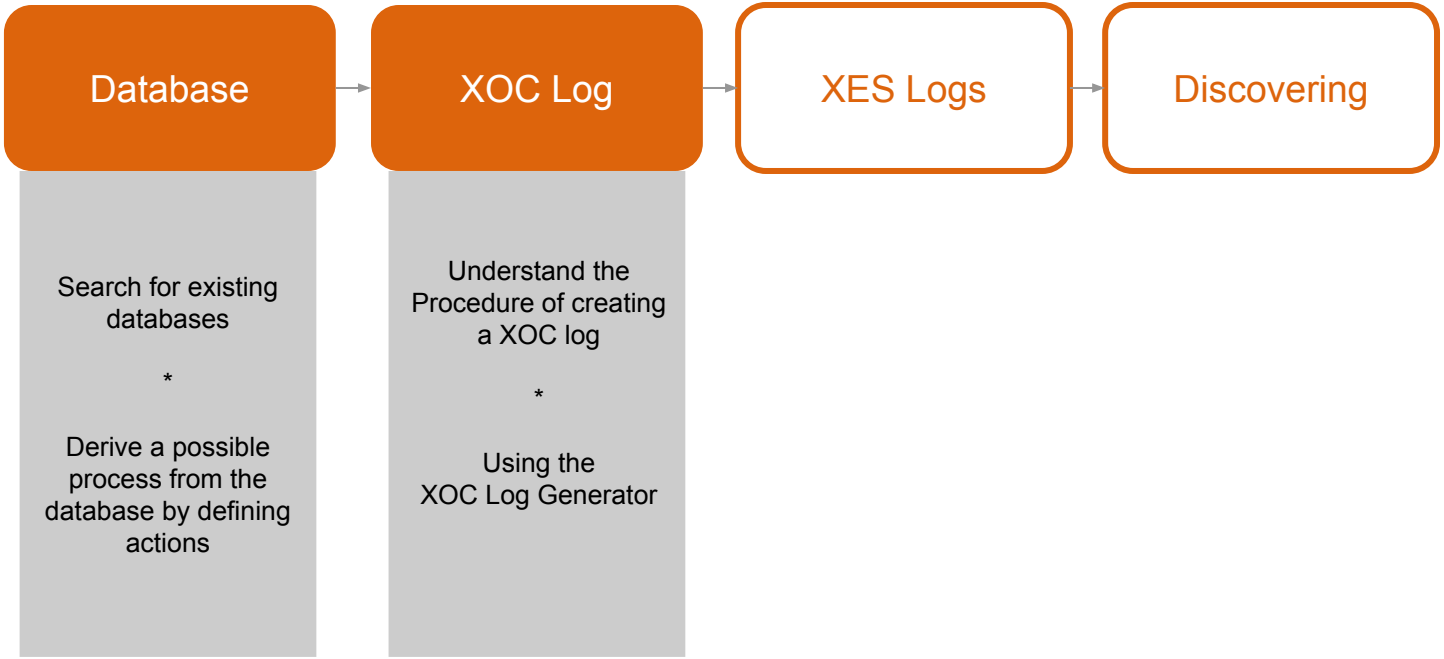
Milestones


Database
XOC Log
XES Logs
Discovering
Search for existing databases
*
Derive a possible process from the database by defining activities

Model of our Process in a Online Rental Shop



Milestones

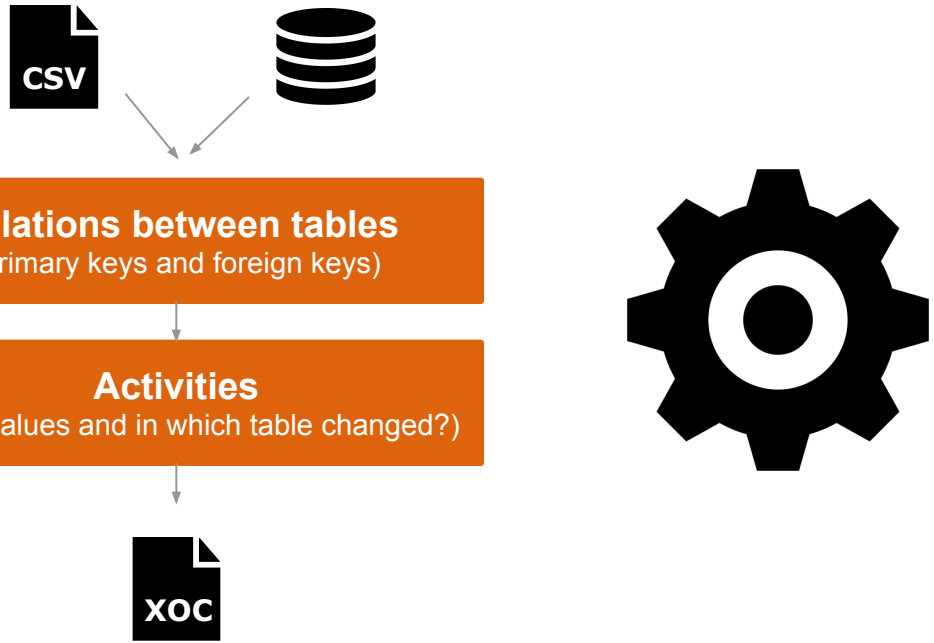

Database
XOC Log
XES Logs
Discovering
Search for existing databases
*
Derive a possible process from the database by defining actions
Understand the Procedure of creating a XOC log
*
Using the XOC Log Generator

XOC Log Generator

Input

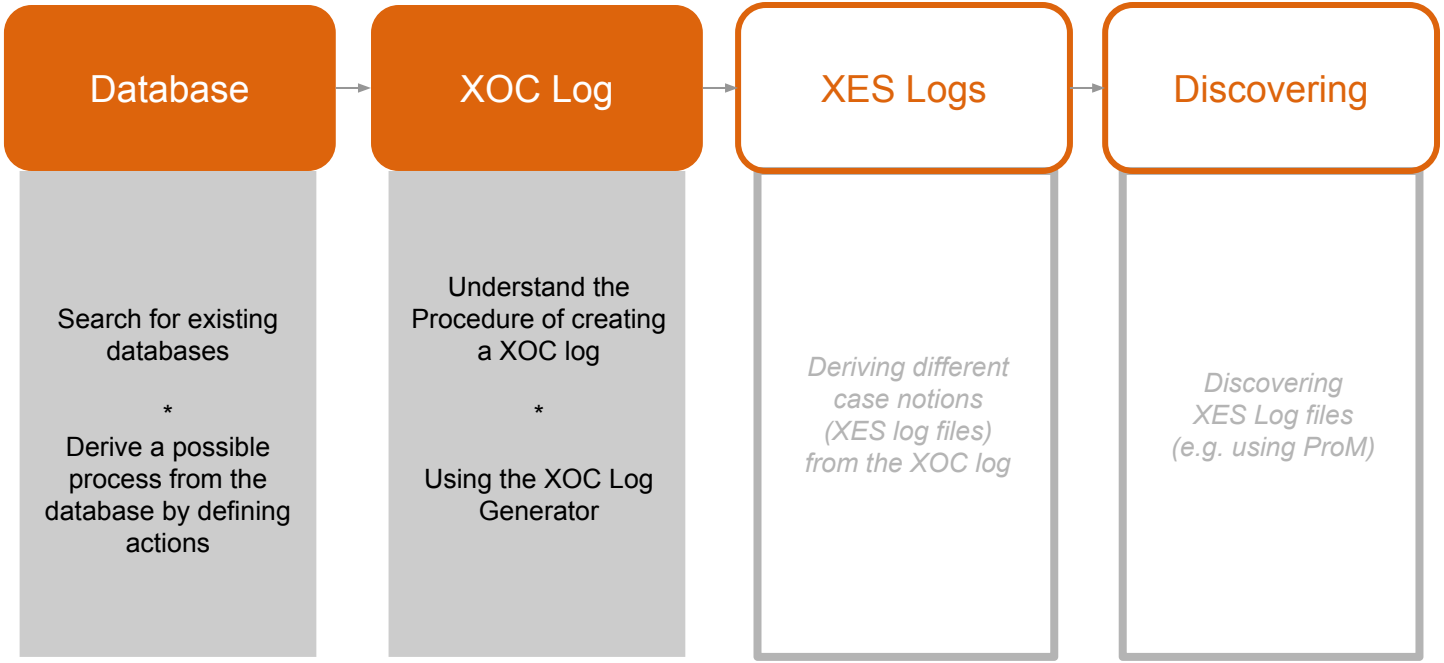
Specify

Output



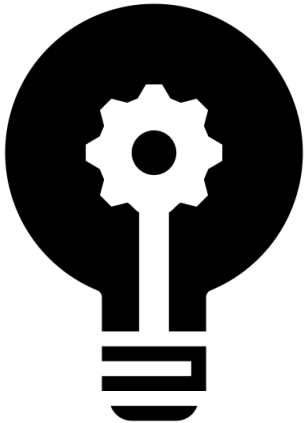
Link to the tool: https://www.win.tue.nl/ocbc/softwares/log_generation.html

Milestones


Database
XOC Log
XES Logs
Discovering
Search for existing databases
*
Derive a possible process from the database by defining actions
Understand the Procedure of creating a XOC log
*
Using the XOC Log Generator
Deriving different case notions (XES log files) from the XOC log
Discovering XES Log files (e.g. using ProM)

Take away

- **Challenges** of dealing with object-centric data from databases
  - **Extract Case ID**
  - **Loss of interaction between process instances**
- **XOC Log notation for object-centric data**
- Our **next steps**
  - Enrich our database
  - Derive XES logs from the XOC log
  - Discover XES logs

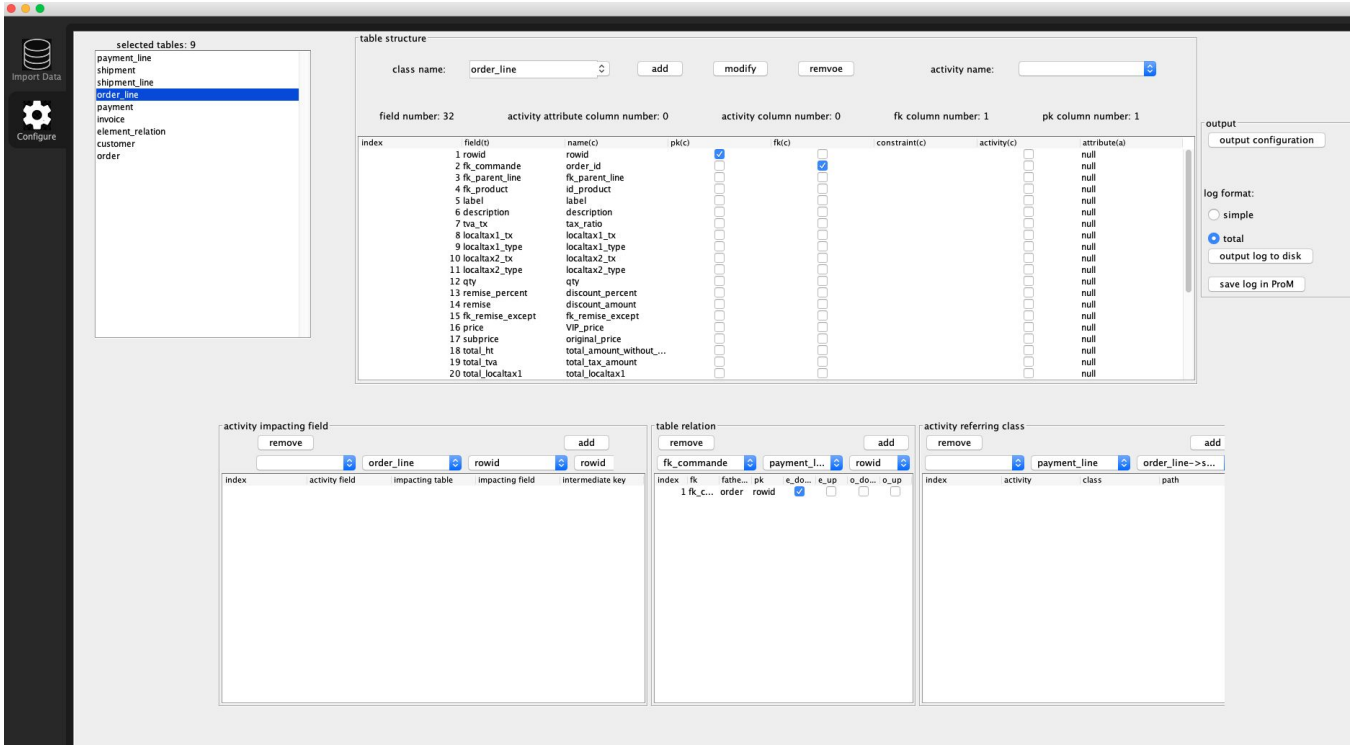


Bibliography

[1] Li G., de Murillas E.G.L., de Carvalho R.M., van der Aalst W.M.P. (2018) Extracting Object-Centric Event Logs to Support Process Mining on Databases. In: Mendling J., Mouratidis H. (eds) Information Systems in the Big Data Era. CAiSE 2018. Lecture Notes in Business Information Processing, vol 317. Springer, Cham

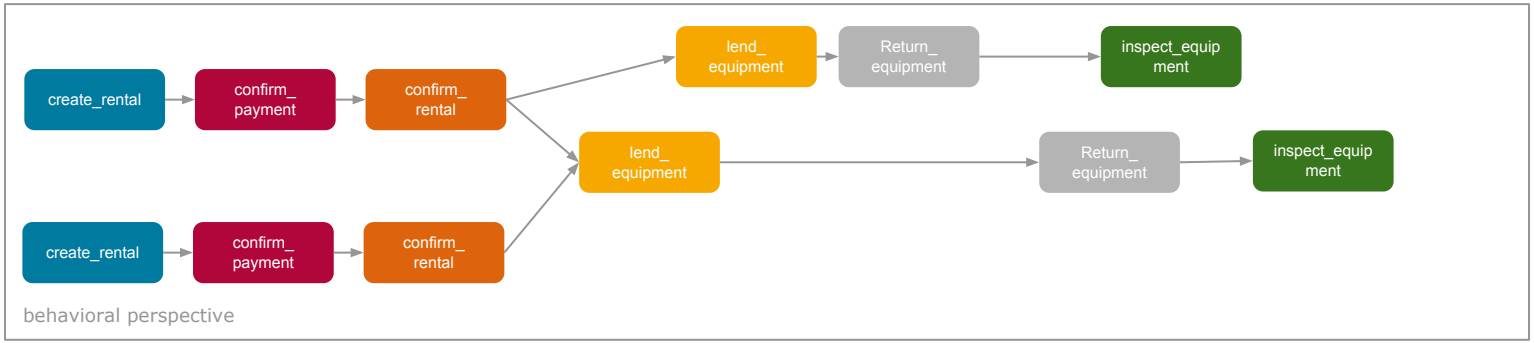
[2] van der Aalst W.M.P. (2019) Object-Centric Process Mining: Dealing with Divergence and Convergence in Event Data. In: Ölveczky P., Salaün G. (eds) Software Engineering and Formal Methods. SEFM 2019. Lecture Notes in Computer Science, vol 11724. Springer, Cham

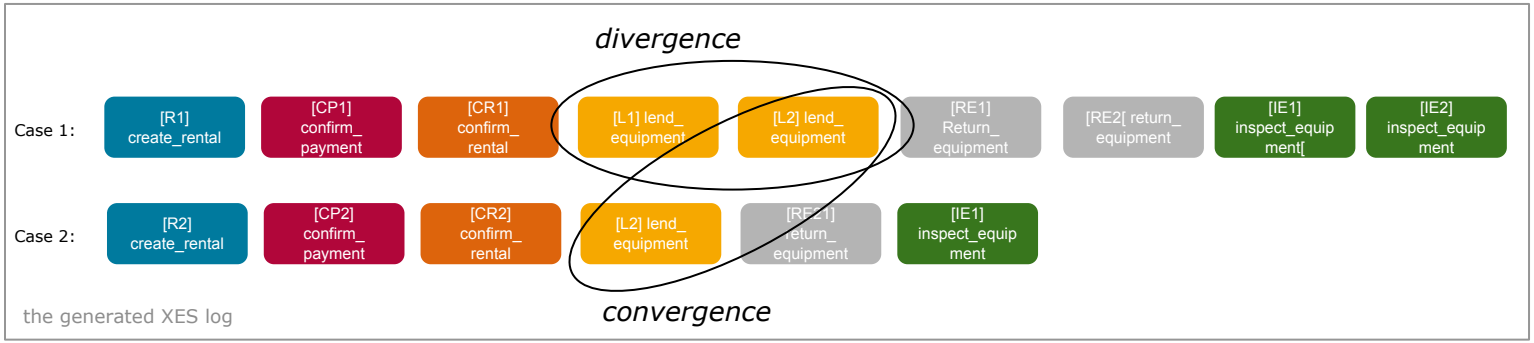
XOC Log Generation



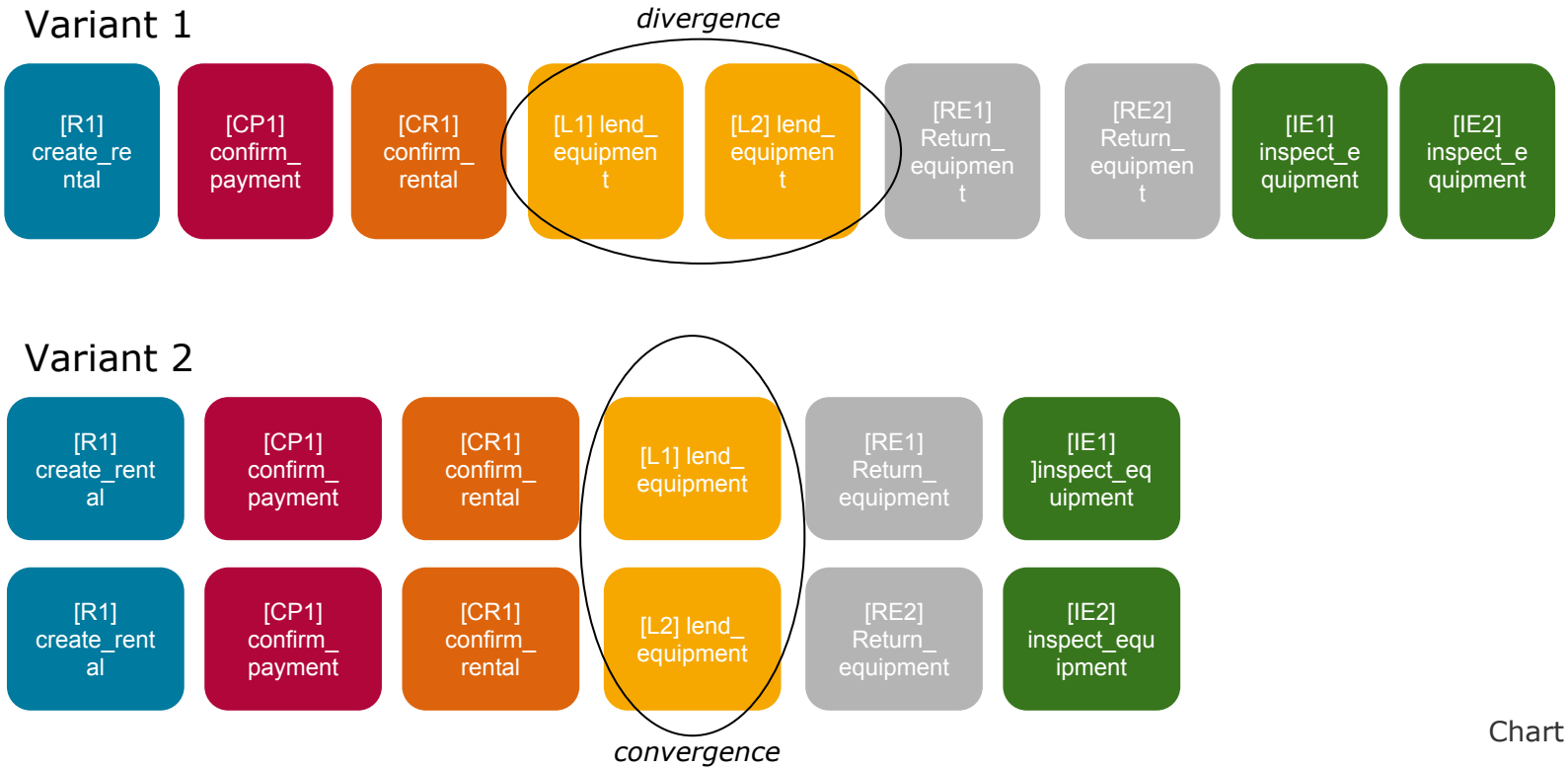
Link to the tool: https://www.win.tue.nl/ocbc/softwares/log_generation.html

The Convergence and Divergence Problem [2]





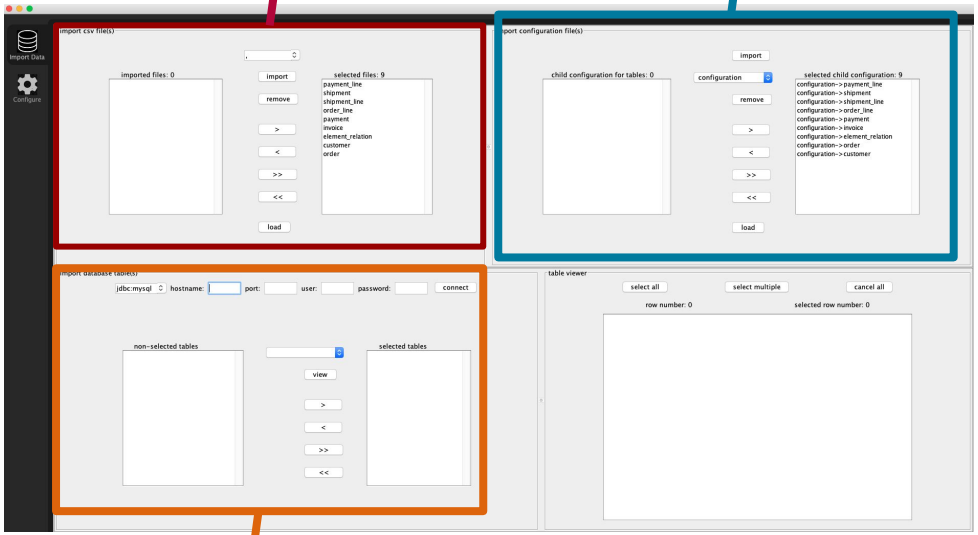
The Convergence and Divergence Problem [2]



XOC Log file generator

Import csv files

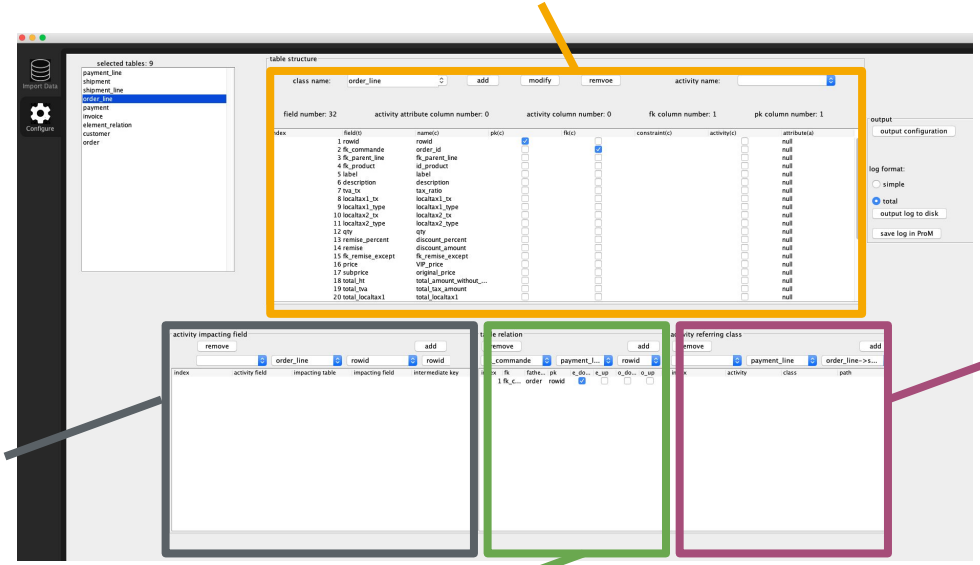
Load configuration file



Connect to a database

Define:

- A table always refers to an object and define for this the PK and the FKs
- Set a tick for a column in the column ‘activity’ to create an activity and the column should be holding a timestamp indicating when that activity happened
- Column ‘attribute’ has been never used in the example dataset



Describing effects of activities: map a column to its impacted column (modifying the value)

- Activity field = all columns marked as activity
- Impacting table = table to be influenced by this activity
- Impacting field = columns of the table impacted by the activity
- Intermediate field = key as a bridge to get to the impacting field

This area is only important when by an action does some changes in more than one table. For each of them create an entry and declare the order.

Defines the relation between the tables

- E_up means that this key is a FK
- E_down means that the id is a PK and is used in another table as a FK