Scientific Journals
of the Maritime University of Szczecin

Zeszyty Naukowe
Akademii Morskiej w Szczecinie

2016, 47 (119), 61–65
ISSN 1733-8670 (Printed)
ISSN 2392-0378 (Online)
DOI: 10.17402/149

Received: *29.04.2016*
Accepted: *23.08.2016*
Published: *20.09.2016*

Extensible event stream format for navigational data

Mariusz Dramski

Maritime University of Szczecin
1–2 Wały Chrobrego St., 70-500 Szczecin, Poland, e-mail: m.dramski@am.szczecin.pl

Key words: eXtensible Event Stream, XES standard, process mining, navigational data, data mining, modeling, XML

Abstract

The eXtensible Event Stream (XES) format is a new approach to illustrate the process data. Every ship journey is a sequence of some activities which can be read using different sources of data such ARPA, AIS etc. So we can say that this is a kind of process and its data can be organized in ordered and simple form. The most popular data formats to show the process data were of course XML and CSV. Currently, we can observe huge progress in the domain of process mining. Every year, new tools appeared and the need for some data standard became necessary. This standard is called Extensible Event Stream. In this paper, the use of XES format in navigational data is described.

Introduction

Modern data analysis requires suitable tools, such as software, methodology etc. Data can be acquired in different forms and from a variety of sources. The type of data is significant for further research and interpretation. The experiences of data scientists showed that the format of the data is the most important factor for success in data analysis (Aalst, 2011).

The same fact can be observed in the context of process mining. A few years ago the main file formats in this field were XML and CSV files. XML is a natural way to express the dependencies in the data and is still used today. This format is very widely used by software developers, scientists and more. CSV is a nice way to create a spreadsheet in MS Excel. It does not even require having MS Office installed. It is easy to use and send through the network. This is only the simple text file which contains the table of data. The last few years of research in process mining resulted in the creation of a new XML-based data format – eXtensible Event Stream (XES). The XES Working Group has mainly been established thanks to the Technical University of Eindhoven (The Netherlands). XES became the official IEEE standard (www.xes-standard.org).

The main features

The most important thing in process mining is the event logs – the basic form of the data. It contains the information about the process, such as timestamp, activities, resources, comments etc. The event log is the description of the process recorded during its life cycle. The natural way to express this kind of data was with CSV and XML formats, but could not show the desired structure of the process. The new XES data format (see UML diagram in Figure 1) is works better due to numerous advantages:

- Simplicity – the simplest possible way to represent the information about the process.
- Flexibility – the standard is able to capture event logs from any background.
- Extensibility – it must be easy to add some features to the standard in the future.
- Expressivity – the loss of the information should be as low as possible, so all information elements must be strongly typed. It must be easy for others to interpret the information.

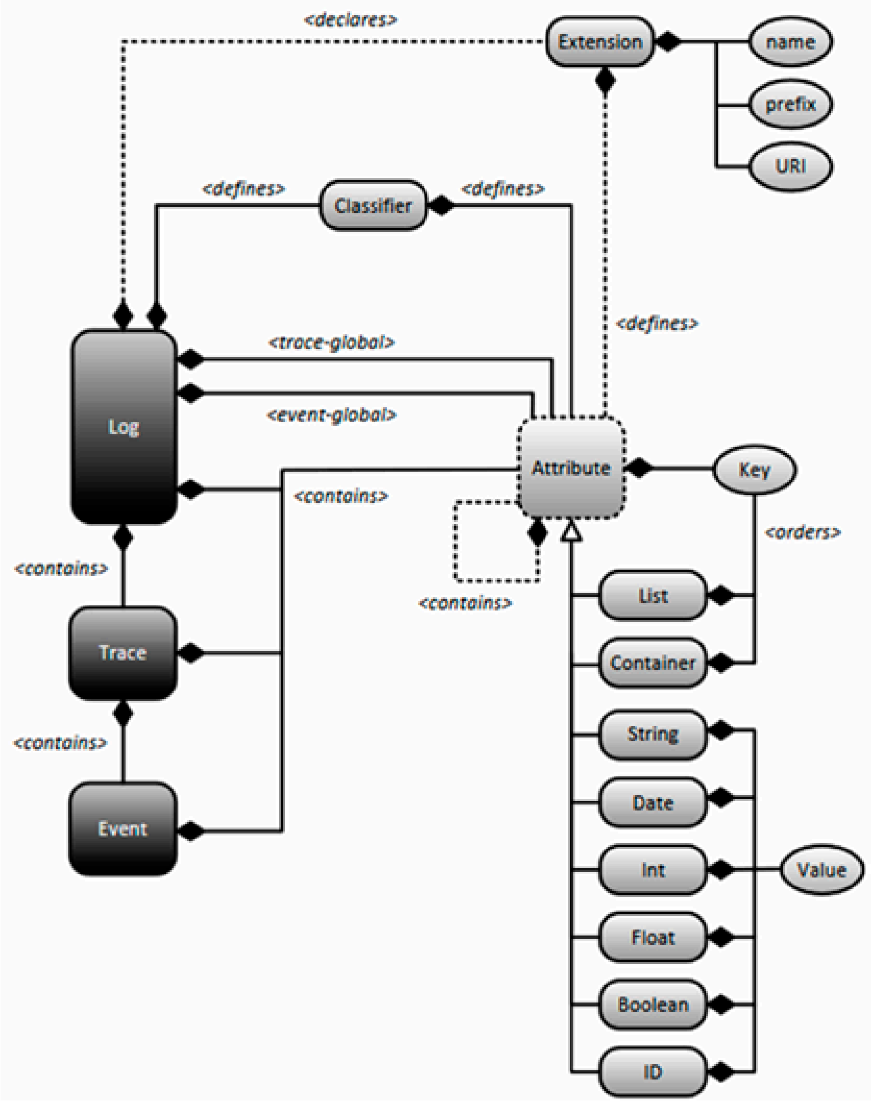


Figure 1. The UML diagram of XES format (source: www.xes-standard.org)

Why is XES data format so important? The reason is obvious and simple. The majority of tools used in process mining support mainly this kind of data (ProM, Disco). Indeed, some of them are able to convert the event log to XES, generally it is recommended to use only this format.

Table 1. An example event log (Aalst, 2011)

Case id	Event id	Properties			
		Time stamp	Activity	Resource	Cost
1	3654423	30-12-2010:11.02	Register request	Pete	50
	3654424	31-12-2010:10.06	Examine thoroughly	Sue	400
	3654425	05-01-2011:15.12	Check ticket	Mike	100
	3654426	06-01-2011:11.18	Decide	Sara	200
	3654427	07-01-2011:14.24	Reject request	Pete	200
2	3654483	30-12-2010:11.32	Register request	Mike	50
	3654484	30-12-2010:12.12	Check ticket	Mike	100
	3654485	30-12-2010:14.16	Examine casually	Pete	400
	3654486	05-01-2011:11.22	Decide	Sara	200
	3654487	08-01-2011:12.05	Pay compensation	Ellen	200
…	…	…	…	…	…

The event log

The event log consists of some cases. These cases are the ordered sets of some activities (not always unique). Each activity has some attributes. The most important attributes are its timestamp and the identifier. Other attributes can represent other types of information – numbers, characters etc.

Where do the event logs come from? The event log can be recorded during the process, can be obtained from the organization etc. It can be also generated artificially (randomly) but it is necessary to look carefully at the accuracy of the data. Absences of data are not allowed. One cannot see the time of the process if does not have any timestamps. The activity cannot be identified if it does not have even the name or identifier. Of course some data can be ignored. If the information about the resources is not necessary, then it can be skipped in some situations. If data recovery is possible, it should be done before the process mining procedures.

In Table 1, a simple example fragment of the event log is given. Three main parts of it are clearly marked. Case id, event id and properties. The most important property is the timestamp. There is no event log without this element.

Navigational data

The facts mentioned above naturally lead to the following question: can we treat the ship's motion along a given route as a process? The answer is, of course, yes. A ship is not only an object which moves from the departure point to the destination. There is a sequence of activities to reach the final port. Navigational data can be obtained in different ways. There is a lot of equipment on board, such as AIS or ARPA. Moreover, there are some web services such Marinetraffic.com tracking the ships. In Dramski (Dramski, 2016) the α-algorithm was used to build a process model for a ship's route. The data was obtained from the web and then transformed into an event log in XES format.

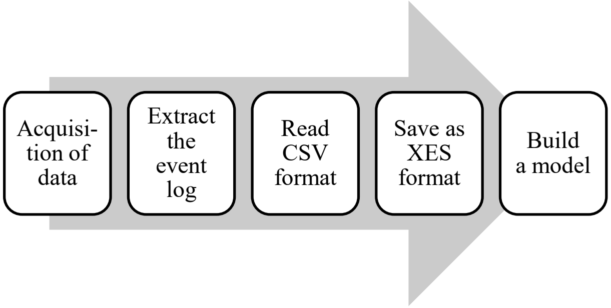


Figure 2. From data acquisition to the process model

Figure 2 illustrates all the necessary steps from the data acquisition to the process model. This time the process data was recorded using MS Excel and CSV files. Then the event log was extracted and saved in XES format using Python programming language.

Data conversion

What does the conversion look like? The sequence diagram in Figure 3 illustrates the job of the software created for this purpose.

The first step in creating a process model is the analysis process. This is the starting point of the whole procedure. Next, the event log is created and then the data is recorded. At the given point, the data recording is stopped, but the process can still work

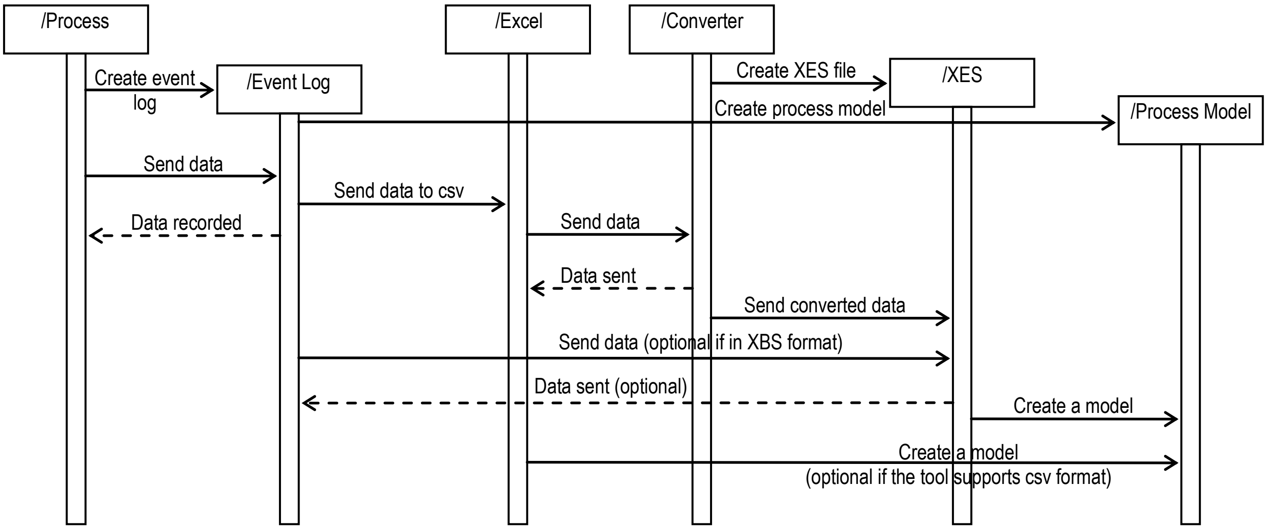
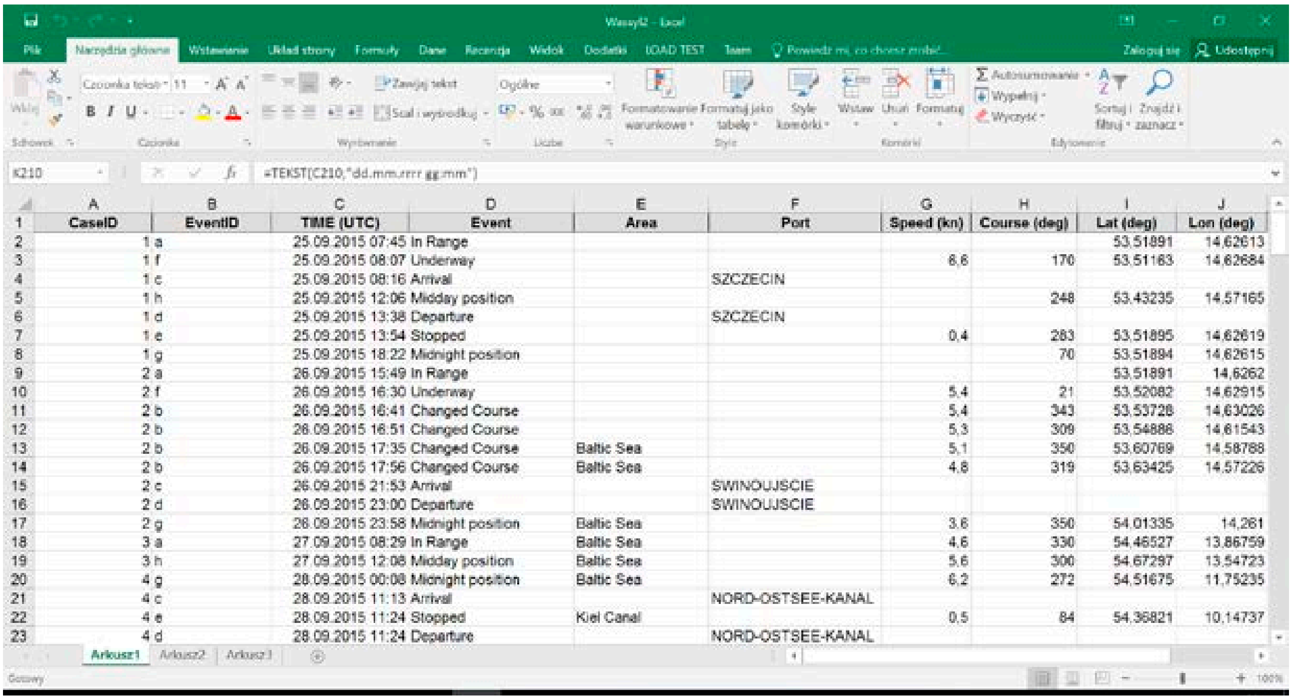


Figure 3. The sequence diagram of CSV to XES conversion

(it is not dependent on the model). Now we have two possibilities. If the recorded data format is CSV, it can now be sent directly to MS Excel, for instance, and some procedures can be carried out (e.g. in the case of data absences). If the data was originally recorded in XES standard, then it can be sent to the XES file and used for the creation of the model. However, let us go back to the situation where the CSV format is considered. The data needs to be converted and the converter is used. After the conversion the data can be sent to the XES file and, of course, used by the model.

In Figure 3 it can also be seen that the event log does not exist at the same time as the process. This is natural and easy to explain. If we do not record data, the event log is not needed. The same situation is observed with an XES file. It does not exist if it is not created. The creation time comes after the converter's job has begun. In this diagram, it is also clear that no object is destroyed. The process exists in the time and it does not depend on any other object from the system. The same situation occurs with the Excel and converter objects. Evidently it can be assumed that the converter is not needed after the conversion, but it can still be used in the future.

In Figure 4 the real collected data from Marinetraffic.com is illustrated. Clearly, due to the large number of samples, the spreadsheet is limited to the



=TEKST(C210,"dd.mm.rrrr gg:mm")

	A	B	C	D	E	F	G	H	I	J
1	CaseID	EventID	TIME (UTC)	Event	Area	Port	Speed (kn)	Course (deg)	Lat (deg)	Lon (deg)
2	1	a	25.09.2015 07:45	In Range					53,51891	14,62613
3	1	f	25.09.2015 08:07	Underway			8,6	170	53,51163	14,62684
4	1	c	25.09.2015 08:16	Arrival		SZCZECIN				
5	1	h	25.09.2015 12:06	Midday position				248	53,43235	14,57165
6	1	d	25.09.2015 13:38	Departure		SZCZECIN				
7	1	e	25.09.2015 13:54	Stopped			0,4	283	53,51895	14,62619
8	1	g	25.09.2015 18:22	Midnight position				70	53,51894	14,62615
9	2	a	26.09.2015 15:49	In Range					53,51891	14,6262
10	2	f	26.09.2015 16:30	Underway			5,4	21	53,52082	14,62915
11	2	b	26.09.2015 16:41	Changed Course			5,4	343	53,53728	14,63026
12	2	b	26.09.2015 16:51	Changed Course			5,3	309	53,54888	14,61543
13	2	b	26.09.2015 17:35	Changed Course	Baltic Sea		5,1	350	53,60769	14,58788
14	2	b	26.09.2015 17:56	Changed Course	Baltic Sea		4,8	319	53,63425	14,57226
15	2	c	26.09.2015 21:53	Arrival		SWINOUJSCIE				
16	2	d	26.09.2015 23:00	Departure		SWINOUJSCIE				
17	2	g	26.09.2015 23:58	Midnight position	Baltic Sea		3,6	350	54,01335	14,261
18	3	a	27.09.2015 08:29	In Range	Baltic Sea		4,6	330	54,46527	13,86759
19	3	h	27.09.2015 12:08	Midday position	Baltic Sea		5,6	300	54,67297	13,54723
20	4	g	28.09.2015 00:08	Midnight position	Baltic Sea		6,2	272	54,51675	11,75235
21	4	c	28.09.2015 11:13	Arrival		NORD-OSTSEE-KANAL				
22	4	e	28.09.2015 11:24	Stopped	Kiel Canal		0,5	84	54,36821	10,14737
23	4	d	28.09.2015 11:24	Departure		NORD-OSTSEE-KANAL				

Figure 4. The collected navigational data in MS Excel

```
d = dict()

for row in range(first_sheet.nrows):
   L = list()
   for column in range(first_sheet.ncols):
      L.append(first_sheet.cell(row,column).value)
   if (L[0] != "CaseID"):
      d[L[0]] = d.get(L[0],0)
   globalList.append(L)

numberOfCases = len(d.keys())

for i in range(1,numberOfCases + 1):
   tmp = list()
   for j in range(1,first_sheet.nrows):
      if (globalList[j][0] == i):
         tmp.append((globalList[j][1],globalList[j][10]))
         d[i] = tmp
```

Create global dictionary

Read data from MS Excel file and make a global list (first_sheet variable contains the file path)

Determine the number of cases

Convert the global list and save it as a dictionary which will be the base for XES data format

Listing 1. The fragment of CSV to XES converter in Python programming language

```
<?xml version="1.0" encoding="UTF-8" ?>
<log xes.version="2.0" xes.features="arbitrary-depth" xmlns="http://www.xes-standard.org/">
   <trace>
      <string key="CaseID" value="1" />
      <event>
         <string key="concept:name" value="a"/>
         <date key="time:timestamp" value="2015-09-25T07:45:00.000+00:00"/>
      </event>
      <event>
         <string key="concept:name" value="f"/>
         <date key="time:timestamp" value="2015-09-25T08:07:00.000+00:00"/>
      </event>
      <event>
         <string key="concept:name" value="c"/>
         <date key="time:timestamp" value="2015-09-25T08:16:00.000+00:00"/>
      </event>
…
```

Listing 2. The initial fragment of the XES file

first visible fragment from the screen. It is observed that some of the data is lacking. Empty fields do not always denote missing data because they can be easily updated. One of the MS Excel features is the possibility to export the data directly into CSV format. Unfortunately, MS Office does not support the XES format, so it is necessary to write a converter. This was done using Python (www.python.org) programming language. The most important part of the code is shown on the Listing 1.

The presented listing shows the main initial procedures of the converter's job. First of all there is a need to read the data from the MS Excel spreadsheet and then save it to a special data structure, being a combination of Python's list, tuple and the dictionary. At the end (not presented in this listing), the data is directly saved into XES format. The content of the XES file looks like on the Listing 2.

Listing 2 shows only the initial fragment of the final XES file. As aforementioned, the file structure is similar to XML, but the most important thing is that it can now be easily used to create a process model using process mining techniques and tools.

Conclusions

In this paper, the short description of the usability of XES file format for navigational data is shown. This data structure lets us treat the ship's route (or other processes related to marine industry) like a sequence of certain activities. The XES format allows easy understanding of what the most important activities are; it shows the time of each activity and also can contain some other, maybe not always necessary, but usable information.

The research development in the process mining domain allows the creation of process models in every domain of the modern economy. Thanks to these tools, there are a lot of advantages, e.g. operational support (Dramski, 2015), which makes the predictions, or correction of some incorrect functionalities of the process, possible.

The XES data format makes the analysis of the process easier. It is supported by most tools, such as ProM or Disco.

Acknowledgments

This research outcome has been achieved under the research project No. 1/S/ITM/16 financed from a subsidy of the Ministry of Science and Higher Education for statutory activities of Maritime University of Szczecin.

References

1. Aalst V.D. (2011) *Process mining – discovery, conformance and enhancement of business processes*. Springer.
2. Dramski M. (2015) Wsparcie operacyjne w transporcie w kontekście process mining. *Logistyka* 4.
3. Dramski M. (2016) *The alpha algorithm in the modeling of the ship's route*. TST International Conference, Ustroń 16–19 March 2016.
4. www.python.org
5. www.xes-standard.org