

Micro Event Extension

Background

Many current Business Process Management Suites (BPMS) can define **nested activities** in order to provide a summary view and an abstract flow by hiding the nested activities. The nested activities may also be called **sub activities, micro activities, sub process**, and so on. Several BPMSs support sub activities [1-3]. Examples of nested activities include executions of web services, web pages, and manual processes. For example, in IBM BPM [1], a process designer first defines an abstract flow of activities. Then he selects one of the activities, moves to the nested activity view, and defines web pages that actually conduct the selected parent activity.

In order to analyse the behaviours of such nested activities, the history of the nested activities must be recorded in event logs. We call an event of the nested activity a **micro event**. By using some kind of workaround, we can record the micro events to an existing XES based event log. However, they tend to contaminate the xml element of the parent activity. Before we become awash with several variations of workaround based event logs that record micro events, we have to demonstrate how micro events are appropriately stored in event logs.

Definition

The micro event extension defines a nested element for storing information about the micro events associated with activities within a log.

The extension is defined as shown in the table below:

Extension definition	
Name	Micro
Prefix	micro
Extension URI	http://www.xes-standard.org/micro.xesext
XML representation	<extension name="Micro" prefix="micro" uri="http://www.xes-standard.org/micro.xesext" />

The extension defines the following attributes:

No	XES Attribute key	Definition	XES datatype	Occ.	Allowed values, examples, other constraints
1	events	Container element for micro events	List	1	It contains a list of micro events and attributes of a set of micro events.
2	event	Container element for micro event	List	1-n	It contains attributes of one micro event.

XES extension

```
<xesextension
  name="Micro"
  prefix="micro"
  uri="http://www.xes-standard.org/micro.xesext">
  <event>
    <list key="events">
      <alias mapping="EN" name="List of micro events for this event"/>
    </list>
    <int key="length">
      <alias mapping="EN" name="Number of micro events for this event"/>
    </int>
  </event>
</xesextension>
```

```

</event>
<meta>
  <string key="name">
    <alias
      mapping="EN" name="Generally understood name for a micro event"/>
    </string>
  <date key="timestamp">
    <alias
      mapping="EN"
      name="The date and time at which the micro event has occurred"/>
    </date>
  </meta>
</xesextension>

```

Example

Assume that a top level activity 'Register' consists of three nested activities, 'input_customer_info', 'input_insurance_app', and 'input_etc'. The history of micro events are stored under the parent event. An event that is parent of micro events can have attributes of the micro event list such as "the length of micro events was three". A container element micro event contains attributes of the micro event such as the name of a web page, a timestamp, and so on.

```

<log>
  <extension
    name="Concept"
    prefix="concept"
    uri="http://www.xes-standard.org/concept.xesext"/>
  <extension
    name="Time"
    prefix="time"
    uri="http://www.xes-standard.org/time.xesext"/>
  <extension
    name="Micro"
    prefix="micro"
    uri="http://www.xes-standard.org/micro.xesext"/>
  <trace>
    <event>
      <string key="concept:name" value="Register" />
      <date key="time:timestamp" value="2015-04-13T14:02:30.287Z" />
      <list key="micro:events">
        <string key="micro:name" value="input_customer_info">
          <date key="micro:timestamp" value="2015-04-13T14:02:31.199Z"/>
        </string>
        <string key="micro:name" value="input_insurance_app"/>
          <date key="micro:timestamp" value="2015-04-13T14:02:41.032Z"/>
        </string>
        <string key="micro:name" value="input_etc"/>
          <date key="micro:timestamp" value="2015-04-13T14:02:49.729Z"/>
        </string>
      </list>
      <int key="micro:length" value="3"/>
    </event>
  </trace>
</log>

```

References

[1] IBM BPM, "Modeling your business processes with IBM WebSphere Lombardi Edition, Part 3: Advanced modeling"

http://www.ibm.com/developerworks/websphere/library/techarticles/1112_wang/1112_wang.html

[2] Bizagi , "Creating a sub-process"

http://help.bizagi.com/processmodeler/en/index.html?creating_a_sub_process.htm

[3] Oracle BPM, "Using Subprocesses to Organize Your Process"

http://docs.oracle.com/cd/E25178_01/doc.1111/e15176/model_bus_procs_bpmpd.htm#CJADJBHC