XML as a basis for interoperability in Real Time Distributed Systems

Polly M.S. Poon[1], Tharam S Dillon[1], Elizabeth Chang[2]
[1]*Faculty of Information Technology, UTS,*
Sydney, Australia
{polly, tharam}@it.uts.edu.au

[2]*School of Information Systems,*
Curtin University of Technology,
Australia
change@cbs.curtin.edu.au

Abstract

Many real time applications consist of components that can situate in a centralized/distributed environment. Typically due to different system requirements these components could be developed in different technologies which require a bridge for communication. With the increasing popularity of XML, it has become an alternative solution for data exchange in the real time application domain. XML has become the most important mechanism for data exchange between heterogeneous data sources. This paper presents a methodology for real time application data in XML. By defining a solid XML Schema for the real time domain attributes, applications from diverse platforms can be based on the data definition and create instances of XML documents to exchange data.

1. Introduction

Real Time Systems have been appearing in a variety of domains from industrial, commercial, health care to defense applications. Some examples of real time systems include dealing room software for share trading, network management systems, telecommunication and electric power system computer control centers, industrial process control system and aerospace applications. Real time systems often involve interaction with an external environment and transmission of data to databases that store data for further processing. They frequently also involve safety critical and fault tolerant aspects.

When considering real time systems currently one can distinguish two broad types: [1]

(i) Isolated embedded systems that perform a narrow set of functions

(ii) Distributed real time systems that interact with a widely geographically dispersed external environment

Systems of type (i) are often hard real time systems with sharp and definite deadlines that must be met. Systems of type (ii) consist of both hard real time systems and soft real time systems.

In this paper we will be primarily concerned with systems of type (ii), which could be widely distributed with multiple data sources. The key factors that characterize real time systems of type (ii) are:

i) Collection and transmission of data from sensors to data concentrators
ii) Determination of the time validity of data
iii) Storage of data at several levels and different databases that must interoperate with each other
iv) The timeliness of computation

When considering factors i) to iii) above one notes it is important to have:

a) A suitable and relatively widely accepted format for exchange of data between different data sources
b) A suitable mechanism for storing 'annotated' data that has clearly defined semantics

With regards to a), one should note that XML Messaging is becoming a widely accepted approach for data exchange and interoperability for web based and Business to Business (B2B) e-Commerce systems. There is, therefore, a strong case to examine it as the basis for interoperability for real time systems. However, in order to do this, we need a careful XML Schema definition and tag definition which allows proper characterization of the features of real time data.

(b) requires suitable annotation of real time data to indicate time validity, constraints on use, etc. Again an XML repository containing real time data in the form of XML documents would be appropriate for this. The semi-

Proceedings of the Second IEEE Workshop on Software Technologies for Future Embedded and Ubiquitous Systems (WSTFEUS'04)
0-7695-2123-1/04 $ 20.00 © 2004 IEEE

structured nature of XML data, will more easily allow extensibility of such a data store. This is of considerable importance as the number and type of sensors may evolve

These two factors provide a strong motivation for exploration of the creation of a Real Time Markup Language (RTML), which is the subject of this paper. As a start, we will concentrate in this paper on characterization of the 'time' aspects of real time data. In order to create the schema we will first use a valid model for the required features and then carry out a transformation to XML Schema.

2. Architectural framework for XML based Real Time Messaging System

A typical architectural framework is shown here in Figure 1. XML data might be exchanged between several databases and systems are shown in Figure 1. Note that all the databases would be utilizing a common XML Schema which would facilitate this exchange of data. Note this could require transformation of either relational or object data into XML format which conforms to the common XML Schema.

3. Transformation of Real Time UML to Real Time Markup Language

The use of UML has been appearing in the design of many real time systems. In 1999, the Object Management Group (OMG) issues the request for proposal (RFP) for the standardization of precise semantics and modeling of real-time systems in UML.[4, 5] This proposal quantifies the analysis in terms of schedulability, performance and time issues in real time domain. Version 1.0 title "UML™ Profile for Schedulability, Performance, and Time Specification" [2] (Will name it OMG Real Time Profile from here) was released in September 2003.

In this paper we will concentrate on discussing the transformation of the General Time Modeling into XML

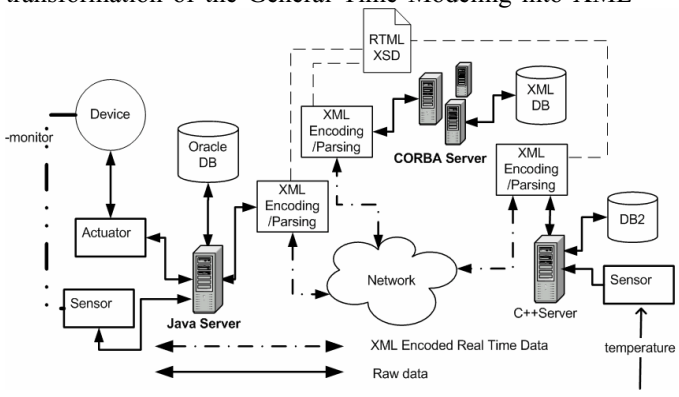

Figure 1 Real Time Messaging using RTML

Schema representation, as the first step towards defining a markup language suitable for Real Time Systems. Based on the above argument, we have transformed the OMG Real Time Profile, into a set of XML Schema [6, 7, 8, 9].

3.1 Timing Mechanism Model

The timing mechanism UML diagram is reproduced from the OMG Real Time UML Standard as shown in Figure 2. We transformed these using the rules for transformation defined in OMG Real Time Profile. The specific relationships that needed transformation here are generalization and association.

In the follow sections, we are going to transform the above model and classes into XML Schema based on the transformation in [3]. It will provide a XML Schema representation as well as a Document Object Model (DOM) [10] visualization respectively. Amongst the concepts above we will be considering are:
1) Timing Mechanism; 2) Time Value; 3) Clock and 4) Timer.

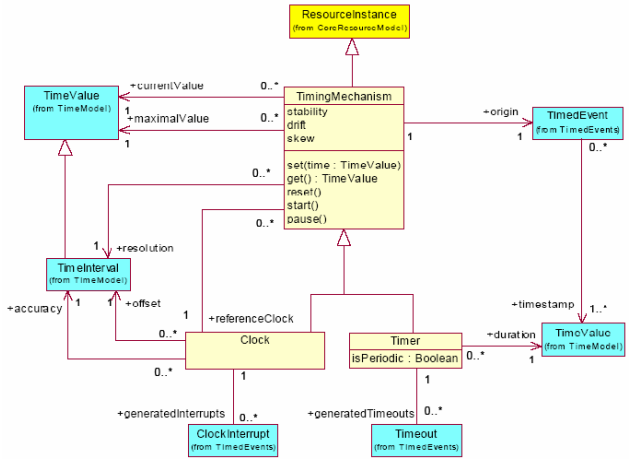

Figure 2 Timing Mechanism Concepts (Reproduced from OMG Real Time Profile)

The XML Schema and the DOM model for timing mechanism and time value are shown in Fig. 3 and 4 respectively.

```
  <xs:element name="TimingMechanism"
type="TimingMechanismType">
      <xs:key name="TM_Key">
         <xs:selector
xpath="TimingMechanism"/>
         <xs:field xpath="@TM_ID"/>
      </xs:key>
      <xs:keyref name="TM_KeyRef"
refer="TM_Key">
         <xs:selector xpath="Clock"/>
         <xs:field xpath="@TM_ID"/>
```

```
      </xs:keyref>
   </xs:element>
   <xs:complexType name="TimingMechanismType">
      <xs:sequence>
         <xs:element name="stability"
type="xs:float" minOccurs="0"/>
         <xs:element name="drift"
type="xs:float" minOccurs="0"/>
         <xs:element name="skew"
type="xs:float" minOccurs="0"/>
         <xs:element name="maximalValue"
type="TimeValueType" minOccurs="0"/>
         <xs:element name="origin"
type="xs:string" minOccurs="0"/>
         <xs:element name="resolution"
type="TimeValueType" minOccurs="0"/>
         <xs:element name="offset"
type="TimeValueType" minOccurs="0"/>
         <xs:element name="accuracy"
type="TimeValueType" minOccurs="0"/>
         <xs:element name="currentVal"
type="TimeValueType" minOccurs="0"/>
      </xs:sequence>
      <xs:attribute name="TM_ID"
type="xs:string"/>
   </xs:complexType>
```

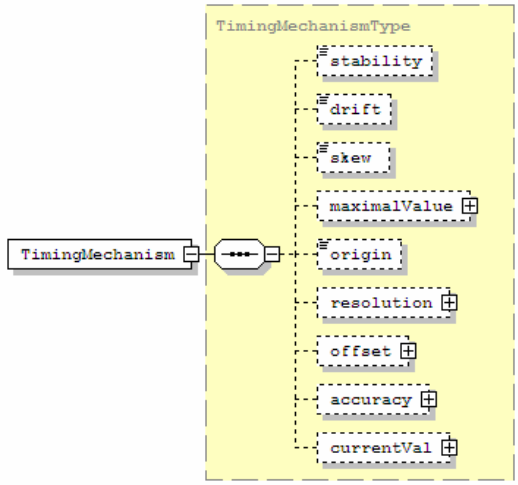

Figure 3 Timing Mechanism

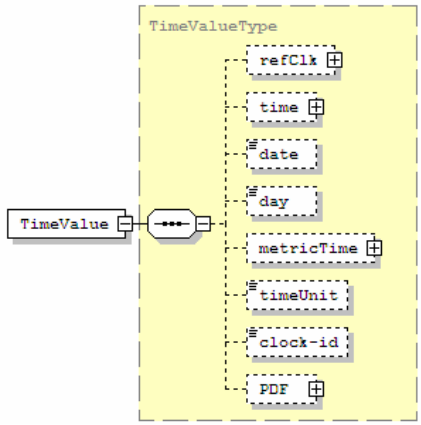

Figure 4 Time Value

```
  <xs:element name="TimeValue"
type="TimeValueType"/>
   <xs:complexType name="TimeValueType">
      <xs:sequence>
         <xs:element name="refClk"
type="ClockType" minOccurs="0"/>
         <xs:element name="time"
type="timeStrType" minOccurs="0"/>
         <xs:element name="date"
type="dateStrType" minOccurs="0"/>
         <xs:element name="day"
type="dayStrType" minOccurs="0"/>
         <xs:element name="metricTime"
type="metricTimeStrType" minOccurs="0"/>
         <xs:element name="timeUnit"
type="timeUnitType" minOccurs="0"/>
         <xs:element name="clock-id"
type="ClockIDType" minOccurs="0"/>
         <xs:element name="PDF"
type="PDFStringType" minOccurs="0"/>
      </xs:sequence>
      <xs:attribute name="kind"
type="timeKind"/>
      <xs:attribute name="Time_ID"
type="xs:string"/>
   </xs:complexType>
   <xs:simpleType name="timeKind">
      <xs:restriction base="xs:string">
         <xs:enumeration value="discrete"/>
         <xs:enumeration value="sense"/>
      </xs:restriction>
  </xs:simpleType>
<xs:simpleType name="hrType">
      <xs:restriction base="xs:string">
         <xs:pattern value="([0-1][0-9]|2[0-
4])"/>
      </xs:restriction>
   </xs:simpleType>
   <xs:simpleType name="minType">
      <xs:restriction base="xs:string">
         <xs:pattern value="([0-5][0-9])"/>
      </xs:restriction>
   </xs:simpleType>
   <xs:simpleType name="secType">
      <xs:restriction base="xs:string">
         <xs:pattern value="([0-5][0-9])"/>
      </xs:restriction>
   </xs:simpleType>
   <xs:simpleType name="centisecType">
      <xs:restriction base="xs:string">
         <xs:pattern value="([0-9][0-9])"/>
      </xs:restriction>
   </xs:simpleType>
   <xs:complexType name="timeStrType">
      <xs:sequence>
         <xs:element name="hr" type="hrType"/>
         <xs:element name="min"
type="minType"/>
         <xs:element name="sec"
type="secType"/>
         <xs:element name="centisec"
type="centisecType"/>
      </xs:sequence>
   </xs:complexType>
   <xs:simpleType name="dateStrType">
      <xs:restriction base="xs:string">
```

```
         <xs:pattern     value="([0-9][0-9][0-
9][0-9])/([0-0][1-9]|[0-1][1-2])/([0-0][1-
9]|1[0-9]|2[1-9]|3[0-1])"/>
      </xs:restriction>
   </xs:simpleType>
   <xs:simpleType name="dayStrType">
      <xs:restriction base="xs:string">
         <xs:enumeration value="Mon"/>
         <xs:enumeration value="Tue"/>
         <xs:enumeration value="Wed"/>
         <xs:enumeration value="Thu"/>
         <xs:enumeration value="Fri"/>
         <xs:enumeration value="Sat"/>
         <xs:enumeration value="Sun"/>
      </xs:restriction>
   </xs:simpleType>
   <xs:complexType name="metricTimeStrType">
      <xs:sequence>
         <xs:choice>
            <xs:element name="PDFString"
type="PDFStringType"/>
         </xs:choice>
         <xs:element name="timeUnit"
type="timeUnitType"/>
      </xs:sequence>
      <xs:attribute name="MetricTimeStr_ID"
type="xs:string"/>
   </xs:complexType>
  <xs:simpleType name="timeUnitType">
      <xs:restriction base="xs:string">
         <xs:enumeration value="ns"/>
         <xs:enumeration value="us"/>
         <xs:enumeration value="ms"/>
         <xs:enumeration value="s"/>
         <xs:enumeration value="hr"/>
         <xs:enumeration value="wks"/>
         <xs:enumeration value="mos"/>
      </xs:restriction>
   </xs:simpleType>
   <xs:complexType name="PDFStringType">
      <xs:sequence>
         <xs:choice>
            <xs:element name="bernoulliPDF">
               <xs:complexType>
                  <xs:sequence>
                     <xs:element
name="probability">
                        <xs:simpleType>
                           <xs:restriction
base="xs:float">
                              <xs:maxInclusive
value="1"/>
                           </xs:restriction>
                        </xs:simpleType>
                     </xs:element>
                  </xs:sequence>
               </xs:complexType>
            </xs:element>
            <xs:element name="binomialPDF">
               <xs:complexType>
                  <xs:sequence>
                     <xs:element
name="probability" type="xs:integer"/>
                     <xs:element
name="numberOfTrial" type="xs:integer"/>
                  </xs:sequence>
               </xs:complexType>
            </xs:element>
            <xs:element name="exponentialPDF">
               <xs:complexType>
                  <xs:sequence>
                     <xs:element name="mean"
type="xs:float"/>
                  </xs:sequence>
               </xs:complexType>
            </xs:element>
            <xs:element name="gammaPDF">
               <xs:complexType>
                  <xs:sequence>
                     <xs:element name="k"
type="xs:integer"/>
                     <xs:element name="mean"
type="xs:float"/>
                  </xs:sequence>
               </xs:complexType>
            </xs:element>
            <xs:element name="geometricPDF"/>
            <xs:element name="histogramPDF">
               <xs:complexType>
                  <xs:sequence>
                     <xs:sequence>
                        <xs:element
name="startOfInterval" type="xs:float"/>
                        <xs:element
name="probability" type="xs:float"/>
                     </xs:sequence>
                     <xs:element
name="endInterval" type="xs:float"/>
                  </xs:sequence>
               </xs:complexType>
            </xs:element>
            <xs:element name="normalPDF">
               <xs:complexType>
                  <xs:sequence>
                     <xs:element name="mean"
type="xs:float"/>
                     <xs:element name="std">
                        <xs:simpleType>
                           <xs:restriction
base="xs:float">
                              <xs:minInclusive
value="0"/>
                           </xs:restriction>
                        </xs:simpleType>
                     </xs:element>
                  </xs:sequence>
               </xs:complexType>
            </xs:element>
            <xs:element name="poissonPDF">
               <xs:complexType>
                  <xs:sequence>
                     <xs:element name="mean"
type="xs:float"/>
                  </xs:sequence>
               </xs:complexType>
            </xs:element>
            <xs:element name="uniformPDF">
               <xs:complexType>
                  <xs:sequence>
                     <xs:element name="start"
type="xs:float"/>
                     <xs:element name="end"
type="xs:float"/>
                  </xs:sequence>
```

```
            </xs:complexType>
          </xs:element>
        </xs:choice>
        <xs:element name="unit"
type="timeUnitType"/>
      </xs:sequence>
   </xs:complexType>
```

3.1.3 Clock

The XML Schema and the DOM model for clock and timer are shown in Fig. 5 and 6 respectively.

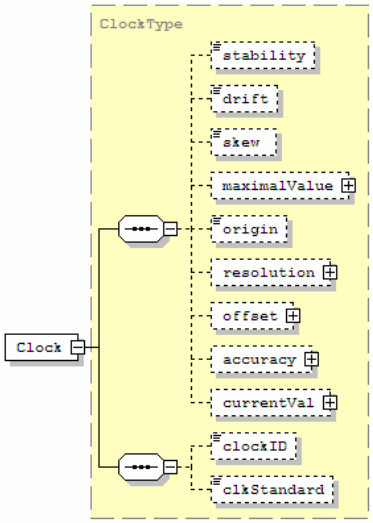

Figure 5 Clock

```
   <xs:complexType name="ClockType">
    <xs:complexContent>
       <xs:extension base="TimingMechanismType">
          <xs:sequence>
             <xs:element name="clockID"
type="ClockIDType" minOccurs="0"/>
             <xs:element name="clkStandard"
type="xs:string" minOccurs="0"/>
          </xs:sequence>
          <xs:attribute name="Clock_ID"
type="xs:string"/>
       </xs:extension>
    </xs:complexContent>
   </xs:complexType><xs:simpleType
name="ClockIDType">
       <xs:union memberTypes="ClockIDString
clock-idChoice"/>
        </xs:simpleType>
```

3.1.4 Timer

```
   <xs:element name="Timer" type="TimerType"/>
    <xs:complexType name="TimerType">
       <xs:complexContent>
          <xs:extension
base="TimingMechanismType">
             <xs:sequence>
                <xs:element name="duration"
type="TimeValueType" minOccurs="0"/>
                <xs:element name="isPeriodic"
type="xs:boolean" minOccurs="0"/>
             </xs:sequence>
          </xs:extension>
       </xs:complexContent>
    </xs:complexType>
```

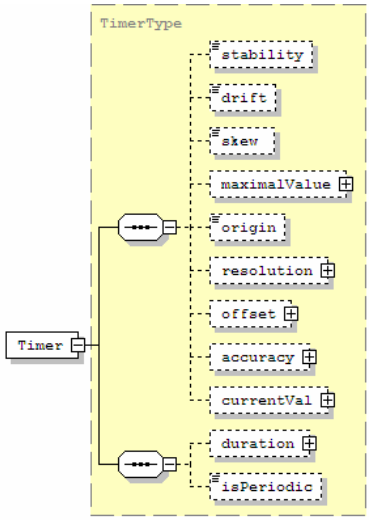

Figure 6 Timer

3.2 Timed Stimulus Model

Following the transformation method introduced in the last section, we are going to derive the Timed Stimulus model and convert it into a XML Schema. The Timed Stimulus Conceptual UML diagram is reproduced from the OMG Real Time Profile which is shown in Figure 7. In some cases, we realize there is need to specify QoS parameters to ensure the reliability of the data. For instance, in Stimulus class, we have added an extra attribute 'validity' to indicate the timeliness of data, which determine its usefulness in regards to time from a particular 'Stimulus' instance. In this section, we are mostly concerned with:

1) Stimulus; 2) Timed Stimulus; 3) Clock Interrupt and 4) Timeout

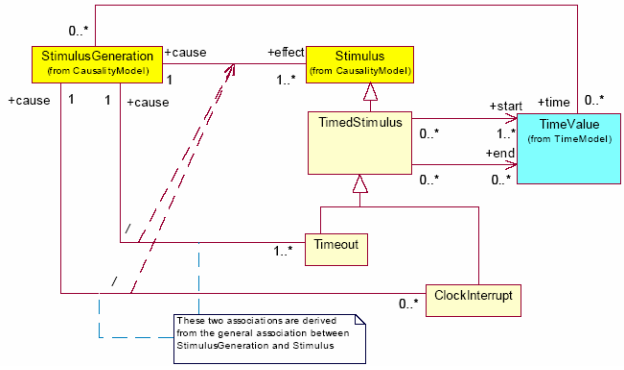

Figure 7 Timed Stimulus Concept (Reproduced from OMG Real Time Profile)

The XML Schema and the DOM model for Stimulus is shown in Figure 8. The validity of Stimulus is typed Time Value.

3.2.1 Stimulus

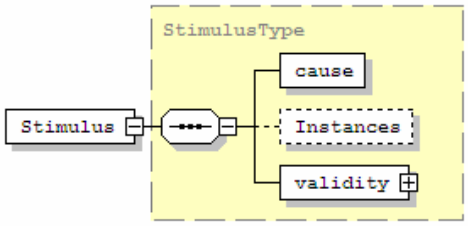

Figure 8 Stimulus

```
   <xs:element name="Stimulus"
type="StimulusType"/>
    <xs:complexType name="StimulusType">
       <xs:sequence>
          <xs:element name="cause">
             <xs:key name="Stimulus_Gen_Key">
                <xs:selector
xpath="StimulusGeneration"/>
                <xs:field
xpath="@Stimulus_GEN_ID"/>
             </xs:key>
             <xs:keyref
name="Stimulus_Gen_KeyRef"
refer="Stimulus_Gen_Key">
                <xs:selector
xpath="StimulusGeneration"/>
                <xs:field
xpath="@Stimulus_GEN_ID"/>
             </xs:keyref>
          </xs:element>
          <xs:element name="Instances"
minOccurs="0">
             <xs:complexType/>
             <xs:keyref name="Instance_KeyRef"
refer="Instance_Key">
                <xs:selector xpath="Instance"/>
                <xs:field
xpath="@Instance_ID"/>
             </xs:keyref>
             <xs:key name="Instance_Key">
                <xs:selector xpath="Instance"/>
                <xs:field
xpath="@Instance_ID"/>
             </xs:key>
          </xs:element>
          <xs:element name="validity"
type="TimeValueType"/>
       </xs:sequence>
       <xs:attribute name="Stimulus_ID"
type="xs:string"/>
    </xs:complexType>
```

3.2.2 Timed Stimulus

The XML Schema and the DOM model for Timed Stimulus is shown in Figure 9.

```
   <xs:complexType name="TimeStimulusType">
      <xs:complexContent>
         <xs:extension base="StimulusType">
            <xs:sequence>
               <xs:element name="start"
type="TimeValueType"/>
               <xs:element name="end"
type="TimeValueType"/>
            </xs:sequence>
         </xs:extension>
      </xs:complexContent>
   </xs:complexType>
```

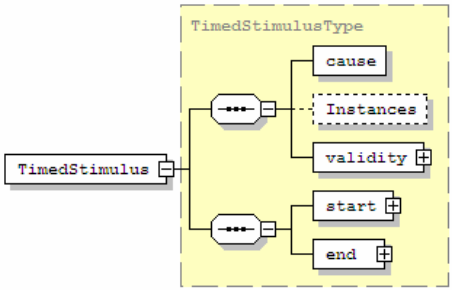

Figure 9 Timed Stimulus

3.2.3 Clock Interrupt

The XML Schema and the DOM model for Clock Interrupt is shown in Figure 10.

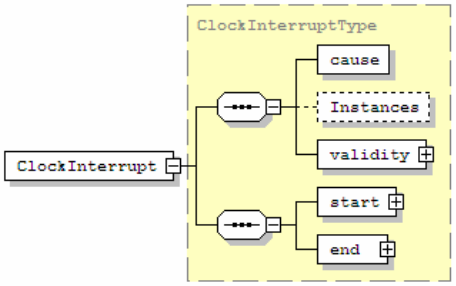

Figure 10 Clock Interrupt

```
   <xs:element name="ClockInterrupt"
type="ClockInterruptType"/>
    <xs:complexType name="ClockInterruptType">
       <xs:complexContent>
          <xs:extension
base="TimeStimulusType">
             <xs:sequence>
                <xs:element name="cause"
minOccurs="0" maxOccurs="unbounded"/>
             </xs:sequence>
          </xs:extension>
       </xs:complexContent>
   </xs:complexType>
```

3.2.4 Timeout

The XML Schema and the DOM model for Timeout is shown in Figure 11.

```
   <xs:element name="Timeout"
type="TimeoutType">
      <xs:key name="Timeout_Key">
         <xs:selector
xpath="StimulusGeneration"/>
         <xs:field xpath="@Stimulus_GEN_ID"/>
      </xs:key>
      <xs:keyref name="Timeout_KeyRef"
refer="Timeout_Key">
         <xs:selector xpath="Timeout"/>
         <xs:field xpath="@Timeout_ID"/>
      </xs:keyref>
   </xs:element>
   <xs:complexType name="TimeoutType">
      <xs:complexContent>
         <xs:extension
base="TimedStimulusType"/>
      </xs:complexContent>
   </xs:complexType>
```

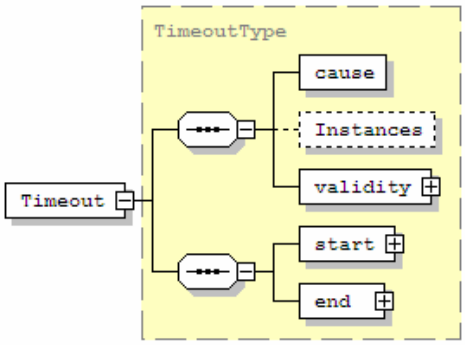

Figure 11 Timeout

4. Conclusion

In this paper, we outline a technique for generating the XML Schema and tags necessary for XML Messaging and XML repository in real time systems. This is an important step towards defining RTML.

References

[1] Chang, E. Annal, D. and Grunta, F. "A Large Scale Distributed Object Architecture - CORBA & COM for Real Time Systems". IEEE conference on Object-Oriented Real Time Distributed Systems. March 12-17, 2000 CA, USA.

[2] Object Management Group, UML™ Profile for Schedulability, Performance, and Time Specification. Available at http://www.omg.org/technology/documents/formal/schedulability.htm

[3] L. Feng, E. Chang, T. Dillon, "Schemata Transformation of object-oriented conceptual models to XML", International Journal of Computer Systems Science & Engineering, CRL Publishing Ltd, 2003, P. 45-60

[4] B. Selic, "The emerging real-time UML standard", International Journal of Computer Systems Science & Engineering, CRL Publishing Ltd, 2002, P. 67-76

[5] Object Management Group, RFP for Scheduling, Performance and Time, OMG document number ad/99-03-13 (March 1999)

[6] W3C Consortium; "XML Schema Part 0: Primer"; http://www.w3.org/TR/xmlschema-0/

[7] W3C Consortium; "XML Schema Part 1: Structures"; http://www.w3.org/TR/xmlschema-1/

[8] W3C Consortium; "XML Schema Part 2: Datatypes"; http://www.w3.org/TR/xmlschema-2/

[9] W3C Consortium; "XML Schema"; http://www.w3c.org/XML/Schema

[10] W3C Consortium; "Document Object Model"; http://www.w3.org/DOM