# An Event Notification Service based on XML Messaging on Different Transport Technologies

Ravinder Thirumala[1] , Usha Rani[2], Timothy A. Gonsalves[1]
[1]Department of Computer Science and Engineering,
Indian Institute of Technology Madras, Chennai, India
[2]NMSWorks Software, Chennai, India
{ravinder,tag}@lantana.tenet.res.in, usha@nmsworks.co.in

## Abstract

*With the size and increasing complexity of telecom networks, there is a need to interconnect management systems at different levels. This requires information flow across many applications in a domain independent way. XML is a widely deployed standard which is being used for integration of Network Management System(NMS) with other applications. We examine the performance of different transport mechanisms, JMS, CORBA, HTTP and RMI for an XML message based event Notification Service. To improve the performance of XML message based event notifications, event grouping is examined and is found to perform well.*

## 1. Introduction

The Introduction of convergent services and new technologies in telecom networks and the rapid increase in number of services have resulted in an increase in the number of network management processes. Automated and cost effective management system for services and communication network requires interconnection among different operation processes and linkage between service operations and Network Management Systems.

In the Telecommunications Management Network (TMN) layered architecture, the Element Management Layer (EML) manages technology and vendor specific subnetworks. The NML provides a complete network view and aggregates information from EML and passes to the SML. It receives requests from the SML, process them and pass on relevant commands and data to the appropriate EML. The SML is concerned with manging services provided to customers.

XML is a widely deployed standard which is supported by a large number of applications. XML provides a domain independent, interoperable cost effective, open and standardized management interface. It is platform independent. Large amounts of data can be represented and transmitted in a single XML message. XML messages are easy to generate, parse and process.

Event Notification is critical for Network Management. We design an XML-based event notification service. As events are time-sensitive, we examine the performance of different transport options. We use JMS [13], CORBA [4], HTTP [12], and RMI [14] as transport options. We also compare the performance of XML based event Notification Services with SNMP [3] Inform Request, which is used to notify a peer manager in SNMP based Network Management. Unlike SNMP traps which are mostly used for event notification between agent and manager, for every SNMP Inform Request sent a Response will be received.

XML encoding is known to be verbose, which can affect performance. Therefore, to improve performance of Event Notification Service based on XML messaging, we propose to use event grouping. In this method we encode multiple events in a single XML message for transmission to the event receiver.

Section 2 gives an overview of related work. Section 3 discusses the different transport options used. Section 4 gives details about the experiment setup and presents the performance results. Section 5 discusses event grouping. Conclusions and future work are given in Section 6.

## 2. Related Work

In this section, we describe standardization activities and research work done in the area of application of XML technologies for network management and XML based event notification service.

### 2.1. Standardization Activities

**NetConf:** The NetConf [6] Working Group is chartered by IETF to produce a protocol suitable for configuration Management. The NetConf protocol uses XML for data encoding. Work is in progress to define a framework for sending event notifications. They have defined the operations necessary to support event Notifications, and also discuss implications for the mapping to application protocols. Events and operations to support event notifications are encoded as XML messages.

**Multi-Technology Operations Systems Interface (MTOSI) [15]:** The Telecom Management Forum's (TMF's) effort to define a unified open interface to be used between Operations Systems (OSs) for network and service management. MTOSI supports XML/Web Service interactions between various types of Operations Systems. MTOSI defines Notification service whose interface is based on Service Oriented Architecture (SOA). Operation messages and events are encoded in XML.

**Operations Support Systems through Java Initiative (OSSJ) [11]:** The OSSJ Initiative defines and implements an open, standard set of Java technology-based APIs that facilitate the implementation of end-to-end services on next-generation communications networks. OSS through Java supports XML/JMS style of interactions between OSS Components.

**Web Services Notification (WSN) [10]:** Web Services Notification (WSN) is a family of related specifications proposed by the Organization for the Advancement of Structured Information Standards (OASIS), that defines a standardized way for Web services to interact using Notifications or Events based on topic-based publish/subscribe pattern. The Web Services Base Notification specification describes the notification interfaces, and details the message exchanges required to fulfill these interfaces. The Web Services Topics specification defines a mechanism to organize and categorize items of interest for subscription. The Web Services Brokered Notification specification identifies the interfaces, message exchanges, and semantics associated with intermediaries known as notification brokers.

### 2.2. Research work

Hong et al. propose an XML-based Management (XBM) architecture [9] that is based on XML/HTTP as its management protocol. To allow the integration of SNMP managed devices, they propose XML/SNMP gateway, they compared performance of XML based agent with SNMP agent and also compared message size of SNMP Trap with XML encoded notification message. They transported notifications encoded as XML messages over HTTP.

George Pavlou et al. compared the performance of Web services with SNMP and CORBA [5]. They measured the response times and traffic for retrieval of TCP MIB variables and found Web services is promising technology but, being XML-based, has more overhead than SNMP and CORBA.

Lawrence Menten [8] explored possibilities of XML-based Device Management, In the process they have created an architecture, tools, and reusable libraries that can help significantly reduce the development cycle and the cost to develop and maintain remotely managed devices.

Jagadish et al. [2] propose intermittent dial-up as a cost-effective management network for rural telecom operators. They classify alarms by priority and group alarms to be transmitted upon dial-up.

Several papers address the application of XML technologies for device management [8], comparison of XML and SNMP [1] and XML/SNMP gateways [9, 7].

## 3. Event Notification Service based on XML Messaging

As a part of open interface Event Notification Service is essential for management of networks. Many service operations processes like Inventory Management, Service Impact Analysis, Failure Auditing, etc need event notifications from Network Management Systems.

In this paper we address application of XML technologies at Network management layer, which allow easy integration with other applications. As XML messaging is transport-independent, An Event Notification Service based on XML Messaging on different transport mechanism is examined. The select transport mechanisms are JMS, CORBA, HTTP, RMI. We also compare the performance of XML with SNMP Inform Request, which is used to notify peer manager in SNMP based Network Management. We encoded Events as XML messages. For this we defined

```
<Alarm>
  <notifId>1239</notifId>
  <objName>
    <mdName>NMSWorks/CygNetEMS</mdName>
    <meName>ADM</meName>
    <eqName>shelf=2/slot=3/port=1</eqName>
  </objName>
  <aType>EQUIP</aType>
  <aTime>20060106210627.3</aTime>
  <isClearable>true</isClearable>
  <layerRate>LR_Line_OC3_STS3_and_MS_STM1</layerRate>
  <perceivedSeverity>MAJOR</perceivedSeverity>
  <serviceAffecting>SERVICE_AFFECTING</serviceAffecting>
  <probableCause>LOS</probableCause>
  <aText></aText>
</Alarm>
```

**Figure 1. Sample Alarm encoded as XML Message**

XML Schema which is based on ITU-T X.733 standard. Figure 1 shows a sample event. For SNMP Inform Request we mapped XML Schema elements to OID's.

### 3.1. Transport Technologies used

**JMS** Java Message Service is an API, that can be used to asynchronously send and receive events. JMS supports both message queuing and publish-subscribe styles of messaging. We used the Sun Java System Application Server as the messaging engine. Both producer and consumer machines run application server. The Producer publishes notifications to the topic which in turn sends to all subscribed consumers.

**Corba Event Service** This is a Corba service that allows multiple event suppliers to send events to multiple event consumers via an event channel. We use the OpenORB-1.3 implementation. Naming and Event services are running in the same machine as the event Notification Producer. Events encoded as XML messages are mapped to type any. We use Push Model for event forwarding.

**HTTP** Hyper Text Transport Protocol is a widely deployed request-response protocol for client-server applications. For sending Asynchronous XML messages over HTTP we implemented event Producer as client with Apache commons-httpclient API. Notification Consumer is implemented as a servlet with a doPost method which receives events as HTTP post message. We used the Jakarta-tomcat-4.1.31 servlet engine.

**RMI** Remote Method Invocation is a Java mechanism for access to a remote object. To send Asynchronous XML messages over RMI, Notification Producer performs a lookup on subscribed Notification Consumer and performs remote method call by passing XML message as a parameter to the method call.

## 4. Performance Evaluation

The Notification Producer and Consumer communicate asynchronously. For uniformity the Notification Producer and Consumer were implemented in Java. Tests were conducted with one producer and one consumer. Latency is defined as the time between sending the first event at the producer and receiving last event at the consumer. We synchronized date and time on producer and consumer systems using rdate with a common source. Latency evaluation was done with varying number of alarms. This is to simulate real time scenario where burst of alarms is common. To minimize errors we average ten measurements.

### 4.1. Experiment Setup

We use Java J2SDK 1.4.2 version on Linux RedHat 7.1. The Event Producer is placed in Intel Celeron 860 MHz processor with 512 MB of RAM and Event Consumer is placed in Intel Pentium III 600 MHz processor with 256 MB of RAM, both were connected through a dedicated 100 Mbps Ethernet.

**Table 1. Latency on different Transport Mechanisms (ms)**

| | **Alarms** | | | |
|---|---|---|---|---|
| | **1** | **100** | **1000** | **5000** |
| **JMS** | 787 | 2,418 | 7,833 | 29,085 |
| **CORBA** | 656 | 1,856 | 4,986 | 20,395 |
| **HTTP** | 40 | 2,090 | 20,000 | 99,158 |
| **RMI** | 7 | 222 | 1,101 | 5,117 |

**Table 2. Latency with SNMP Inform Request (ms)**

| | **Alarms** | | | |
|---|---|---|---|---|
| | 1 | **100** | **1000** | **5000** |
| SNMP | 4 | 212 | 1,258 | 4,996 |

### 4.2. Latency on Different Transport Mechanisms

From Table 1, we see that latency response to transfer one event encoded as XML message on CORBA and JMS is 100 times that of RMI and 15 times more than HTTP. This is because in Corba events are exchanged over a channel and in JMS over topics which act as brokering mechanism between producer and consumer. Latency for HTTP protocol is higher as number of alarms increases, because there is a response for every post message.

### 4.3. Latency with SNMP Inform

From Table 2 latency for SNMP Inform Request is seen to be low compared to XML messaging on different transports. This is because SNMP Inform request uses UDP as its transport protocol and it has smaller message size.

### 4.4. Bandwidth Usage

We measured the traffic generated in transporting one event encoded as an XML message on the different transports. These measurements do not include connection overhead but do include protocol overheads due to headers. These results were compared with SNMP Inform Request.

The Overhead for transporting event encoded as XML Message on JMS is high when compared to other transports (Table 3). This is because more overhead is involved between messaging engines for every alarm transfer. The Overhead with RMI is twice that of Corba because Java uses 16-bit Unicode encoding, HTTP generates more traffic than

**Table 3. Traffic Generated for one Event in bytes**

| **JMS** | **CORBA** | **HTTP** | **RMI** | **SNMP** |
|---|---|---|---|---|
| 1,517 | 658 | 1,035 | 1,055 | 422 |

RMI and Corba due to the HTTP Header. Traffic generated with SNMP Inform is very low because SNMP uses compact BER encoding.

## 5. Event Grouping

In An Network Management System managing tens of thousands of Managed Objects, bursts of events are common because of failure propagation among Managed Elements. To improve the latencies of XML based event Notification Service we investigate Event Grouping. Event Grouping sends multiple events in a single XML Message, invoking a single send operation. This is similar to event batching proposed by the CORBA Notification Service. Event Grouping differs from Corba's event batching. In Event Grouping multiple events are grouped into a single XML, but with Event Batching events are sequenced to be sent as an event burst. Similar to Event Batching, Event Grouping is controlled by two parameters maximum group size in a single message and pacing interval. The Pacing interval is the maximum time to wait for events to be collected. If fewer events than the group size are available when the pacing timer expires, the Notification Service just groups whatever events are available and sends them.

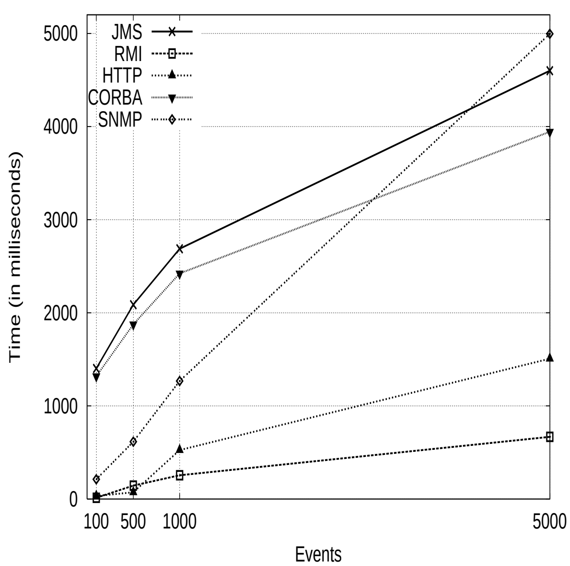


**Figure 2. Latency with Event Grouping**

### 5.1 Latency with Event Grouping

Results as a function of number of Events Grouped are shown in graph (Figure 2). This experiment was performed with varying burst sizes of 100, 500, 1000, 5000 per 30 minutes, and pacing interval is fixed at 30 minutes. It turned out that latencies for RMI and HTTP are 100 times less than that of CORBA and JMS. This is because JMS and CORBA Notification services use broker to transport events to the consumer. With SNMP it steadily increases and at around 4000 events it crosses JMS which has highest latency. This is because SNMP Inform Requests are sent sequentially. It is obvious from the graph that if the grouping size is small the advantage of having event grouping is less.

## 6. Conclusion and Future Work

This paper compared the performance of Event Notification Service based on XML messaging on different transport technologies and found RMI better, but it lacks Publish and Subscribe Mechanism. The verbose nature of XML consumes more bandwidth when compared with SNMP. To improve performance we used Event Grouping which invokes a single send operation to send multiple events in a single XML message. The performance implications of having event filter, multiple event subscribers and compression of Event Grouping Message will be studied in future.

## References


[1] Aiko Pras et al. Comparing the Performance of SNMP and Web Services-Based Management. *IEEE eTNSM (Trans. on Net. and Service Mgmt.)*, 1(2):1–11, Dec 2004.

[2] C. Jagadish, V. Kumar, and T. A. Gonsalves. Low-cost DCN for Rural Telecom Network Management. Technical report, DON Lab, Computer Science Dept., Indian Institute of Technology, Madras, 2006.

[3] J. Case, M. Fedor, M. Schoffstall, and J. Davin. RFC 1157 - A Simple Network Management Protocol (SNMP), May, 1990.

[4] G. Coulouris, J. Dollimore and T. Kindberg. *Distributed Systems: Concepts and Design*. Addison-Wesley, 2001.

[5] Geogre Pavlou et al. On Management Technologies and the Potential of Web Services. *IEEE Commun. Mag.*, 42(7):58–66, July, 2004.

[6] IETF. Network Configuration (Netconf). www.ietf.org/html.charters/netconf-charter.html.

[7] T. Klie and F. Strauss. Integrating SNMP Agents with XML-Based Management Systems. *IEEE Commun. Mag.*, 42(7):76–83, July 2004.

[8] L. Menten. Experiences in the Application of XML for Device Management. *IEEE Commun. Mag.*, 42(7):92–100, July 2004.

[9] Mi-Jung Choi et al. XML-Based Network Management for IP Networks. *ETRI J.*, 25(6):445–458, Dec, 2003.

[10] OASIS. Web Services Notification (WSN). http://www.oasis-open.org.

[11] OSSJ. Operations Support Systems through Java Initiative. www.ossj.org.

[12] R. Fielding et al. RFC 2616 - Hypertext Transfer Protocol (HTTP), June, 1999.

[13] Sun Microsystems. Java Messaging Service (JMS). www.java.sun.com/products/jms.

[14] Sun Microsystems. Java Remote Method Invocation (Java RMI). www.java.sun.com/products/jdk/rmi/.

[15] TMForum. Multi-Technology Operations Systems Interface MTOSI. www.tmforum.org.