

---

# Interaction Translation Methods for XML/SNMP Gateway Using XML Technologies

Yoon-Jung Oh, Hong-Taek Ju and James W. Hong  
{bheart, juht, jwkhong}@postech.ac.kr

Distributed Processing & Network Management Lab.  
Graduate School for Information Technology  
POSTECH

## Abstract

XML-based network management has been proposed as an alternative or to compliment SNMP-based network management, which has constraints in scalability and efficiency. But the XML-based network management cannot provide a method to manage networks equipped with legacy SNMP agents in the integrated management system. This integrated management system must include an XML/SNMP gateway, which translates and relays messages between the XML-based manager and the SNMP agent. The gateway must provide both specification translation and interaction translation between the two management applications.

In this paper, we propose several methods for interaction translation in the gateway using XML technologies (namely DOM, XPath, XQuery and SOAP). In our approaches to the translation, we first distinguish levels of translation into three: process level, message level, and protocol level. In the process level translation, we translate standard DOM interfaces to SNMP operations, which provides a method for XML-based manager to directly access management information through the DOM interfaces. In the message level translation, we translate HTTP messages through URI extension with XPath and XQuery, which provide methods to define detailed request message for XML/HTTP communication. Finally, we apply the SOAP protocol, which is accepted as a standard protocol for XML, and propose method for the gateway to advertise the translation services to the manager using the SOAP RPC. We also compare the advantages and disadvantages of the proposed translation methods.

# Introduction

---

## ❖ Backgrounds

- XML-based network management has been proposed as an alternative and to compliment SNMP-based network management.
- But XML-based network management does not yet provide standard methods to manage network devices, equipped with SNMP agents.
- An XML/SNMP gateway is needed for an XML-based integrated management
- The XML/SNMP gateway must provide both specification translation and interaction translation between the two management applications.
- No standardized interaction translation method for the gateway exists yet.

## ❖ Proposed Interaction Translation Methods for XML/SNMP Gateway

- **Process Level Method:** Translation between **DOM interfaces** and **SNMP operations**
- **Message Level Method:** Translation of **HTTP messages** using **URI** extensions with **XPath** and **XQuery**
- **Protocol Level Method:** Translation based on **SOAP RPC** between manager and gateway

XML-based network management (XNM) [2, 3] has been proposed as an alternative or to compliment existing SNMP-based network management [20], which has constraints in scalability and efficiency. XNM has many advantages. First, XML [9] provides powerful modeling features for structured management information in network management, and XNM can transfer large amount of data over HTTP. Also XNM can be easily implemented using standard API and freely available software. However, XNM does not provide standard or established methods to manage widely deployed network devices, which are already equipped with SNMP agents through XML-based managers. Therefore, a method to manage SNMP agents in XML-based integrated management is needed.

An XML/SNMP gateway provides a method to manage networks equipped with SNMP agents in an XML-based integrated management system. The gateway must provide both specification translation and interaction translation between the two management applications. Recent research work on MIB (SMI) to XML (XML Schema) translation [3, 4] provides the foundation for the XML/SNMP Gateway. However, there is no concrete or standardized interaction translation method for the gateway exists yet. Most XML-based applications implement various communication methods between peer systems because XML does not specify any standard communication protocol. Therefore, research on concrete and verified methods for the XML-based systems to efficiently exchange messages is necessary.

In this paper, we propose three methods for efficient interaction translation in the XML/SNMP Gateway for XML-based integrated network management. In our approach to the translation methods we first distinguish the levels of translation into three: process level, message level, and protocol level. In the process level translation, we describe mappings between Document Object Model (DOM) [6] interfaces and SNMP operations. The gateway uses the DOM structure and its interfaces to translate management information between XML and SNMP protocols. We analyze the functions and meanings of the DOM interfaces on management information, and translate the interfaces into appropriate SNMP operations. This method provides a method for XML-based manager to directly access and manipulate management information through the DOM interfaces for XML-based manager. In the message level translation, we translate HTTP messages through URI [15] extensions with XPath [8] and XQuery [9]. This method provides an efficient mechanism for querying management information in XML/HTTP communication, which is the most common method for exchanging XML documents. Finally, we apply Simple Object Access Protocol (SOAP) [17], which is accepted as a standard protocol for XML. In this protocol level translation, we propose SOAP-based communication between XML-based manager and XML/SNMP gateway. The gateway advertises its translation services to the manager using SOAP RPC [19].

# Related Work

---

## ❖ XML Technologies

- **DOM (Document Object Model)**: Platform- and language-independent interface that allows applications to dynamically access and manipulate the content, structure, and style of documents.
- **XPath (XML Path Language)**: Language for addressing specific parts of an XML document.
- **XQuery (XML Query Language)**: Query language for XML, designed to be broadly applicable across various types of XML data sources.
- **SOAP (Simple Object Access Protocol)**
  - Simple and lightweight protocol for exchanging structured and typed information in a distributed environment using XML.
  - Defines the use of XML and HTTP to access services, objects and servers in a platform-independent manner.
  - Consists of an envelope, the encoding rules, and the RPC representation.

We first briefly explain XML technologies that are applied to our proposed translation methods.

Document Object Model (DOM) [6] is a platform- and language-independent interface that allows applications to dynamically access and manipulate the content, structure, and style of documents. DOM also allows reconstruction of the complete document from the input XML document, and access to any portions of the document.

XPath [8] is a language for addressing parts of an XML document. XPath has a compact, non-XML syntax, and can be used within URIs and XML attribute values. It operates on the abstract and logical structure of an XML document. One important type of XPath expression is a location path, which selects a set of nodes relative to the context node, and can recursively contain sub-expressions that are used to filter a set of nodes.

XQuery [9], a query language for XML, is designed to be broadly applicable across all types of XML data sources, such as structured and semi-structured documents, relational databases, and object repositories. XQuery contains an XPath as a subset, and provides several kinds of expressions, which may be constructed from keywords, symbols, and operands. XQuery provides constructors that can create XML structures within a query. In a constructor, curly braces “{}” delimit enclosed expressions, distinguishing them from literal text. Enclosed expressions are evaluated and replaced by their value.

Simple Object Access Protocol (SOAP) [16, 17, 18] provides a simple and lightweight protocol for exchanging structured and typed information between peers in a decentralized, distributed environment using XML. SOAP defines the use of XML and HTTP to access services, objects and servers in a platform-independent manner. SOAP consists of an envelope, the encoding rules and the RPC representation. The SOAP envelope defines the overall framework for expressing what is in a message, who should deal with it, and where it is optional or mandatory. The SOAP encoding rules define a serialization mechanism that can be used to exchange instances of application-defined data types. The SOAP RPC [19] representation defines a convention that can be used to represent remote procedure calls and responses. SOAP can be used in combination with a variety of other protocols such as HTTP and SMTP.

# Related Work (cont'd)

## ❖ XML-based Network Management

- Uses XML DTD or Schema for management information model.
- Exchanges management data in the form of XML document.
- Uses standard XML technologies for processing the data.
- Web-based Integrated Management Architecture (WIMA) : J.P. Martin-Flatin, EPFL, 2000
  - Presented the HTTP-based management communication model
- XML-based Network Management (XNM) : H. T. Ju, POSTECH, 2001
  - Extended the use of EWS from direct Web user interface to Web-based management agent by using standard Web and XML technologies

## ❖ SNMP to XML Translator

- Automatic translation from SNMP MIB to XML.
- Translation Models: J.P. Martin-Flatin, EPFL, 2000
  - Model-level mapping & Metamodel-level mapping
- SNMP MIB to XML Schema mapping: J. H. Yoon, POSTECH, 2001
  - Validated by developing an XML-based SNMP MIB browser.
- Library to access SMI MIB (libsmi): Frank Strauss, 2000

XML-based network management (XNM) [2, 3] defines management information using XML DTD or XML Schema [11, 12, 13]. It exchanges management data in the form of XML documents, and uses standard XML technologies for processing the data. XNM has many advantages. First, XML provides powerful modeling features for structured management information in network management, and XML-based network management can transfer large amount of data over HTTP.

J.P. Martin-Flatin presented an idea to use XML for integrated management in his research on Web-based Integrated network Management Architecture (WIMA) [3]. He described the advantages of HTTP/XML-based communication, and presented the basic idea concerning SNMP MIB to XML conversion. As another approach to XNM, we proposed an XML-based network management using EWS (Embedded Web Server) [2] in our previous research. In this work, we extended the use of EWS for element management to network management. We also proposed network management using XML to transfer management information over HTTP between an agent and a manager.

As mentioned above, XML/SNMP gateway must provide both a specification translation and an interaction translation. For the specification translation, the gateway translates SNMP MIB into XML. J.P. Martin-Flatin, in his paper where presented WIMA, proposed SNMP MIB to XML translation models, namely Model-level mapping and Metamodel-level mapping [3]. Model-level mapping is a method where DTD or XML Schema is made out specifically to an SNMP MIB, which make elements and attributes in XML DTD by using SNMP variable identifiers just as they are. The advantage of using XML documents based on this model is that readability is superior and it is easy to intuitively understand. The disadvantage is that this model needs as many DTDs or XML Schema as MIB modules. The Metamodel-level mapping method defines one generalized DTD or XML Schema, and applies to all MIBs. This model does not use variables defined in MIB but uses general keywords for defining XML elements.

In our previous work on the XML/SNMP Gateway, we defined an SNMP MIB to XML translation algorithm, and also implemented an automatic SNMP MIB to XML translator using this algorithm [1]. For validation of the algorithm, we had implemented an XML-based SNMP MIB browser using this SNMP MIB to XML Translator.

Frank Strauss presented a library to access SMI MIB information, “libsmi” [4], which translates SNMP MIB to other languages, such as JAVA, CORBA, C, XML etc. This library provides a tool for MIB dump into an XML document based on metamodel-level schema mapping.

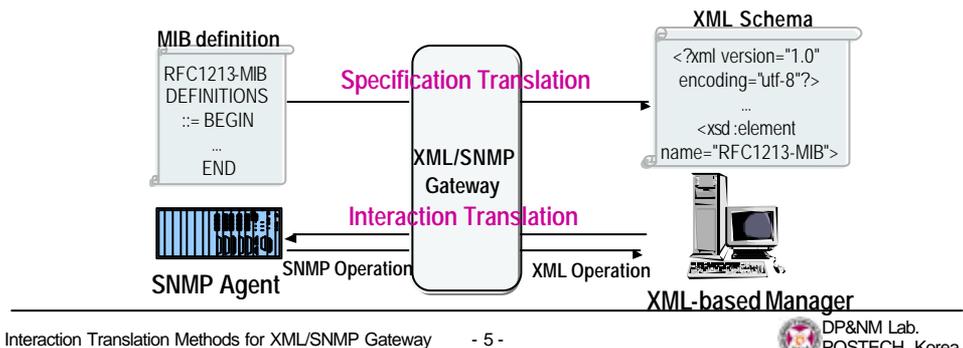
# XML/SNMP Gateway

## ❖ What is XML/SNMP Gateway?

- Relays management information and operations between SNMP agent and XML-based manager system.
- Provides a method to manage networks equipped with SNMP agents using XML-based manager.

## ❖ Basic Requirements for the gateway

- Specification translation and Interaction translation



An XML/SNMP gateway provides a method to manage networks equipped with SNMP agents in an XML-based integrated management system [1]. The XML-based manager transfers XML messages using HTTP, and the SNMP agent passes SNMP messages using SNMP. The gateway converts and relays management data between the two management applications.

The XML/SNMP Gateway provides the SNMP MIB to XML Translator, which is called the specification translation [1]. The specification translator translates the SNMP MIB definition into an XML Schema definition without loss of management information. The translator generates an XML Schema file and an XML DOM tree structure. The XML Schema file is used for validation of each instance of the XML document, and DOM is used for generating and processing XML documents which contain management data. This specification translation algorithm is the basis of an interaction translation in our current work. For interaction translation from SNMP operations to XML operations, we propose interaction translation methods for the XML/SNMP gateway.

These translation methods will be described in the following pages.

# Interaction Translation Methods

❖ Interaction Translation is based on Specification Translation Result by SNMP MIB to XML Translator

❖ Levels of the translation methods

– **Process Level Translation**

- Translation between DOM Interface and SNMP Operation
- provides a method to directly access management information through the DOM interfaces for XML-based manager.

– **Message Level Translation**

- Translation of **URI-based HTTP requests** from XML-based manager into **SNMP requests**.
- provides an efficient mechanism for querying managed objects in XML/HTTP communication.

– **Protocol Level Translation**

- Interaction Translation based on **SOAP RPCs** between XML-based manager and XML/SNMP Gateway

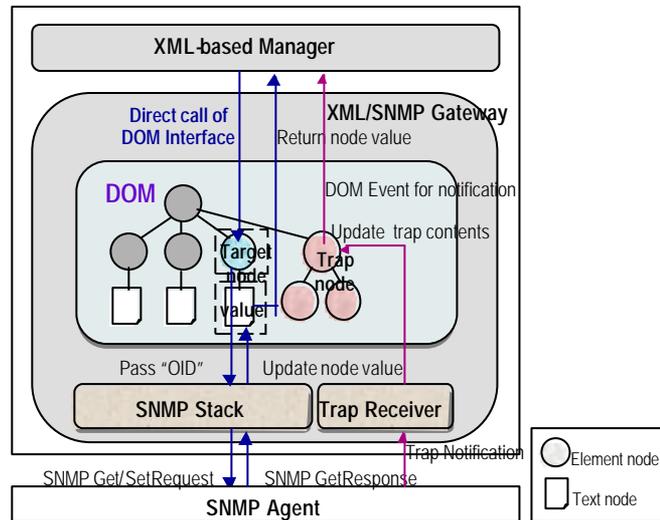
Our approach to interaction translation methods for the XML/SNMP Gateway distinguish the levels of the approach into three: a process level, a message level, and a protocol level. In the process level translation, we translate standard DOM interfaces to SNMP operations, which provides a method to directly access management information through the DOM interfaces for XML-based manager. In the message level translation, we extend URI with XPath and XQuery, which provide methods to define detailed request message for XML/HTTP communication, and thus better efficiency in the communication. Finally, we apply the SOAP protocol, which is accepted as a standard protocol for XML, and propose a method for the gateway to advertise the translation services to the manager using SOAP RPC.

The XML/SNMP Gateway transfers management information to the XML-based manager converting SNMP agent's MIB into an XML format. The gateway uses DOM as a structure for information translation and as an intermediate storage for management data. Using standard DOM interfaces, the gateway manipulates the DOM tree. The translator of the gateway converts each node of SNMP MIB into an element of the XML Schema, and the name of the MIB node into the name of the element. Interior clauses of the MIB node such as "access", "status", etc. are translated into the attributes of the XML element. The table in the bottom shows the conversion of document structure between SNMP SMI and XML Schema. Each MIB node in the DOM tree of the gateway can be accessed and manipulated using its corresponding DOM interface.

SNMP SMI	XML Schema	DOM Interface
MIB Module	XML document	Document
MIB Module name	Root Element name	Element::tagName
Leaf node (macro definition)	Element with (a) child text node(s)	Element
Node name	Element name	Element::tagName
Clauses in node	Attributes of an element	Attr
Object Identifier(OID)	Attribute of type "ID"	Attr

# Process Level Translation

## ❖ Translation between DOM Interface and SNMP Operation



In our first approach to interaction translation of the gateway, we propose the process level translation, in which a DOM interface call from the XML-based manager is translated into an SNMP operation. This method is very useful in case of an internal gateway, which is integrated within the management system. The manager can directly access management data in the DOM using DOM API provided by the gateway. In such case, a key role of the gateway is to maintain DOM contents consistent with the SNMP agent's MIB.

As a result of a specification translation, a DOM tree in the gateway consists of elements mapped to nodes in a MIB. The value of a leaf node in the MIB is stored in the text node of an element in the DOM. The text node for the value of the leaf node is initialized with a null value. To provide an up-to-date SNMP MIB value from an agent to the manager, it is desirable that the gateway updates the DOM tree whenever the MIB in the agent is modified. However, it is both inefficient and difficult to implement. In our approach, the gateway sends an SNMP request and updates a node value with returned data, only if the manager accesses the child text node of a leaf node: that is, the manager invokes `Element::firstChild` or `Element::nextSibling`. If the target node is non-leaf node, the interface simply returns null. In this way the gateway maintains efficiently the DOM tree consistent with the agent's MIB. In case of a columnar object, which has variable numbers of instances, an element for the columnar object in the DOM consists of a list of child text node as the same number of the instances. And these children are created and deleted dynamically as the instances of the columnar object are changed.

The figure in the slide illustrates how the manager retrieves a MIB value using DOM API involving an SNMP operation. A DOM interface call from the XML-based manager is translated into an SNMP `GetRequest` or `SetRequest`, and returns a node value from SNMP response to the manager after modifying the DOM content. The SNMP `GetBulk` operation, which includes multiple accesses to different nodes in the MIB tree, can be simply implemented by iteration of a DOM interface. For notification delivery, the Trap Receiver in the gateway receives a trap notification from the SNMP agent, and updates trap contents in the DOM. DOM Event interface invokes an appropriate event handler in order to notify modification of the trap content, and delivers it to the manager.

The DOM interfaces provide methods to access a node in the DOM tree by its attribute of type "ID" along with an element name, and it is easy to traverse the hierarchy without the "oid" operation using the interfaces. The DOM Traversal Interface provides different logical views using a filtering interface, and the DOM Range Interface provides methods to access and manipulate the document tree within a specified range [7]. This translation method can extend its functionality through these additional features of DOM interfaces and the XML/SNMP Gateway can easily integrate legacy SNMP agents into XML-based management through the translation between DOM interface-based requests to SNMP requests and vice versa.

# Message Level Translation

## ❖ Translation of URI-based HTTP requests from XML-based manager into SNMP requests.

- Extends URI string with XPath and XQuery expressions
- Provides efficient ways to retrieve MIB objects in XML/HTTP communication
- Examples of XPath and XQuery expression included in URI string

### •XPath with URI

http://example.org/gateway/device[@type="server"]?community=public&version=1&operation=get...

### •XQuery string

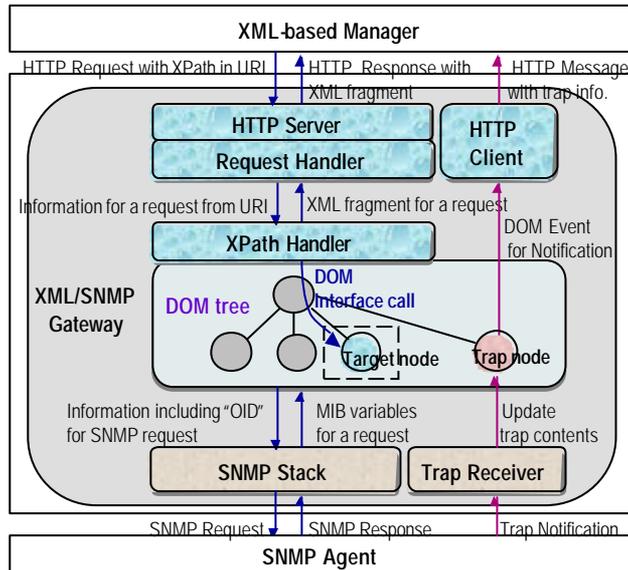
```
<result> {  
Let $t := input() //ifTable/ifEntry/ ifType[contains( ./text(), "ethernet")]  
Return  
<totalInOutOctets count="{count($t) }">  
<in> { sum($t/ifInOctets/text()) } </in>  
<out> { sum($t/ifOutOctets/text()) } </out>  
</ totalInOutOctets > }  
</result>
```

In the message level translation method, the XML/SNMP gateway translates URI-based HTTP requests from XML-based manager to SNMP requests. We extend URI [15] with XPath [8] and XQuery [9]. This method provides efficient ways to retrieve MIB objects in XML/HTTP communication, which is the most common for the exchange of XML documents.

When a manager requests management information, addressing managed objects is important in the message between the manager and the gateway. We applied the XPath language in an URI string to indicate a target object. The XPath is a standard for addressing parts of an XML document and provides a rich addressing mechanism for an efficient and effective query on management information.

Another powerful and structured facility is provided by Xquery, which is a query language that provides query expressions across various types of data sources, such as structured and semi-structured documents, relational databases, and object repositories. XQuery uses XPath as a subset, and easily expresses a complicated query. For example, when a manager retrieves the total number of in/out octets of a network interface with type "Ethernet" in the interface table, the manager must send a number of requests to retrieve all instances of the objects in the table, and then calculate the total number of octets from the returned values. The example below simplifies the requests for this case and reduces the amount of data transferred between a manager and a gateway using XQuery.

# Message Level Translation (cont'd)



The figure in the slide illustrates the translation of HTTP requests into SNMP requests on the basis of XPath included the URI string. A CGI-like interface called “Request handler” in the gateway receives and parses XPath expression as input with several arguments in the URI, and delivers the request to the gateway application. XPath Handler in the gateway application analyzes the XPath expression, and returns a list of target nodes. The DOM interface call on the target nodes is translated into SNMP request in the same way as in the previous translation method. For notification delivery, HTTP Client of the gateway sends an asynchronous event message from the DOM Event interface to the manager.

XQuery also provides such features as filtering a document to produce a table of contents, joins across multiple data sources, grouping and aggregates the contents, and queries based on sequential relationships in documents. The XML-based manager can reduce the number of requests and data transfer for further processing in the manager such as complicated statistical analysis. Thus, it can improve management efficiency by adopting XPath and XQuery in a message format.

# Protocol Level Translation

## ❖ Interaction Translation based on SOAP RPCs between XML-based manager and XML/SNMP Gateway.

- SOAP defines a standard method to transfer XML-encoded messages over HTTP.
- Basic XML elements for SOAP RPC messages :

**getRequest / setRequest / getResponse / Trap**

### ▪ **getRequest**

```
<m:getRequest xmlns:m="http://example.org/gateway ">
<m:community>public</ m:community><m:version>1</m:version>
<m:path>// ifSpeed[1]</m:path></m:getRequest>
```

### ▪ **setRequest**

```
<m:setRequest xmlns :m="http://example.org/ gateway ">
<m:community>media</m:community><m:path>//hostInfo/hostname</m:path>
<m:value>myHost</m:value></m:setRequest>
```

### ▪ **getResponse**

```
<m:getResponse xmlns:m="http://example.org/ gateway ">
<rpc:result xmlns:rpc="http://www.w3.org/2001/12/soap-rpc">
<ifSpeed>64000</ifSpeed></rpc:result></m:setResponse>
```

SOAP [16] is a protocol for exchanging XML-based messages over HTTP or SMTP. At the basic functionality level, SOAP can be used as a simple messaging protocol, and also can be extended to an RPC protocol [23]. In this section, we propose a SOAP-based communication method between XML-based manager and XML/SNMP Gateway.

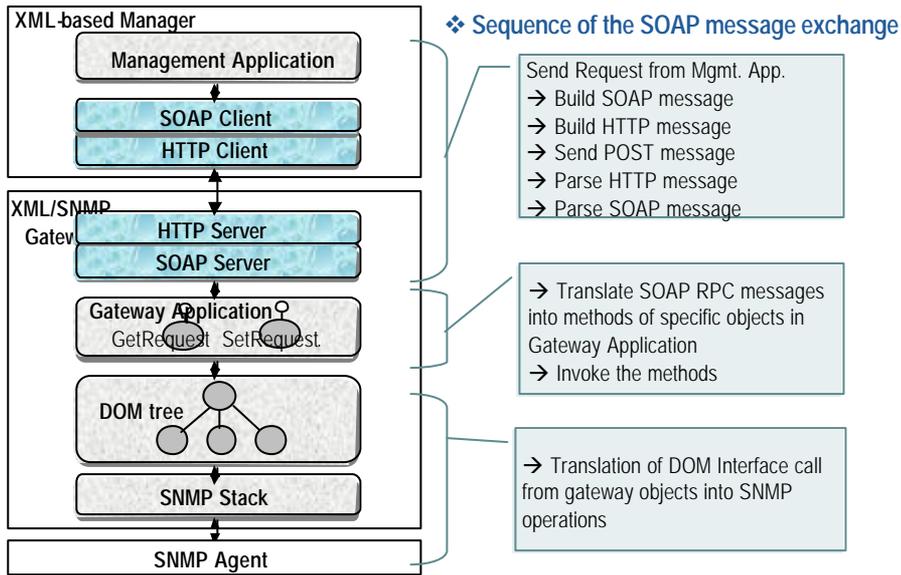
XML-based manager exchanges an XML encoded message with communication peers, such as an XML-based agent and an XML/SNMP gateway, which means that a request message from the manager and a response message from the gateway are all formatted as an XML document. SOAP defines a standard method to transfer XML-encoded messages over HTTP. In this slide, we describe the message format for the SOAP-based RPC [19] between the manager and the gateway.

Basically, we define three types of XML elements for the SOAP RPC messages between an XML-based manager and an XML/SNMP Gateway. These elements are both named and typed identically to the name of methods advertised by the gateway, and have sub-elements corresponding to the parameters of the methods. “getRequest” or “setRequest” has a “version” element indicating the version of SNMP, “community” to be used for identification, and either an “oid” element for object identification or a “path” element for addressing one or more object nodes in the DOM tree using the XPath expression. The manager can access one or more nodes by specifying the “path” element as well as an “oid”. A “query” element is defined to contain XQuery expression for a complicated query. The “setRequest” element uses the “values” element to set a value of a node to be modified. The “getResponse” element is also defined as a response message for “getRequest” and “setRequest”. “getResponse” has the “result” element as the only sub-element. The “result” element is already defined in the SOAP specification version 1.2 [19]. The manager finds a method to invoke and to pass appropriate parameters to the method using XML Schema including definitions of these elements.

As mentioned above, we defined two elements, “getRequest” and “setRequest” for the request from the manager. The SNMP getBulk operation or other complicated types of request can be defined using “getRequest” with XPath or XQuery. We defined a method equivalent to SNMP getBulk, using “getRequest” with XPath or XQuery, and introduced a new element called “getBulkRequest”.

As described in previous slide, XPath and XQuery provide an efficient way of indicating managed objects to be retrieved. XPath and Xquery can also be applied in the SOAP request message as a parameter of methods.

# Protocol Level Translation (cont'd)



One of the design goals of SOAP is to encapsulate and exchange RPC calls using an extensibility of XML. In this slide, we describe SOAP based communication over HTTP between XML-based manager and XML/SNMP gateway using the SOAP messages defined in the previous page.

The figure in the slide illustrates a SOAP-enabled architecture of XML-based manager and the gateway, and the sequence of the SOAP message exchange. In this architecture, a SOAP client generates XML-encoded SOAP messages, conversely, a SOAP server in the gateway parses the message and invokes an appropriate procedure in the gateway application. The SOAP client takes in RPC information from the manager application and generates a SOAP message. The SOAP message is passed to the HTTP Client, which then sends the HTTP POST request to the HTTP Server in the gateway. The HTTP Client must specify a value of "SOAPAction" field in the HTTP Request header in order to route the HTTP request to the Web server identified in the field. The HTTP Server in the gateway converts the text POST to the HTTP message and sends the HTTP message to the SOAP server. Then the SOAP server parses the HTTP message into a properly formatted RPC call and invokes the appropriate object served by the gateway application. The SOAP server receives the results of the method call and generates a well-formed SOAP response message. The response message backtracks to the SOAP Client in the manager. Finally, the manager application receives the results of the method call.

It is very common to exchange XML-encoded data over HTTP. The SOAP provides a better solution than a proprietary XML/HTTP method because SOAP is an open standard with a growing body of developers and vendors supporting it. The SOAP defines a standard vocabulary and a structure for messages between communication peers, which eliminates an overhead of parsing and processing messages by a proprietary method in each gateway. The XML Schema provides self-documenting features of defining data and interface type like IDL in CORBA and DCOM, and which gives the basis of an RPC mechanism. One disadvantage of the SOAP is the overhead in converting the native format of application data into an XML-based SOAP message [22]. However, this overhead is eliminated in this approach, because the XML/SNMP Gateway already deals with XML documents serialized from DOM tree as a communication message. As more vendors offer SOAP products and services, the advantages of using SOAP will increase.

# Analysis of the Proposed Methods

- **Process level translation** is the basis for the other translation methods
- **Message level translation** can be easily adopted into XML/HTTP
- **Protocol level translation** includes the features of other translation methods

Translation Methods	Advantages	Disadvantages
<b>Process Level Translation</b>	<ul style="list-style-type: none"> <li>- No need for request handler between GW and manager.</li> <li>- Can be applied to both internal GW and external GW.</li> <li>- Uses DOM as intermediate storage for manager.</li> </ul>	<ul style="list-style-type: none"> <li>- Impose a burden on manager of invoking a series of interfaces for request processing.</li> </ul>
<b>Message Level Translation</b>	<ul style="list-style-type: none"> <li>- Simple to implement SOAP over HTTP.</li> <li>- Simple message format.</li> <li>- Provides an efficient mechanism for querying managed objects.</li> </ul>	<ul style="list-style-type: none"> <li>- No standard in the use of URI string as a request specifier.</li> <li>- Need of XPath or XQuery parser.</li> </ul>
<b>Protocol Level Translation</b>	<ul style="list-style-type: none"> <li>- Inherits advantages in message level translation.</li> <li>- provides a standard method to implement an RPC</li> </ul>	<ul style="list-style-type: none"> <li>- Overhead of packaging SOAP messages.</li> </ul>

In our first approach to interaction translation in XML/SNMP Gateway, we focused on the DOM in the gateway. The gateway uses the DOM structure and its interfaces to translate management information between XML and SNMP. We analyzed the functions and meanings of the DOM interfaces on management information, and translated the interfaces into appropriate SNMP operations. This translation provides a method to directly access management information through the DOM interfaces for XML-based manager. This method is targeted especially for an internal gateway [21], which is integrated within the manager system. In an internal gateway, the manager's request using the DOM interface is directly translated into an SNMP operation without any request parser application. Instead, the manager is responsible for using DOM interfaces, and thus must understand entire functions of interfaces. This method imposes a burden on the manager of invoking a series of methods in appropriate order. This method is also applicable to an external gateway. In such a case, however, the request parser or handler is needed between the manager and the gateway because the manager cannot directly use the DOM interface in the gateway. The request handler parses a request from the manager, and translates it to the DOM interface.

In the second approach, we proposed an efficient method to indicate target nodes using a query language for XML namely XPath and XQuery. This method gives an advantage of efficient and effective communication between a manager and a gateway by reducing the amount of request messages, resulting in lower data transfer. This approach also can be easily implemented in XML/HTTP with little effort, extending URI with a query expression. To support XPath and XQuery, both the manager and the gateway must include XPath and XQuery enabled module, and which is already supported by commonly used XML parsers [14].

Finally, we proposed SOAP-based communication as a translation method through RPC calls. A structure defined in a SOAP message is translated into a specific method served by gateway application. This method takes in the advantages of the second approach because SOAP is based on HTTP, and also SOAP messages can contain XPath and XQuery expressions in its body. The SOAP provides a better solution than any proprietary XML/HTTP method because SOAP is an open standard and SOAP messages are easily created. SOAP is an emerging XML standard protocol for invoking methods, objects, components, and services on remote servers. The SOAP specification provides a standardized XML vocabulary for representing input parameters, return values, and errors. Using SOAP, the gateway can receive requests from and send responses to XML-based manager in a standardized way, and which eliminates an overhead of parsing and processing messages by a proprietary method on XML/HTTP. The manager also has an advantage that it can access remote gateway codes without knowing how to package up XML message and make an HTTP request. Finally, the gateway can communicate with any type of managers, who can understand the interface definition in the XML Schema, and can generate a SOAP RPC message from the definition.

# Summary & Future Work

---

## ❖ Proposed Three Interaction Translation Methods for Integrated Management using the XML/SNMP gateway

- Translation in DOM process level provides the manager direct access to the DOM in the gateway using DOM interfaces.
- Translation of HTTP Messages provides efficiency improvement in XML/HTTP communication using URI with XPath and XQuery.
- SOAP-based translation provides a flexible and standardized method for interaction between the manager and the gateway.

## ❖ Future Work

- Now in the process of implementing the XML/SNMP Gateway using the proposed interaction translation methods
- Integrating it into an XML-based manager for global element management system.
- Performance evaluation of the gateway.
- Efficiency improvement of the translation mechanism.
- Considerations on the scalability of the gateway.

In this paper, we have proposed several interaction translation methods for the XML/SNMP Gateway, which resides between an XML-based manager and managed devices with SNMP agents. The gateway uses a DOM tree to process a request from the manager, and to transfer management data corresponding to the MIB variable in an SNMP agent. First, we proposed a translation method in the process level, which provides the manager direct access to the DOM in the gateway using DOM interfaces in order to exchange management information with the SNMP agent. In the translation of HTTP messages, we extended a URI string, which contains information on a request with XPath and XQuery. XPath and XQuery can be easily applied to URIs to provide a location path of target objects and a query language in the request message. This method provides efficiency improvement in XML/HTTP communication, which is the most common in the exchange of XML documents.

We also proposed a SOAP-based communication between XML-based manager and XML/SNMP Gateway as a translation method in protocol level. Using SOAP, the gateway provides a flexible and standardized method for interaction with XML-based manager in a distributed environment. The gateway can communicate with any type of XML-managers, who can understand the interface definition in the XML Schema, and can generate a SOAP message from the definition.

We are now in the process of implementing the XML/Gateway and plan to use it for developing an XML-based manager for global element management system. A possible future work currently being considered involve improving the efficiency, performance, and scalability of the gateway.

## References

- [1] Jeong-Hyuk Yoon, "Development of SNMP-XML Gateway for XML-based Integrated Network Management", MS. Thesis, POSTECH, Dec, 2001.
- [2] Hong-Taek Ju, Sehee Han, Yujung Oh, Jeong-Hyuk Yoon, Hyojin Lee, James W. Hong, "An Embedded Web Server Architecture for XML-Based Network Management", Proc. of the IEEE/IFIP Network Operations and Management Symposium (NOMS 2002), Florence, Italy, April 2002, pp. 5-18.
- [3] J.P. Martin -Flatin. "Web-Based Management of IP Networks and Systems", Ph.D. Thesis, Swiss Federal Institute of Technology, Lausanne (EPFL), Oct 2000.
- [4] Frank Strauss, "A Library to Access SMI MIB Information", <http://www.ibr.cs.tu-bs.de/projects/libsmi/>.
- [5] W3C, "Extensible Markup Language (XML) 1.0", W3C Recommendation, October 2000.
- [6] W3C, "Document Object Model (DOM) Level 2 Core Specification", W3C Recommendation, Nov 2000.
- [7] W3C, "Document Object Model (DOM) Level 2 Traversal and Ranges Specification", W3C Recommendation, Nov 2000.
- [8] W3C, "XQuery 1.0: An XML Query Language", W3C Working Draft April 2002.
- [9] W3C, "XML Path Language (XPath) Version 2.0", W3C Working Draft, April 2002.
- [10] W3C, "XQuery 1.0 and XPath 2.0 Functions and Operators", W3C Working Draft April 2002.
- [11] W3C, "XML Schema Part0: Primer", W3C Recommendation, May 2001.
- [12] W3C, "XML Schema Part1: Structures", W3C Recommendation, May 2001.
- [13] W3C, "XML Schema Part2: Data Types", W3C Recommendation, May 2001.
- [14] Apache XML project, "Xerces Java parser", <http://xml.apache.org/xerces-j/>.
- [15] T. Berners-Lee, R. Fielding, and L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax". IETF RFC 2396, August 1998.
- [16] IBM, "SOAP : Simple Object Access Protocol", <http://www-106.ibm.com/developerworks/library/soap/soapv11.html>
- [17] W3C, "SOAP Version 1.2 Part 0: Primer", W3C Working Draft, December 2001.
- [18] W3C, "SOAP Version 1.2 Part 1: Messaging Framework", W3C Working Draft, December 2001.
- [19] W3C, "SOAP Version 1.2 Part 2: Adjuncts", W3C Working Draft, December 2001.
- [20] W. Stallings, "SNMP, SNMPv2, SNMPv3, and RMON 1 and 2" Third edition, Addison-Wesley, Reading, MA, USA, 1999.
- [21] Heinz-Gerd Hegering, "Integrated Network and System Management", Addison-Wesley, 1999.
- [22] O'Connell, P.; McCrindle, R., "Using SOAP to clean up configuration management", Computer Software and Applications Conference, 2001. pp. 555-560.
- [23] Curbera, F.; Duftler, M.; Khalaf, R.; Nagy, W.; Mukhi N.; Weerawarana, S., "Unraveling the Web services web: an introduction to SOAP, WSDL, and UDDI", IEEE Internet Computing , Volume: 6 Issue: 2 , March-April 2002 , pp.86-93.
- [24] Rick Strahl, "Building and using a SOAP Web Service with Visual FoxPro and Web Connection", <http://www.west-wind.com/presentations/xmlmessaging/soapwebservices.htm>