

## Design of Web Service by Using OPC XML-DA and OPC Complex Data for Automation and Control Systems

Vu Van Tan, Dae-Sung Yoo, and Myeong-Jae Yi

School of Computer Engineering and Information Technology,

University of Ulsan, San-29, Moogu-2 Dong, Namgu, Ulsan 680-749, Republic of Korea,

Email: {vvtan, ooseyds, ymj}@mail.ulsan.ac.kr

### Abstract

Web technologies are gaining increased importance in automation and control systems. However, the choice of Web technologies depends on the use cases in the application environment. In industrial systems, the data can be got not only from many different field systems and devices but also from different OPC (OLE for Process Control) Servers. Current OPC Client might be able to read simple data from OPC Server, but there are some problems to get structured data and to exchange structured information between collaborating applications. Therefore, OPC Foundation has defined interfaces to OPC XML-DA (OPC XML Data Access) and OPC Complex Data that aim to solve those problems. The OPC XML-DA can facilitate the exchange of plant data across the internet, and upwards into the enterprise domain. In addition, the OPC Complex Data will extend the OPC DA specification to allow the OPC Client to read and decode any type of data from measurement and control systems on the plant floor. This paper will describe the concept of OPC XML-DA and OPC Complex Data. And then it proposes a mechanism to implement the OPC Complex Data into OPC XML-DA Server. Additionally, the paper also discusses the security aspects.

### 1. Introduction

Web technologies have been adopted not only in the e-business but also in the automation and control systems. In addition, the XML technology is a standard way of representing and exchanging the structured data. Therefore, OPC Foundation formed the OPC XML-DA technical working group to define a new specification to move the same type of plant floor data as the existing OPC DA COM-based specification. The OPC XML-DA provides vertical integration between the plant floor and condition monitoring system, maintenance system etc. using such as industry standards XML, HTTP and SOAP (Simple Object Access Protocol) [1], [2]. Since the XML and SOAP are the fundamental parts of Web Services, the OPC XML-DA actually is described as a Web Service [1], [2], [3].

The scalar data might present machine operating parameters from analogue measurements such as pressure, temperature, flow, level, and vibration used to represent on/off state or abnormal condition. However, OPC Clients might be able to read the data as a simple data. Therefore, OPC Foundation has defined the OPC Complex Data to overcome these limitations [4]. The OPC Complex Data will provide a way for OPC Client application to read and decode any type of data from OPC Servers. Actually, The OPC Complex Data is defined by complex data items that compose of a combination of structured data, simple items and complex items [2], [4].

Using the concepts explained above, this paper shows the model of Web integration for Automation System. It also describes the proposal to implement the OPC Complex Data into the OPC XML-DA Server. In addition, the security aspects are discussed.

This paper is organized as follows: Section 2 provides the basic concept of OPC XML-DA, OPC Complex Data and SOAP protocol. Section 3 gives a brief overview of the design of Web Service. Section 4 discusses about framework for Web Service. And some security aspects are discussed in the final section.

### 2. Background

A general architecture of Web integration is shown in Figure 1 [5]. It consists of three layers.

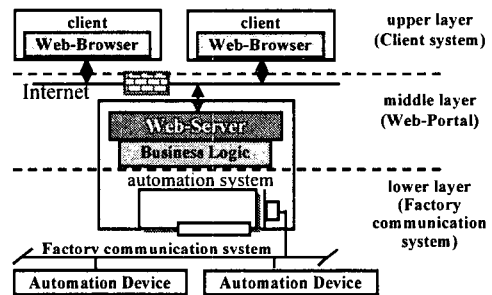


Figure 1. General architecture of Web integration

The lower layer is a factory communication system. It provides information from automation devices to controller level of an automation system. The middle layer Web-portal contains the functionality of business logic and it also performs as an application gateway between the upper clients and the lower automation system. In this paper, we propose the implementation of the OPC Complex Data in this layer. The OPC XML-DA Server can be used to assign information from the automation and control system to an object model. This model can be accessed via DCOM (Distributed Component Object Model). However, using the Web-technology for industrial systems, it means integrating a multitude of different technologies. The upper layer is based on standard information technology such as Client/Server model, using Web server as data source and browsers as clients.

#### 2.1. SOAP Protocol

SOAP is a lightweight protocol for exchange of information in distributed environments. It is an XML based protocol that consists of three parts: an envelop that defines a framework for describing what is in a message and how to process it, a set of encoding rules for expressing instances of application-defined data types, and a convention for presenting remote procedure

calls and response. The SOAP can be potential used to in combination with a variety of other protocols [3], [6].

### 2.2. OPC XML-DA

The OPC XML-DA [1] is a standard Web Service interface for reading and writing data from and to plant floor automation systems. OPC XML-DA data model is based on OPC Items that are named and organized in a hierarchy. Each item stores a single value. It is defined with the combination of two strings: *item path* and *item name*. A set of dynamically retrievable properties is associated with every item containing its metadata including human readable description, access rights, timestamp, change rate, and data type. Operations for accessing item values are *Read* and *Write*. Both allow accessing several items with a single call. To optimize periodic reads of the same set of items as subscription mechanism is provided. The set of items is subscribed by calling the operation *Subscribe* and then periodically polled using *SubscriptionPolledRefresh*. The operation *Browse* and *GetProperties* are used to query which OPC Items are available and values of their properties. The operation *GetStatus* is used to retrieve the OPC Server's status. The OPC XML-DA is specifically designed to allow existing OPC DA COM based products to be wrapped by the OPC XML-DA interface and in effect support both interface from the same OPC Server. Any group can develop a generic OPC XML-DA wrapper to internet-enable existing OPC DA Servers allowing them to publish plant floor data to the Web. Figure 2 shows the roles of OPC XML-DA.

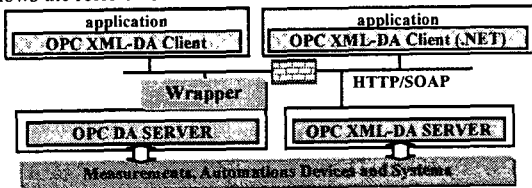


Figure 2. The roles of OPC XML-DA

### 2.3. OPC Complex Data

The OPC Complex Data [4] initiative will provide a way to the OPC Client application to read and decode any type of data form OPC Servers. Actually, the OPC Complex Data contains OPC complex data items that can include non-structured items, abstract elements, abstract items, structured items, strings, integers, XML data, OPC binary, and other OPC data items. Each data item is accompanied by a Data Type Description that defines the structured item, and a Dictionary that describes each of the Data Type. The dictionary contains all information that the OPC Client needs to understand the complex data item it is receiving. The OPC specifications defined two type systems that provide this level of capability, XML schema and OPC binary. The XML schema describes complex data values that presented in the XML, and the OPC binary schema describes complex binary values.

## 3. Design of Web Service

As explained above, we proposed the implementation of the OPC Complex Data into the OPC XML-DA Server. Therefore, the clients can understand and decode any type of data that provided by OPC XML-DA Server. The clients can retrieve the WSDL (Web Services Description Language) file from the OPC

XML-DA Server and can use it to create an appropriate user interface. The WSDL file contains service information such as data types, value ranges, display information, parameters etc. Figure 3 shows the request and response information between OPC XML-DA Server and Client by using XML messages.

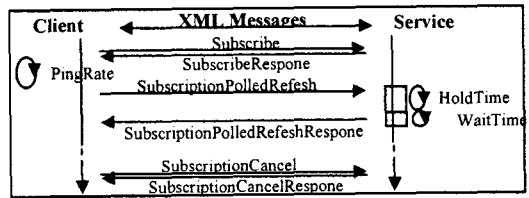


Figure 3. Communication between OPC XML-DA Server and Client

The XML Schemas provide a mean for defining the structure, content and semantics of XML documents and add data type information to XML data. The complex data type can be defined by the WSDL. Actually, the WSDL files are XML files that contain the description of all data types using in OPC. Therefore, the OPC Complex Data Dictionaries are defined by using the WSDL. The WSDL both creates the interface for client application and exchanges all kind of data between the OPC XML-DA Server and Clients. The implementation of the OPC Complex Data into the OPC XML-DA Server is shown in Figure 4. The Web Service Listener is invoked, when a request has been passed to the Web Server. It interacts with the objects and returns the results of the function as an XML files. The Service Aggregator consists of modules to perform the data transformation, authentication/encryption, configuration etc. In order to describe above, the complex data types are defined by using XML files, the WSDL is the XML files and the temporary XML files can be transformed by using the transformation rules (XSLT : eXtensible Style Language for Transformation). Therefore, the WSDL files and XSLT style sheet [7] should be implemented in the Service Aggregator. While the required format is specified in the WSDL files, the XSLT style sheet defines the transformation rules that define mappings between different syntactic information. The XSLT transformation is performed to create the WSDL response messages. The OPC XML-DA subscription makes use of a "polled-pull" style of subscription. The client can invoke the subscription method and receive the result by using a loose contract with server (Figure 3).

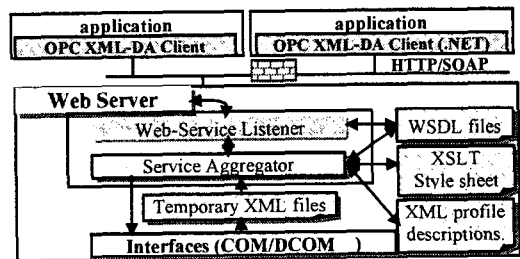


Figure 4. Web Service implementation

## 4. Framework for Web Service

As we discussed above, a Web based framework for industrial communication system relies on an XML based set of interface descriptions. These descriptions represent interaction

schemas between specific applications with the specific contexts. Conversations between the schemas are performed according to transformation rules. The application framework uses content model to request data source and data formats for the values that have to be read out of the factory communication systems and its components. Since the XML schemas contain information on the data types, value ranges, display information, access path, parameters etc, the application can determine and present values in the client side. Consequently, using the WSDL files, we can construct a standard XML-based application. For example, the WSDL file can be created by the transforming a device-specific file with data description and WSDL template. As we know, the OPC Complex Data as a dictionary that can be defined and described by using XML schemas. In addition, using XSLT with transformation rules, the temporary files can be transformed with the output format. The XML-based content model [8] which consists of a distributed set of XML files, schemas, transformation rules. In order to prevent multiple definitions, the files are linked together and created as a consistent set of descriptions.

### 5. Security Consideration

The security is an important point in Web application. Therefore, it has to be considered carefully when designing an application. In order to achieve required security criteria, the concepts and solutions developed for general IT systems have to be applied. Furthermore, appropriate technologies like encryption, Secure Socket Layer technology, Secure HTTP, transaction based concepts, certificates and digital signature should be used. The security in Web-based applications on automation and control systems has to be seamlessly integrated into the system when designed. This paper applies XML Encryption/Decryption approach, as described in XML Encryption and XML Signature specification [9-12]. This solution strictly utilizes the XML way and provides more control over authentication and read-write access. It can also manage access right on the OPC Item level. Both the processing rules and the algorithms for encryption and decryption are provided in [13]. In this paper, the XML distributed signature approach is used for encryption and decryption. This solution reduces the data to transfer between OPC XML-DA Server and Client. Moreover, it is not necessary to build a special environment for XML signature creation on the client. The processing flows are shown in Figure 5.

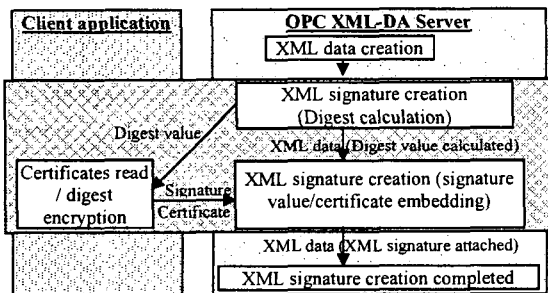


Figure 5. XML distributed signature processing flow between OPC XML-DA Server and Client

Furthermore, the WSDL descriptions can be used to check and verify data passed to the services request. This reduces errors caused by inconclusive parameters. On the other hand, the middle layer of the Web integration concept shown in the Figure

1 can be treated as a specific type of application proxy. In addition, we can use gateway for security in the lower layer. Other security solution is shown in [14] that used both symmetric secret and asymmetric key for security.

### 6. Conclusion

We have introduced the design and implementation of Web Service by using OPC XML-DA and OPC Complex Data for Automation and Control Systems. In our research, we proposed the implementation of the OPC Complex Data into OPC XML-DA Server. Therefore, the client can read and decode any type of data to be got from the OPC Servers. The implementation can be characterized to be extremely flexible, reusable, and optimally adopted required functionality. Besides the potential of the WWW can be enabled, combined with easy integration into existing solutions and with additional security features based on validated XML message. Moreover, this paper provided the optimal security model using XML distributed signature. This solution can manage the access right on the OPC Item level and reduce the data processing flows between the OPC XML-DA Server and Client. Thus, this solution also ensures truly thin clients. Additionally, this paper also proposed some solutions for security problems in Automation Systems.

### 7. Reference

- [1] OPC XML-DA 1.01 Specification, <http://opcfoundation.org/>, Dec. 2004
- [2] D. W. Holley, "Understanding and Using OPC for maintenance and reliability applications", *Journal, IEE Computing and control engineering*, Mar. 2004.
- [3] A. Shonnard, "Understanding SOAP", Mar. 2003.
- [4] OPC Complex Data 1.0 Specification, <http://opcfoundation.org/>, Dec. 2003
- [5] M. Wollchlaeger, P. Neumann and Th. Bangemann, "Web service for remote maintenance of fieldbus based automation systems", *IEEE-AFRICON*, Oct. 2002, p. 247-252.
- [6] SOAP Version 1.2 Recommendation 2003/6/24 <http://www.w3c.org/TR/SOAP/>
- [7] XSL Transformation (XSLT) version 1.1, 2001/8/24, <http://www.w3.org/TR/2001/WD-xslt11-20010824/>
- [8] M. Wollchlaeger, and T. Bangemann, "XML based Description Model as a Platform for Web-based Maintenance, *IEEE-INDIN*, Jun. 2004, p. 125-130.
- [9] Miyachi, "XML Signature/Encryption - The Basic of Web service security," *NEC Journal of advanced technology*, Vol.2, No.1, Winter 2005
- [10] XML-Signature XPath Filter 2.0: W3C Recommendation 2002/11/08, <http://www.w3.org/TR/xmlenc-core/>
- [11] Decryption Transform for XML Signature: W3C recommendation 2002/10/10, <http://www.w3.org/TR/xmlenc-decrypt>
- [12] XML-Signature Syntax and Processing: W3C Recommendation 2002/2/12, <http://www.w3.org/TR/xml-dsig-core/>
- [13] XML Encryption Syntax and Processing: W3C Recommendation 2002/10/10, <http://www.w3.org/TR/2002/REC-xmlenc-core-0021210/>
- [14] S. Chilingaryan and W. Eppler, "High speed data exchange protocol for Modern distributed data acquisition systems based on OPC XML-DA", *IEEE-NPSS*, Jun. 2005, p. 352-356

본 연구는 산업자원부/울산광역시 지원 울산대학교 네트워크 기반 자동화연구센터의 지원에 의한 것입니다.