OPC UA summary

Zhi-Ning Wang[1,a], Yi-Yang Liu[†,1,b], Feng Gao[2,c], Meng Tang[2,d],
Yun Shi[2,e] and Xiao-Ming Zhou[3,f]

[1]Industrial Control Networks and Systems Department,
Shenyang Institute of Automation Chinese Academy of Sciences,
Shenyang, 110016, China
[2]PetroChina Fushun Petrochemical Company,
Fushun, 113006, China
[3]Liaoning Electric Power Company Limited of State Grid,
Shenyang, 110016, China
[a]wangzhining_cau@126.com; [†,b]sialiuyiyang@sia.cn;
[c]gaofeng_fspc@163.com; [d]tangmeng@petrochina.com.cn;
[e]shiyun123@petrochina.com.cn; [f]zxmlw1017@126.com

This paper illustrates the OPC technology's shortcomings of the platform portability, remote data access and security aspects of industrial automatic production. For the limitations of the OPC on these aspects, the OPC Foundation presented the improved concept of OPC UA. This paper described the OPC UA specifications and technology, which were focused on the OPC UA Client/Server architecture, OPC UA technology to data encodings, secure conversations and data transport technology, address space model, object model and services. Compared with the existing OPC specifications, OPC UA has significant advantages with its platform independent, interoperability, security and other aspects. With the above advantages, OPC UA will play an important role in the control system based on the Internet communications.

Keywords: OPC UA; Address Space Model; Unified Object Model

1. Introduction

OPC is a set of standards developed by the OPC Foundation, it can easily get data from devices in the production of industrial automation by a uniform way in the different internal network. OPC technology is based on Microsoft's COM and DCOM technology from the late 1990s. Because of the widely use of Windows PC, OPC is widely used in industrial fields by Windows operating platform. OPC client / server (C/S) mode defines how to get data from the server and the mode of the client to receive data from the server, including read, write, test data variables and other operations. By reading different kinds of data, such as alarms, events, historical data and some special data types, OPC can achieve functions of event notification, alarm notifications and historical data storage access.

However, OPC has great limitations, for example:

- Platform limitations: First, due to the continuous development of computer technology, Unix and Linux continues to use, OPC relied on Windows operating platforms and is not compatible, so availability is greatly reduced.
- Remote data access limitations: The data cannot be transferred between different networks, and it makes that high layer cannot get production data from low layer for making decisions.

Complex configuration and low security: It needs more ports to communicate between DCOM and computer. Each port is a complex process for configuration, so data is too cumbersome to make it out past firewalls. Finally, because of the cumbersome configuration process and the shortages of relevant technical staff, they have to adopt simple configuration, which makes the access protocols more relaxed, more simplified access authorization, data loss or vulnerable.

Subsequently, the Web services and HTTP-based SOAP were released, and OPC Foundation releases OPC XML-DA. Despite an improvement in platform independence, the XML message format needs for more space compared with the previous DCOM packets which makes it difficult for existing network XML format data transmission at the same time. So we can only send individual smaller operations. And in other respects, it still has not been changed, so this version is not widely used.

2. The improvement of OPC UA

The emergence of OPC UA, which solves the limitations of the OPC, and it has fully inherited the advantage of OPC [1]. OPC UA establishes the address space in order to unify the different types of data acquisition. Server obtains different types of data from different devices, and the client can be obtained by calling the different functions of the data directly in the address space. OPC UA added data models and semantic models to form a new data element model, so it can deal with complex data structures and clients do not need to understand the different data to identify. OPC UA also offers a variety of SDK, so you can use OPC technology on different platforms, breaking the platform limitations. Finally, OPC UA uses a safe, fast and advanced transmission mechanism to create a secure messaging environment through TCP, HTTP technology. OPC UA greatly improves the safety by using continuous improvement of the UA security protocols and PKI.

3. The system architecture of OPC UA

OPC UA system is similar to classic OPC in using a client / server architecture. The one who wants to expose its own information to other applications is called the UA server. And the other who wants to use other applications is called the UA client. The Figure 1 shows the OPC UA technology in various organization networks.

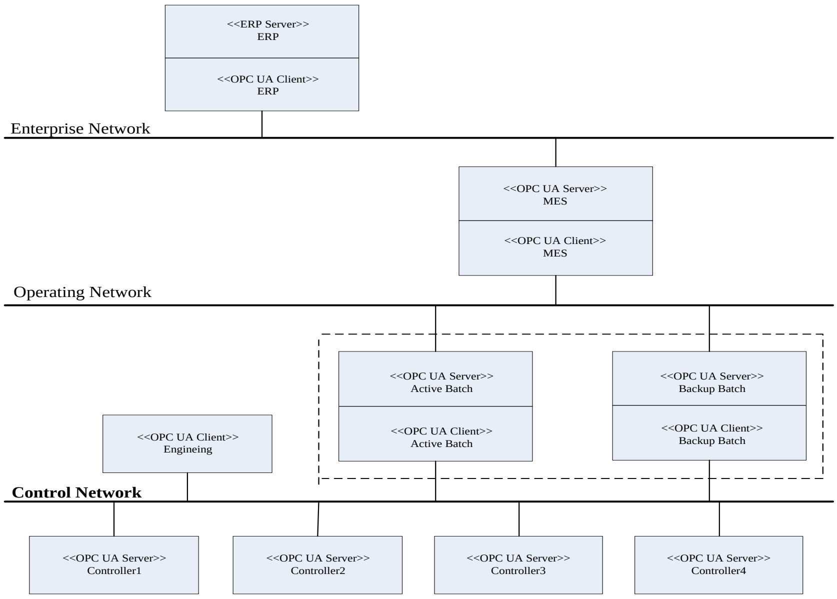

Figure 1. The applications of OPC UA in various organization's network

As can be seen from Figure 1, OPC UA client always be used as interface for service in an ERP system in the enterprise network. Among them, the controller needs real-time operating system; the batch system and MES system may be based on Windows operating system; and the ERP system may be deployed on a UNIX platform [2]. OPC UA architecture was modeled with OPC UA clients and servers by the way of interrelating objects. Each system may contain multiple clients and servers. Each client may be connected to one or more servers at the same time, and each server can be at the connection with one or more customers at the same time. An application may also contain two parts which are both of the server and client components, in order to achieve connect to other server and client requirements.

3.1. Client

OPC UA client architecture includes customer terminal client/server interaction. OPC UA client structure includes OPC UA client application, OPC UA communication stack, OPC UA client API, as shown in Figure 2. It uses the OPC UA client API to contact with OPC UA server for sending and receiving OPC UA service request and response. OPC UA client API is an internal interface, which separates the client application code from the OPC UA communication stack. OPC UA communication stack converts the OPC UA client API calls into a message. It sends to the server through the underlying message layer, and informs the server the client application's requests. OPC UA communication stack also obtains responses and notification messages from the server, and passes message to the client application through API.

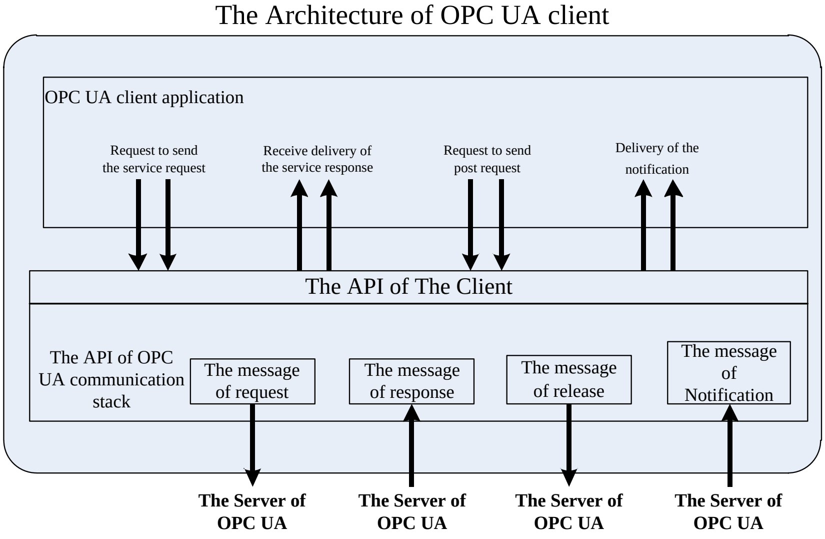
The Architecture of OPC UA client
OPC UA client application
Request to send the service request
Receive delivery of the service response
Request to send post request
Delivery of the notification
The API of The Client
The API of OPC UA communication stack
The message of request
The message of response
The message of release
The message of Notification
The Server of OPC UA
The Server of OPC UA
The Server of OPC UA
The Server of OPC UA

Figure 2. The architecture of OPC UA client

3.2. Server

OPC UA server architecture models the part of the C/S structure. OPC UA includes OPC UA server applications, the real objects, OPC UA address space, the publish/subscribe entity, OPC UA server interface API and OPC UA communication stack, as shown in Figure 3. The real objects are physical devices which are directly accessed by OPC UA server application program or internal software programs. The address space is composed of a series of nodes. Clients can use OPC UA interface and the method to access the node in the service. The node in address space is used to represent real objects, and their definitions and references between each other. OPC UA will establish its information model in server instead of on the client.

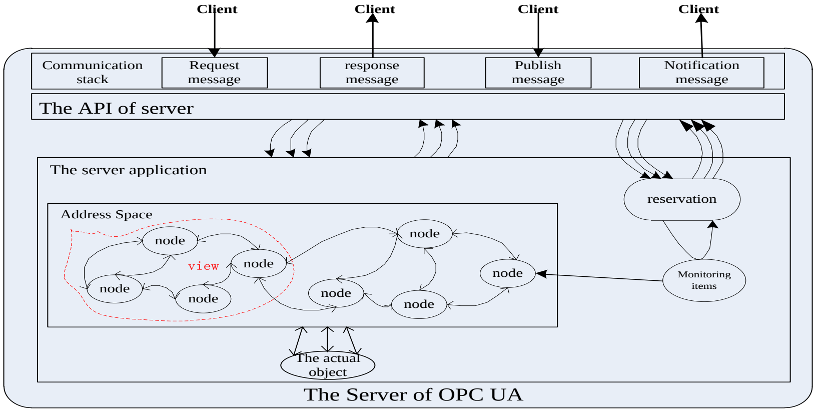

Figure 3. The structure of OPC UA server

OPC UA server API sends and receives messages which sent by the OPC UA client. Through the question and answer mode, the interaction between the client and the server is through the UA communication stack. OPC UA server and the client's main interaction process are as follows:

- The client sends a service request to the OPC UA communication stack via the underlying communication entity. And it invokes a request/response service through the OPC UA server interface. After completed the specified tasks in the address space of one or more nodes, the UA server returns a response.
- The client sends a post request to the OPC UA communication stack via the underlying communication entity. And it sends to the reservation via OPC UA server interface. When reserving specified monitoring item detects data changes or event/alarm occurs, the monitoring item generates a notice sent to the reservation. Then it is sent to the customer by the reservation [3].

4. OPC UA technology

Combined with the characteristics of industrial automation system, technologies necessary for exchanging data and ensuring its interoperability between OPC UA applications are the following three: data encoding, securing the communication, and transporting the data.

4.1. Data Coding

Encoding and decoding of the two types respectively use different encoder and execute on the network transmission layer. OPC UA through the size of the network throughput to determine the optimal selection of the data encoding type, respectively, XML (text format) and UA binary two types [4]. By the size of the throughput of network, OPC UA selects the optimal type of encoding data.

XML format can be on the server of various types of data structured, can be better for data transmission. When the throughput in the network is large, the data is in UA binary format, which ensures the real-time information and does not accumulate [4]. After the coding layer there are two layers, namely the security channel layer (responsible for the integrity of the information, security) and session layer (authentication).

When people want to exchange data in a common format consumed by different applications, platforms, XML documents often play an important role with its standardized structure. XML documents can be parsed with a parser by any application or platform. Based on this fact many operations level applications (such as MES) and some corporate level applications (e.g., ERP systems) use XML to exchange data. Because OPC UA applications will run between two layers, an XML format is necessary [5]. The XML documents are structured to deal with all kinds of data on the server, and it can better for transporting the data.

To save performance and spending in industrial systems the OPC UA foundation defined a data format called OPC UA Binary to provide fast encoding by a small size and efficient format [6]. After coding layer, there are two layers. Wherein the secure channel layer takes charge of keeping information's completion and safety, the session layer is used to make user authentication.

4.2. Securing the communication

4.2.1. OPC UA security architecture

The level of the OPC UA security architecture is shown in Figure 4.

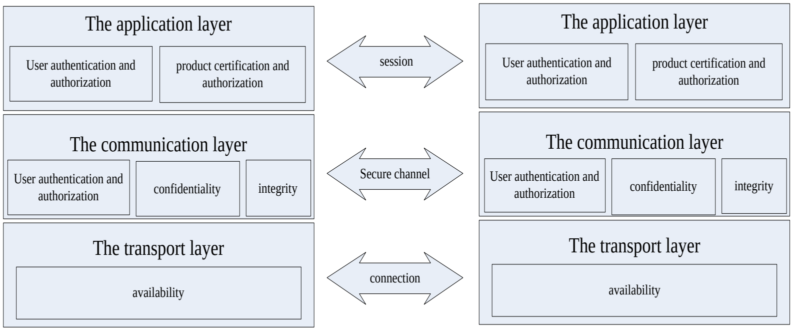

Figure 4. The security architecture of OPC UA

The Application layer uses sessions to get the user's client authentication and authorization. Sessions run on a secure channel, and the safe of secure channel is guaranteed by the communication layer in the way of digital signatures, authentication and authorization. The underlying transport layer’s work is to keep error recovery mechanisms by using socket connection by and a secure way.

4.2.2. Security Protocols

OPC UA has two security protocols: WS-SecureConversation and UA-SecureConversation. Both of them work according to certificate connection.

WS-SecureConversation defines a security algorithm to negotiate and share encrypted data over a secure channel between OPC UA. WS-SecureConversation is the foundation of the OPC UA security policy and security configuration. WS-SecureConversation improve XML on encryption and signature. It makes more automation engineers to choose OPC UA when they develop own systems. Though exchange data by XML has to expend very big onlined overhead, it promotes UA-Secure Conversation.

UA-SecureConversation add additional secure information both in the head or end of service messages, instead of map abstract services directly. UA-SecureConversation which contains other message’s significate works to verify changed status and the entity status whether it has certificate after sending.

4.3. Transporting the data

Data transmission is mainly carried out in the network layer and session layer. In OPC UA, the method of communication between the client and the server is message mechanism.

OPC UA standard defined two transport protocols: UA TCP and SOAP/HTTP. Both of them used in network level to make a connection between an OPC UA server and client. OPC UA session and secure channel just be established on both two transport protocols [7].

The first transmission protocol is UA TCP. To establish an open reliable communication in network between the client and server cannot lack it. This protocol has message head to make sure secure and message body to carry information. There are two kinds of information in the message body. One kind is the necessary socket connecting information for building TCP protocol. The other is encrypted information transported to UA server in networks, such as hello messages (to monitor status of server), acknowledge messages (servers used to return hello messages) and error massages (when the connection is wrong between the server and client).

The other kind of network transmission protocol is HTTP/SOAP which is widely accepted by web service since its simple and firewall-friendly characteristic. HTTP uses TCP protocols to allow the client and server information transmission and identifies the data in the application layer for the client API calling to execute its function. UA client sends a request to the server, then the server will return information back to the client. SOAP technology combines with HTTP network technology and XML documents so that it can be implemented in such a heterogeneous hierarchical network to achieve interoperability between the various applications and platforms. It has the ability to exchange information in the network, but also has the flexibility of XML.

5. OPC UA Integrated Address Space Model

In traditional OPC specification, each independent OPC specification defines its own address space and service. Usually, the client sends a request to the server to complete a visit which needs multiple address space cooperation, so that it causes a low efficiency of the system. In order to solve this problem, the OPC Foundation puts forward the concept of integration of OPC UA address space model, that is, all kinds of specification will be (such as DA, A&E, HDA) integrated into single address space of the OPC UA server. The client just makes one call so it can obtain the real-time data, alarm events, historical data, etc.

The set of information in the OPC UA server can be seen by the client called address space. The basic unit of address space is the node. According to the objective, nodes can be of different NodeClasses. Some of them represent instances, and others represent types. [4]. When the node is defined in address space, NodeClasses should be instantiated. Node category consists of node attributes and references. Attributes are responsible for describing nodes, and a node can have one or several attributes even a different set of attributes [8]. And attribute is the basic component of the node class, which determines the characteristics of the node. The client can get the attribute value of the node through reading, writing, query, subscription/monitoring items. Every node has its attributes from its NodeClass. However, there are some common attributes, and the common attributes of the node are shown in Table 1 [8].

Table 1 Common Attributes Of Node

Attribution	Description
NodeId	In OPC UA server's address spaces, the only one identifies a node.
NodeClass	The type is an enumeration.
BrowseName	It's generated by browsing OPC UA server. It is not localized.
DisplayName	It used to display nodes' names to users. It is localized.
Description	Description for localized nodes. It is optional.
WriteMask	It tells which nodes can be revised by OPC UA clients. It is optional.
UserWriteMask	It tells which nodes can be revised by connecting OPC UA server. It is optional.

The NodeId which makes a most important roll in addressing information and exchanging identifies a certain node in the server. When servers finish browsing or querying their address space NodeIds will return. Clients just use NodeIds for service calls after addressing a node [8].

One who only has the common attributes of the node in address space is called the BaseNode. It is the metadata model of the address space. The BaseNode is abstract and it cannot be instantiated. All other nodes in the address space must be inherited from the BaseNode. It's necessary to be instantiation for call its attributes and methods.

Connections between two nodes call a reference. A reference can only access by browsing instead of directly accessed. References can't be represented as nodes, thus they have no attributes or properties. However, references have its own type called reference types to expose semantic, and how to connect nodes with different semantics [8].

Although references without attributes aren't nodes, the ReferenceTypes are exposed as nodes in the address space. It allows clients obtain the information from nodes in server's address space. The same with attributes, reference is also a basic component of NodeClass. References used between nodes compose hierarchy structure in address space, which improve the interoperability between the client and the server.

In order to simplify the access from the client to address space, the OPC UA server puts forward the concept of view. View is a subset of address space with the default value of entire address space. The Address Space is divided by view into several blocks. The same with address space, view is organized into a hierarchical structure by references between the nodes. The server can hide unnecessary data by view to limit the scope of address space that is visible to the specific user or tasks. Some content limited the client to visit by view makes it easier than before. In addition, A&E、HDA and other services of OPC UA are managed through view. For instance, the A&E source node creates an instance of the A&E object by inheriting the base node, and then it is connected to the node type of the ConditionEventType object type by the HasEventSource reference type. Set HasEventSource reference type to the A&E view node filter parameters, and you can organize all the A&E information together for effective management with facilitate access for the client. The management of HDA data in OPC UA server and the client's access to the data are the same [9].

6. OPC UA United Object Model

Object, variable and method are top 3 important in OPC UA. Objects have variables and methods to trigger events [4]. Objects were individual in existing OPC definitions, and it's difficult to visit or call for the client. OPC UA unified object model provides a set of complete, consistent NodeClasses to describe objects in address space, and the three main functions of servers are data access (DA), alarm events (A&E), and historical data access (HDA). For example, OPC UA server described the pressure sensor as an object, which comprised parameters of a pressure value, maximum pressure and alarm. It realized the integration of each object services.

Nodes which called NodeClass Variable represent a value. Its type depends on the variable. Clients have functions of reading, subscribing, revising and writing the value [8]. The method, that is, something of representing the method in the object model called by the client in the OPC UA server and executed by the server to return the results to the client. Each method specifies the parameters to be passed to the client and results from the server to expect that the return value. The client calls the method execution by calling the service call, and contains the result returned in the service call response.

Event indicated the important things which the system thinks of. Among events, the abnormal event is called alarms. For example, when the value of the temperature sensor exceeds the limit value, the alarm event will be issued. Through the unified object model, it integrates data, event and alarm etc., which is integrated into a single OPC UA server, which is convenient for the client to access the data.

7. OPC UA Service

The OPC UA service is defined in an abstract way, defined as data communication of the application level that is, the provider of the information model - the server and the user of the information model - the interface between the clients.

Services provided by an OPC UA server are roads to a client to get the information model. The traditional OPC standards only defines the API between applications. The API is related to a specific transmission mechanism which is component object model (COM) which is one of software component technologies belonged to Microsoft. Different with traditional OPC standards, OPC UA service defines the communication interface between UA applications. It is independent of transmission protocols and application development environment. Service is defined by abstractly different transport protocols, which are implemented by a specific API in the UA stack of different development languages.

8. Conclusion and Prospect

OPC UA has a great advantage in the platform portability, remote data access and security. Therefore, OPC UA can be widely used in equipment, controller, DCS, MES and ERP system. OPC UA will be widely used in industrial intelligent manufacturing with its own standardization of the information model and the security of the transmission mechanism. Therefore, OPC UA can be widely used between factories and enterprises owing to its independence of platform and system. OPC UA is a safer and more reliable technology so that the manufacturers can make possible seamless integration from the upper layer of enterprise network to the device layer.

Acknowledgement

The authors acknowledge the financial support of the Strategic Priority Research Program of the Chinese Academy of Sciences under Grant No.XDA06021100.

References

[1] The OPC Foundation, “OPC Unified Architecture Part 1: Concept Version 1.00”, http://www.opcfoundation.org/, accessed July 2006

[2] Wolfgang Mahnke, Stefan-Helmut Leiner, Matthias Damm. OPC Unified Architecture. Berlin Heidelberg. Springer-Verlag. 2009

[3] Zhao Yanhui, Nie Yajie, Wang Yongli, Li Haiping. Research on OPC UA Technology[J]. CHEMICAL DEFENCE ON SHIPS,2010,02:33-37.

[4] The OPC Foundation, “OPC Unified Architecture Part 8: Data Access Version 1.01”, http://www.opcfoundation.org/, accessed August 2009

[5] The OPC Foundation, “OPC Unified Architecture Part 6: Concepts, Technology Mapping, Version 1.00”, http://www.opcfoundation.org/, accessed January 2008

[6] W.Mahnke, S.H.Leitner, M.Damm, “OPC unified architecture” (Springer Press, 2009).

[7] C. Diedrich, M. Muhlhause, M. Riedl, and T. Bangemann, “Mapping of smart field device profiles to web services,” in Proc. IEEE Workshop on Factory Commun. Syst. (WFCS), Dresden, May 21–23, 2008, pp.375–381.

[8] The OPC Foundation, “OPC Unified Architecture Part 5: Information Modeling: Concepts Version 1.01”, http://www.opcfoundation.org/, accessed January 2009

[9] Lu Huiming, Yan Zhifeng..Research and development of key technology for address space of OPC UA server [J],Electric Power Automation Equipment,2010,07:109-113.