A Framework for Distributed Fieldbus Remote Supervisory Control System

Yingzi Shi and Jiangang Lu

Abstract—**In order to solve the problems of applying OPC (OLE for Process Control) technology to remote supervision, this paper proposes a new framework for distributed fieldbus remote supervisory control system based on the Windows DNA (Windows Distributed interNet Applications Architecture) developing model, which adopts the XML (eXtensible Markup Language) and SOAP (Simple Object Access Protocol) technologies as communication mechanisms of distributed components. This new framework has good extensibility and feasibility, and can help programmers design the distributed fieldbus remote supervisory control solutions.**

I. Introduction

FIELDBUS Control System (FCS) is a new control technology developed in1990s', which introduces the ideas of modern network communication and management into control field and is known as "Framework of Control System in 21 Century". With the development of computer science, communications and networks, FCS connects all kinds of sensors, actuators, and controllers by fieldbus control network instead of the traditional connection of hardware. Meanwhile, computers with windows operating systems have been found more and more applications in process industry. With the combination of these two trends, a more open framework appears, where various fieldbus equipments from different companies can be connected with each other, and all kinds of application programs of equipment management, supervisory and configuration also can cooperate with each other. As a middleware between hardware and software, the Object Linking and Embedding (OLE) for Process Control (OPC), sets an industrial computing environment standard to make different client software able to access any data source[1]. The communication standard for the supervisory control system based on OPC technology is the Component Object Model (COM) and the Distributed Component Object Model (DCOM).

The above framework realizes the modularization and avoids the repetition work in the system development. However, there exist several problems. The first one is that although the system is modularized and accesses between modules are standardized, it is still not open enough and its extension is also difficult, due to the tight coupling between modules. The second one is that DCOM as a wired heavyweight protocol cannot pass through a firewall. It is difficult for DCOM to be used in heterogeneous systems and to get service from one platform to another platform. The third one is the frangibility of the supervisory control system based on OPC. The server and client must run synchronously, where any little error may interrupt the server's application programs. The last one is the lack of a unique data exchange technology for the seamless integration of different systems into the same framework.

To overcome the above shortcomings, this work proposes a new framework for distributed fieldbus remote supervisory control system based on the Windows DNA (Windows Distributed interNet Applications Architecture) developing model, which adopts the XML (eXtensible Markup Language)[2,3] and SOAP (Simple Object Access Protocol) technologies as communication mechanisms of distributed components.

II. Fundamental Technologies Of The New Framework

A. *Windows DNA developing model*[4]

Windows DNA developing model creates a seamless environment for developing multitier distributed computing applications by unifying PC, client-server, and web-based application development around a common, component-based application architecture. It has the true interoperability and support industrial standard protocols, such as HTML, XML, and SOAP. It can solve the problems resulting from the heterogeneous systems and different platforms.

B. *XML data exchange technology*[4,5]

XML is a markup language for documents containing

Manuscript received September 27, 2004. This work was supported by the National Natural Science Foundation of China (No. 20206027), the Natural Science Foundation of Zhejiang Province (No. 202046) and the National 973 Program of China (No. 2002CB312200).

Yingzi Shi is with the School of Education Science, Hangzhou Teachers College, Hangzhou 310012, China (e-mail: syz89@163.com).

Jiangang Lu is with the National Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou 310027, China (phone: 86-571-87952829; fax: 86-571-87951879; e-mail: jglu@iipc.zju.edu.cn).

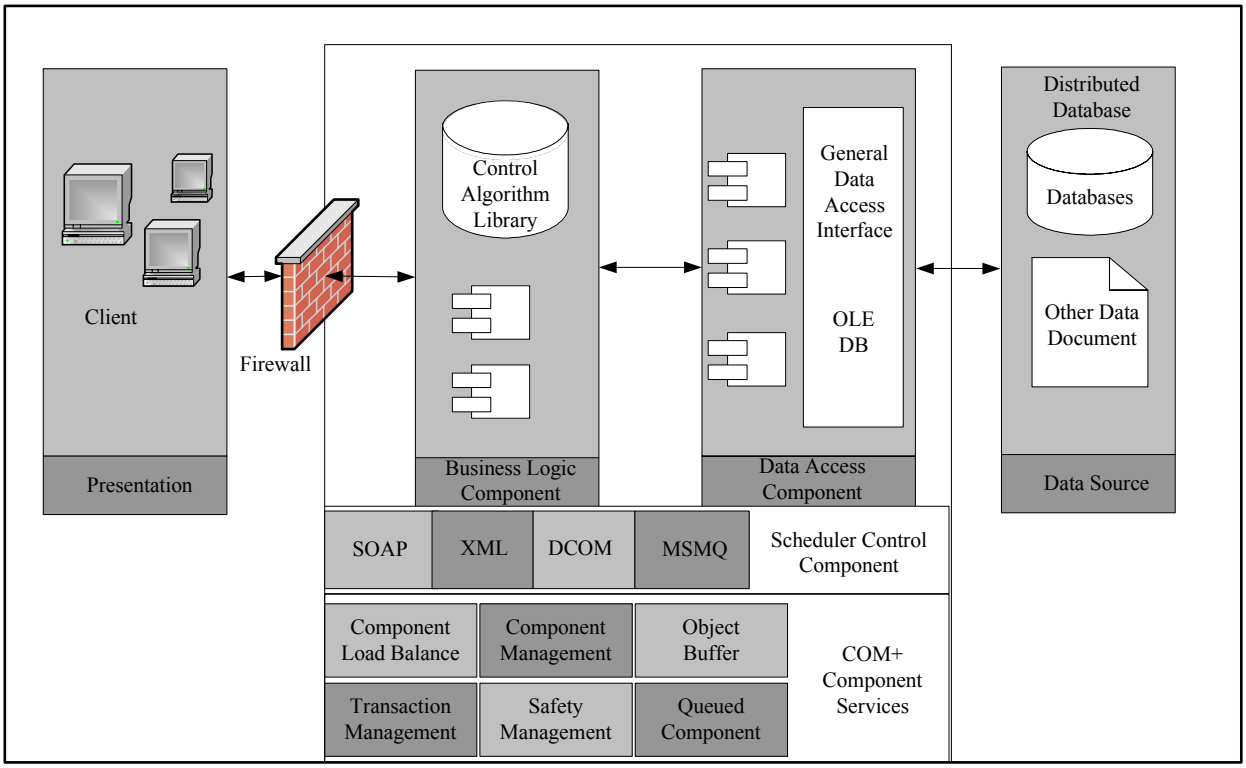

Fig. 1. Framework for distributed fieldbus remote supervisory control system

structured information. The XML specification defines a standard way to add markup to documents. XML as a general-purpose description language has an easy access to interfaces for electronic data exchange between different platforms.

C. *SOAP protocol* [5]

SOAP is a lightweight protocol for exchange of information in a distributed environment. 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 representing remote procedure calls and responses. SOAP can be potentially used in combination with a variety of other protocols.

III. Framework for the Distributed Fieldbus Remote Supervisory Control System

According to Windows DNA architecture, multitier architecture is used to build the framework of distributed fieldbus remote supervisory control system. A tier is a logical separation of functionality from another piece of functionality. Between the business logic tier and the presentation tier, there is a firewall. Information is transferred from the data source tier to the data access tier, then to the business logic tier, and finally to the presentation tier where the application interacts with the user. To partition the system into independent tiers can reduce the complexity of the whole system and enable the application extensible.

As shown in Figure 1, in this new framework of distributed fieldbus remote supervisory control system, there are five tiers: data source, data access, business logic, scheduler control, and presentation. In this architecture, COM+ (the next generation of COM) component services are responsible for the basic COM and transaction functionality and Internet Information Server (IIS) is responsible for web application services.

IV. Data Flow Model of the Distributed Fieldbus Remote Supervisory Control System

In distributed fieldbus remote supervisory control system, the data flow models can be classified into two categories: network management data flow models as shown in Figure 2 and intelligent device running data flow models as shown in Figure 3. The presentation tier uses SOAP protocol to realize the remote scheduling of the business logic components, and the business logic tier picks up the useful information from XML files through the corresponding Document Object Model (DOM) analyzer.

A brief description of the functional modules implemented in each tier is given as follows:

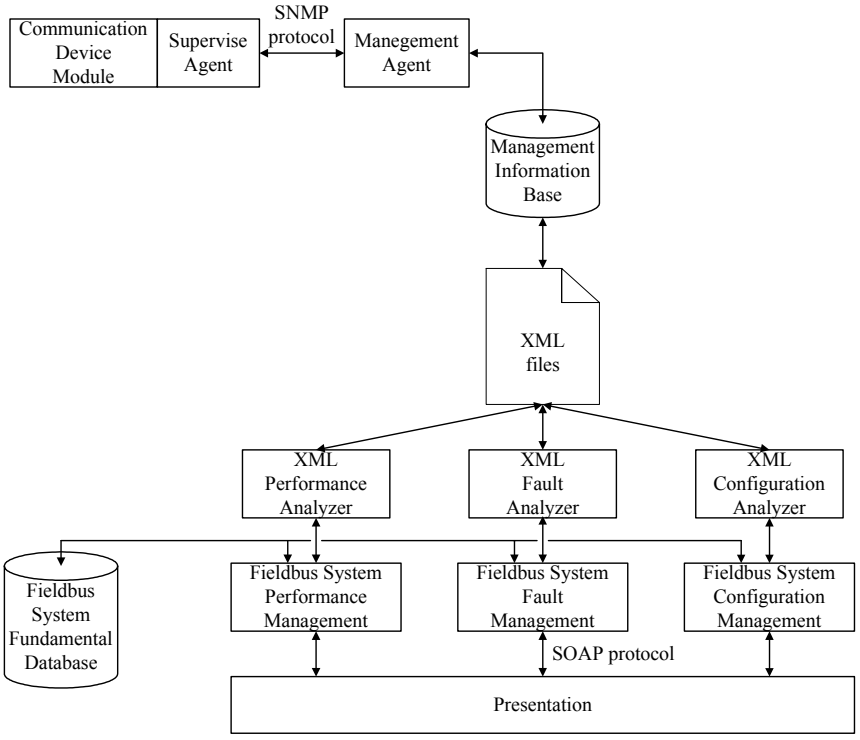

Fig. 2. Network management data flow model of the distributed fieldbus system

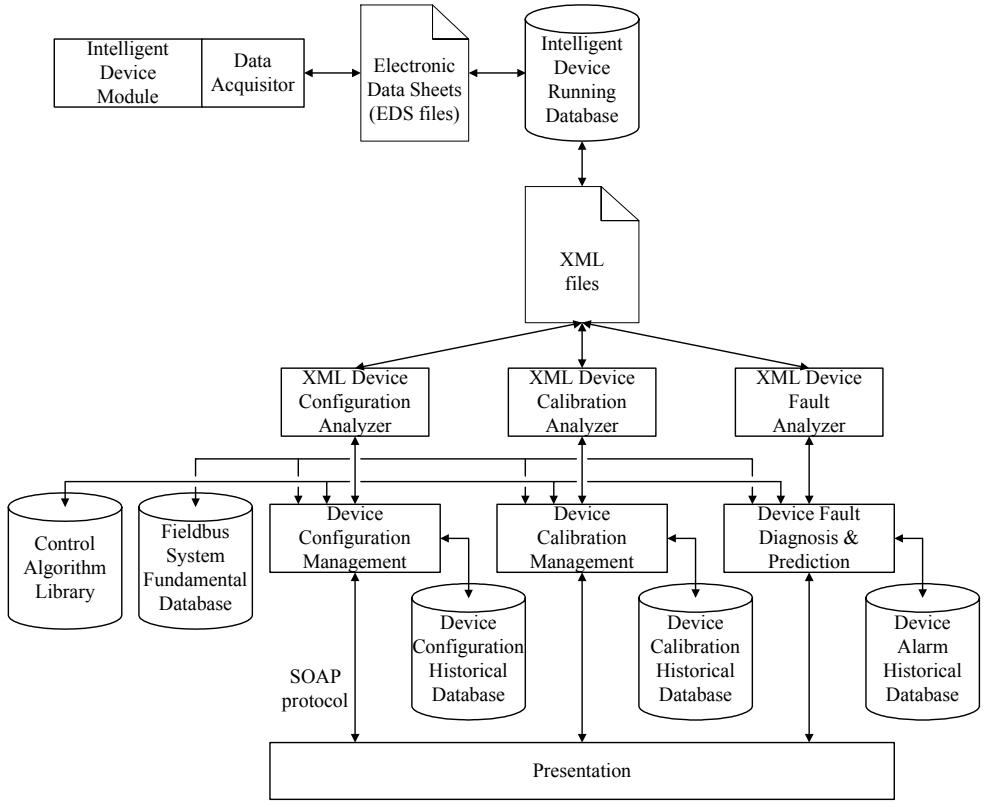

Fig. 3. Intelligent device running data flow model of the distributed fieldbus system

(1) Data source tier. It includes Fieldbus System Fundamental Database, Network Management Information Base, Intelligent Device Real-Time Databases, Device Configuration Historical Databases, System Fault Historical Database, and so on. The Unified Modeling Language (UML) is used to design these database models.

(2) Data access tier. It includes Fieldbus System Data Access Middleware, Management Information Base Access Middleware, Intelligent Device Real-Time Data Access Middleware, Device Configuration Historical Data Access Middleware, System Fault Historical Data Access Middleware, and so on. Data Access Middlewares are responsible not only for accessing the encapsulated data, but also for making the data position clear to the applications. The data access tier enable the unified access to different data source regardless of different database content and structure. The technologies of Object Linking and Embedding Database (OLEDB) and ActiveX Data Objects (ADO) are used to develop the data access component.

(3) Scheduler control tier. It includes data exchange mechanism based on XML, distributed communication mechanism based on SOAP, and traditional communication mechanism based on DCOM. These scheduler control components make the fieldbus remote supervisory control system a true network distributed application, which has advantages of loose coupling component scheduler, seamless integration of components, and easy extensibility.

(4) Business logic tier. It includes Fieldbus System Performance Management, Fieldbus System Fault Management, Fieldbus System Configuration Management, Device Configuration Management, Device Calibration Management, Device Fault Diagnosis and Prediction, and so on. The design of business logic tier conceals the complex interactions requested by a series of business process rules. The business logic tier processes and manipulates the data. The Unified Modeling Language (UML) is used to design the business logic components.

(5) Presentation tier. It includes Network Topology Subsystem, Network Performance Analysis Subsystem, Device Configuration History Subsystem, Device Debugging History Subsystem, Real-Time Supervisory Control Subsystem, Device Configuration Management Subsystem, Network Configuration Management Subsystem, Device Debugging Subsystem, System Fault History Subsystem, and so on.

V. CONCLUSION

A new framework for distributed fieldbus remote supervisory control system based on Windows DNA developing model, which adopts the XML and SOAP technologies as communication mechanisms of distributed components, is proposed to solve the problems of the traditional remote supervisory system. It has good extensibility and feasibility, and can help programmers design the distributed fieldbus remote supervisory control solutions.

REFERENCES

[1] OLE for Process Control Data Access Automation Interface Standard Version 2.02[S]. OPC Foundation, 1999.

[2] Wollschlaeger, M., 2000. A framework for fieldbus management using XML descriptions. Proceedings of the 2000 IEEE International Workshop on Factory Communication Systems, 6-8 Sept. 2000, pp. 3-10.

[3] Buhler, D., Kuchlin, W., 2000. Remote fieldbus system management with Java and XML. Proceedings of the 2000 IEEE International Symposium on Industrial Electronics, 4-8 Dec. 2000, pp. 1-6 (vol. 1).

[4] Gross, C., 2000. A Programmer's Introduction to Windows DNA. Springer-Verlag, New York.

[5] Travis, B.E., 2000. XML and SOAP Programming for BizTalk Servers. Microsoft Press, Redmond, Washington.