DEVELOPMENT OF SERVICE ORIENTED WEB-BASED SCADA APPLICATION

Traian Turc, Adrian Gligor
"Petru Maior" University of Tg.Mures
traian.turc@engineering.upm.ro, agligor@engineering.upm.ro

ABSTRACT

This paper presents some considerations on the development and use of Web services in Web-based SCADA applications . SCADA systems are becoming more and more comon allowing the user to monitor and control local and remote technological processes. SCADA Web-based applications are SCADA applications for web clients, that run in web browsers. The use of Web services enables the development of web SCADA applications in a consistent way, by using the standardized methods for data transmission. Web services use XML standard for data transfer. This paper also proposes the use of the XML standard for describing virtual instruments.

Keywords: web service, virtual instrumentation, SCADA, web application

1. Introduction

SCADA web applications used for monitoring and controlling processes implies the development of HMI(human machine interface) used in order to mimics as closely as possible the technological process and to allow a user-friendly interface.

In consequence, virtual instruments that are as diverse as possible and as similar to the real ones are required.

The drawback of this diversity is that it makes the process of creating the applications more difficult, imposing thus the need of a consistent way for the implementation of a range as wide as possible of virtual instruments.

The transmission of data acquired from technological systems to SCADA applications also requires the use of some standardized methods of data transfer.

Web services provide such a method by using standards based on the XML format.

2. Considerations on web services

Web services are applications provided by service servers. The request for a service implies the call for a URL (Uniform Resource Locator).

Web servers provide their clients with Web services through the use of the HTTP protocol.

Clients requests are received by the service server which in return gives the client an answer in an XML format through the use of the HTTP protocol.

The use of Web services offers numerous advantages to both applications developers and users.

The use of Web services implies the use of simple protocols that are easy to implement in comparison with other methods.

Client applications can be in the form of desktop, mobile or Web-based applications. Any type of programming language that supports this protocols can be used, regardless of the way in witch the Web services were implemented.

3. Web services used for the transfer of data gathered from the ethnological system.

Lets assume that we have an technological system that consists from 7 actuators placed in the points ACT01-ACT07.

Each actuator provides data such as: input pressure, the actuator's status, the position of the actuator and the torque of the actuator.

The SCADA web application needs the data gathered from the actuators ACT 01 – ACT 07, data related to the input pressure TP 01 – TP 07, the status of each actuators, the position of this actuators and the torque of the actuators in order to display a HMI in accordance to this actuators.

This data will then be collected through the use of Web services. Web services will be required for every set of data.

The Web service that sends the pressure recorded in one of the points TP01 – TP07 will have to return the value of the pressure in accordance to the parameter that represents the number of the actuator, parameter which is sent by the client application.

Let us assume that the name of the service used for the recording of the pressures is 'service1.asm'.

If the service: *service1.asmx* is installed on the server localhost, port 51655, then the URL used for the service call is http://localhost:51665/service1.asmx.

The service client sending a service request message to a service provider. The service provider returns a response message to the service client. The request and subsequent response connections are defined in some way that is understandable to both the service client and service provider

A client application that requests the value recorded in the point TP03 for example, must provide besides the URL a parameter that will specify the TP.

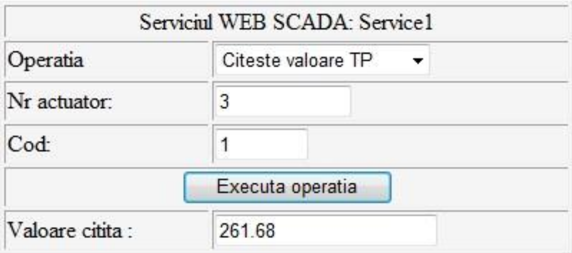

Fig. 1 – Client request for TP03

The clients request is sent to the web service provider in a XML format in the following format:

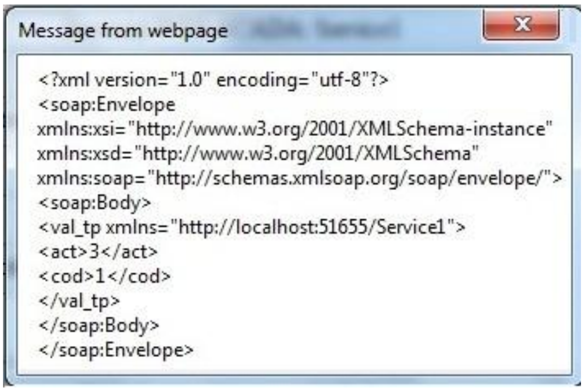

Fig. 2 – XML client request

The communication can involve either simple data passing or it could involve two or more services. In our case simple data is passing and the Web service provider, constructs its answer in the same XML standard, in the following format:

Fig. 3 – XML server provider answer

In the case of an SCADA Web application, the services used for reading the values gathered are called for repeatedly at regular time intervals for each value recorded, thus a client application will be capable of requesting the values of all the points TP01-TP07, in a similar way to the client application below:

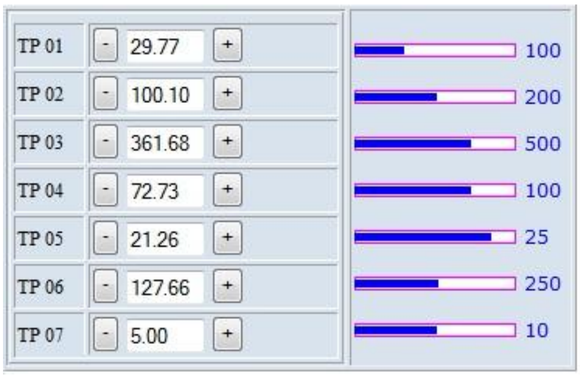

Fig. 4 –TP01-TP07 - virtual instrument

We can assume that the application shown above is a virtual instrument used for displaying the values recorded in the points TP01-TP07 , instrument which can be built in a more complex HMI.

Following the same principle, one can build services and applications that will allow the user to monitor the other parameter values such as the position of the actuators or the torque required for them to function.

For the process of monitoring the status of the actuators, we need a type of service which will provide us with information about the status of the actuator, information such as: actuator open/closed, actuator opening/closing.

The client application that requests the status of actuator ACT04 for example, must provide besides the URL, the parameter that indicates the actuator.

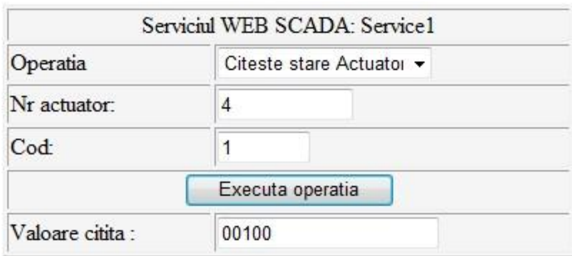

Fig. 5 – Client request for actuator status

As one can observe, the value returned by the Web service is built in the form of a string that contains boolean elements (0 or 1) which in fact encodes the four possible states : actuator open/closed and actuator opening/closing.

By running the service for each actuator in turn, an HMI can be created that contains the status of all the actuators. The following figure shows an example of an application that monitors the status of the actuators.

Fig. 6 –Actuators status -virtual instrument

In the eventuality that an complex HMI is build which uses virtual instruments of the fashion of those previously shown, we must define a mechanism that will enable the virtual instrument to find on its own his designated service by calling an service broker. After this process, in accordance to the description provided by the broker, the virtual instrument will request the service from the service provider.

4. Software Architecture of a virtual instrumentation

A proposed architecture of a virtual instrumentation application for SCADA, is shown in Figure 7.

Virtual instrument type components are built to make available an interface that provides data for initialization, then, built a template and process data and related orders if necessary. Each instrument that complies with this architecture will implement a method that will require the service it wants to meet in a SCADA system Services Broker. SCADA Services Broker will provide information about the components that control hardware instrumentation, virtual instruments that finally will work throughout the period of operation.

Broker Services SCADA system is an entity that registers providers of tools, data and services. Basic reason for that came from the need to allow modification of the SCADA system on the fly.

Implementing the service requires an interface known by the virtual instrument and Service Provider. This interface is made known through a standard application.

Data management facilities are proposed to call the database.

Data transfer in this system is done via messages encapsulated in XML structures.

Service Provider makes available the interface to hardware devices with based system services. It must be known the implementation details of the device's handle, the physical instrument and how they interact with virtual instruments and SCADA Services Broker.

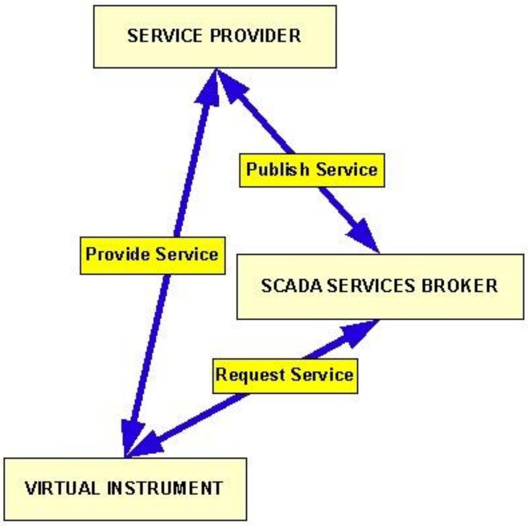

Fig. 7 – SCADA services broker

5. The use of the XML standard for describing virtual instruments

To develop such a system, an API for instrumentation was performed on a Java platform that contains the minimum set of tools used to monitor and control gas plant. Characteristics of instruments are stored in a database, accessible through the SCADA Service Broker. These characteristics are used to manufacture components and manipulate them. Their description was done using the XML standard [2]. For example, the description of system pressure regulation virtual instrument (shown in Figure8) and which can specifies the percentage of opening. Order sent codified in accordance with the XML standard is:

<vimeter type=”action”>
<addr>ACT01 </addr>
<type> percent </type>
<val> 20</val>
</vimeter>

During the initialization of the component for pressure, the later is recorded as being active in SCADA Service Broker on a message with the following form:

<vimeter type=”request-for-monitor”>
<caption> Actuator </caption>
<code> BIFI-276 </code>
<addr>ACT01 </addr>
<status>ready</status>
<locale>Instr. replaced by maintanance op.</locale>
<date-of-ready>12:05:2010 </ date-of-ready >
</vimeter>

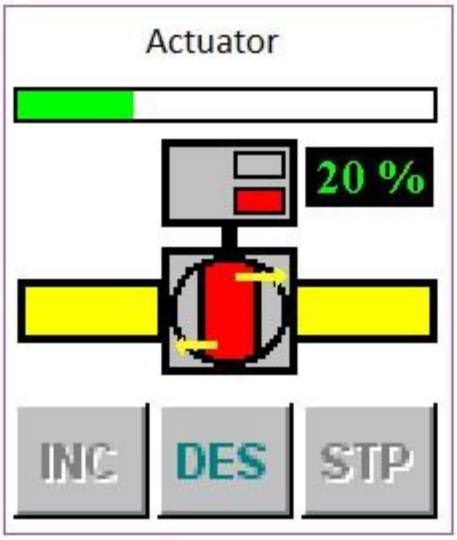

Fig. 8 Actuator-virtual instrument

Through the use of virtual instruments and web services, SCADA web based applications that contain HMI interfaces which mimic as accurate as possible the technological processes can easily be obtained.

In general, HMI presents the processed data to a human operator, and through this, the human operator may monitors and controls the process.

By providing information alerts, commands and other tools, an HMI connects the user with the process being controlled. The HMI as a graphical user interface includes plug-ins controls, which reflects the status of the SCADA system at a given time.

The SCADA client application offers an intuitive and user friendly HMI.

The SCADA client application also offers support on multiple screens, which can contain combinations of synoptic diagrams and text in order to display an event lists, alarm lists and trends.

1. Conclusion and future work

This paper presents an improvement to SCADA systems, consisting of classical applications that can migrate to Web-based technologies.

The solution presented enables heterogeneous integration solutions using XML standard and operation of the system software components.

An important point to mention is the concept of service software in SCADA systems. It provides extensibility and standards support by helping to streamline the integration of heterogeneous components.

A novelty is the introduction of interfacing through SCADA Remote Clients, which facilitates the implementation of the multilevel management systems in geographically widely distributed applications with WAN connectivity.

SCADA Remote Clients benefit from similar facilities to SCADA Local Clients multilevel, allowing efficient management.

The purpose of the present work is to develop a standard interface API and Virtual Instrumentation web interface to ensure efficient development of new applications such as SCADA and easy updating of the existing ones.

Through the concept of virtual instrument component, we ensure the future possibility of developing platform independent systems.

As a future work, the achieving of a uniform description based on XML for virtual instrumentation used is intended in order to allow development of new systems or to convert the existing ones to open systems.

References

[1] He Lingsong; Dengpa Zhang, *XVIML: an extensible markup language virtual instrument*, IEEE Autotestcon, 2005., 26-29 Sept. 2005

[2] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler and F. Yergeau, *Extensible Markup Language (XML) 1.0 (Fifth Edition).* W3C Recommendation 26 November 2008, http://www.w3.ooorg/TR/2008/REC-xml-20081126/, 2008.

[3] Schneider Electric Ltd, *SCADA's Powerful System Architecture* , Australia 2003

[4] Traian Turc, Gligor Adrian, *Gas Plant SCADA Software Application*, Intering - Interdisciplinary in Engineering , Tg. Mureş, 2009.

[5]Traian Turc, *Industrial processes automation and monitoring system includes a central data collector and a computer for modular control via Microsystems*, ISI Web of Knowledge -JO7B1-2003.