Application for Communication

Visual Basic .NET OPC Client for the XML-DA Interface of the SIMATIC NET OPC Server

Introduction
Warranty, Liability and Support

We accept no liability for information contained in this document. Any claims against us - based on whatever legal reason - resulting from the use of the examples, information, programs, engineering and performance data etc., described in this document shall be excluded. Such an exclusion shall not apply in the case of mandatory liability, e.g. under the German Product Liability Act (“Produkthaftungsgesetz”), in case of intent, gross negligence, or injury of life, body or health, guarantee for the quality of a product, fraudulent concealment of a deficiency or breach of a condition which goes to the root of the contract (“wesentliche Vertragspflichten”). However, claims arising from a breach of a condition which goes to the root of the contract shall be limited to the foreseeable damage which is intrinsic to the contract, unless caused by intent or gross negligence or based on mandatory liability for injury of life, body or health. The above provisions does not imply a change in the burden of proof to your detriment.

The application examples do not purport to cover all details or variations in equipment, nor do they provide for every possible contingency. They are not customer-specific solutions. They are only intended to provide support for typical tasks. You are responsible for ensuring that the described products are used correctly. These Application Examples do not relieve you of the responsibility in safely and professionally using, installing, operating and servicing equipment. By using this application example you accept that Siemens is not liable for any damages except for those specified in the above liability clause. We reserve the right to make changes to these Application Examples at any time without prior notice. If there are any deviations between the recommendations provided in this application example and other Siemens publications – e.g. Catalogs – the contents of the other documents have priority.

Copyright© 2004 Siemens A&D. Any form of duplication or distribution of these application examples or excerpts hereof is prohibited without the expressed consent of Siemens Energy & Automation, Inc.

For questions about this document please use the following e-mail address:

mailto:csweb@ad.siemens.de
Foreword

Objectives of the application

Exchanging process data between logic controllers and computers via OPC is a core element of automation technology.

The application on hand shows a possible procedure for independently generating individual OPC clients using the programming language Visual Basic .NET by Microsoft.

A visualization of a mixed process is hereby realized in Visual Basic .NET. The programming paradigm ".NET" by Microsoft is used here.

Main contents of this application

The following teaching materials are provided with this application:

- Installation and configuration of a web-server (in this example the web-server of the internet information services of Windows). This web-server is used for accessing the OPC server.
- Highlighting safety aspects (authentication and authorization of clients) for accessing the web-server and the OPC XML-DA web service.
- Demonstration in dealing with web services in .NET, especially the OPC XML-DA web service.
- Demonstration of the most important OPC XML-DA methods for accessing the process variables
- Demonstration of error handling with .NET
- Generating an OPC client as Windows application as well as ASP.NET web application (for calling a web browser) with similar function range.

Delimitation

This application does not contain a complete description

- of network safety aspects (encoded communication via SSL, application of firewalls, using VPNs)
- of the .NET framework,
- of the OPC specification as well as
- deeper level ASP.NET mechanisms

Basic knowledge in the area of object oriented programming is required. Further knowledge in UML (Unified Modelling Language) is an advantage.
Structure of the document

The documentation of this application is divided into the following main parts:

- **Introduction**
- **Extension**
- **Demonstration**

In addition, two setup programs (for Windows application and for ASP.NET-application respectively), which also contain the source code of the application, as well as the STEP 7 code, are available.

This first document, the **Introduction**, is aimed at persons who just want to gain a quick overview.

<table>
<thead>
<tr>
<th>Part</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Introduction** | Application Description and Function Principles  
Provides a general overview of the contents. You will learn about the components used (standard hardware and software components and the specially created software). |
| **Extension** | Function Principles in detail and Program Structure  
Discusses the detailed function processes of the involved hardware and software components, the solution structures, and where sensible the concrete implementation of this application. This part is necessary if you want to learn about the interaction of the solution components, for example in order to use them as the basis for own development. |
| **Demonstration** | Structure, Configuration and Operation of the Application  
This part leads you step by step through the structure, important configuration steps, commissioning and operation of the application. |

An additional component available is the S7 program code.

<table>
<thead>
<tr>
<th>Part</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setup Program of the Windows Application</td>
<td>Installs the Windows application on your PC. The source code of the application is available in the program directory.</td>
</tr>
<tr>
<td>Setup Program of the ASP.NET Application</td>
<td>Installs the ASP.NET application on your PC. The source code of the application is available in the program directory.</td>
</tr>
<tr>
<td>S7 program code</td>
<td>The S7 program code contains the code and a user interface.</td>
</tr>
</tbody>
</table>
Reference for Automation and Drives Service & Support

This entry originates from the internet application portal of the A&D Service and Support. Clicking the link below directly displays the download page of this document.

Table of Contents

Table of Contents ......................................................................................................... 6
Application Description ............................................................................................... 7

1 Basic Information on OPC .................................................................................. 7

2 Automation Task....................................................................................................... 9
  2.1 Overview ........................................................................................................... 9
  2.2 Requirements ................................................................................................... 10

3 Automation Solution ............................................................................................... 11
  3.1 Overview of the overall solution ................................................................. 11
  3.2 Description of the core functionality ......................................................... 12
  3.3 Required hardware and software components ........................................ 16
  3.4 Basic performance data .............................................................................. 17
  3.5 Alternative solutions ................................................................................... 18

Appendix and List of Further Literature ................................................................. 20

4 Glossary ........................................................................................................ 20

5 Literature ....................................................................................................... 21
  5.1 Bibliographic References ........................................................................... 21
  5.2 Internet links ................................................................................................ 21
Application Description

Content

Here, an overview over using the XML-DA interface of the SIMATIC NET OPC server is provided. You will learn about the components used (standard hardware and software components and the specially created software).

The basic performance data show how powerful this application is.

1 Basic Information on OPC

Here you will find information on …

further basics for the application of the XML-DA interface and on the tasks solved with this interface.

What is OPC?

OPC serves for data exchange between hardware and software components of different producers in automation technology. OPC offers a collection of software interfaces for data exchange between PC applications and process devices.

Originally based on the COM (Component Object Model) or DCOM (distributed COM) technology by Microsoft, OPC in its current specification offers a number of interfaces for different platforms. The OPC specifications are available on the internet at \3\.

Data exchange via OPC occurs from an OPC server to an OPC client. Via standardized interfaces, the OPC server provides process information of a device at its interface. They can be read and described with an OPC client.

What is OPC Data Access?

The Data Access (abbrev. "DA") specifications of OPC define, how process data are exchanged between an OPC server and an OPC client. Data exchange is here based on the COM or DCOM technology by Microsoft and can only be applied on Microsoft Windows-based systems. This specification was the very first specification by OPC.

In the mean time, there are specifications for transferring process alarms (OPC Alarms & Events), for the data exchange between OPC servers (OPC Data eXchange) and many more.
What is the OPC XML-DA interface?

OPC XML-DA (Data Access to an OPC server via XML) is, like the “classic” OPC Data Access, an interface for accessing process data of an OPC server. This is, however, completely platform independent (as opposed to the COM technology used for OPC Data Access V2.05) and is also supported by the development environment Visual Studio .NET.

Data access by means of OPC XML-DA has a function scope based on OPC Data Access, however, only write and read services are available. Change controlled feedback messages via data changes, as for COM or DCOM OPC DA interfaces are not possible for OPC XML-DA.

Data exchange via the OPC XML-DA interface is based on the open standards by XML and SOAP (see Glossary).

Why should the XML-DA interface be employed?

The exchange of process data with OPC via a local network (LAN), an intranet or the internet is, in theory, also possible with the Data Access interface based on COM / DCOM. The XML-DA interface offers an almost equal function scope along with a much simpler configuration, parameterization and administration.

The OPC XML-DA interface being based on open standards, its application is not restricted to a certain system (e.g. Microsoft Windows).

Advantages and disadvantages of the interfaces

<table>
<thead>
<tr>
<th>Interface</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>XML-DA</td>
<td>• Platform independent</td>
<td>• Not suitable for transferring large data volumes</td>
</tr>
<tr>
<td></td>
<td>• Simple administration as based on SOAP and XML</td>
<td>• A web server is required on the OPC server side</td>
</tr>
<tr>
<td>Data Access via DCOM</td>
<td>• Enables feedback from server to client</td>
<td>• Windows-based</td>
</tr>
<tr>
<td></td>
<td>• Higher data throughput</td>
<td>• Not directly supported by .NET</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Complicated administration</td>
</tr>
</tbody>
</table>
2 Automation Task

Here you will find information on the automation task discussed in the documentation on hand.

2.1 Overview

Overview of the automation task

The following figure gives an overview of the automation task.

Figure 2-1

Description of the automation task

Using the example of a simulated mixing process this application shows how this process can be visualized. The connection of the visualization station with the plant should either be made via the internet, intranet, or a local network.
2.2 Requirements

Requirements to the visualization solution

- Access to the XML-DA interface of any OPC server
- Access to the OPC server of the head-end station via internet, intranet or a local network
- Symbolic addressing of process data and setpoint values
- Cyclic (synchronous and asynchronous) reading of process data from the controller
- Synchronous writing of setpoint values to the controller
- Realizing with programming language Visual Basic .NET
- Displaying and controlling process variables with a Windows application and an ASP.NET application (for calling by means of any web browser), where both versions have approximately the same function scope.

Requirements for the head-end station

- Access protection for the OPC server via login
- Provision of symbolic address information

Requirements for the controller

- Saving current process values in a data block.
- Saving the setpoint values for the process in a data block.
3 Automation Solution

Here you will find information on …

the solution selected for the automation task, the components used, and
the tasks of the individual components.

3.1 Overview of the overall solution

Display

The following figure displays the most important components of the
solution:

Figure 3-1
3.2 Description of the core functionality

User interface

The following figure contains the user interface of the ASP.NET application (left) and the Windows application (right). The elements are described below.

Figure 3-2

Description of user interface

The following table explains the individual elements of the user interface.

Table 3-1

<table>
<thead>
<tr>
<th>No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Buttons for starting and terminating the visualization process. This establishes the connection to or disconnects from the OPC server.</td>
</tr>
<tr>
<td>2</td>
<td>Area for entering and writing the setpoint values.</td>
</tr>
<tr>
<td>3</td>
<td>Display of process values (actual values)</td>
</tr>
<tr>
<td>4</td>
<td>Buttons for starting and terminating the mixing process.</td>
</tr>
<tr>
<td>5</td>
<td>Graphical visualization of the mixing process (only Windows application)</td>
</tr>
</tbody>
</table>
Process sequence of main functionality

This figure shows the internal communication structure of the application on hand. It is assumed, that the connections between the components have already been established.

Figure 3-3

Explanation of main functionality

Here you will learn how the individual components of this application communicate with each other and which technologies are being used.

Table 3-2

<table>
<thead>
<tr>
<th>No.</th>
<th>Instructions</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Data exchange between controller and OPC server occurs via a configured S7 connection.</td>
<td>The configuration is made with STEP 7 (NetPro).</td>
</tr>
<tr>
<td>2</td>
<td>The OPC XML web service of SIMATIC NET will be included in the IIS Web Server (part of the Windows operating system). It provides data of the OPC server to the OPC Client.</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>The data exchange between the OPC XML web service and the connected OPC client (in this case the Windows application and the ASP.NET application) occurs via the SOAP protocol.</td>
<td></td>
</tr>
</tbody>
</table>
Describing the process in the controller

The S7 control program simulates a mixing process. Liquids are hereby fed into a tank via two inlet valves. If the tank is full, the liquid is mixed and heated up. After the setpoint temperature has been reached, the liquid is cooled down again and then drained via an outlet valve.

The process variables for the data exchange with the OPC server are saved to three different blocks.

Advantages of applying the OPC XML-DA interface

The application of the OPC XML-DA interface of the SIMATIC NET OPC server for plant visualization via a local network, intranet or the internet, has the following advantages:

- The SIMATIC NET OPC server supports all communication protocols and services provided by the communication modules.
- Simple transmission of process data via a local network, intranet or the internet.
- Simple configuration of the OPC server (as usual in STEP 7)
- Simple commissioning of OPC XML web service (incl. encoding and access protection)
- OPC clients are platform independent (due to the employed SOAP protocol being platform independent)

Advantages of the Windows application

The implementation of the OPC client as Windows application using .NET has the following advantage for you:

- Generating user interfaces and functions of any level of complexity which can be tailored to a specific case.
- Simple and cost effective distribution of the visualization application
Advantages of the ASP.NET application

For an ASP.NET application, the entire program intelligence is located on the server (here: head-end station). In the Internet Browser of the Clients, which calls the ASP.NET application, only static information is displayed. This has the following advantages:

- Very cost effective solution low in maintenance, as no installation required on the client side (only Internet Browser)
- Minimum maintenance costs due to central software administration
### 3.3 Required hardware and software components

#### Hardware components for visualization station and head-end station

**Table 3-3**

<table>
<thead>
<tr>
<th>Component</th>
<th>No.</th>
<th>MLFB / Order number</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIMATIC Field PG</td>
<td>1</td>
<td>6ES7 711-2BA21-3LB3</td>
<td>Incl. STEP 7 software and network card</td>
</tr>
<tr>
<td>SIMATIC RACK PC IL 40 S V2</td>
<td>1</td>
<td>6AG4011-1CA21-0XX0</td>
<td>Incl. Network card and Windows XP Professional</td>
</tr>
<tr>
<td>Standard HUB or Switch (e.g. NETGEAR FS105 5-Port 10/100 MBit/s)</td>
<td>1</td>
<td>Depending on product</td>
<td><a href="http://www.netgear.de">www.netgear.de</a></td>
</tr>
<tr>
<td>Ethernet lines</td>
<td>3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Hardware components for the controller

**Table 3-4**

<table>
<thead>
<tr>
<th>Component</th>
<th>No.</th>
<th>MLFB / Order number</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>PS307 5A</td>
<td>1</td>
<td>6ES7307-1EA00-0AA0</td>
<td></td>
</tr>
<tr>
<td>CPU 315-2DP</td>
<td>1</td>
<td>6ES7315-2AG10-0AB0</td>
<td></td>
</tr>
<tr>
<td>CP 343-1</td>
<td>1</td>
<td>6GK7343-1EX20-0XE0</td>
<td></td>
</tr>
</tbody>
</table>

#### Standard software components

**Table 3-5**

<table>
<thead>
<tr>
<th>Component</th>
<th>No.</th>
<th>MLFB / Order number</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEP 7 V5.3</td>
<td>1</td>
<td>6ES7 810-4CC07-0YA5</td>
<td>Contained in the above field PG</td>
</tr>
<tr>
<td>SIMATIC NET IE SOFTNET-S7 V6.2</td>
<td>1</td>
<td>6GK1704-1CW62-3AA0</td>
<td>Installation on head-end station</td>
</tr>
<tr>
<td>IISLockdown Tool V2.1 (iislockd.exe)</td>
<td>1</td>
<td></td>
<td>Installation on head-end station. Tool for automatic protection of IIS web server</td>
</tr>
<tr>
<td>.NET Framework V1.1</td>
<td>2</td>
<td></td>
<td>On field PG and head-end station</td>
</tr>
<tr>
<td>Microsoft Visual Studio .NET 2003 Professional</td>
<td>1</td>
<td>Can be ordered via your administrator or at <a href="http://www.microsoft.com">www.microsoft.com</a></td>
<td>Optional on field PG or head-end station</td>
</tr>
</tbody>
</table>

**Note:**
- Network card and Windows XP Professional
- Depending on product
- Installation on head-end station
- Tool for automatic protection of IIS web server
- Optional on field PG or head-end station
Example files and projects

The following list contains all files and projects used in this example.

Table 3-6

<table>
<thead>
<tr>
<th>Component</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>21402169_OPC_XML-DA_STEP7_v10.zip</td>
<td>Contains the archived STEP 7 project.</td>
</tr>
<tr>
<td>21402169_OPC_XML-DA_CODE_WIN_v10.zip</td>
<td>Contains the setup program for the Windows application and the source code.</td>
</tr>
<tr>
<td>21402169_OPC_XML-DA_CODE_ASP_v10.zip</td>
<td>Contains the setup program for the ASP NET application and the source code.</td>
</tr>
<tr>
<td>21402169_OPC_XML-DA_Introduction_DOKU_v10_e.pdf</td>
<td>Application Description and Function Principles</td>
</tr>
<tr>
<td>21402169_OPC_XML-DA_Extension_DOKU_v10_e.pdf</td>
<td>Function Principles in detail and Program Structure</td>
</tr>
<tr>
<td>21402169_OPC_XML-DA_Demonstration_DOKU_v10_e.pdf</td>
<td>Structure, Configuration and Operation</td>
</tr>
</tbody>
</table>

3.4 Basic performance data

Application software

Due to the combination of different components in this example it is not possible to specify concrete performance data. The following statements on the SIMATIC NET OPC server can be made:

- The “classic” Data Access interface (COM-based) is 10% – 25% faster than the XML interface. The total speed also depends on the S7 communication between OPC server and S7 station.
- When using the internet, no general statement on transfer velocity can be made, as it may fluctuate strongly.
- Via the XML interface up to 10,000 OPC items can be read or written (see FAQ ID 15227599).
3.5 Alternative solutions

Alternatives for using the OPC XML-DA interface of the SIMATIC NET OPC server are introduced, evaluated and compared (advantages, disadvantages, ...).

Using the OPC-DA interface of the SIMATIC NET OPC server

The OPC-DA (Data Access) interface is the standard interface for exchanging process data. This interface is based on the COM technology. The implementation of the OPC-DA interface occurs with a COM-capable programming language (e.g. C++ or Visual Basic).

Programming of an OPC client with C# using the OPC-DA interface is described in the following application:

Advantages and disadvantages of the OPC-DA interface

Table 3-7

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>• High data throughput and high performance</td>
<td>• Being based on COM, it can only be applied on Windows operating systems</td>
</tr>
<tr>
<td>• Generating user interfaces and functions of any level of complexity which can be tailored to a specific case.</td>
<td>• High implementation workload (programming with high-level language; C++, C#, Visual Basic)</td>
</tr>
<tr>
<td></td>
<td>• Setting up distributed systems is work intensive (using DCOM)</td>
</tr>
</tbody>
</table>

Using WinCC with Web Navigator

Web Navigator is an option for WinCC for operating and monitoring plants and systems via the internet or the company intranet or LAN. The Web Navigator consists of a web server on a PC with WinCC software (single-user, client or server version). A web client can operate and monitor a running WinCC project via an Internet browser.
Advantages and disadvantages of the Web Navigator

Table 3-8

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Quick update rates due to event controlled communication</td>
<td>• Based on ActiveX technology which is exclusively supported by Windows systems.</td>
</tr>
<tr>
<td>• Optimal tailored clients for Operator Control &amp; Monitoring, Evaluation, Service &amp; Diagnostics</td>
<td>• License costs depending on the number of clients</td>
</tr>
<tr>
<td>• Taking over configuration data from an existing WinCC project for the web, generally without changes.</td>
<td>• Installation of the WebNavigator clients necessary on client side</td>
</tr>
<tr>
<td>• Minimum maintenance costs due to central software administration</td>
<td></td>
</tr>
<tr>
<td>• Integrated WinCC User management</td>
<td></td>
</tr>
</tbody>
</table>

Using the IT-CP

On top of the functionality of an S7 communication processor (PG/OP communication, S7 communication, ...), the IT-CP (CP 343-1 IT or CP 443-1 Advanced) offers the possibility of communicating via standardized protocols of the Internet world. This enables access to process data of a controller via HTTP and a web browser, access to data blocks of the controller via FTP, sending of e-mails via an integrated mail server.

Using the IT-CP is described in the application "Operator Control and Monitoring with the SIMATIC IT-CP via different WAN configurations" (see link \4\).

Advantages and disadvantages of the IT-CP

Table 3-9

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>• No additional PC (head-end station) necessary, as the web server is an integral part of the CP</td>
<td>• Extensive implementation of complex html-pages</td>
</tr>
<tr>
<td>• Quick implementation of simple user interfaces</td>
<td>• Currently no secure communication via https.</td>
</tr>
<tr>
<td>• Simple configuration via STEP 7</td>
<td>• Only suitable for small quantity frameworks (access to data of an S7-CPU via the backplane bus)</td>
</tr>
<tr>
<td>• Cost effective</td>
<td></td>
</tr>
</tbody>
</table>
Appendix and List of Further Literature

4 Glossary

The following explanations and definitions were taken from the Internet encyclopedia "Wikipedia", see \5\.

Authentication

Authentication refers to the process of verifying the identity of a person or a program based on certain features. This can, for example, occur by means of a fingerprint, a certain user name and password, or any other legitimation.

Authorization

Refers to the assignment and verification of access rights to data and services of a system by a user. Authorization mainly occurs after successful authentication.

HTTP

The Hypertext Transfer Protocol (HTTP) is a stateless data exchange protocol for transfer of data. It is one of the protocols provided by the TCP/IP protocol stack. Primarily it is used within the framework of the world wide web for transferring web-sites.

SOAP

SOAP is a protocol which can be used for exchanging data between systems, and calling functions on remote systems.

SSL

SSL (Secure Sockets Layer) refers to a transfer protocol which enables encoded communication using tunneling.

TCP/IP

Transmission Control Protocol / Internet Protocol. The internet protocol suite is a family of network protocols which form the basis for network communication in the internet.
5 Literature

5.1 Bibliographic References

This list is by no means exhaustive and only gives a selection of appropriate sources.

Table 5-1

<table>
<thead>
<tr>
<th>Topic</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>/1/</td>
<td>STEP 7 Automatisieren mit STEP 7 in AWL und SCL (Automating with STEP7 in STL and SCL) Hans Berger published by: Publicis MCD Verlag ISBN 3-89578-113-4</td>
</tr>
<tr>
<td>/2/</td>
<td>SIMATIC NET, Industrial communication SIMATIC NET, Industrielle Kommunikation mit PG/PC Band 1 – Grundlagen, Systemhandbuch (auf SIMATIC NET CD)</td>
</tr>
</tbody>
</table>

5.2 Internet links

This list is by no means exhaustive and only gives a selection of appropriate sources.

Table 5-2

<table>
<thead>
<tr>
<th>Topic</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>/3/</td>
<td>OPC Foundation <a href="http://www.opcfoundation.org">http://www.opcfoundation.org</a></td>
</tr>
</tbody>
</table>