SIEMENS

SIMATIC

PCS 7 process control system OpenPCS 7 (V8.2)

Function Manual

Valid for PCS 7 as of V8.2

03/2016
A5E36185039-AA

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

⚠ DANGER

indicates that death or severe personal injury **will** result if proper precautions are not taken.

⚠ WARNING

indicates that death or severe personal injury **may** result if proper precautions are not taken.

⚠ CAUTION

indicates that minor personal injury can result if proper precautions are not taken.

NOTICE

indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

⚠ WARNING

Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Siemens AG
Division Process Industries and Drives
Postfach 48 48
90026 NÜRNBERG
GERMANY

A5E36185039-AA
℗ 03/2016 Subject to change

Table of contents

Security information

1

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, solutions, machines, equipment and/or networks. They are important components in a holistic industrial security concept. With this in mind, Siemens' products and solutions undergo continuous development. Siemens recommends strongly that you regularly check for product updates.

For the secure operation of Siemens products and solutions, it is necessary to take suitable preventive action (e.g. cell protection concept) and integrate each component into a holistic, state-of-the-art industrial security concept. Third-party products that may be in use should also be considered. For more information about industrial security, visit http://www.siemens.com/industrialsecurity

To stay informed about product updates as they occur, sign up for a product-specific newsletter. For more information, visit http://support.automation.siemens.com.

Preface

2

Purpose of this documentation

PCS 7 is an open system that allows access to process values, archived process values and hardware interrupts by third-party applications. PCS 7 uses the following industry standards of the OPC Foundation for this purpose:

- OPC Unified Architecture (DA, HA, A&C)
- OPC Data Access
- OPC Historical Data Access
- OPC Alarm and Events

PCS 7 also supports access by third-party applications using the WinCC OLE DB database mechanism.

This documentation describes access to one or more PCS 7 OS servers using the OpenPCS 7 station. The information provided here will show you the configuration options with OpenPCS 7 and help you get started with the configuration of the OpenPCS 7 station in PCS 7.

To familiarize yourself in detail with the topics relating to the OPC interfaces, you will also require the OPC-documentation from the OPC Foundation. You can find the documentation on the Internet at: www.opcfoundation.org (www.opcfoundation.org)

Required basic knowledge

You require general knowledge of automation engineering to understand this manual. Because OpenPCS 7 is based on the PCS 7 process control system, you should also be familiar with the operation of PCS 7.

Basic knowledge of the general use of the PC/programming device and of the use of the Windows operating system is required.

Target Audience of this Documentation

This documentation is intended for PCS 7 users and SIMATIC S7 specialists.

Options for accessing PCS 7 documentation

The documentation required for PCS 7 includes the following types:

- **PCS 7 Readme**
  The readme file is available in two versions:
  - **PCS 7 Readme (offline)**
    This version is installed by PCS 7 Setup. The file only contains general information and links to documents on the Internet.
  - **PCS 7 Readme (online)**
    This version contains all information on the installation and use of PCS 7 in the format which is already familiar to you. The file is only available on the Internet to keep it always up-to-date.

  Note

  The information provided in the *PCS 7 Readme* (online) on the Internet takes precedence over **all** other PCS 7 documentation.

  Read this *PCS 7 Readme* carefully, because it contains important information and amendments on PCS 7.

- **PCS 7 System Documentation**
  System documentation contains information that covers several products, such as configuration manuals and Getting Started manuals. This documentation serves as a guideline for the overall system and explains the interaction between the individual hardware and software components.
  Note the information on "PCS 7 Documentation Portal Setup" in the product overview *Process Control System PCS 7; PCS 7 - Documentation*; Section "Options for accessing the documentation".
- **PCS 7 Product Documentation**
  Product documentation contains information about special hardware and software components. The individual documents provide detailed information on the specific component.

Full versions of the documentation are available from the "Technical Documentation SIMATIC PCS 7" website:
www.siemens.com/pcs7-documentation (www.siemens.com/pcs7-documentation)

You can find additional information in the product overview *Process Control System PCS 7; PCS 7 - Documentation*; Section "Options for accessing the documentation".

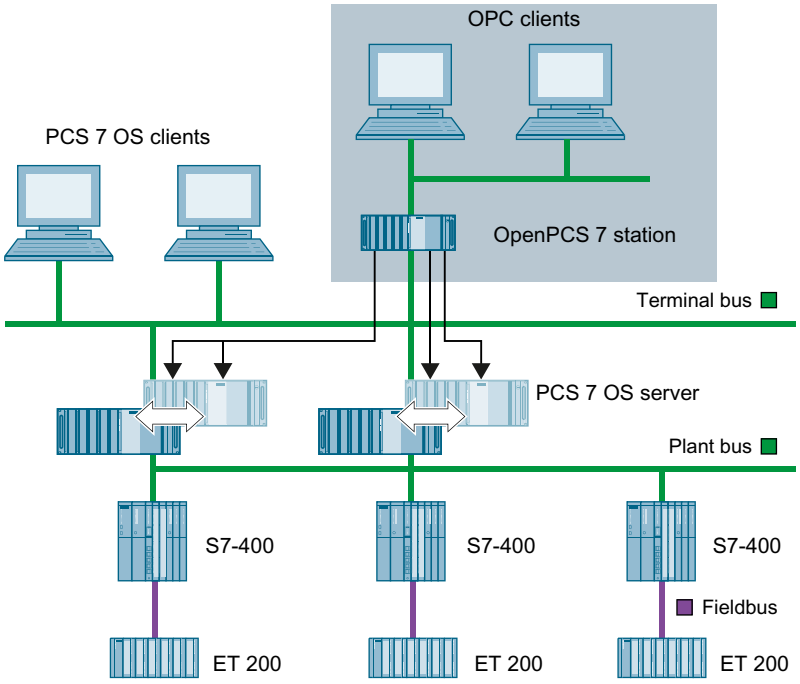
Basics

3

3.1 General information

Overview

Higher-level process control systems for production planning, process data evaluation, and process data management can access SIMATIC PCS 7 process data via the OpenPCS 7 station. The higher-level systems are OPC clients of the OpenPCS 7 station.



Depending on the system configuration, the OpenPCS 7 station can provide data from various OS servers and from a Process Historian. The OpenPCS 7 station hides the distribution of the data in terms of:

- Time period (OS1, OS2, …, Process Historian)
- Location (OS1, OS2, …, Process Historian) and
- Redundancy (OS1 master, OS1 standby …)

OpenPCS 7 replaces @PCS 7previously available in the context of SIMATIC PCS 7 and is, at the same time, the equivalent of the Connectivity Station and the Connectivity Pack of the WinCC SCADA system. Direct access to SIMATIC BATCH and SIMATIC Route Control data is not possible with OpenPCS 7.

Layout

The OpenPCS 7 station can be operated in various configurations:

- OpenPCS 7 station without OS client, based on a SIMATIC PCS 7 Industrial Workstation in the client version. This is the recommended preferred configuration.
- OpenPCS 7 station on an OS client.
- OpenPCS 7 station on an OS single station.
- OpenPCS 7 station on the OS server.

Note

The hardware and software requirements of the PC station on which the OpenPCS 7 station is to be installed must be met.

3.2 Microsoft basics

Microsoft basics

The basis of OPC "Classic" is provided by several Microsoft technologies. These technologies are explained in the following section.

OPC

OPC stands for "OLE for Process Control".

OPC is a standardized, vendor-independent software interface that allows data to be exchanged between hardware and software. One system can provide another system with process data via OPC .

OLE

OLE means "Object Linking and Embedding", the technology for embedding objects in documents.

COM

COM means "Component Object Model" and is necessary for the use of OPC.

COM is a central component of Windows operating systems and controls the interaction between multiple software components. By using COM , the OPCserver effectively becomes part of the Windows operating system and is therefore not dependent on file names, storage locations and versions.

COM defines a standard that allows objects to be defined as self-contained units in Windows and to access these units beyond the limits of a process.

COMobjects can be understood as extensions of the operating system. They are not dependent on programming languages and are available in principle to all applications.

The data and code of the object are not directly accessible to the user of the COMobject.

DCOM

DCOM is the acronym for "Distributed Component Object Model". It is based on COM technology and provides the additional option of communicating over a network.

Object

Objects are defined by properties and methods that can be used on objects.

Events

An event is used to control program flow. Program execution is not linear, instead special event handling routines are executed whenever a specific event occurs.

In terms of OpenPCS 7 and the OPC standard, events mean messages/operation messages of the PCS 7-system.

DLL

It is possible to use DLLs with Microsoft Visual Basic (VB) and Microsoft Visual Basic for Applications (VBA). DLL stands for Dynamic Link Library. A DLL is a dynamic link library. Users can link the functionalities of a DLL into their own applications. If you want to program an OPCclient with VB, make sure that the relevant automation interface DLL is selected in the VB Editor in "Project > References".

Collection

Collections are objects that support count and item properties. A collection consists of a certain number of items. An item can be collection-specific, any data structure, or an object. The count property specifies the number of items in a collection. In Microsoft Visual Basic, each item in a collection can be identified using loops.

3.3 OPC Foundation

OPC Foundation

The aim of the OPC Foundation is to ensure compatibility between different subsystems in automation and process control engineering. Compatibility is achieved by creating and maintaining open specifications for the standardization of communication between subsystems of different manufacturers. This allows process data, alarms and messages as well as archived process data to be exchanged between subsystems regardless of their manufacturer.

The OPC Foundation has more than 300 members worldwide. These include the most important manufacturers of control systems, process instrumentation and process control systems worldwide. SIEMENS is a member of the OPC Foundation.

The declared goal of the OPC Foundation is vertical integration of information from the field level through to the enterprise level. Using the OPC standard, this integration is possible regardless of the system and manufacturer.

The most important standards of the OPC Foundation for OpenPCS 7

- OPC Unified Architecture
  - DataAccess
  - Historical Data Access
  - Alarms&Conditions
- OPC “Classic”:
  - Data Access Custom Interface
  - Data Access Automation Interface
  - Historical Data Access Specification
  - Historical Data Access Automation Interface
  - Alarms and Events Custom Interface
  - Alarm & Events Automation Interface

3.4 Components of OpenPCS 7

Overview

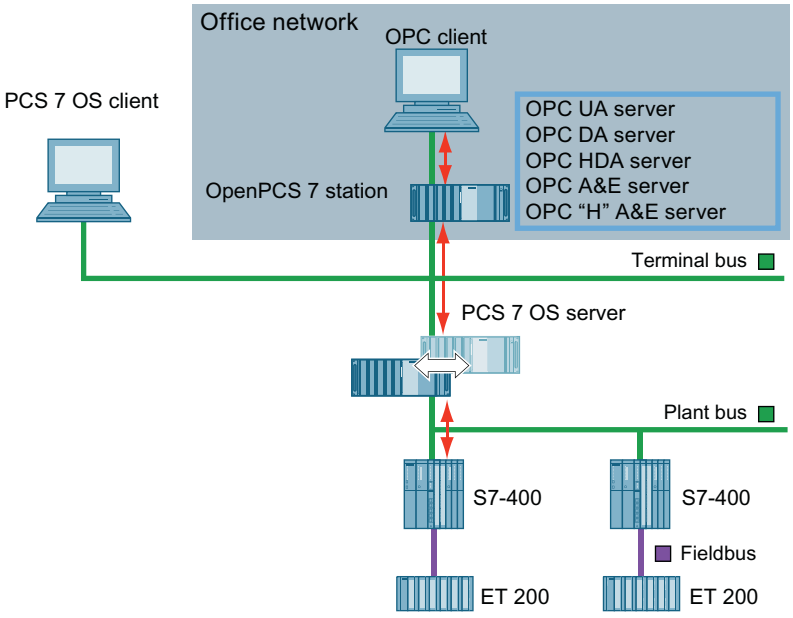
OpenPCS 7 provides the following components:

- OPC Unified Architecture (OPC UA server for OpenPCS 7)
- OPC "Classic":
  - OPC DA (Data Access server)
  - OPC HDA (Historical Data Access server)
  - OPC A&E (Alarms & Events server)
  - OPC "H" A&E (Historical Alarms & Events server)
- OLE DB

Note

For security reasons, we recommend converting the applications to OPC Unified Architecture (UA). The OPC UA server uses the TCP/IP protocol to exchange data. Certificates are exchanged for authorization between OPC UA server and OPC UA client. You can encrypt or sign data traffic in addition.

In the following graphic, you can see all the OPC components that will run on the OpenPCS 7 station. The figure also shows the PCS 7environment. OLE DB starts a direct database query on the PCS 7 OS server.



OPC UA (Unified Architecture server)

For access to the data management, messages and archive system according to the OPC specification OPC UA 1.02.

As OPC UA server, the OpenPCS 7 station offers other applications the following data:

- Current data from the OS data management (data access)
- Historical data from the OS measured value archives (historical access)
- OS messages (alarms & conditions)

Possible uses:

- An OPC UA client can register on the OPC UA server for ongoing changes of process values (subscriptions) or write values.
- An OPC UA client can access measured value archives.
- The OPC UA server can forward OS messages with all associated process values to registered clients on the production and corporate management level. OS messages can also be acknowledged here.
- Filter mechanisms and subscriptions ensure that only selected, changed data or selected messages are transmitted.

OPC DA (Data Access server)

For read and write access to process values according to the OPC specification OPC DA V1.00, V2.05a, V3.00.

As an OPC DA server, the OpenPCS 7 station provides other applications with current data from the OS data management. The OPC DA client can register for current changes or write values.

OPC HDA (Historical Data Access server)

For read access to archived process values according to the OPC specification OPC HDA V1.20.

As an OPC HDA server, the OpenPCS 7 station provides other applications with historical data from the OS archive system. The OPC client - for example a reporting tool - can specifically request the data required by specifying the start and end of a period. Various aggregate functions, such as variance, mean value, or integral, allow preprocessing by the OPC HDA server and contribute to a reduction in the communication load.

OPC A&E (Alarms & Events server)

For read access to messages, alarms and events according to the OPC specification OPC A&E V1.10.

As an OPC A&E server, the OpenPCS 7 station forwards OS messages with all the associated process values to the subscribers at the production and enterprise management level. OS messages can also be acknowledged here. Filter mechanisms and subscriptions ensure that only selected, changed data is transmitted.

OPC "H" A&E (Historical Alarms & Events server)

For read access to archived alarms and messages.

The standard OPC interface has been extended by Siemens. This enables the OpenPCS 7 station to send historical alarms and messages from the archive to subscribers on the production and corporate management level.

OLE-DB

The WinCC OLE DB Provider allows standardized and direct access to the archive data in the Microsoft SQL server database on the OS server.

This setup permits access to all OS archive data, along with the associated process values, message texts, and user texts.

3.5 Security concept for OPC UA connections in PCS 7

OPC Unified Architecture (UA) has an integrated security concept, which is essentially based on the following components:

- Authentication and authorization of the participating applications and users
- Assurance of integrity and confidentiality of messages exchanged between applications

You can find technical details on this in the documentation *OPC Unified Architecture – Specification – Part 2 (Release 1.02)*

Additional information

You can find information on the structure and configuration of plants with OPC UA connections in the following sections:

- Section "Plant configuration with firewall (Page 36)"
- Section "OPC Unified Architecture (OPC UA) (Page 42)"

3.6 OLE DB interface

OLE DB

OLE DB is an open standard for fast access to different databases. The connection between the OLE DB level and the database is established by a database provider.

The WinCC OLE DB Provider has been tested and approved for PCS 7. The WinCC OLE DB Provider allows access to the process value and message archives.

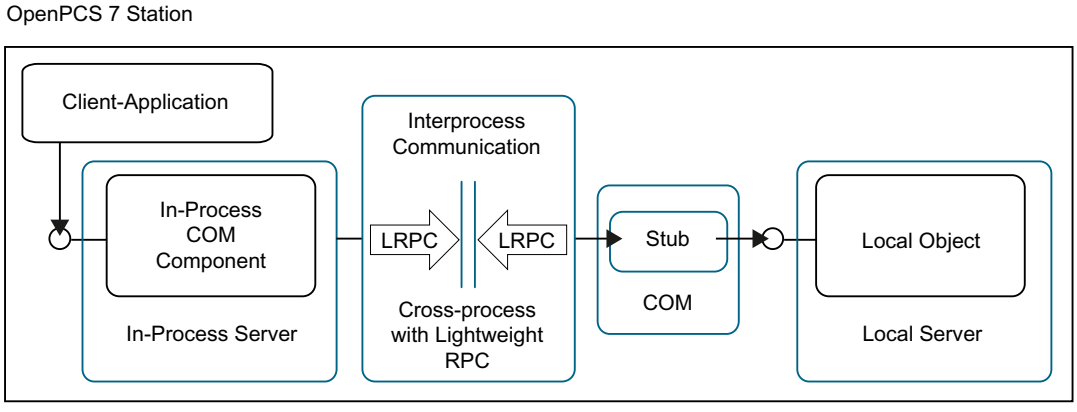
You can find additional information about this in the section "Basics of OLE DB (Page 88)".

3.7 COM / DCOM components of OpenPCS 7

COM / DCOM components used by OpenPCS 7

- OPC client runs on the OpenPCS 7 station

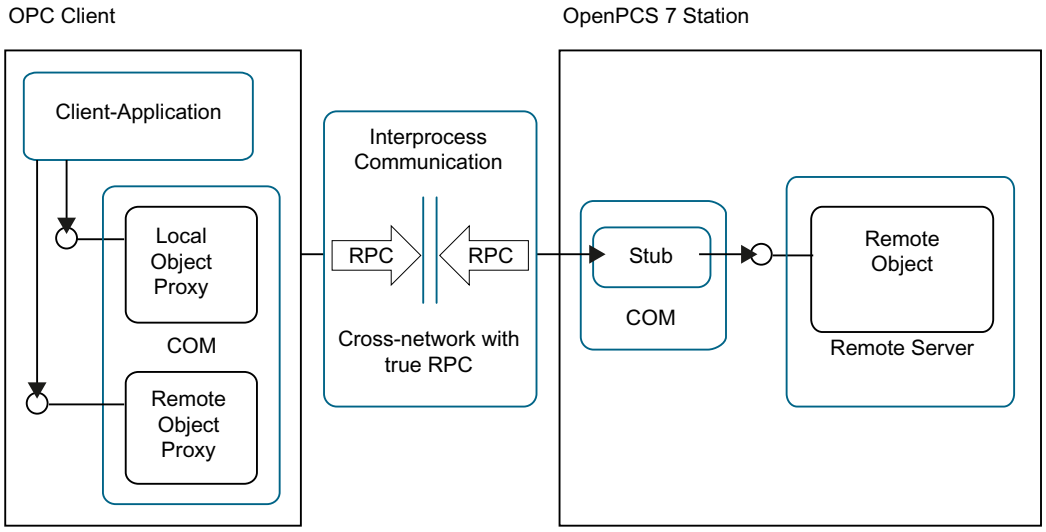
If the OPC client and OPC server process run on one computer, the OPC client accesses the local OPC server object using an "in-process server" component via LRPC and a stub object.



- OPC client runs on a separate computer

If the OPC client and the OPC server process run on separate computers, the OPC client accesses a local and a remote proxy object. These objects communicate via RPC and a stub object, with the OPC server object on a different computer.

Remote Procedure Call (RPC) is a technique used for communication between different processes. The processes typically run on different computers.


OPC Client
OpenPCS 7 Station
Client-Application
Local Object Proxy
COM
Remote Object Proxy
Interprocess Communication
RPC
RPC
Cross-network with true RPC
Stub
COM
Remote Object
Remote Server

Installation and licensing

4

4.1 Hardware requirements

Recommended configuration for PC stations

You can find recommendations for SIMATIC PCS 7 PC stations in the *PCS 7 Readme* (see "Preface (Page 9)").

4.2 Installing the OpenPCS 7 station

OpenPCS 7 stations in different configurations

Configuration	Requirements for installation	Intended use of the software	Program package in the PCS 7 Setup for the installation
OpenPCS 7 station without OS client	Installation requirements for the SIMATIC PCS 7 Workstation with an OS client configuration	This serves as software for a separate OpenPCS 7 station without OS client functionality.	• "OpenPCS 7" package
OpenPCS 7 station with OS client	Installation requirements for the SIMATIC PCS 7 Workstation with an OS client configuration	This serves to expand an existing OS client with the functionality of an OpenPCS 7 station.	• "OpenPCS 7 with OS-Client" package (corresponds to the software packages: "OpenPCS 7" + "OS-Client")
OpenPCS 7 station on the OS single station	Installation requirements for the OS single station	This serves to expand an existing OS single station with the functionality of an OpenPCS 7 station.	• "OpenPCS 7" package • Package "OS Single Station"
OpenPCS 7 station on the OS server	Installation requirements for the OS server	This serves to expand an existing OS server with the functionality of an OpenPCS 7 station.	• "OpenPCS 7" package • Package "OS-Server"

Note

If you only purchase the "OpenPCS 7 (OS-Client)" software package, you will only receive the license for OpenPCS 7 . The required licenses for OS clients must be obtained additionally and must be available on the computer at runtime.

Requirement

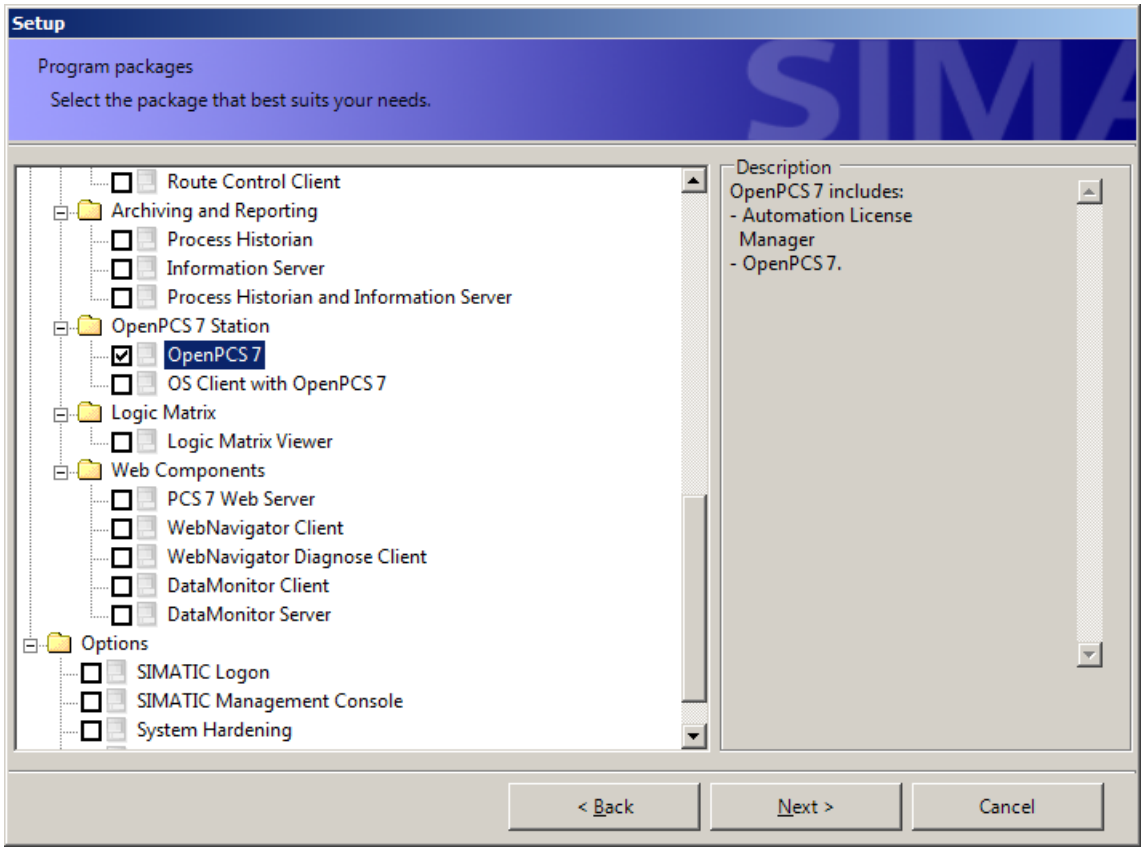
Message queuing must be installed.

Procedure

An example procedure for the standalone variant is explained below:

1. Insert the "SIMATIC Process Control System; PCS 7" DVD in the DVD drive.
2. Run the "SETUP.exe" installation program.
3. Select the setup language and click "Next".
4. Close all programs and click "Next".
5. Read the product instructions and click "Next".
6. Accept the license conditions and click "Next".
7. Select the setup type "Install" and click "Next".
8. Enter the user information and click "Next".
9. Select the package installation and click "Next".
10. Select the program package and click "Next".
    If, for example, you want to select the OpenPCS 7 station without OS client, click on OpenPCS 7".
11. Check the components selected for installation and click "Next".
12. Click the "Install" button.

The following picture shows the selection for the program package:



Additional information

- You can find additional information about installation requirements in the system documentation *Process Control System PCS 7; PCS 7 - PC Configuration*.

4.3 Licensing of the OpenPCS 7 station

General

Two licenses are available for OpenPCS 7. The licenses for the OpenPCS 7 station are license keys that must be transferred directly to the machine on which the OpenPCS 7 station runs using the Automation License Manager. A license is required for each OpenPCS 7 station. The licenses for the OpenPCS 7 station are single licenses. This means that a license server cannot be used for OpenPCS 7.

Note

PCS 7data can be accessed via OpenPCS 7 under the following conditions as regards license keys:

- The OpenPCS 7 station / OS require a license for a PCS 7 OS. This is applicable for all configurations of the OpenPCS 7 station.
- If you use the "OpenPCS 7 Station / OS" license key, the PCS 7 OS must be in process mode.
- If you use the "OpenPCS 7 station" license key, you can always access PCS 7data via OpenPCS 7.
- Access to the corresponding PCS 7data is only ensured when a license is available.

Licenses for OpenPCS 7

The following licenses are available for OpenPCS 7. The article numbers are available in the current PCS 7 catalog or in the online catalog.

License	Description
OpenPCS 7 station / OS	License for the software to expand an existing operator station with the OpenPCS 7 station functionality. This is a single license for one installation.
OpenPCS 7 station	License for one OpenPCS 7 software installation for a separate OpenPCS 7 station based on the hardware of the SIMATIC PCS 7 workstation (client version). This is a single license for one installation.

Note

OPC client - server connections

On a PC with the Microsoft Windows 7 operating system, Microsoft permits up to 20 connections between PCs. COM access between an OPC client and the OpenPCS 7 station does not count as a connection in this sense.

4.4 Licensing of the OS-internal OPC server

Licensing

The PCS 7 OS provides the following internal OPC servers on an OS server or an OS single station:

- SOPCSRVRWinCC.exe for OPC DA
- SOPCAESRVRWinCC.exe for OPC A&E
- SOPCHDASRVRWinCC.exe for OPC HDA

These internal OPC servers can be used with OpenPCS 7-licenses. This is only possible if the "OpenPCS 7 station / OS" or "OpenPCS 7 station" license is available on the OS station.

Note

If the internal OPC servers of an OS station are used, this represents additional load in terms of the OS station's performance.

PCS 7 Engineering

5

5.1 Configuring an OpenPCS 7 station

Introduction

The OpenPCS 7 station is configured as a SIMATIC PC station in the SIMATIC Manager. It includes the "SPOSA application" object. SPOSA is the acronym for Single Point Of System Access. The OPC client application can also be executed on the OpenPCS 7 station. During configuration, the following steps need to be taken:

- Insertion of a SIMATIC PC station
- Insert and configure the SPOSA application in the hardware configuration of the SIMATIC PC station

If you created the multiproject with the PCS 7wizard, you will already have inserted an OpenPCS 7 station if you selected the appropriate option. You can also insert an OpenPCS 7 station by expanding the project later with the "Expand Project" PCS 7wizard. The steps required to create an OpenPCS 7 station manually are explained in the following section.

Requirement

The PCS 7 project is open. You have created the OS servers and generated the server packages.

Procedure

1. Select the project into which you want to insert the OpenPCS 7 station, in the component view of the SIMATIC Manager.
2. Select the menu command **"Insert > Station > SIMATIC PC Station"**.
   A new SIMATIC PC station is inserted into the selected project.
3. Select the SIMATIC PC station, select the menu command **"Edit > Object Properties"** and enter the desired name for the PC station and the computer name.
4. Select the SIMATIC PC station in the component view and open the hardware configuration by double-clicking on the "Configuration" object in the detailed view.
   The SIMATIC PC station hardware configuration opens.
5. If the hardware catalog is not visible, select the menu command **"View > Catalog"**.
6. Select the SPOSA application from "SIMATIC PC Station > HMI" in the hardware catalog and drag it to the configuration table.
7. Save and compile the hardware configuration using the menu command **"Station > Save and Compile"**
8. Expand the newly created PC station and the SPOSA application in the SIMATIC Manager.

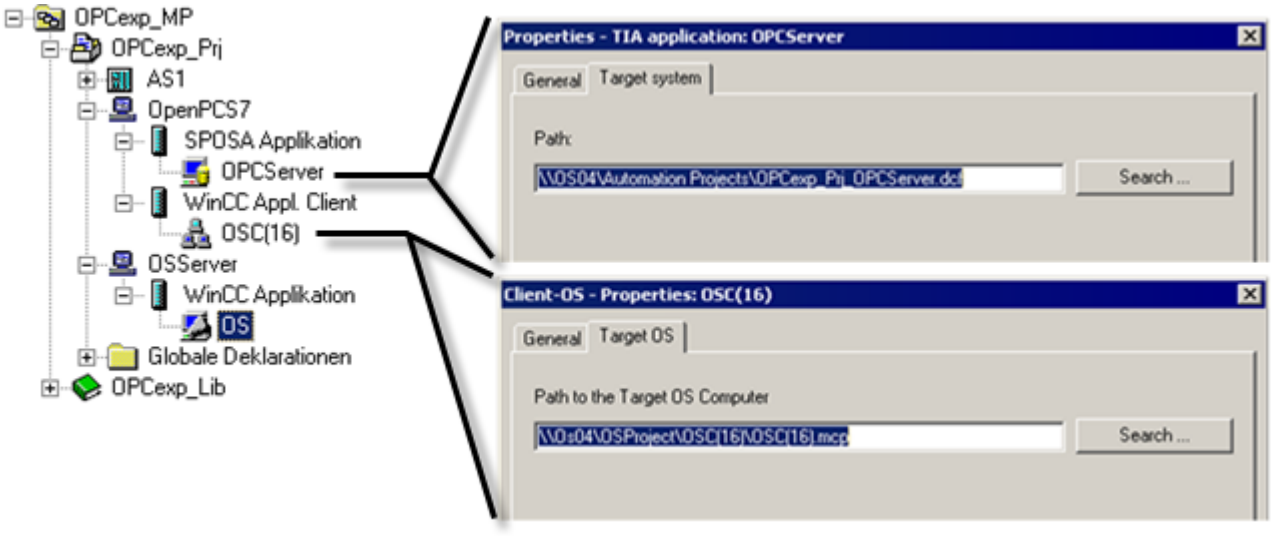
9. Open the object properties of the "OpenPCS7_Station(1)" object and enter the download path to the OpenPCS 7 station in the "PLC" tab.
10. Right-click on the "OpenPCS7_Station(1)" object to open the shortcut menu and select **"Assign OS Server..."**.
11. Assign the PCS 7 OS server packages to the SPOSA application.
12. Open the hardware configuration of the OpenPCS 7 station and select the menu command **"Station > Save and Compile"**.
13. Right-click on the "OpenPCS7_Station(1)" object to open the shortcut menu and select **"PLC > Download"**.

5.2 OpenPCS 7 and OS application on a shared PC station

Configuration

In the hardware configuration of the PC station, you configure a "SPOSA application" and a "WinCC application client". Configuration of an "WinCC application" is possible as well.

- You specify the path to the target system of the OpenPCS 7-computer in the object properties of the "SPOSA application".
- You configure the path to the target OS computer in the object properties of the "OS client application".



Note

In a PCS 7-version lower than V7.1, a SPOSA application and an OS client cannot be configured on the same SIMATIC PC station. In a PCS 7-version lower than V7.1, two PC stations with the same computer name but with different PC station names must be configured.

5.3 Configuring an OPC client

OPC client engineering within PCS 7

OPC client engineering is not part of the PCS 7 engineering. No PC station is inserted and configured in the SIMATIC Manager. In the SIMATIC Manager, only the OpenPCS 7 station is configured.

Note

If you want to use the SIEMENS OPC Automation Interfaces, either the OPC client must be executed on the OpenPCS 7 station or you will have to take manual steps to ensure that the required files exist and are registered on the OPC client. Refer to the section " Configuring an OPC client computer".

OPC client configuration outside PCS 7

An OPC client communicates with the OpenPCS 7 station via a network connection. We recommend that you install a firewall between the OPC client and the OpenPCS 7 station if the OPC client is on a non-secure network. So that communication between the OPC client and OpenPCS 7 station works correctly, you need to configure the firewall and configure the DCOM settings on the PC on which the OPC client is being executed. You can find the relevant settings in the documentation of the OPC client and the firewall.

Note

If you want to use an OPC client for OPC "Classic" on a PC without an OpenPCS 7-installation, you need to install the OPC proxy/stub components of the OPC Foundation. You can install these with the "OPC Core Components 3.00 Redistributable" setup of the OPC Foundation. In this case, you should also check the installation instructions of the OPC client vendor.

Additional information

- You can find more information about PCS 7 security concepts in the manual *Security Concept PCS 7*.

5.4 Loading the OpenPCS 7 station

Requirements

- The following components are installed on the PC station:
  - Operating system
  - OpenPCS 7 station software
- The terminal bus is configured on the OpenPCS 7 station.

- The OpenPCS 7 station is connected to the engineering station via the terminal bus.
- The protocol for the communication on the terminal bus is set to TCP/IP.
- The OpenPCS 7 station is configured in the PCS 7project.
- The user logged in on the ES can access the shared Windows folder.
  Default setting: "\\<OpenPCS 7 Station>\Automation"
- The PCS 7multiproject is open on the ES.

Procedure

1. Expand the PC station of the OpenPCS 7 station.
2. Expand the SPOSA application.
3. Right-click on the object below the SPOSA application.
4. Select " CPU > Download".
5. If no project has yet been loaded on the OpenPCS 7 station, the configuration is applied. Configuration changes only take effect after a restart of the OpenPCS 7 station.

5.5 Configuration support with the PCS 7 project wizard

The PCS 7 project wizard

The PCS 7project wizard supports you when creating a PCS 7project including an OpenPCS 7 station. In the following example, we will create the minimum configuration for a PCS 7project with an OpenPCS 7 station. We will only deal with the configuration steps on the ES that relate to the OpenPCS 7 station. AS / OS engineering and downloading of the project are not dealt with at this point.

Requirement

You have an engineering station with at least a PCS 7 V8.0 installation.

Example

1. Open SIMATIC Manager.
2. Start the PCS 7project wizard in the SIMATIC manager using the " File > ' New Project' Wizard" menu.
3. Click the "Next" button in the "Introduction" dialog.
4. Select the required CPU in the "Which CPU are you using in your project?" dialog and then click the "Next" button.
5. Click the " Preview >>>" button.
6. Select the "OpenPCS 7" option in the "Which objects are you still using?" dialog.
7. Click "Next".

8. Enter a directory name in the "Directory name" box in the "Where do you want to store the multiproject?" dialog.
9. Click the "Browse" button and set the path for the storage location.
10. Click the "Finish" button.
11. Click "OK" in the "Message Number Assignment" dialog.
12. Insert a new PC station in your project.
13. Enter the computer name in the object properties of the PC station.
14. Open the hardware configuration and insert an OS application.
15. Compile the OS.
16. In the shortcut menu of the OpenPCS 7 station, select the menu command **Assign OS Server...** .
17. In the dialog window "Assignment OS server for...", select the check box for OS server from which the OpenPCS 7 requires station data.
18. Click "OK".
19. Following this, you will need to complete the project with the AS and OS engineering and download the project.

Plant configurations

6

6.1 General configuration

General OpenPCS 7configuration

In process mode, the OpenPCS 7 station communicates with the automation systems via the operator station (OS server).

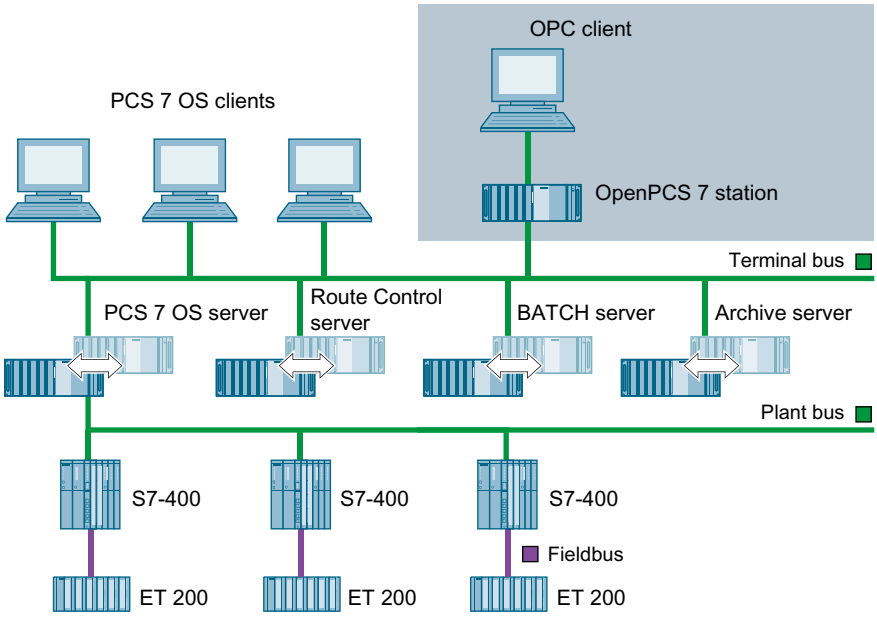
With the OpenPCS 7 station, you can access the data of redundant PCS 7 OS server pairs. If the PCS 7 OS master server fails, the redundant OS server is automatically connected for the next read job.

If the connection aborts during a read job, the OpenPCS 7 station also attempts to read the data from the redundant OS server.

The OpenPCS 7 station cannot directly access the data of the BATCH server and Route Control server. SIMATIC BATCH and SIMATIC Route Control use blocks in the automation system. These blocks have OS tags for operator control and monitoring at the OS level. These OS tags are available via OPC DA or OPC UA. SIMATIC BATCH and SIMATIC Route Control also use an OS server as a message server. These messages are available via OPC A&E or OPC UA (A&C).

The following data of the archive server can be provided to the OpenPCS 7 station:

- Process value archives via OPC UA or OPC HDA
- Message archives via OPC "H" A&E



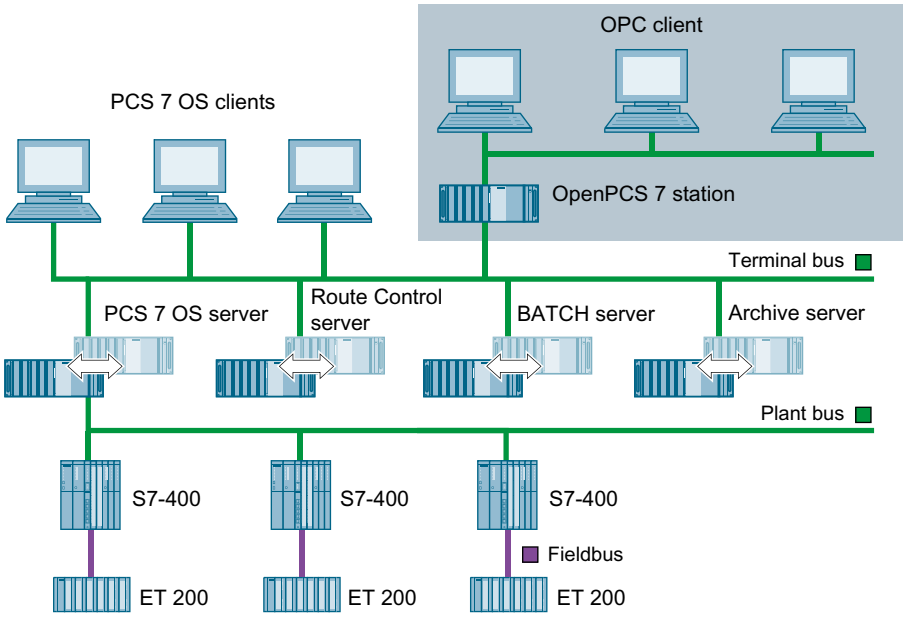
6.2 OpenPCS 7 without OS

OpenPCS 7 station without installed OS client

In this configuration, the OpenPCS 7 station is installed without an OS client. The OPC clients run on a separate PC. All OPC clients access the OpenPCS 7 station. The OpenPCS 7 station contains the following OPC server:

- OPC UA server
- OPC DA server
- OPC HDA server
- OPC A&E server

This configuration is intended for large plants.



Construct redundant connections for OpenPCS 7 stations in PCS 7

No redundant OpenPCS 7 station is implemented in PCS 7.

You can create redundancy of sorts by running several identical OpenPCS 7 stations. This requires that your OPC client has the following functionality:

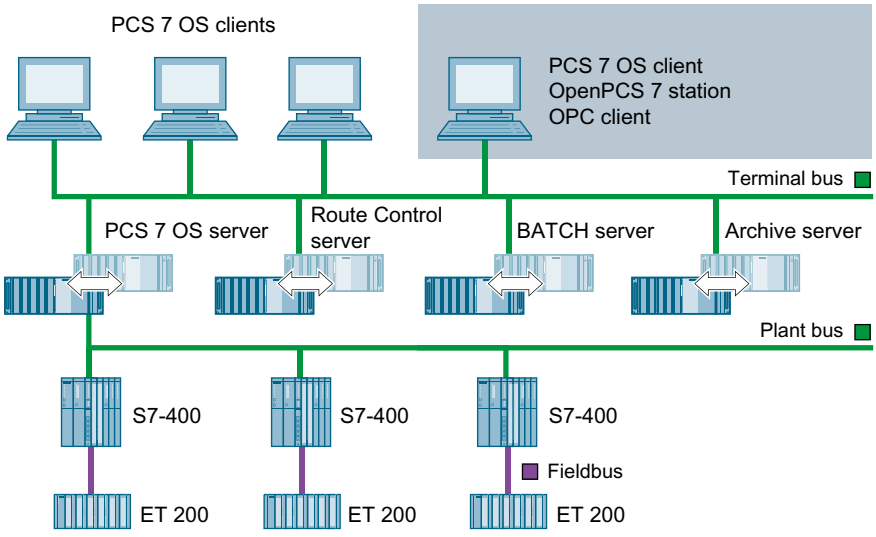
- Determination of the OpenPCS 7 station through which the necessary information is available.
- Detection of failure of an OpenPCS 7 station and failover to an available OpenPCS 7 station

6.3 OpenPCS 7 combined with an OS

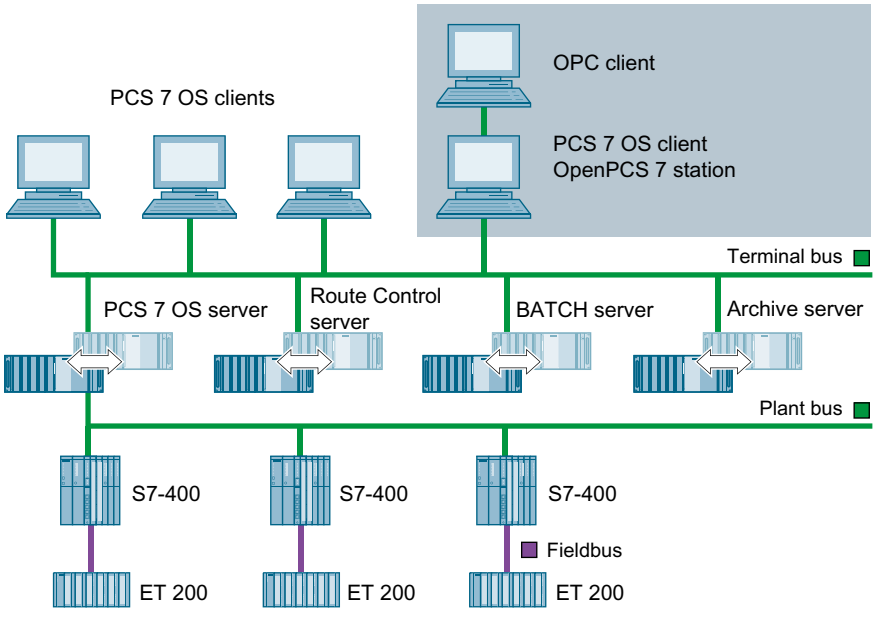
6.3.1 OpenPCS 7 combined with an OS client

OpenPCS 7 combined with an OS client

In this configuration, the OpenPCS 7 station is also installed on an OS client. The OPC client also runs on the OpenPCS 7 station. The OpenPCS 7 station contains the OPC server (OPC UA, OPC DA, OPC HDA and OPC A&E). This configuration is intended for small plants.



The OPC client can also run on a separate computer. This configuration is intended for small to medium-sized plants.



6.3.2 OpenPCS 7 combined with an OS server

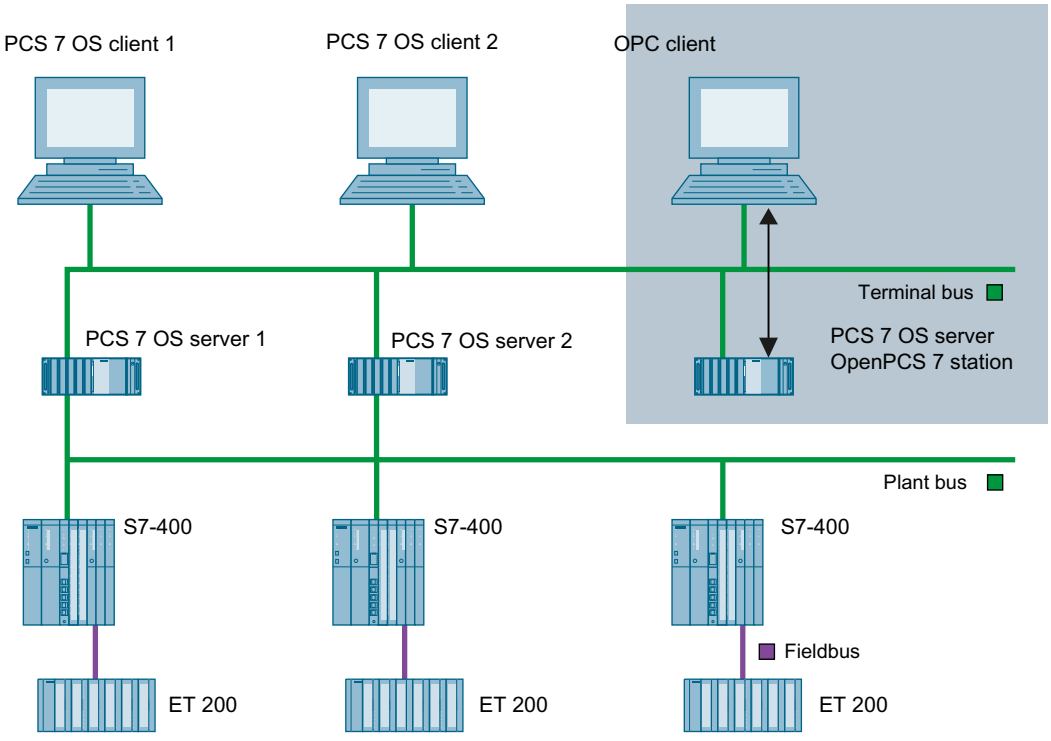
OpenPCS 7 on a server

In this configuration, the OpenPCS 7 station is also installed on an OS server. The OpenPCS 7 station provides data of one or more OS servers to OPC clients.

Note

Note that the OpenPCS 7 station affects the performance of the OS server. For large PCS 7 plants, we recommend the use of a separate OpenPCS 7 station.

Example configuration for the "PCS 7 OS server":

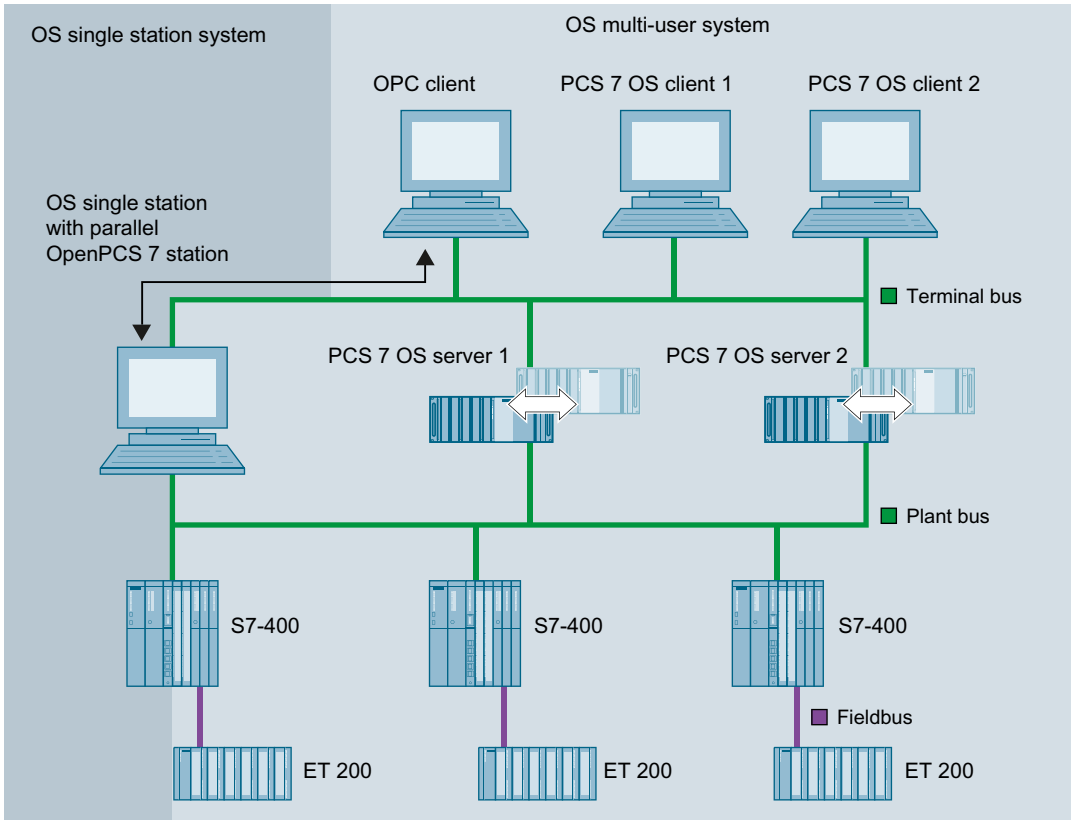


6.3.3 OpenPCS 7 combined with an OS single station system

OpenPCS 7 on an OS single station system

In this configuration, the OpenPCS 7 station is also installed on an OS single station system. The OpenPCS 7 station provides data of one or more OS servers to OPC clients.

Configuration:



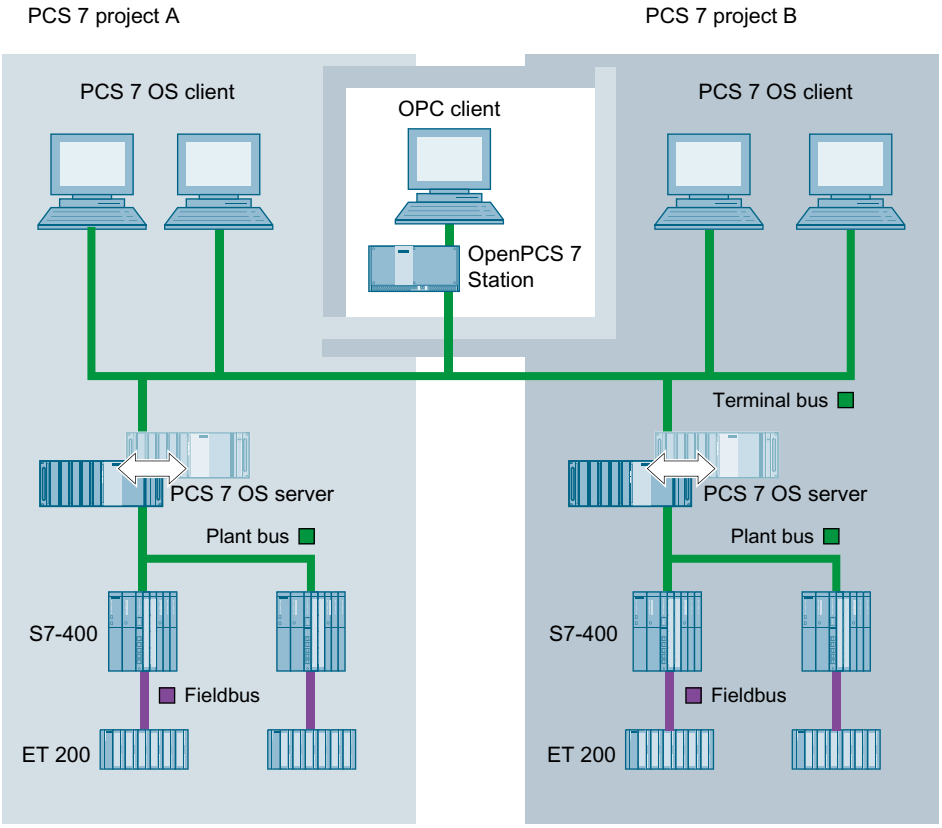
Note

An OS single-station system cannot be assigned to an OpenPCS 7 station. When OpenPCS 7 is used, it is not possible to provide data of the OS single station system to an OPC client.

6.4 OpenPCS 7 station for multiple PCS 7 projects

Example configuration

In this example configuration, an OpenPCS 7 station is used to make OS server data from two PCS 7 projects available. Multiple OS servers can also be used per PCS 7project.



This example configuration has been released for identical PCS 7 versions in the PCS 7projects A and B.

Note

Hybrid configurations with regard to PCS 7versions are disabled.

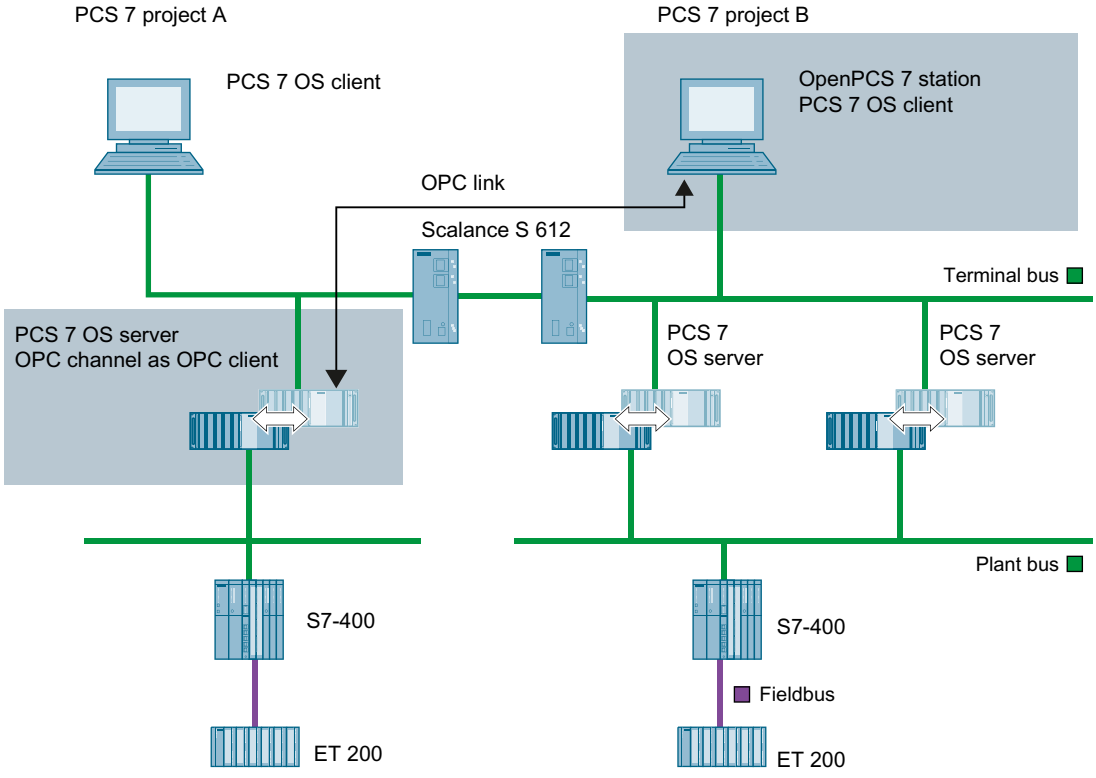
Configuration

A SIMATIC PCS 7 station with a SPOSA application is configured in every PCS 7 project. The SPOSA application is assigned to the OS server of the relevant PCS 7project. The identical path to the target system is configured and downloaded in both SPOSA applications.

6.5 DA linking of two PCS 7 projects using OpenPCS 7

Plant configuration of the OPC DA link

The PCS 7 OS servers can also be used as OPC clients. The "OPC" channel is the OPC client application of the PCS 7 OS. Information on the configuration of the OPC channel is available in the WinCC Information System in the section "WinCC Information System/Communication/ OPC Channel". In this configuration, the OpenPCS 7 station can make data from several OS servers available for Project A via OPC.



Application 1:

There are two PCS 7projects, Project A and Project B, in a plant network. For technical reasons, Project A needs to visualize data of Project B. The OPC channel of the PCS-7 OS can be used for this purpose.

Application 2:

If PCS 7Project A and PCS 7Project B use different PCS 7versions, but support the same OPC version. The OPC channel of the PCS 7 OS can be used for this purpose.

Application 3:

This configuration can also be used if Project B is a third-party product that provides an OPC server and not a PCS 7 project. This requires that both projects support the same OPC standard.

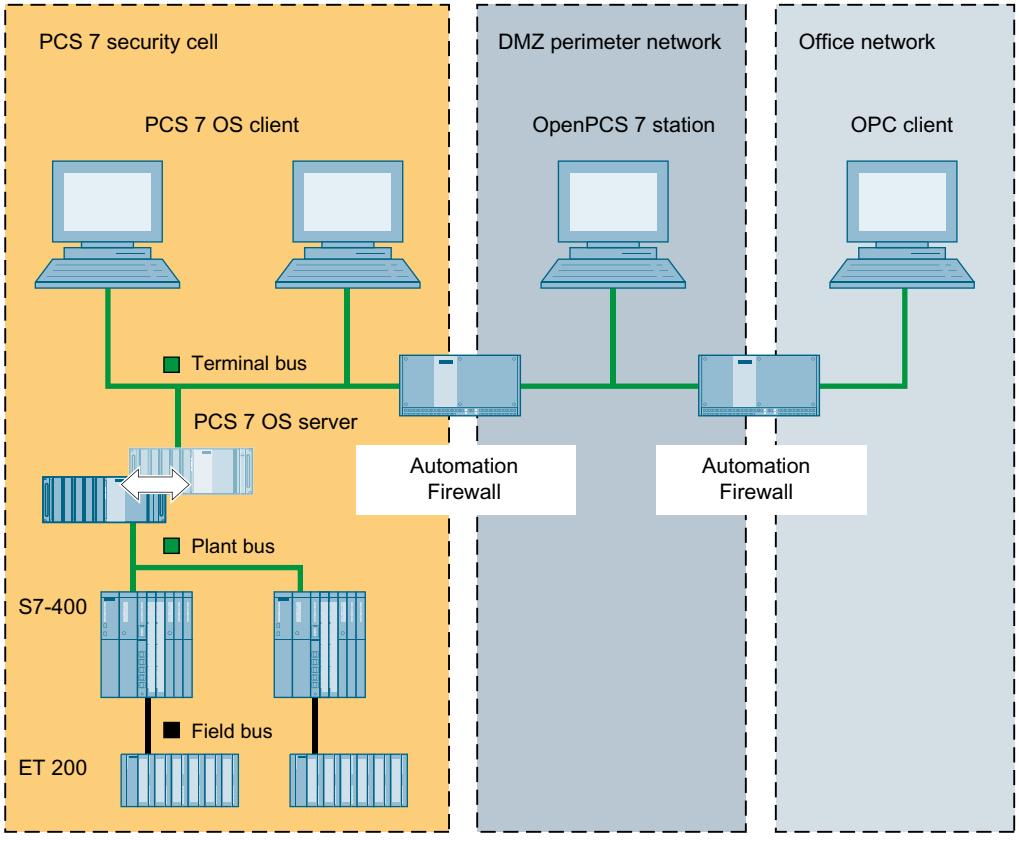
Note

The OPC channel can only be used for OPC DA.

6.6 Plant configuration with firewall

Plant configuration with OpenPCS 7 station and firewall

The following figure shows the schematic structure of a PCS 7 plant with an OpenPCS 7 station and firewall. The PCS 7 security cell is separated from the demilitarized zone (DMZ) by a Microsoft ISA server firewall. The OpenPCS 7 station is placed in the DMZ. The DMZ is also separated from the office network by a firewall. The OPC client that accesses the OpenPCS 7 station is located in the office network.



Additional information

- *Security concept of OPC UA:* Section "OPC Unified Architecture (OPC UA) (Page 42)"
- You can learn about the PCS 7 security concept in the *PCS 7 Security concept* manual
- Documentation *Process Control System PCS 7; PCS 7 - PC Configuration*

6.7 Setting the Windows firewall for OpenPCS 7

Windows Firewall settings

When the OpenPCS 7 station is outside the network (subnet) of the PCS 7 system, settings need to be made in the Windows Firewall on all OS servers which are to access to the OpenPCS 7 station.

Setting location

The following table shows where the settings must be made for the respective operating system:

Operating system	Setting location
• Windows 7 • Windows Server 2008 R2 • Windows Server 2012 R2	Inbound rules in the "Windows Firewall with Advanced Security" dialog

Settings for the rules

- The following rules must be adjusted depending on the type of connection:
  - The "CCEServer" rule for an OPC connection
  - The "SQL Server xxxx" and "SQL Browser" rules for an OLE DB connection
    Note for SQL Server:
    The exact name depends on the version of SQL Server that is approved for your PCS 7 components (see *Process Control System PCS 7; PCS 7 Readme* documentation).

  Expand the scope for corresponding rule(s) mentioned above from which access is to be allowed to the subnet or IP address of the OpenPCS 7 station on the OS server.
- Make sure that bi-directional access is ensured between the OpenPCS 7 station and OS server. Check the availability of both ends with the "ping" command. Adjust the firewall settings if necessary.

Creating a firewall for the OPC client

The Windows firewall blocks inbound RPC connections for OPC callback calls on the OPC client. Create a rule that allows OPC callback calls on the OPC client:

1. In the Windows "Start" menu, select the menu command **Control Panel >Windows firewall > Advanced settings**.
   The "Windows Firewall with Advanced Security" dialog opens.
2. Select the "Inbound Rules" path in the tree structure.
3. Select the menu command **Action > New Rule...**.
   The "New Inbound Rule Wizard" dialog opens.
4. In the "Rule Type" tab, select the "Custom" option.
5. In the "Program" tab, select the "This program path" option.
   Enter the folder and program name of the OPC client application in the input box.
6. In the "Protocols and Ports" step, select the following settings:
   - Protocol type: **TCP**
   - Local port: **RPC Dynamic Ports**
7. Recommendation:
   - In the "Scope" step, select the option "These IP addresses" as the setting for the list "Which local IP addresses does this rule apply to?".
   - In the list, insert the IP addresses of the OpenPCS 7 station.
8. In the following steps, select any necessary plant-specific settings.
9. Enter a name for the profile in the input box in the "Name" step.
10. Click "Finish".

6.8 Users and passwords in a workgroup

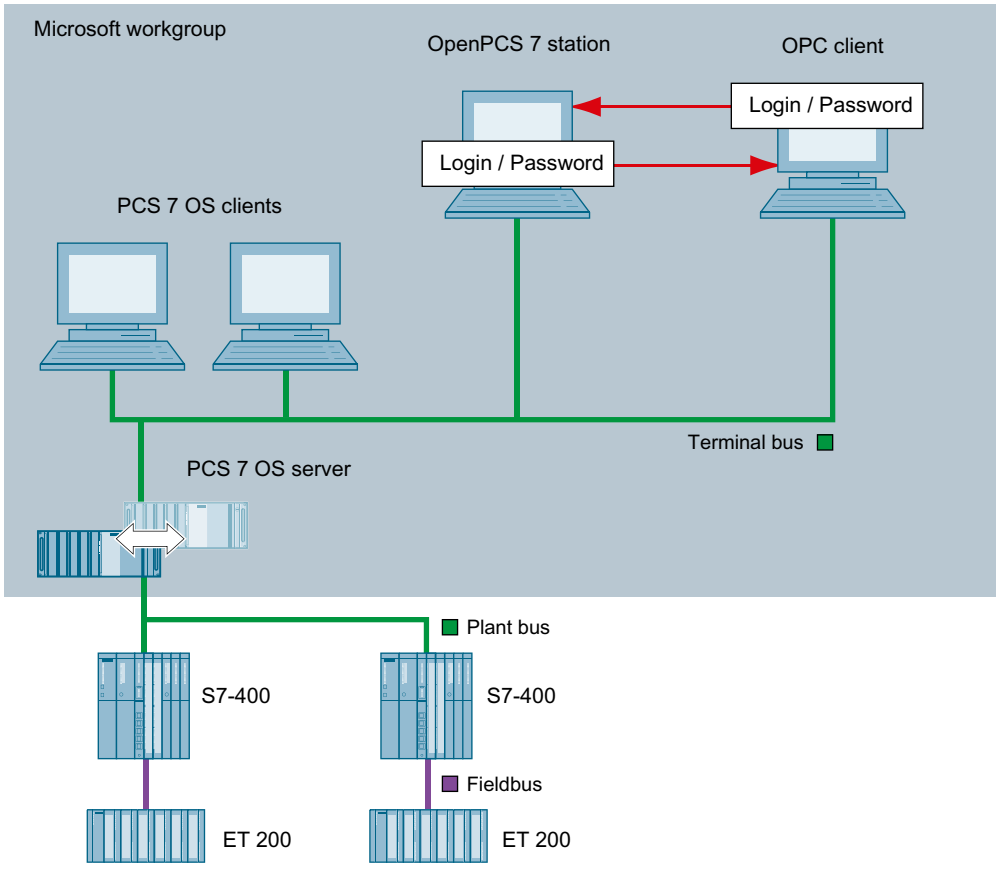
Users and passwords

Users and passwords of logged-on users must be identical on the OpenPCS 7 station and on the OPC client. Otherwise, correct access by the OPC client to the OpenPCS 7 station cannot be guaranteed.

The logged-on user should have at least power user privileges in the operating system.

Example

If the user with the user name "User1" and the password "xyz" is logged on to the OpenPCS 7 station, an identical user must be created on the OPC client. If the user with the user name "User2" and the password "abc" is logged on to the OPC client, an identical user must be created on the OpenPCS 7 station.



OpenPCS 7 interface

7

7.1 Access options

Access options

The following table lists the access options via OPC to the following PC stations in PCS 7:

- PCS 7 OS server
- Maintenance Station
- Process Historian

Access to data from	Access via interface type	Data type	Type of access
OS server or Maintenance Station	OPC UA	Process tags	Read/write
	OPC UA	Alarms and messages	Read
	OPC UA	Archive tags in the measured value archive (TagLogging)	Read
	OPC DA	Tags in process mode	Read/write
	OPC HDA	Archive tags in the measured value archive (TagLogging)	Read
	OPC A&E	Alarms and messages	Read/acknowledge
	OPC "H" A&E	Alarms and messages from the message archive (AlarmLogging)	Read
Process Historian	OPC UA	Archive tags in the measured value archive (TagLogging)	Read
	OPC HDA	Archive tags in the measured value archive (TagLogging)	Read
	OPC "H" A&E	Alarms and messages from the message archive (AlarmLogging)	Read

7.2 Data transmission

Type of data transfer of OPC DA, OPC HDA and OPC A&E

This section explains the various types of data transfer with OPC "Classic".

Event-driven data transfer

With event-driven data transfer, the OPC registers the required data for updating on the OPC server. The OPC server sends the tags to the OPC client when they change.

Asynchronous data transfer

When using asynchronous data transfer, the called method is not completely processed on the OPC server before the OPC client receives data from the OPC server. The OPC server returns the results of the asynchronous operations using events. In this way, it is possible for the OPC server to supply data to the OPC client automatically if the data has changed.

Synchronous data transmission

When calling synchronous methods, the job is processed completely on the OPC server before the information is transferred to the OPC client. The OPC server provides synchronous write and read methods.

Note

You can find the available asynchronous and synchronous read and write methods in the following sections:

- **OPC Data Access (OPC DA)** - (SIEMENS OPC DA Automation Interface 2.0)
- **OPC Historical Data Access (OPC HDA)** - (SIEMENS OPC HDA Automation Interface 1.0)
- **OPC Alarms and Events (OPC A&E)** - (SIEMENS OPC A&E Automation Interface 1.0)

7.3 OPC Unified Architecture (OPC UA)

OPC and OPC UA are standardized and vendor-independent interfaces and information models of the OPC Foundation for reliable communication in automation technology as well as in other fields of application.

Overview

You can find information about the following topics in the sections below:

- How the OPC UA server works (Page 43)
- Configuring the OPC UA server (Page 44)
- Security concept (Page 45)
- Supported OPC UA services and profiles (Page 47)
- Namespace of the OPC UA server (Page 48)
- OPC UA Data Access (Page 51)
- OPC UA Log Access (Page 51)
- OPC UA alarm & conditions (Page 52)
- Mapping the WinCC message system to OPC UA (Page 53)

Introduction to the OPC Unified Architecture interface

OPC UA (Unified Architecture) is the successor technology to OPC "Classic" and has the following main features:

- Platform-independent
- Vendor-independent
- Use of standardized protocols as the communication medium (e.g. tcp, http)
- Integrated security concept (authentication and authorization, encryption and data integrity by signing)
- High-performance information model and provision of all data (DA/HDA/A&C) in a single address space

Detailed information on the OPC UA specifications can be found on the OPC Foundation website on the Internet (http://www.opcfoundation.org).

7.3.1 How the OPC UA server works

Operating principle

The OPC UA server for OpenPCS 7 provides the following data:

- Process values
- Values from measured value archives
- OS messages

Supported specifications

The OPC UA server supports the OPC UA specification 1.02.
You can find more information on UA functionality in the section "Supported OPC UA services and profiles (Page 47)".

Installation

The OPC UA server is installed with OpenPCS 7 as a Windows service. The service is automatically started when the computer is booted. The OPC UA server can only be used when the OpenPCS 7 station has been configured and loaded.

Configuration/accessibility

The OPC UA server is pre-configured during installation.

OPC UA server only supports the OPC UA binary protocol ("UA-TCP UA-SC UA binary" communication profile) for the communication. The port number used can be configured.

You can find more information on this subject in the following sections:

- Section "Configuring the OPC UA server (Page 44)"
- Section "Supported OPC UA services and profiles (Page 47)"

Discovery Server

The "Discovery Server" is provided by the OPC Foundation. With the installation of OpenPCS 7, the "Discovery Server" is installed by default as Windows service. The "Discovery Server" provides OPC UA clients with information about OPC UA servers that are registered on the "Discovery Server".

The OPC UA server for OpenPCS 7 registers itself at startup depending on its configuration with no, one or multiple configured and available "Discovery Servers". Registration is repeated cyclically.
Default setting: Registration on local Discovery Server

7.3.2 Configuring the OPC UA server

NOTICE
Changing the server configuration Changing the server configuration incorrectly could cause malfunctions or security gaps! The server configuration should therefore be performed only by qualified personnel.

"OPCUAServerOpenPCS7.xml" configuration file

You can find the configuration file in the installation folder of OpenPCS 7. The most important settings in the configuration file of the OPC UA server are:

- Port number of the OPC UA server
- Security settings

Note
Applying a change to the server configuration

Changes to the server configuration do not become active until the OPC UA server is restarted.

Changing the port number of the OPC UA server

The OPC UA server is accessible via the following URL: "opc.tcp://[HostName]:[Port]"

Parameter	Description
HostName	Placeholder for the computer name. Used automatically.
Port	TCP/IP - Port number Do not use a port number that is already used by another application. Default setting: 4863

Example for the setting in the configuration file of the OPC UA server:

```
<BaseAddresses>
<ua:String>opc.tcp://[HostName]:4863</ua:String>
<BaseAddresses>
```

Making the security settings

You can find information about the security settings in the section "Configuration of security mechanisms (Page 46)".

7.3.3 Security concept

The security concept of OPC UA is essentially based on the following:

- Authentication and authorization of the participating applications and users
- Assurance of integrity and confidentiality of messages exchanged between applications

For technical details on the configuration, refer to the following documentation:
"OPC UA Specification, Part 2"

OPC UA applications use certificates for mutual authentication.

The stores of the certificates used are set in the configuration file of the OPC UA server.

Access to the stores must be restricted to access by:

- The server itself (default setting)
- The system administrator (default setting)

Server certificate

The server certificate (ApplicationCertificate) and its private key are stored in the certificate store during the installation.

Default:

- Stores of the certificates used: [ApplicationPath]\PKI\CA\certs
- Stores of the private keys: [ApplicationPath]\PKI\CA\private

Note

OPC UA Server certificates and keys

After uninstallation of OpenPCS 7, the OPC UA Server certificates (public and private key) and trusted client certificates (public key) are not deleted.

Trusted client certificates

A client is trusted if one of the following certificates is available in the certificate store for the trusted certificates of the OPC UA server.

- A valid self-signed certificate of the client.
- A valid certificate issued by a trusted certificate authority. Only the certificate of the certification authority needs to be present. The instance certificate of the client does not need to be in the certificate store of trusted certificates.

Default:

Store of the trusted certificates: [ApplicationPath]\PKI\CA\certs

Certificate from a certification authority

The certificates from certificate authorities that are required for the verification of a client certificate chain are stored in the certificate store of the certification authorities.

Rejected client certificates

If the certificate used by a UA client to access the OPC UA server is not trusted, the OPC UA server rejects the secure communication. The OPC UA server copies the certificate of the UA client to the folder for rejected certificates.

Default:

Store of the rejected certificates: [ApplicationPath]\PKI\CA\rejected

7.3.3.1 Configuration of security mechanisms

OPC UA provides the following security mechanisms even on the communication level:

- Authentication of UA applications
- Confidentiality of exchanged messages
- Integrity of exchanged messages

The applied security mechanisms are defined by standardized security policies. You can find information on the requirements (e.g. algorithms for encryption, signing) in the following documents:

- *OPC UA Specification; Part 2*
- *OPC UA Specification; Part 4*
- *OPC UA Specification; Part 7*

You set the security policies supported by the OPC UA server in the configuration file of the OPC UA server:

- SecuredApplication
- ServerConfiguration

User identity

The OPC UA server supports the authentication of the user of client applications.

In the OPC UA server configuration file, you can make the settings for authenticating users below the "ServerConfiguration":

UserTokenPolicy "UserName".

7.3.4 Supported OPC UA services and profiles

OPC UA services

The following table summarizes the functionality supported by the OPC UA server:

OPC UA service sets	Services	Comment
Discovery Service Set	FindServers GetEndpoints	-
Secure Channel Service Session Service Set	All	-
View Service Set	Browse BrowseNext RegisterNodes UnregisterNodes	• Detection of mapped OS data: Process values and archived data
Attribute Service Set	Read	• OS tags only
	Write	• OS tags only
	HistoryRead	• Only archived tags
Subscription Service Set	CreateSubscription SetPublishingMode Publish RePublish DeleteSubscription	-
MonitoredItem Service Set	CreateMonitoredItems SetMonitoringMode DeleteMonitoredItems	• "Value" attribute of OS tags only • EventNotifier upon access to OS messages
Method Service Set	Call	• Acknowledge • ConditionRefresh

OPC UA profile and Conformance Units

The OPC UA server supports the following OPC UA profiles 1.02 without restrictions:

- 6.5.3 Base Server Behavior Facet
- 6.5.12 Standard Event Subscription Server Facet
- 6.5.14 A & C Base Condition Server Facet

- 6.5.24 Method Server Facet
- 6.5.30 Historical Raw Data Server Facet
- 6.5.107 UA TCP UA SC UA Binary
- 6.5.123 SecurityPolicy - None
- 6.5.124 SecurityPolicy - Basic128Rsa15
- 6.5.125 SecurityPolicy - Basic256

The OPC UA server supports the OPC UA profiles shown in the following table with restrictions:

Profile	"Group"	Not supported "Conformance Unit"
6.5.8 Standard DataChange Subscription Server Facet	Monitored Item Services	• ModifyMonitoredItems • DeadBand Filter • Monitor MinQueueSize_02
6.5.9 Enhanced DataChange Subscription Server Facet	Monitored Item Services	• Monitor MinQueueSize_05
6.5.25 Core Server Facet	Attribute Services	• Attribute Write Index
6.5.26 Data Access Server Facet	Data Access	• Data Access Analog • Data Access Multistate • Data Access PercentDeadBand • Data Access Semantic Changes • Data Access Two State
6.5.35 Standard UA Server	Attribute Services	• Attribute Write StatusCode & TimeStamp
6.5.47 Standard UA Server Profile	Attribute Services	• Attribute Write StatusCode & Timestamp

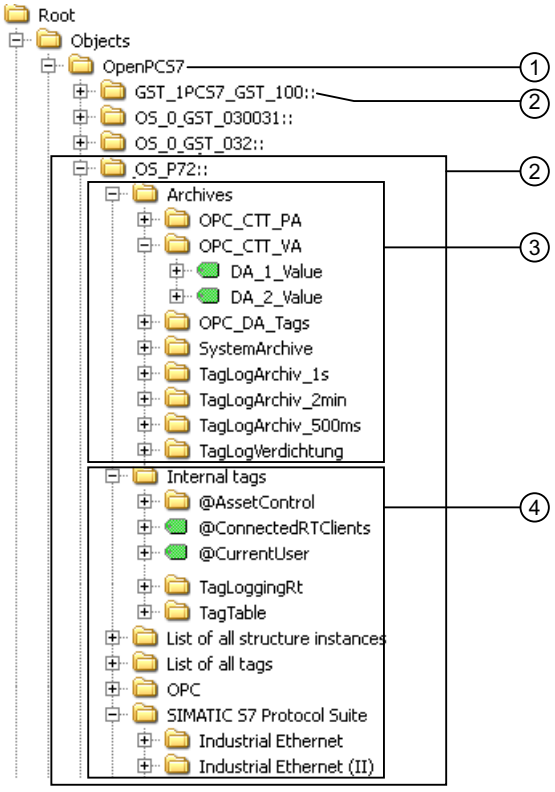
7.3.5 Namespace of the OPC UA server

The OPC UA server for OpenPCS 7 provides OPC UA clients with a hierarchical namespace and offers access to the following data of the operator stations of PCS 7 in process mode:

- Process values (OS tags and OS tag groups)
- Tag archive including archived tags
- OS messages

The namespace of the OPC UA server is attached under the "Objects" default folder.

The following figure shows the namespace of the OPC UA server of an OpenPCS 7 station with several configured PCS 7 OS servers:



1 Start node of the specific namespace of the OpenPCS 7 station.

2 Shows the configured PCS 7 OS server.

3 Shows the tag archives.

4 Shows the tags with a structure corresponding to the structure of the tags in the OS.

Mapping of the PCS 7 OS server

Each configured PCS 7 OS is mapped by a UA object of the "FolderType". The name of this folder corresponds to the server prefix of the OS.

Mapping of the OS tags

Tag groups, communication drivers and connections are mapped by OPC UA objects of the "FolderType". Each of these folders has references of the "Organizes" type to the subordinate objects and tags.

Internal and external tags are mapped with OPC UA tags of the type "DataItemType". If a tag is archived in addition, the mapped OPC UA tag also has a reference of the "HasHistoricalConfiguration" type to an archive configuration. The "Historizing" and "AccessLevel" attributes are respectively set.

The following table shows the most important attributes of the OPC UA tags that represent an OS tag. You can find the complete list of attributes in the "OPC UA Part 3 - Address Space Model 1.02 Specification" document under "§5.6":

Attribute	Description	Comment
NodeId	Unique identifier of the OS tag	-
BrowseName	Name of the OS tag	-
DisplayName	Name of the OS tag	-
Value	Tag value and status	-
DataType	OPC UA data type that corresponds to the OS tag type, for example: • Int32; signed 32 bit value • UInt32; unsigned 32 bit value	-
AccessLevel	"CurrentRead" / "CurrentWrite" "HistoryRead"	depending on configuration of the OS tag
ValueRank	Always "Scalar"	-

Display of the logging tags

Process values and compressed logs are displayed by OPC UA objects of the "FolderType" type. Each of these folders has references of the "Organizes" type to the related logging tags.

Logging tags from process value or compressed logs are displayed by OPC UA tags of the "BaseDateVariableType" type. A logging tag always has a reference of the "HasHistoricalConfiguration" type for a log configuration.

The following table shows the most important attributes of the OPC UA tags that represent a WinCC logging tag. You can find the complete list of attributes in the "OPC UA Part 3 - Address Space Model 1.02 Specification" document under "§5.6":

Attribute	Description	Comment
NodeId	Unique designation of a logging tag	-
BrowseName	Name of the archive tag	-
DisplayName	Name of the archive tag	-
Description	Node description	-
Value	Not available	For a logging tag, this attribute cannot be read nor changed.
DataType	OPC UA data type that corresponds to the tag type in the OS, for example: • Double; 64-bit floating point number • UInt32; unsigned 32 bit value	-
AccessLevel	"HistoryRead"	-
ValueRank	Always "Scalar"	-

Access to OS messages

OPC UA clients can receive OS messages if you register the following node with the "EventNotifier" attribute in a subscription:

- The start node of the OpenPCS 7 namespace ("OpenPCS7") allows you to receive the messages from all the configured operator stations of PCS 7.
- Each node of a PCS 7 operator station allows you to receive the messages of this PCS 7 operator station.

7.3.6 OPC UA Data Access

The OS tags are mapped to OPC UA tags of the "DataItemType" type. Other DataAccess tag types are not supported.

The OPC UA server supports the types of access:

- Read access
  Access to OPC UA tag attributes (e.g., "DataType" or "AccessLevel")
- Write access
  Access to the "Value" attribute of a tag (if permitted)
- Client logon
  Access to ongoing changes of the "Value" attribute of a tag (subscription/monitored item)

7.3.7 OPC UA Log Access

"OPC Historical Access" enables access to archives and includes the "Historical Data" and "Historical Events" services. With OpenPCS 7, the OPC UA server exclusively supports the "Historical Data" service for accessing archives.

The OPC UA server offers the OPC clients access to the raw data of tag archives via the following service:

- HistoryRead (READRAW)

Note

Modification of archived data is not desired in a process control system. Therefore, only read access to archived data is enabled for OPC UA in PCS 7.

You can read the values of archive tags in tag archives with an OPC UA client. Depending on the configuration of the tag archive, the archive tag can contain either raw data or process values already processed by the OS.

Characteristics of archive tags

A process tag in a PCS 7 OS can be located in multiple tag archives. In this case the process tag is linked to one of the corresponding archive tags.

Properties of archive configurations

The following table shows the properties of an OPC UA tag configuration of the "HistoricalConfigurationType". The "Description" property maps the archive tag comment configured in WinCC. You can find the complete list of properties in the "OPC UA Part 11 - Historical Access 1.02 Specification" document under "5.2.2":

Property	Description / Value	Comment
Definition	WinCC process tag name	For a process value archive
Stepped	True	-

The following optional properties are not supported:

- MaxTimeInterval
- MinTimeInterval
- ExceptionDeviation
- ExceptionDeviationFormat

7.3.8 OPC UA alarm & conditions

The OPC UA server for OpenPCS 7 provides access to OS messages.
The OPC UA clients can be notified about OS messages using subscriptions and monitored event items. The notifier node event to be used for this is either the start node of the OpenPCS 7 namespace ("OpenPCS 7") or one of the OS nodes.

The client can filter the messages and define the list of message attributes returned.

The OPC UA server support the specification "OPC UA Alarms & Conditions 1.02".
In the "Mapping the WinCC message system to OPC UA (Page 53)" section, you can find an overview of the specific information on the topics:

- Mapping of the WinCC message system to OPC UA for OpenPCS 7
- Attributes supported by the OPC UA server

For more information, refer to the OPC UA 1.02 Specification, particularly "Part 9: Alarms and Conditions".

7.3.8.1 Supported event methods

Acknowledgment

The "Acknowledge" method of the default OPC UA type "AcknowledgeableConditionType" is used to acknowledge a message.

Only messages of the "WinCCAlarmConditionType" can be acknowledged.

ConditionRefresh

The "ConditionRefresh" method of the default OPC UA type "ConditionType" is used to detect pending messages.

Additional information

For more information, refer to the OPC UA Specification, "Part 9: Alarms & Conditions".

7.3.8.2 Filters

The OPC UA client can define a filter for monitored event items.

The following operators are not supported by the OPC UA server:

- FilterOperator_Cast
- FilterOperator_BitwiseAnd
- FilterOperator_BitwiseOr
- FilterOperator_RelatedTo
- FilterOperator_InView

7.3.9 Mapping the WinCC message system to OPC UA

7.3.9.1 OPC UA event types

WinCC messages are mapped to the following OPC UA event types:

OPC UA event types	Description
WinCCEventType	This type is based on BaseEventType and maps "simple" messages, i.e. WinCC messages, with the following acknowledgment policy: • "Message without status went out" is activated. • "Acknowledgment came in" is not activated. Examples of this type of message are starting and stopping motors.
WinCCAlarmCondition-Type	This type is based on AlarmConditionType and maps all messages which cannot be mapped to WinCCEventType, for example, acknowledgeable messages and messages with the "incoming" and "outgoing" status. The event is linked to a condition, e.g. limit violation of a tag, for a WinC-CAlarmConditionType message. With WinCC, a message is generated as soon as such a limit violation occurs. This message corresponds to an alarm condition in OPC UA.

The event types add WinCC-specific message attributes to the basic type. These message attributes are mapped 1:1 as UA properties and are described in more detail below.

The OPC UA server can pass WinCC event notifications to OPC UA clients (about subscriptions and monitored event items), but does not have condition instances in its namespace.

7.3.9.2 WinCC attributes

Overview

The following table contains the attributes of the OS message system that are relevant and configurable for OPC. These attributes are mapped 1:1 as UA Event Properties .

WinCC message attribute	Meaning	Data type
CLASSNAME	Name of message class	String
TYPENAME	Name of message type	String
FORECOLOR	Foreground color for incoming, outgoing and acknowledged messages	Int32
BACKCOLOR	Background color for incoming, outgoing and acknowledged messages	Int32
FLASHCOLOR	Flash color	Int32
FLAGS	Indicates mandatory message acknowledgment	Int32
TEXT01…TEXT10	Content of user text block #1....#10	String
PROCESSVALUE01…PROCESSVAL-UE10	Content of process value block #1....#10	
STATETEXT	Status message	String
INFOTEXT	Information text for the message	String
LOOPINALARM	Indicates whether LoopInAlarm was configured	Int32
CLASSID	Message class ID	Int32
TYPEID	Message type ID	Int32
MODIFYSTATE	Value of message status tag	Int32
AGNR	Outputs the number of the automation system that generated the message	Int16
CPUNR	Outputs the number of the CPU that generated the message	Int16
DURATION	Outputs the time period between the incoming state, outgoing state and acknowledgment of a message	Int32
COUNTER	Number of messages after the start of runtime	Int32
QUITSTATETEXT	Indicates whether the message has been acknowledged	String
QUITCOUNT	Number of open, unacknowledged messages	Int32
PARAMETER	Configuration parameter of the message	Int32
BLOCKINFO	Current content of the message block	String
ALARMCOUNT	Number of pending messages	Int32
LOCKCOUNT	Number of locked messages	Int32
PRIORITY	Priority of the message	Int32
APPLICATION	Outputs the application which triggered the message	String
COMPUTER	Outputs the name of the computer which processed the message	String
USER	Outputs the name of the user who processed the message	String
COMMENT	Message comment	String
HIDDEN-COUNT	Number of hidden messages	Int32
OS-HIDDEN	Indicates that the message is hidden	Boolean

WinCC message attribute	Meaning	Data type
OS_EVENTID	Message number	Int32
BIG_COUNTER	Message counter	Int64

7.3.9.3 Message class and message type

The WinCC message system informs the user of disturbances and operating conditions in the process. A WinCC message always belongs to a specific message class and message type, which are specified in the CLASSID, TYPEID, CLASSNAME, TYPENAME properties of the corresponding UA events.

7.3.9.4 Priority

When configuring alarms in the message system, you can configure a priority from "0" to "16". The OPC UA specification defines a value range of "1" to "1000" for the severity. "1" stands for the lowest and "1000" for the highest severity.

For this reason, the values of the priority are mapped according to the OPC severity. In the standard mapping, priority "0" is assigned to OPC severity "1" and priority "16" to OPC severity "1000". All other priority values are obtained by linear interpolation between "0" and "1000".

7.3.9.5 OPC UA mapping rules

During the configuration of the WinCC message system, settings are made to determine which process events generate a message. This message is generally shown as an event in OPC UA.

Attributes/properties of events

The following table shows the most important attributes/properties of an event and how the WinCC message system provides the information:

OPC UA attribute or property	Content/mapping in the WinCC message system
For all event types:	
EventId	Unique message identifier
EventType	Event type: Node ID of the WinCCAlarmConditionType or WinCCEventType node
SourceNode	Irrelevant
SourceName	Indicates the source of the message. You can find additional information about this in the section "Settings for mapping the WinCC message system (Page 57)".
Message	Message text for the corresponding message number
Time	Time of the event. The time stamp is given in UTC (Coordinated Universal Time).
Severity	Mapping of the priority of the WinCC message
Only with WinCCAlarmConditionType:	
ConditionName	Set text that is output as well as the message. Contains the message type, for example "Process control message"

OPC UA attribute or property	Content/mapping in the WinCC message system
Quality	Displays the quality of the message.
ConditionClassId	Node ID of the "ProcessConditionClassType" node
ConditionClassName	"ProcessConditionClassType"
Retain	True, with pending messages
NodeId	Condition ID: • Uniquely identifies a UA condition (e.g. an alarm). • Required for acknowledgment, even if no condition instances are supported.
EnabledState	True if the message is enabled.
ActiveState/Id	True if the message is has been received.
AckedState/Id	True if the message has been acknowledged.
ClientUserId	Indicates the user that is logged on.

Note

The following OPC UA condition or alarm properties are not supported by the OPC UA server:

- BranchId
- LastSeverity
- InputNode
- ConfirmedState
- SuppressedState
- ShelvingState
- SuppressedOrShelved
- MaxTimeShelved

7.3.9.6 Message statuses / acknowledgment statuses

The following table shows WinCC message status mapping to the respective WinCCAlarmConditionType properties:

Message status	EnabledState/Id	ActiveState/Id	AckedState/Id
Locked message	false	-	-
Unlocked message	true		
Message came in	true	true	false
Sent message with acknowledgment	true	false	true
Sent message without acknowledgment	true	false	false
Acknowledged messages (message pending)	true	true	true
Acknowledged messages (message no longer pending)	true	false	true
Received, acknowledged message	true	true	true
Received, sent message with acknowledgment	true	false	true
Received, sent message without acknowledgment	true	false	false
Message acknowledged by system (message still pending)	true	true	true

Message status	EnabledState/Id	ActiveState/Id	AckedState/Id
Message acknowledged by the system (message no longer pending)	true	false	true
Emergency-acknowledged message (message pending)	true	true	true
Emergency-acknowledged message (message no longer pending)	true	false	true

7.3.9.7 Settings for mapping the WinCC message system

The configuration of the OPC A&E server also applies to the OPC UA server as regards the mapping of the "SourceName" and "Message" properties of a message.

- If the OPC A&E server operates with hierarchical access, the following mapping also applies to OPC UA:

SourceName	Indicates the source of a message with the following formatting: "<Server prefix>::Area\origin" The server prefix of a local computer is "@LOCALMACHINE".
Message	Displays the message text of the corresponding message number.

- If the OPC A&E server operates without hierarchical access, the following mapping also applies to OPC UA:

SourceName	Indicates the source of a message. The Source has the format "<Server prefix>::localhost::". The server prefix of a local computer is "@LOCALMACHINE".
Message	Displays the message text of the corresponding message number.

Additional information

You can find information on this in the section "OPC A&E with hierarchical access (Page 75)".

7.3.9.8 Alarm groups

In this version, the WinCC alarm groups are not displayed in the namespace.

7.4 OPC Data Access (OPC DA)

Overview

You can find information about the following topics in the section below:

- Introduction to the OPC Data Access interface (Page 58)
- How the OPC DA server works (Page 59)

7.4.1 Introduction to the OPC Data Access interface

Introduction

The data access interface is a vendor-independent worldwide standard for reading, writing and monitoring process data. Communication is based on the Microsoft COM protocol. This standard has gained acceptance both with users and manufacturers. The user programs, for example, range from office applications to sophisticated HMI (Human Machine Interface) or SCADA (Supervisory Control and Data Acquisition) systems.

OPC Data Access (OPC DA)

The OPC Data Access specification defines the interface between client and server programs for process data communication.

Here, the OPC DA servers allow one or more OPC DA clients transparent access to an extremely wide variety of data sources (for example temperature sensors) and data sinks (for example controllers). The following options are typical for connecting data sources and sinks to automation systems in a PCS 7 plant:

- Connection via PROFIBUS
- Connection using input modules or output modules

OPC DA client

Data access clients can, for example, be Excel tables in conjunction with Visual Basic for Applications (VBA). They may also be extensive applications created with Visual Basic or Visual C++.

OPC DA server

Basically an OPC DA server makes process data available to an OPC DA client for read and write access.

OPC DA servers can be programs that, for example, allow access to an automation system via a serial interface. More complex programs are possible that provide access to large numbers of tags on numerous devices using extensive communications mechanisms. This is the case with OpenPCS 7.

Access options using OPC DA

OPC Data Access is a specification for access to process data using process variables. An OPC DA server manages the process variables and the various options for access to these variables. The following types of access are possible:

- Reading the value of one or more process variables (tags)
- Modifying the value of one or more process variables (tags) by writing a new value
- Monitoring the value of one more process variables (tags)
- Reporting value changes

7.4.2 How the OPC DA server works

Requirement

To be able to set up successful OPC communication, the following requirements must be met:

- The PCS 7 OS project and the OPC DA server must be loaded and started.
- The computer of the OPC DA server can be reached by the OPC DA client via its IP address.

Note

In your PCS 7 OS project, variables can be grouped together in variable groups to provide structuring. The variables must not have the same name as a variable group.

Operating principle

The OPC DA server supports OPC data access with the following specifications:

- 1.0a
- 2.0.
- 3.0.

The OPC DA server is a DCOM application. Using this software interface, the OPC DA server provides the OPC DA client with the necessary information about PCS 7 OS tags. The OPC DA server becomes active when it is accessed by the OPC DA client over a connection to the OPC DA server.

7.5 OPC Historical Data Access (OPC HDA)

Overview

You can find information about the following topics in the sections below:

- How the OPC HDA server works (Page 60)
- Data structure of the OPC HDA server (Page 61)
- Overview of supported attributes (Page 61)
- Overview of supported aggregates (Page 62)
- Overview of supported functions (Page 63)
- Time format of the OPC HDA server (Page 63)
- Quality codes of the OPC HDA server (Page 65)
- Write access supported by the OPC HDA server (Page 65)

The following sections show the data structure and the attributes, aggregates and functions supported by the OPC HDA server. This is not a detailed description but rather an overview.

Detailed information is available in the "OPC Historical Data Access Specification" of the OPC Foundation.

7.5.1 How the OPC HDA server works

Operating principle

Using OPC HDA, it is possible to access archived data from the PCS 7servers. The OPC HDA server is a DCOM application that provides the OPC HDA client with the required data from the PCS 7 OS archive system. Data is accessed via item handles. Only read access to archived data is approved for PCS 7. The data can also be analyzed.

The OPC HDA server supports the OPC Historical Data Access 1.20 specification. This was confirmed by the compliance test. All OPC HDA clients complying with the OPC Historical Data Access 1.20 specification can access the OPC HDA server. The use of individually programmed OPC HDA clients is the best way to meet the requirements.

Using the OPC HDA client

There are numerous potential applications for an OPC HDA client. The following uses are possible:

- Analysis and evaluation of archived data.
- Static process control via archives from various OPC HDA servers.

Rules

If you request historical values with the OPC HDA client, remember the following during configuration:

- Select the cycle for a query so that the client has received the requested data before the next query starts. If the cycles are too short, the result can be a large time offset when receiving the data.
- The CPU load of the PCS 7 OS server depends on the number of variables per query.

7.5.2 Data structure of the OPC HDA server

Data structure

The data of the OPC HDA server is structured. The following table describes the data structure.

Element	Description
Raw Data	The raw data is the data transferred from the PCS 7 OS archive system for a specified period. This data has a time stamp and a quality.
Attribute	Return additional quality characteristics of the raw data. Attributes include data type, information on archiving. Additional information is available in the section "Overview of the supported attributes".
Aggregate	Return a single value based on the raw data of a particular period. Aggregate functions include average value, minimum and maximum. Additional information is available in the section "Overview of the supported aggregate functions".
StartTime/EndTime	Specify the start and end time for the period.
Bounding Values	Bounding values are the values recorded at the start and end time. If these do not exist, the values closest to the time are used as the bounding values.
Item Handle	The item handle is a unique assignment to a PCS 7 OS archive tag.
ItemID	The ItemID is the unique identification of the PCS 7 OS archive tag. An item handle can be fetched using the ItemID.

Additional information

- Additional information on the data structure of the OPC HDA is available in the specification *OPC Historical Data Access Specification V1.2* of the OPC Foundation.

7.5.3 Overview of supported attributes

Supported attributes

The following table lists the attributes supported by the OPC HDA server. You can find additional information in the "OPC Historical Data Access Specification 1.20" of the OPC Foundation.

Attribute	Attribute ID	Description
ItemID	OPCHDA_ITEMID	Specifies which PCS 7 OS archive tag is accessed.
Item data type	OPCHDA_DATA_TYPE	Specifies the data type of the PCS 7 OS archive tag.
Description	OPCHDA_DESCRIPTION	Outputs the description of the PCS 7 OS archive tag. The description is specified in PCS 7 OS TagLogging.
Engineering Units	OPCHDA_ENG_UNITS	Specifies the labeling of the unit shown in the display. The labeling is specified in PCS 7 OS TagLogging.

7.5.4 Overview of supported aggregates

Supported aggregate functions

The following table lists the aggregate functions supported by the OPC HDA server. You can find additional information in the "OPC Historical Data Access Specification 1.20" of the OPC Foundation.

Aggregate function	Description
OPCHDA_COUNT	Obtains the number of raw data for the specified period.
OPCHDA_START	Obtains the start value of the raw data at the start time.
OPCHDA_END	Obtains the end value of the raw data at the end time.
OPCHDA_AVERAGE	Obtains the average value of the raw data for the specified period.
OPCHDA_TIMEAVERAGE	Obtains the time-weighted average value of the raw data for the specified period.
OPCHDA_TOTAL	Obtains the total value for the specified period.
OPCHDA_STDEV	Obtains the standard deviation of the raw data for the specified period.
OPCHDA_MINIMUMACTUALTIME	Obtains the highest value and the time stamp of the raw data for the specified period.
OPCHDA_MINIMUM	Obtains the lowest value of the raw data for the specified period.
OPCHDA_MAXIMUMACTUALTIME	Obtains the highest value and the time stamp of the raw data for the specified period.
OPCHDA_MAXIMUM	Obtains the highest value of the raw data for the specified period.
OPCHDA_DELTA	Obtains the difference between the first and last value of the raw data of the specified period.
OPCHDA_REGSLOPE	Obtains the slope of the linear regression of the raw data for the specified period.
OPCHDA_REGCONST	Obtains the value of the linear regression of the raw data at the start time.
OPCHDA_REGDEV	Obtains the standard deviation of the linear regression of the raw data for the specified period.
OPCHDA_VARIANCE	Obtains the variance of the raw data for the specified period.
OPCHDA_RANGE	Obtains the difference between OPCHDA_MAXIMUM and OPCHDA_MINIMUM of the raw data for the specified period.
OPCHDA_DURATIONGOOD	Obtains the period in which the quality of the raw data was good. The period is entered in seconds.
OPCHDA_DURATIONBAD	Obtains the period in which the quality of the raw data was bad. The period is entered in seconds.
OPCHDA_PERCENTGOOD	Obtains the portion during which the quality of the raw data was good as a percentage.
OPCHDA_PERCENTBAD	Obtains the portion during which the quality of the raw data was bad as a percentage.
OPCHDA_WORSTQUALITY	Obtains the worst quality of the raw data for the specified period.

7.5.5 Overview of supported functions

Introduction

The following tables list the functions supported by the OPC HDA server. These functions can be used by the OPC HDA client for data exchange. You can find additional information in the "OPC Historical Data Access Specification 1.20" of the OPC Foundation.

Function	Description
ReadRaw	Specifies the raw data, its quality and its time stamp for the specified period.
ReadProcessed	Returns the calculated value, the quality of the value and the time stamp for the specified period. The calculated value depends on the selected aggregate.
ReadAtTime	Specifies the raw data, its quality and its time stamp for a specific point in time. If no value exists, it is interpolated for this point in time.
ReadAttribute	Returns the attributes of the item and the time stamp for the specified period.

7.5.6 Time format of the OPC HDA server

Period of the historical data

The period is specified by the start and end time on the OPC HDA server. The specified period defines the period examined for the historical data. When specifying the times, you will have to maintain certain formats. A point in time can be specified in the following ways:

- Absolute in UTC.
- Relative to the local time of the OPC HDA server.

Specified in absolute form according to UTC

As default, the OPC HDA server works with the coordinated world time UTC as the timebase. The time corresponds to the Greenwich time zone (= central European standard time minus one hour).

Time format

YYYY/MM/DD hh:mm:ss.msmsms

Parameter

YYYY = year

MM = month

DD = day

hh = hour

mm = minute

ss = second

ms = millisecond

Example of an entry

2011/08/10 09:27:30.000

Specifying the point in time relative to the local time

Here, the point in time is specified relative to the local time of the OPC HDA server. You set the local time zone in the Control Panel of your computer in "Date/Time".

Time format

Keyword +/-offset1 +/-offset(n)

The offset is the deviation from local time of the OPC HDA server.

Keyword

NOW = current local time of the server

SECOND = current second

MINUTE = current minute

HOUR = current hour

DAY = current day

WEEK = current week

MONTH = current month (0-11)

YEAR = current year

Offset

+/-S = deviation in seconds

+/-M = deviation in minutes

+/-H = deviation in hours

+/-D = deviation in days

+/-W = deviation in weeks

+/-MO = deviation in months

+/-Y = deviation in years

Example

DAY - 1D = previous day

DAY-1D + 7H30 = previous day at 7:30 a.m.

MO-1D+5H = last day of the previous month at 5.00 a.m.

NOW-1H15M = 1 hour and 15 minutes ago

YEAR+3MO= April of this year

7.5.7 Quality codes of the OPC HDA server

Quality codes

The quality code is required to check the status and quality of the raw data. The following table shows the quality codes of OPC HDA.

Code	OPC	Description	Quality
0x00040000	OPCHDA_RAW	Provides information on the quality of the value transfer of raw data.	GOOD BAD UNCERTAIN
0x00080000	OPCHDA_CALCULATED	Provides information on the quality of the value transfer of calculated data.	GOOD BAD UNCERTAIN
0x00100000	OPCHDA_NOBOUND	No bounding values were found at the start or end time.	BAD
0x00200000	OPCHDA_NODATA	No raw data was found for the specified period.	BAD
0x00400000	OPCHDA_DATALOST	Raw data was not completely archived during the selected period.	BAD

You can find additional information in the "OPC Historical Data Access Specification 1.20" of the OPC Foundation.

7.5.8 Write access supported by the OPC HDA server

Introduction

The OPC HDA specification of the OPC Foundation also defines write access to archived data.

Note

PCS 7 is a process control system. In a process control system, archived data must not be modified. For this reason, write access to archived data using OPC HDA has not been enabled.

7.6 OPC Alarms and Events (OPC A&E)

Overview

You can find information on the following topics in the sections below:

- Introduction to OPC A&E (Page 66)
- Mapping the PCS 7 OS message system on OPC A&E (Page 68)
- Mapping the message classes and message types of PCS 7 OS on OPC A&E (Page 70)
- Mapping priorities of PCS 7 OS messages on OPC A&E (Page 70)
- Attributes of the PCS 7 OS message system (Page 71)
- Acknowledgment scheme (Page 72)
- Quality codes for OPC A&E (Page 74)
- OPC A&E with hierarchical access (Page 75)
- Upgrading OPC A&E (Page 81)

Note

You can find general information on these topics in the sections. Detailed information is available in the "OPC Alarms and Events Custom Interface Standard V1.0" and "OPC Alarm and Events Automation Interface Standard V1.01" specifications of the OPC Foundation.

7.6.1 Introduction to OPC A&E

Functionality of the OPC A&E server

The OPC A&E server is a DCOM application. The OPC A&E client will receive information on status changes of PCS 7 OS messages using subscriptions. Using the subscription, the OPC A&E client can set a filter. This filter specifies which messages and attributes will be displayed. The OPC A&E server supports the specification OPC Alarm&Event 1.10. The following section explains the mapping of the PCS 7 OS message system on OPC A&E and the attributes supported by the OPC A&E server. This is not a detailed description but rather an overview of the specific information. Detailed information on this topic is available in the specification of "OPC Alarms & Events 1.10".

Supported events

The OPC A&E server supports the following events:

- Condition-related event
- Simple event
- Tracking event

Condition-related events

With a condition-related event server, the event is associated with a condition. A condition may be the limit violation of a tag. On the PCS 7 OS, a message will be generated as soon as a limit violation occurs. This message is shown as an alarm in OPC A&E.

Simple event

Simple events are messages that inform the OPC A&E client about events. Simple events include, for example, the launching and closing of programs.

Note

Note the following when using redundant PCS 7 OS servers:

Simple events that are linked to internal tags are sent twice when comparing tags. The first message is triggered by the OS master server, the second by the OS standby server.

Tracking event

If a change is made in the process control system, the OPC A&E client will receive a message. A change can, for example, be a change to a control parameter in the faceplate of the controller or the suppression of messages in the message system.

Note

When filtering for all alarms of a plant section, make sure that you replace the source with a wildcard in the filter text, because the source is only generated in runtime for a tracking event. For example, the filter text for all events from the "Plant1\Unit1\Tank1" area is "Server prefix::*Plant1\Unit1\Tank1*" and not "Server prefix::Plant1\Plant1\Unit1\Tank1*".

OPC A&E client

All OPC A&E clients complying with the OPC Alarms & Events 1.10 specification can access the OPC A&E server. On the OPC A&E client, this can involve, for example, simple or complex Microsoft Visual Basic or Microsoft Visual C++ applications. The use of individually developed OPC A&E clients is the best way to meet the requirements. An OPC A&E client can, for example, be used to analyze alarms or for the common archiving of alarms from different OPC A&E servers.

When acknowledging messages, a distinction must be made between "pending" messages and "historical" messages.

- **Pending messages**
  Messages received over the path specified by OPC A&E (for example, "Refresh") can be acknowledged.
- **Historical messages**
  Messages received over the "historical messages" path (extension of the Siemens OPC A&E server) cannot be acknowledged.

Additional information

- For additional information about "OPC - OLE for Open Connectivity" refer to the *WinCC Information System*.
- Additional information about the OPC A&E can be found in the specification *OPC Alarms & Events 1.10*.

7.6.2 Mapping the PCS 7 OS message system on OPC A&E

Introduction

In PCS 7, program functionality in the automation system is configured using messaging capable CFCblocks from the PCS 7library. After compiling the PCS 7 OS, the configuration messages exist in the PCS 7 OS message system. These messages are mapped to the OPC A&E standard by the OpenPCS 7 station implementation. The OS configuration defines which event in the process triggers a message.

Mapping the OS message system on OPC A&E with hierarchical access

In PCS 7, the following standard settings in the OS message system are used for mapping messages on OPC A&E:

- The source of a message is mapped on the OPC source.
- The message text of a message is mapped on the OPC message.

Overview

The OPC attribute class OPCEvent is used in OPC A&E to display events. The following table shows selected OPC attributes and their meaning in the OS message system.

The events that use the configured attributes are shown in the third column of the table:

- "S" means a simple event
- "C" means a condition-related event
- "T" means a tracking event

OPC	OS message system	Event type
Area	In PCS 7, an "area" is an area, a diagnostics area or an alarm hiding group. If there is no area, no diagnostic area or no alarm hiding group configured for the message, only the OPC area corresponding to the server prefix will be available.	S, C, T
Source	Indicates the source of a message. The source has the format "<Server prefix>::Area \Source". The server prefix of a local computer is "@LOCALMACHINE". The server prefixes always show the top areas in the hierarchy of the server.	S, C, T
Time	Issues a time stamp for received, sent and acknowledged messages. Issues a time stamp in UTC (Universal Time Coordinated).	S, C, T
Type	Indicates whether the event is a simple, tracking or condition-related event.	S, C, T
Severity	Displays the priority of the message.	S, C, T

OPC	OS message system	Event type
EventCategory	Displays the message class. "Event Category" is made up of the "CategoryID" and the "Category Description". "CategoryID" corresponds to the internal ID of the message class. "Category Description" corresponds to the name of the message class.	S, C, T
Message	Displays the message text of the corresponding message number.	S, C, T
Condition	Displays the message type. The message types "Alarm", "Warning" and "Tolerance" are combined and called "Level" in PCS 7.	C
Sub Condition	Corresponds to the "Condition" parameter. In PCS 7, this is identical to Condition in single state conditions. The following subconditions are mapped for the "Level" multi-state condition in PCS 7: • Alarm low • Alarm high • Warning low • Warning high • Tolerance low • Tolerance high	C
ChangeMask	Specifies the change of the condition.	C
NewState	Displays the current status of the condition.	C
ConditionQuality	Displays the quality of the message. For additional information about quality codes refer to the system manual *MDM - WinCC/Connectivity Pack*.	C
AckRequired	Indicates whether the message requires acknowledgment.	C
EventAttribute	Lists the attributes required for this message.	C
Quality	Indicates the quality code of the message.	C
Cookie	Does not include any usable information for the client	C
ActorID	Indicates which user acknowledged the message.	T

Note

The message classes and message types **must** be configured identically on the connected OS servers if you operate the OPC A&E server as follows:

- On an OS client
- on a WinCCconnectivity station
- within the scope of OpenPCS 7

If the OS server is not configured identically, the OPC client that is used must access the respective OS server directly.

7.6.3 Mapping the message classes and message types of PCS 7 OS on OPC A&E

Mapping of the message classes

The PCS 7 OS message system provides information on fault and operational states in the process. A PCS 7 OS message always belongs to a particular message class and message type that is in turn related to an event category.

Event category

Each combination of message class and message type is mapped to an event category on the OPC A&E server. An event category is identified by a CategoryID and a category description.

The CategoryID is made up of the internal PCS 7 OS ID of the message class and message type.

The category description is made up of the name of the message class and message type.

The names of the message classes and message types can be obtained explicitly using the alarm attributes CLASSNAME and TYPENAME.

7.6.4 Mapping priorities of PCS 7 OS messages on OPC A&E

Mapping priorities

The priority of messages is mapped by the OPC A&E server to the "Severity" attribute. When configuring alarms in the message system, you can configure a priority from "0" to "16". The OPC A&E specification defines a range of values from "1" to "1000" for the severity. "1" stands for the lowest and "1000" for the highest severity. For this reason, the values of the priority are mapped according to the OPC severity. In the standard mapping, priority "0" is assigned to OPC severity "1" and priority "16" to OPC severity "1000". All other priority values are obtained by linear interpolation between "0" and "1000".

7.6.5 Attributes of the PCS 7 OS message system

Attributes of the PCS 7 OS message system

The following table lists the OPC attributes and their meaning in the PCS 7 OS message system. A PCS 7 OS message belongs to a message class and has system attributes, user texts and process values. Some of the attributes of a PCS 7 OS message are only for internal use in the PCS 7 OS and are therefore not relevant for an OPC A&E client. These attributes are not listed in the table.

OPC attribute	Meaning in the PCS 7 OS message system
ClassName	Displays the message class name.
Type name	Displays the message type name.
ForeColor	Displays the text color for the display of received, sent and acknowledged messages.
BackColor	Displays the background color for the display of received, sent and acknowledged messages.
FlashColor	Displays the flashing color.
Flags	Indicates whether the message requires acknowledgment.
Text01	Displays the content in the source.
Text02	Displays the content of the area.
Text03	Displays the content in the event.
Text04	Displays the batch name.
Text05	Displays the content of the operation message.
Text06	Displays the content of the text block "free 1".
Text07	Displays the content of the text block "free 2".
Text08	Displays the content of the text block "free 3".
Text09	Displays the content of the text block "free 4".
Text10	Displays the content of the text block "free 5".
ProcessValue01	Displays the content of ProcessValueBlock01.
ProcessValue02	Displays the content of ProcessValueBlock02.
ProcessValue03	Displays the content of ProcessValueBlock03.
ProcessValue04	Displays the content of ProcessValueBlock04.
ProcessValue05	Displays the content of ProcessValueBlock05.
ProcessValue06	Displays the content of ProcessValueBlock06.
ProcessValue07	Displays the content of ProcessValueBlock07.
ProcessValue08	Displays the content of ProcessValueBlock08.
ProcessValue09	Displays the content of ProcessValueBlock09.
ProcessValue10	Displays the content of ProcessValueBlock10.
StateText	Displays the status message.
InfoText	Displays the information text for the message.
LoopInAlarm	States if LoopInAlarm has been configured.
ClassID	Displays the message class ID.
Type ID	Displays the message type ID.
AG_Number	Displays the number of the automation device that generated the message.

OPC attribute	Meaning in the PCS 7 OS message system
CPU_Number	Displays the number of the CPU that generated the message.
Duration	Displays the period of time between message received, sent and acknowledged.
QuitStateText	Indicates whether the message has been acknowledged.
Priority	Indicates the configured priority of the message.
OS EVENT ID	Displays the message number.
IEVENT UNIQUE EVENT ID	Displays the unique event ID across all messages.
OS-HIDDEN	Indicates whether a message configured with the "alarm hiding" function is shown (yes/no).
HIDDEN-COUNT	Indicates the number of messages currently hidden with the "alarm hiding" function (HIDDEN).

7.6.6 Acknowledgment scheme

Mapping the acknowledgment concept

In the PCS 7 OS, the acknowledgment concept is how a message is displayed and processed from "came in" to "went out". On the OPC A&E server, this message status is displayed in the "ChangeMask" and "NewState" parameters.

Condition-related event, simple event and tracking event

Messages with acknowledgment are sent from the system to the client as condition-related events. Before a message is treated as a simple event, the message class of the message must meet the following conditions:

- "Acknowledgment came in" is not activated.
- Message without status "went out" is activated.

The following messages are sent as tracking events in PCS 7:

- Messages of the "Operation message" message class.
- Messages of "System, does not require acknowledgment" message class with the "Operation message" message type.

Note

Messages with "System, does not require acknowledgment" message class and "Process control system" message type are transferred as simple events with the "System message" event category.

ChangeMask

The "ChangeMask" parameter keeps track of where the message status was changed.

Parameter values of ChangeMask:

- OPC_CHANGE_ACTIVE_STATE
- OPC_CHANGE_ENABLE_STATE
- OPC_CHANGE_ACK_STATE

NewState

The "NewState" parameter indicates the message status after a change.

Parameter values of NewState:

- OPC_CONDITION_ACTIVE
- OPC_CONDITION_ENABLED
- OPC_CONDITION_ACKED

PCS 7 OS	NewState	ChangeState
Message came in	OPC_CONDITION_ACTIVE OPC_CONDITION_ENABLED	OPC_CHANGE_ACTIVE_STATE
Message went out with Acknowledgment	OPC_CONDITION_ACKED OPC_CONDITION_ENABLED	OPC_CHANGE_ACTIVE_STATE
Message went out without Acknowledgment	OPC_CONDITION_ENABLED	OPC_CHANGE_ACTIVE_STATE
Acknowledged messages (message still pending)	OPC_CONDITION_ACTIVE OPC_CONDITION_ACKED OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
Acknowledged messages (message no longer pending)	OPC_CONDITION_ACKED OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
Locked message	-	OPC_CHANGE_ENABLED_STATE
Unlocked message	OPC_CONDITION_ENABLED	OPC_CHANGE_ENABLED_STATE
Came in, acknowledged message	OPC_CONDITION_ACTIVE OPC_CONDITION_ACKED OPC_CONDITION_ENABLED	OPC_CHANGE_ACTIVE_STATE
Came in, went out message with acknowledgment	OPC_CONDITION_ACKED OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
Came in, went out message without acknowledgment	OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
Message acknowledged by system (message still pending)	OPC_CONDITION_ACTIVE OPC_CONDITION_ACKED OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE

PCS 7 OS	NewState	ChangeState
Message acknowledged by system (message no longer pending)	OPC_CONDITION_ACKED OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
Emergency acknowledged message (message still pending)	OPC_CONDITION_ACTIVE OPC_CONDITION_ACKED OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
Emergency acknowledged message (message no longer pending)	OPC_CONDITION_ACKED OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE

Note

Historical alarms and events are not acknowledged. The OPC A&E historical events interface has read access only.

The following table shows the mapping of the "ChangeMask" and "New State" parameters to the properties of the OPCEvent class of the "OPCSiemensAlarmEventAutomation" type library.

ChangeMask / NewState	Properties of the OPCEvent class
OPC_CHANGE_ACTIVE_STATE	ChangeActivState
OPC_CHANGE_ENABLE_STATE	ChangeEnableState
OPC_CHANGE_ACK_STATE	ChangeAckState
OPC_CONDITION_ACTIVE	ConditionActive
OPC_CONDITION_ENABLED	ConditionEnabled
OPC_CONDITION_ACKED	ConditionAcknowledged

7.6.7 Quality codes for OPC A&E

Quality codes

The quality code is required to check the status and quality of a message. The following table shows the quality codes of OPC A&E.

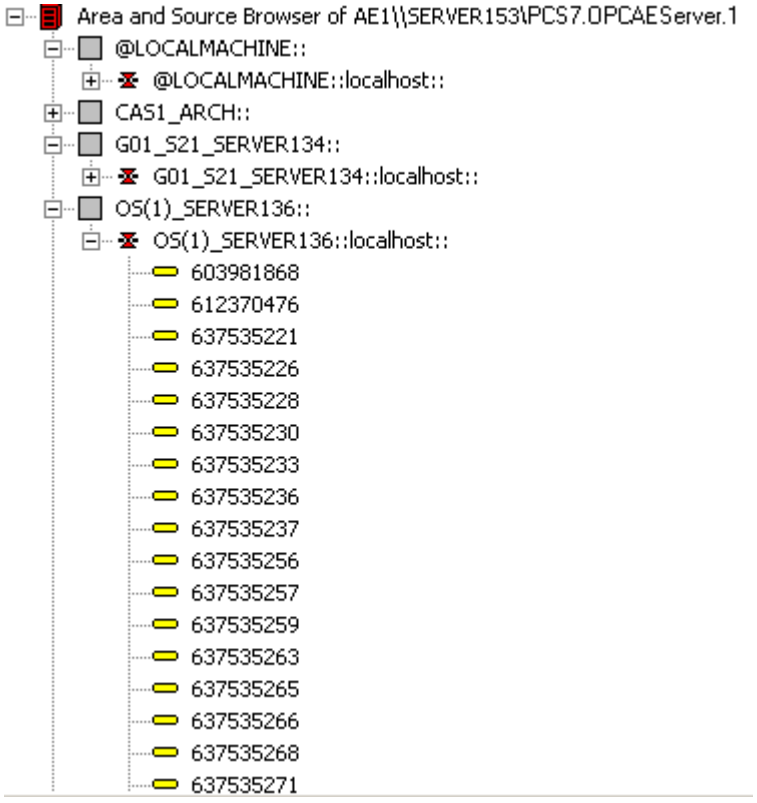
Code	Quality	Status
0xC0	OPC_GOOD	OK
0x40	OPC_UNCERTAIN	Output if there are discrepancies, for example with a late acknowledgment.
0x00	OPC_BAD	Output if the connection to the source is interrupted.

7.6.8 OPC A&E with hierarchical access

7.6.8.1 Differences between OPC A&E and OPC A&E with hierarchical access

Representation of messages with OPC A&E

The OPC A&E server supports "conditional events" and "simple events" during access to the message system. With "conditional events", the message numbers are shown for each source. Since an PCS 7 OS server can hold many more message numbers, it is hard to keep an overview of the messages. The following figure shows an example of the display in an OPC browser:



Representation of the messages with OPC A&E and hierarchical access

As of PCS 7 V7.0 SP1, you can use an OPC A&E server with hierarchical access in PCS 7. The OPC A&E server with hierarchical access supports the following event types:

- Condition-related event
- Simple event
- Tracking event

With a "condition-related event," the source of the messages is determined by the area and the source. The area can be a system area, a diagnostics area or an alarm hiding group. If the "Source" text block in WinCC is dynamic or not configured, the character string ""EventID xx"

is used for the source in OPC instead of this text. "xx" here stands for the unique WinCC message number. This syntax is used for the source in the Area Browser and for the message itself. A tracking event occurs when an operation message is triggered in the system. The following figure shows an example of the representation of a condition-related event in an OPC browser. The "Condition" is shown in addition to "Area" and "Source":

Recommendation

Use an OPC A&E server with hierarchical access when creating new projects. If you upgrade an existing project, the OPC A&E server can be used as before or the OPC A&E server can be converted to hierarchical access. The conversion can be undone again without any loss of data.

7.6.8.2 Example 1: Messages are not assigned to any area

Introduction

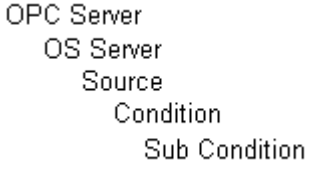
Process control messages, messages for Batch servers and Route Control servers are not assigned to an area or group. For these messages, no designated area is assigned in the structure of the OPC A&E access.

Requirement

- Process control messages are created on the OS servers.
- Messages for Batch servers and Route Control servers are created on the dedicated message servers.
- No diagnostics area is present.

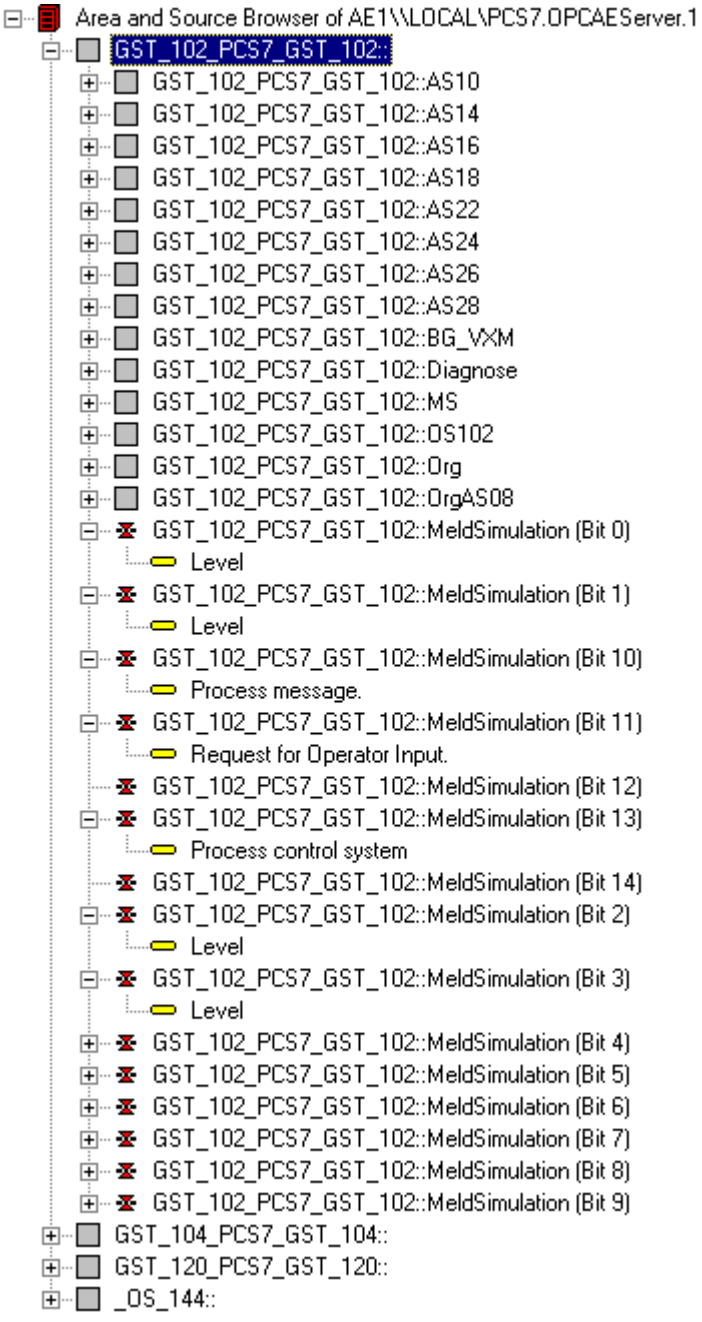
Structure for access to messages via the OPC A&E interface

The general mapping to the OPC A&E interface appears as follows:



Example of a hierarchy without assignment

The following picture shows the hierarchy of messages in a browser that can be sent to the client as a "conditional event".



7.6.8.3 Example 2: Messages are assigned to an area

Introduction

A PCS 7project is generally divided into several system areas and diagnostic areas. This means that messages are assigned to the areas.

The areas are shown as a hierarchy level in OPC A&E for hierarchical mapping.

Requirement

- The PCS 7project contains areas or diagnostic areas.

Structure for access to messages via the OPC A&E interface

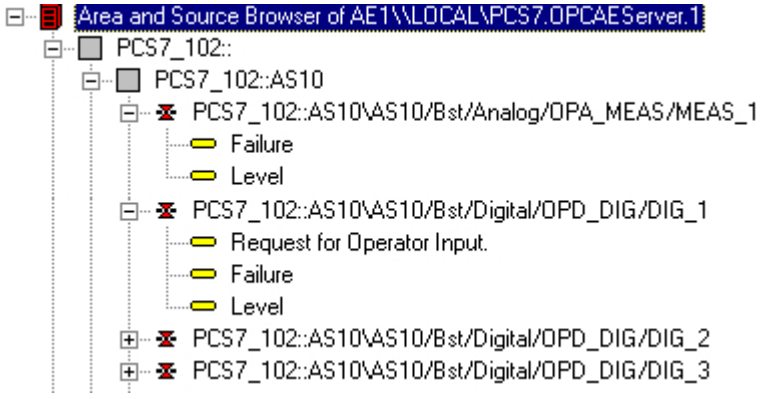
The general mapping to the OPC A&E interface appears as follows:

OPC Server
  OS Server
    Area / Maintenance Area
      Source
        Condition
          Sub Condition

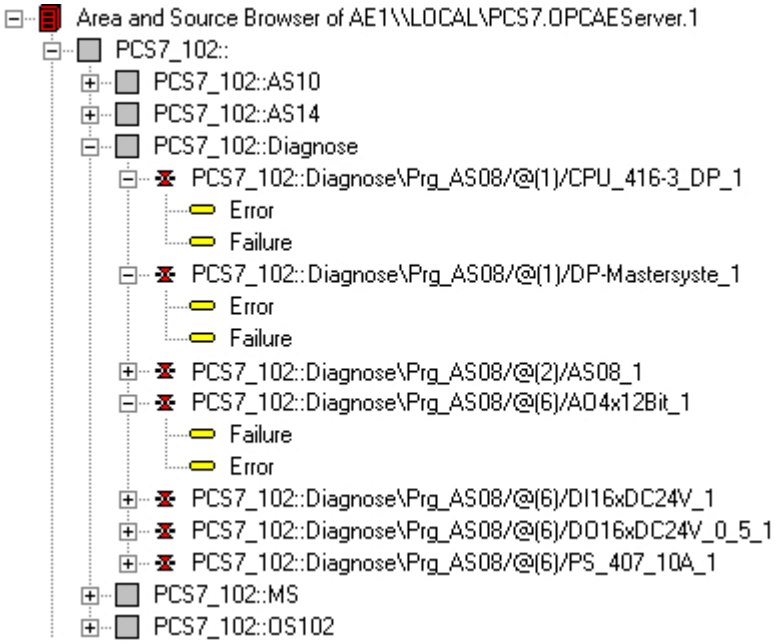
Example of a hierarchy with an area

The following pictures show the hierarchy of messages in a browser that can be sent to the client as a "conditional event".

Example with areas:



Example with diagnostic areas:



7.6.8.4 Example 3: Messages of an area are assigned to an alarm hiding group

Introduction

In the PCS 7project, alarm hiding groups are used to automatically hide messages. An alarm hiding group can contain messages from several areas.

This section describes how OPC A&E accesses these messages.

Requirement

- Alarm hiding groups are configured in the OS project.

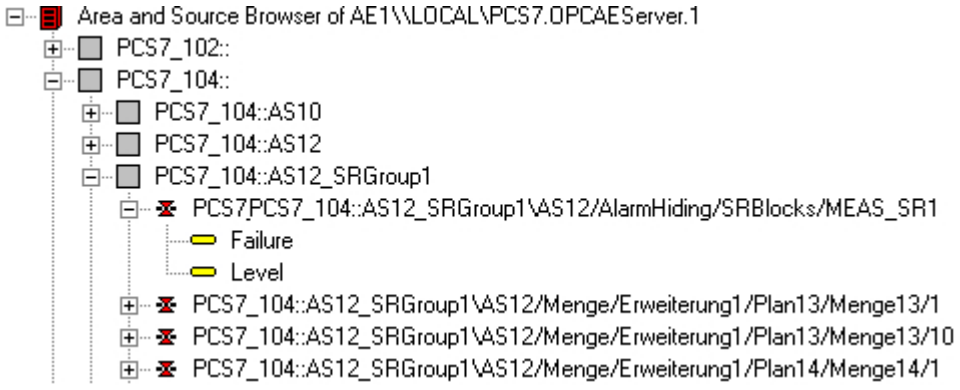
Access to the messages via the OPC A&E interface

The messages of an area are only shown in the alarm hiding group. The messages are no longer visible in the area.
The general mapping to the OPC A&E interface appears as follows:

OPC Server
OS Server
Alarm Hiding Group
Source
Condition
Sub Condition

Example of a hierarchy with an alarm hiding group

The following picture shows the hierarchy of messages in a browser that can be sent to the client as a "conditional event".



7.6.9 Upgrading OPC A&E

7.6.9.1 Updating PCS 7 projects with OPC A&E

As of PCS 7 V7.0 SP1, OPC A&E has been expanded to include hierarchical access to the message system.

You can continue to use the non-hierarchical OPC A&E server.

Updating OPC A&E

If you use OPC A&E with hierarchical access and want to use all functions, an expansion of the currently used OPC A&E client is necessary.

Starting from the PCS 7version of your project, various scenarios for updating projects with OPC A&E are described below.

7.6.9.2 How to update an OPC project with PCS 7 V8.2

Options

The following options are available for updating a PCS 7 project with configured access via OPC A&E:

- Conversion to OPC A&E with hierarchical access
- Retain OPC A&E without hierarchical access

Requirement

The PCS 7version of the project to be updated is PCS 7 V6.1 SP2 or higher.

Preparation

Before performing the update, delete any "CCAeProvider.ini" file on the OS server or on the ES directly in the OS project folder.

You can find the OS project folder in "wincproj" subfolder on the engineering station.

Conversion to OPC A&E with hierarchical access

No adjustments need be made for the conversion to OPC A&E with hierarchical access. OPC A&E with hierarchical access is the default setting.

Retain OPC A&E without hierarchical access

If you want to work with the OPC A&E server without hierarchical access, follow these steps:

1. Insert the following lines into the "CCAeProvider.ini" file in the "OPC/AlarmEvent" subdirectory of the OS project folder on the ES:
```
[OpcMapping]
OpcSource =
OpcMessage =
```
2. Perform a full download.

OS projects in PCS 7 V6.1 SP2

Note

Modified server prefix of the local computer

For OS projects in PCS 7 V6.1 SP2, note the following change in OPC A&E:
The server prefix of the local computer is now "@LOCALMACHINE" instead of "localhost".

7.7 OPC Historical Alarms and Events (OPC "H" A&E)

Overview

You can find information about the following topics in the section below:

- Introduction to OPC "H" A&E (Page 83)
- Reading archived messages (Page 83)
- Syntax for access to archived messages (Page 84)
- Read modes for archived messages (Page 86)
- Identifying archived messages (Page 87)

7.7.1 Introduction to OPC "H" A&E

OPC Historical Alarms and Events

Alarms and operating messages are stored in the database of the PCS 7 OS. With an OPC A&E client, you can access archived messages via the OPC A&E server. When accessing archived messages, the following methods are supported:

- Output of archived messages from a period in the past (history)
- Output of archived messages from a period in the past without an end time.

After outputting the archived messages, all additional newly generated messages are automatically sent to the OPC A&E client.

Note

Reading archived messages using OPC "H" A&E is a Siemens expansion of the OPC A&E standard.

Rule

After reading archived messages, you must not use the returned "ActiveTime" of a message either for acknowledging the message or for tracking the transitions of the message. The "ActiveTime" is not correct for archived messages. To make sure this is recognized, the OPC A&E client must check the "EventType" of a message for the additional flag "OPC_HAE_HISTORICAL_EVENTFLAG".

Note

You will find information on the additional flag in "Identifying archived messages".

7.7.2 Reading archived messages

Reading archived messages using OPC A&E

Archived messages of the PCS 7system are read by the OPC A&E client via the OPC A&E server using filters. In this situation, the term OPC "H" A&E is also used in this documentation. Apart from the standard filters, the following filters are also available with OPC "H" A&E:

- OPC_HAE_FILTER_BY_TIMEFRAME; corresponds to the function "ReadRaw" in OPC Historical Data Access
- OPC_HAE_FILTER_BY_STARTTIME; corresponds to the function "AdviseRaw" in OPC Historical Data Access

Source filter and request for archived messages

To be able to request the archived messages, the OPC A&E client must support the "SetFilter" functionality for a subscription. If you add the additional keyword "OPCHAEServer" in the array of the "Source Filter" of a subscription, the OPC A&E server sends archived messages. Along with this keyword, you can use the following parameters to specify how messages are read:

- Method
- Time period
- With or without limits

The list of sources specified in the filter can include other source names in addition to the "OPCHAEServer" source. In this case, the subscription returns the historical events of the specified sources. The order of the source names has no effect. After configuring the source filter, you can call up the selected period from the OPC A&E client with a "Refresh" call.

7.7.3 Syntax for access to archived messages

Syntax for access to archived messages

OPCHAEServer hMode=(read|advise) htStartTime=szTime

[htEndTime=szTime] [bBounds=(TRUE|FALSE)]

Parameter hMode = [read|advise]

This parameter specifies how the archived messages and events are read.

htStartTime parameter

This parameter specifies the time at which the messages and events are read from the archive.

htEndTime parameter

This parameter specifies the time up to which messages and events are read from the archive.

bBounds = [TRUE|FALSE] parameter

This parameter specifies how messages close to the start and end time are handled.

Example

The following table lists the parameters and their meaning and shows suitable examples.

Parameter	Meaning	Example	Require-ment
hMode = read	Outputs archived messages and events of a defined period in the past (comparable with ReadRaw in OPC Historical Data Access).	Setting a filter to read archived messages over the last 30 minutes: OPCHAEServer hMode=read htStartTime=NOW-30M bBounds=TRUE	Yes
hMode = advise	Outputs archived messages and events starting from a defined time. After receiving all archived messages, new messages are sent as with an active subscription. This is comparable with AdviseRaw in OPC Historical Data Access.	Reading archived messages as of the last 30 minutes: OPCHAEServer hMode=advise htStartTime=NOW-30M The subscription must be active for this purpose.	Yes
htStartTime	Specifies the time at which the messages and events are read from the archive.	htStartTime=NOW-30M	Yes
htEndTime	Specifies the time up to which messages and events are read from the archive. if "hMode = read" ist is set, "NOW" is used as the default setting.	htEndTime = 2011-09-10T10:00:00.000Z	optional
bBounds = FALSE	• The time stamp of the first transferred message >= htStartTime • The time stamp of the last transferred message <= htEndTime	bBounds=FALSE	optional
bBounds= TRUE	• The time stamp of the first transferred message <= htStartTime • The time stamp of the last transferred message >= htEndTime FALSE is the default setting.	bBounds=TRUE	optional

Note

The following notation is supported for the "htStartTime" and "htEndTime" parameters:

- Relative notations, for example NOW
- Symbolic values, for example NOW, YEAR, MONTH
- Absolute UTC date/time specified according to XML notation: "2011-09-10T10:00:00.000Z".
  The use of symbolic notation corresponds to the syntax of OPC Historical Data Access.

7.7.4 Read modes for archived messages

Read modes for archived messages

When reading archived messages, you can use the following read modes:

- Read
- Advise

"read" mode

With the "read" mode, archived messages are read from a defined period in the past. The read order of the messages is always chronological for each OS server from which alarms are read. By setting the start and end time, you can specify whether the oldest message is output first or last. If the start time is lower than the end time, the oldest message is output last.

If you want to use the "read" mode, execute the following functions on the subscription:

1. SetFilter
2. Refresh

A "SetFilter" during the "Refresh" is discarded.

If you activate the subscription during "Refresh", it will not affect the running of the refresh. The historical events are still transferred with the refresh ID. The newly generated events are transferred according to the standard behavior of an active subscription:

- Taking into account the set filter values with the exception of the "historical" source "OPCHAEServer"
- Without refresh ID

The following distinction is made based on the refresh ID:

- Event packages with the refresh ID contain only historical events. These events can also still be pending.
- Event packages without the refresh ID contain only newly generated events.

Note

An event package never contains historical and new events at the same time.

"advise" read mode

With the "advise" mode, archived messages are read starting at a defined time in the past. After reading all archived messages, new messages are sent as with an active subscription. The archived messages are transferred in chronological order related to each OS server. First the archived messages starting at the start time are transferred. The new archived messages are then transferred. An active subscription is used for the "advise" read mode. If you execute the "SetFilter" function on an active subscription, the historical alarms are transferred immediately. If you execute the "SetFilter" function on an inactive subscription, the archived messages are transferred only after the subscription is activated.

If you want to use the "advise" read mode with an inactive subscription, follow the steps below:

1. SetFilter
2. Set the subscription to active with SetState

If you deactivate the subscription, the transfer is interrupted. If you set the subscription to "inactive", the transfer is stopped. A "SetFilter" is discarded while the subscription is active. A "Refresh" on an active "historical" subscription in "advise" mode works in exactly the same way as a with a standard subscription:

All pending condition-related events are transferred in packets with the refresh ID. The last packet also has the "Last Refresh" ID. In "advise" mode, a "refresh" call has no influence on the reading of historical alarms.

Note

Note that you must not specify an end time with "advise".

7.7.5 Identifying archived messages

Principle

Archived messages are identified by an additional flag in the EventType.

This flag is connected with the real EventType via an OR link:

Designation	EventType	EventType (archived message)
OPC_SIMPLE_EVENT	0x01	0x81
OPC_CONDITION_EVENT	0x04	0x84
OPC_TRACKING_EVENT	0x02	0x82
OPC_HAE_HISTORICAL_EVENTFLAG		0x80

Example 1

The archived messages and events of the last 30 minutes are output in "read" mode using the following source filter. The oldest message per OS server is output first. The low limit value is also returned:

OPCHAEServer hmode=read htStartTime=NOW-30M bBounds=TRUE

Example 2

The archived events of 1 September 2006 from 10.00 a.m. to 12.00 a.m. are output in "read" mode using the following source filter. The newest message per OS server is output first. The limit values of this time range are also returned:

OPCHAEServer hMode=read htStartTime=2006-09-01T12:00:00.000Z

htEndTime=2006-09-01T10:00:00.000Z bBounds=TRUE

Example 3

The archived messages and events of the last 30 minutes are output in "advise" mode using the following source filter. After archived messages are read, the newly generated messages are transferred as with an active subscription:

OPCHAEServer hmode=advise htStartTime=NOW-30M

7.8 OLE DB

Overview

You can find information about the following topics in the section below:

- Basics of OLE DB (Page 88)
- Establishing the connection to the database (Page 89)
- Access to the OLE DB provider (Page 90)
- Representation of the process value archive (Page 91)
- Querying process value archives (Page 92)
- Representation of the message archives (Page 95)
- Querying the message archives (Page 97)

7.8.1 Basics of OLE DB

Introduction

You can access process value and message archives using the OLE DB interface made available by the PCS 7 OS and the corresponding database provider.

OLE DB

OLE DB is an open standard for fast access to different databases. It is irrelevant whether or not the database is relational. The connection between the OLE DB level and the database is established by an OLE DB Provider. OLE DB interfaces and providers are available from various vendors. OLE DB provides remote access to the database.

The WinCC OLE DB Provider has been tested and approved for PCS 7.

WinCC OLE DB Provider

The WinCC OLE DB Provider allows access to the process value and message archives.

The WinCC OLE DB Provider, which is integrated in the OpenPCS 7 station, enables transparent access to archived process data of the PCS 7 OS server.

With the WinCC OLE DB provider, you have direct access to PCS 7 OS archive data stored in the MS SQL server database. Depending on the configuration, process data of the PCS 7 OS is stored in compressed form.

Table 7-1 Options for accessing PCS 7 data with the WinCC OLE DB Provider:

Access options	WinCC OLE DB Provider
Transparent access to the process data of the PCS 7 OS	possible
Read data via multiple database segments	possible

Note

When the OS server closes a full archive and opens a new archive, there is a brief time when no data is read from the message and process value archives via the WinCC OLE DB Provider.

7.8.2 Establishing the connection to the database

ConnectionString

The connection between the application, which reads data via the OLE DB, and the archive database, is established via the Connection object with ActiveX Data Objects (ADO). One of the important parameters here is the ConnectionString. The ConnectionString contains all information necessary for access to the database using the OLE DB Provider.

Structure of the ConnectionString

```
"Provider = name of the OLE DB Provider; Catalog=database name;
Data Source=Package name + "\WinCC";"
```

The "Package name" is made up of the symbolic computer name and "::", e.g. "OSPro_1_Prj_OS::".

Parameter	Description
Provider	Name of the OLE DB Provider: e.g. WinCCOLEDBProvider
Catalog	For transparent access, enter the name of the PCS 7 OS project for "Catalog": Example: Catalog=WinCC_Project_Name
Data Source	For transparent access to a redundant operator station via the OLE DB Provider, enter the following for "Data Source": <Symbolic computer name>::\WinCC **Note** If you access an archive tag in an OS server "OS_Server_1", use the name of the archive tag. The OS server "OS_Server_1" returns the required ID and not the ID of the archive tag as the ID: <Symbolic computer name>\\<Archive_Var_Name>

Example of a VBA application:

In the following example, a Connection object is generated and then the connection to the WinCC database (process value or message archive) is opened.

```
Dim sPro As String
Dim sDsn As String
Dim sSer As String
Dim sCon As String

sPro = "Provider=WinCCOLEDBProvider.1;"'Provider
SDsn = "Catalog=CC_OS_06_09_25_13_36_53;"'Catalog
SSer = "Data Source=OSPro_1_Prj_OS::\WinCC"'Data Source
SCon = sPro + SDsn + SSer 'Connection String

'Connection
Set conn = CreateObject("ADODB.Connection")
conn.ConnectionString = SCon
conn.CursorLocation = 3
conn.Open
```

7.8.3 Access to the OLE DB provider

Access options

In PCS 7, the WinCC OLE DB Provider provides access to message and process value archives of the PCS 7 OS.

WinCC OLE DB Provider

The WinCC OLE DB Provider provides transparent access to all PCS 7 OS archive data. The message and process values of the PCS 7 OS are stored in compressed form, depending on the configuration.

Configuration options

To access the databases with the WinCC OLE DB Provider, you can write your own applications. For communication with the WinCC OLE DB Provider, ADO DB is used in applications. ADO DB can, for example, be used with Visual Basic or VBA.

Note

Special characters in tag names

In terms of the tag names, remember that programming languages such as Visual Basic, VBScript or VBA only permit the following characters in tag names:

"A...Z", "a...z", "0...9" and "_". If you use special characters such as "," or ";" in tag names in WinCC , the script will abort with an error message. In this case, use the "Tag ID" to address a tag in the script if it has special characters in the name. We also recommend that you use the "Tag ID" to improve performance.

Basic procedure

1. The OLE DB application must always run on the OpenPCS 7 station. Remote access is not possible.
2. With swapped archives, establish the connection between the SQL database and the swapped archives with the WinCC Archive Connector.

   Note

   The WinCC RT archives in the folder "<Project folder> \ ArchiveManager" and the subdirectories belonging to it must not be connected or disconnected using the Archive Connector because the connection to the SQL Server is managed by the WinCC basic system.

3. Establish the connection to the database, for example, by using MS Excel or your own application. Specify the required selection criteria and read the archive data.
4. For example, you can display the results of the query in MS Excel or export it as a CSV file.

7.8.4 Querying archive data

7.8.4.1 Representation of the process value archive

Recordset for process value archives

The result of the query is returned as a Recordset. This section describes the structure of the Recordset for process value archives.

Field name	Data type	Comment
ValueID	Integer 4 bytes	Unique identification of the value
TimeStamp	Datetime	Time stamp

Field name	Data type	Comment
RealValue	Real 8 bytes	Tag value
Quality	Integer 4 bytes	Quality code of the value (e.g. "good" or "bad")
Flags	Integer 4 bytes	Internal control parameter

7.8.4.2 Querying process value archives

Principle

You can access a process value archive with the following query. The data can be selected using filter criteria. The queries are transferred to the database with the command object.

Syntax

```
TAG:R, <ValueID oder ValueName>,<TimeBegin>,<TimeEnd>[,<SQL_clause>]
[,<TimeStep>]
```

Table 7-2 Parameter

Parameter	Description
ValueID	ValueID from the database table. Can be named more than once, for example: "TAG:R,(ValueID_1;ValueID_2;ValueID_x),<TimeBegin>,<TimeEnd>"
ValueName	ValueName in the format 'ArchiveName\Value_Name'. The <ValueName> parameter must be enclosed in single quotes. Multiple naming is possible, for example: "TAG:R,('ValueName_1';'ValueName_2';'ValueName_x'), <TimeBegin>,<TimeEnd>" **Note** With tag names, remember that programming languages such as Visual Basic, VBScript or VBA only permit the following characters in tag names: "A...Z", "a...z", "0...9" and "_". If you use special characters such as "," or ";" in tag names in WinCC , the script will abort with an error message. In this case, use the "Tag ID" to address a tag in the script if it has special characters in the name.
TimeBegin	Start time in the format: 'YYYY-MM-DD hh:mm:ss.msc' If you use <TimeStep> , then <TimeBegin> must be specified as real time. Relative times or "0000-00-00 00:00:00.000" are not possible.
TimeEnd	End time in the format: 'YYYY-MM-DD hh:mm:ss.msc'

Parameter	Description
SQL_Clause	Filter criterion in SQL syntax: [WHERE search_condition] [ORDER BY {order_expression [ASC|DESC] }] The criterion "ORDER BY" can only be used with a specified sort order "{order_expression [ASC|DESC] }" .
TimeStep	Values in the specified time interval are grouped together beginning at the start time <TimeBegin>. Format: 'TIMESTEP=x,y' x = period in seconds y = aggregate function type The interval result is returned for an interval depending on the type of aggregate function. The following values are possible for the type of aggregate function: • **Without interpolation:** If there are no values in the period, no period result will be returned. 1 (FIRST): First value 2 (LAST): Last value 3 (MIN): Minimum value 4 (MAX): Maximum value 5 (AVG): Average 6 (SUM): Sum 7 (COUNT): Number of values • **With interpolation:** An interval result is returned for every interval. This is done with linear interpolation. There is no extrapolation. 257 (FIRST_INTERPOLATED): First value 258 (LAST_INTERPOLATED): Last value 259 (MIN_INTERPOLATED): Minimum value 260 (MAX_INTERPOLATED): Maximum value 261 (AVG_INTERPOLATED): Average 262 (SUM_INTERPOLATED): Sum 263 (COUNT_INTERPOLATED): Number of values Example: With TIMESTEP=60,257 , an interpolated value is returned for each 60 second period: "TAG:R,1,'2004-07-09 09:03:00.000','0000-00-00 00:10:00.000','TIMESTEP=60,257'"

Rule

Note the following:

- <TimeBegin> and <TimeEnd> may not both be "ZERO" = "0000-00-00 00:00:00.000".
- In order to read data, <TimeBegin> must come before <TimeEnd>. For reverse sorting, use the parameter "Order by-Clause" .
- To improve performance, use the "ValueID" in the query instead of "ValueName". You can get the "ValueID" from the "Archive" table.
- With process values, some applications cannot handle the time with a resolution of 1 ms and this can lead to inaccuracies.

Selecting an absolute time period

Read from start time <TimeBegin> to end time <TimeEnd>.

Example 1:

Reads the values of ValueID 1 from start time 9:03 a.m. to end time 9:10 a.m.

```
"TAG:R,1,'2004-07-09 09:03:00.000','2004-07-09 09:10:00.000'"
```

Selecting a relative time period

Reading from the start of the recording:

```
<TimeBegin> = '0000-00-00 00:00:00.000'
```

Reading to the end of the recording:

```
<TimeEnd> = '0000-00-00 00:00:00.000'
```

However, <TimeBegin> and <TimeEnd> may not both be "ZERO" = '0000-00-0000:00:00.000' .

Note

If you want to query a relative period from a connected archive database, enter this in the following format:

0000-00-DD hh:mm:ss.msc

If you specify a period in months, the content may be incorrect since a month can have between 28 and 31 days.

Example 2:

Reads starting from the absolute time of "TimeBegin" to the end of the recording, in other words, to the last archived value.

```
<TimeBegin> = '2003-02-02 12:00:00.000',
<TimeEnd> = '0000-00-0000:00:00.000'
```

Example 3:

Reads 10 seconds further starting from the absolute time of "TimeBegin" .

```
<TimeBegin> = '2003-02-02 12:00:00.000',
<TimeEnd> = '0000-00-00 00:00:10.000'
```

Example 4:

Reads 10 seconds back starting from the absolute time of "TimeEnd".

```
<TimeBegin> = '0000-00-00 00:00:10.000',
<TimeEnd> = '2003-02-02 12:00:00.000'
```

Example 5:

Reads the values of the last hour for several ValueIDs (1;3;5;6) , starting at the time of the last archived value.

```
"TAG:R,(1;3;5;6),'0000-00-00 01:00:00.000',
'0000-00-0000:00:00.000'"
```

Multiple return values for one query by filtering the tag value

Example 6:

The following query also uses the <SQL_Clause> parameter and returns all values of tags with the ValueID "3" and "6" that are below 50 or over 100.

```
"TAG:R,(3;6),<TimeBegin>,<TimeEnd>,'WHERE RealValue > 100 OR
RealValue < 50'"
```

Query with the <TimeStep>parameter

Example 7:

The following query uses the <TimeStep> parameter and returns all values of the ValueID "1" , starting at "TimeBegin" for the next 5 minutes at intervals of "60" seconds with aggregate function type "5" = "average value without interpolation".

```
"TAG:R,1,'2004-10-13 17:00:00.000','0000-00-00 00:05:00.000',
'TIMESTEP=60,5'"
```

Example 8:

The following query uses the <TimeStep> parameter and returns all values of theValueIDs "1" and "2" , starting at "TimeBegin" until 2 minutes later at intervals of "15" seconds with aggregate function type "261" = "average value without interpolation".

```
"TAG:R,(1;2),'2004-10-13 17:00:00.000','0000-00-00 00:02:00.000',
'TIMESTEP=15,261'"
```

7.8.4.3 Representation of the message archives

Structure of the recordset

The result of the query is returned as a Recordset. The following table describes the structure of the recordset for message archives. Information on the status of messages is available in the WinCC Information System in "Working with WinCC > ANSI-C Function Descriptions > Appendix > Structure Definitions > MSG_RTDATA_STRUCT Structure Definition".

Position	Field name	Type	Comment
1	MsgNo	Integer 4 bytes	Message number
2	State	Small integer 2 Byte	Message status

Position	Field name	Type	Comment
3	DateTime	DateTime 8 bytes	Time stamp of the message (date/time without milliseconds)
4	Ms	Small integer 2 Byte	Time stamp of the message (milliseconds)
5	Instance	VarChar(255)	Instance name of the message
6	Flags1	Integer 4 bytes	(for internal use only)
7	PValueUsed	Integer 4 bytes	Process values used
8 to 17	PValue1 to PValue10	Real 8 bytes	Numerical process value 1 to 10
18 to 27	PText1 to PText10	VarChar(255)	Process value text 1 to 10
28	Computer name	VarChar(255)	Computer name
29	Application	VarChar(255)	Application name
30	Comment	VarChar(255)	Comment
31	Username	VarChar(255)	User name
32	Counter	Integer 4 bytes	Continuous message counter
33	TimeDiff	Integer 4 bytes	Time difference to the "came in" status
34	Classname	VarChar(255)	Name of the message class
35	Type name	VarChar(255)	Name of the message type
36	Class	Small integer 2 Byte	Message class ID
37	Type	Small integer 2 Byte	Message type ID
38 to 47	Text1 to Text10	VarChar(255)	Message text 1 to 10
48	AG_NR	Small integer 2 Byte	Number of the PLC
49	CPU_NR	Small integer 2 Byte	Number of the CPU
50	CrComeFore	Integer 4 bytes	Foreground color for the "came in" status
51	CrComeBack	Integer 4 bytes	Background color for the "came in" status
52	CrGoFore	Integer 4 bytes	Foreground color for the "went out" status
53	CrGoBack	Integer 4 bytes	Background color for the "went out" status
54	CrAckFore	Integer 4 bytes	Foreground color for the "acknowledged" status
55	CrAckBack	Integer 4 bytes	Background color for the "acknowledged" status
56	LocaleID	Integer 4 bytes	Location of the alarm
57	Priority	Integer 4 bytes	Priority
58	AP_type	Integer 4 bytes	Loop in Alarm
59	AP_name	VarChar(255)	Loop in Alarm function name
60	AP_PAR	VarChar(255)	Loop in Alarm screen
61	InfoText	VarChar(255)	Info text
62	TxtCame	VarChar(255)	Text came in
63	TxtWent	VarChar(255)	Text went out
64	TxtCameNWent	VarChar(255)	Text came in and went out

Position	Field name	Type	Comment
65	TxtAck	VarChar(255)	Text acknowledged
66	AlarmTag	Integer 4 bytes	Message tag
67	AckType	Small integer 2 Byte	Acknowledgment type
68	Params	Integer 4 bytes	Parameter

7.8.4.4 Querying the message archives

Principle

Use the following query to access a message archive. The data can be selected using filter criteria. The query is transferred to the database with the command object.

Information on the status of messages is available in the WinCC Information System in "Working with WinCC > ANSI-C Function Descriptions > Appendix > Structure Definitions > MSG_RTDATA_STRUCT Structure Definition".

Syntax

```
ALARMVIEW:SELECT * FROM <ViewName>[WHERE <Condition>...., optional]
```

Parameter	Description
ViewName	Name of the database table. The table must be specified in the required language. The "ViewName" for the five European languages is, for example: ALGVIEWDEU: German message archive data ALGVIEWENU: English message archive data ALGVIEWESP: Spanish message archive data ALGVIEWFRA: French message archive data ALGVIEWITA: Italian message archive data **Note** The languages that are installed on the PCS 7 OS or configured in the PCS 7 OS Text Library are supported. Information on the possible query languages and the corresponding "ViewName" in the SQL server is available in the connected message archives in "Views". Here, all languages and their identifications, such as "ALGVIEWENU", are displayed that are supported in the particular archive.
Condition	Filter criterion e.g.: DateTime>'2003-06-01' AND DateTime<'2003-07-01' DateTime>'2003-06-01 17:30:00' MsgNo = 5 MsgNo in (4, 5) State = 2 With DateTime, the time can only be specified in absolute format.

Example 1:

Reads all entries of message number 5 that were recorded after 5 July 2003.

```
"ALARMVIEW:SELECT * FROM ALGVIEWENU WHERE MsgNo = 5 AND
DateTime>'2003-07-05'"
```

Example 2:

Reads all messages with the time stamp between 3 July 2003 and 5 July 2003.

```
"ALARMVIEW:SELECT * FROM ALGVIEWENU WHERE DateTime>'2003-07-03' AND
DateTime<'2003-07-05'"
```

Appendix A

A.1 Commissioning

Commissioning of a connection from OPC client to the OpenPCS 7 station

The following list contains the key tests for commissioning a connection from the OPC client to the OpenPCS 7 Station:

1. Check that all necessary OpenPCS 7 licenses are available and valid.
2. Check that the OpenPCS 7 station can be reached from the OPC client computer using a ping.
3. Check whether network access to the path "\\<OpenPCS 7 Station>\Automation Projects" is possible and whether the following files exist:
   - SPOSA.dcf
   - <Project_name_PC_station_name>.dcf
4. Check that the OS servers and automation systems are working properly.
5. Check the installation and version of the OpenPCS 7 station.
6. Check the installation of the OPC client. Information on the correct installation of the OPC client must be supplied by the vendor of the OPC client.
7. Check the Windows firewall settings. The standard settings are made during installation. The following OpenPCS 7 components must be entered in the "Exceptions" tab:
   - OPC UA Discovery
   - SIMATIC PCS7 OPC UA Server
   - OPC Server Enumerator
   - SIMATIC PCS7 OPC AE Server
   - SIMATIC PCS7 OPC DA Server
   - SIMATIC PCS7 OPC HDA Server
8. Based on the following tables, check that the required processes are executed on the OpenPCS 7 station during OPC client runtime.

Process on the OpenPCS 7 station with OPC UA

Process	Information on the process
opcualds.exe	Window service that is started automatically by the operating system. This process is not absolutely necessary. It is used to determine the OPC servers available on this station. An OPC server on this station can be addressed directly.
OpcUaServerOpenPCS7.exe	Window service that is started automatically by the operating system. If the service is in "Paused" state, this can have the following causes: • The required license key cannot be identified. • No project dcf is available

Processes on the OpenPCS 7 station with OPC Classic

Process	Information on the process
OpcEnum.exe	Windows service that is started by an OPC client during the determination of the OPC server list. This process is not absolutely necessary. It is used to determine the OPC servers available on this station. An OPC server on this station can be addressed directly.
SOPCAESVRPCS7.exe	This process is started by the coupling with an OPC client.
SOPCDASVRPCS7.exe	This process is started by the coupling with an OPC client.
SOPCHDASVRPCS7.exe	This process is started by the coupling with an OPC client.

Lists and folders

B

B.1 List of sources

Sources

[1] OPC Foundation, OPC Unified Architecture Release 1.02, Parts 1..11

[2] OPC Foundation, Data Access Custom Interface Specification 3.0

[3] OPC Foundation, Data Access Automation Interface Standard 2.02

[4] OPC Foundation, Alarms and Events Custom Interface Standard 1.10

[5] OPC Foundation, Alarm & Events Automation Interface Standard 1.01

[6] OPC Foundation, Historical Data Access Specification 1.20

[7] OPC Foundation, Historical Data Access Automation Interface Standard 1.0

B.2 List of abbreviations/acronyms

List of abbreviations/acronyms

Abbreviation/acronym	Description
APC	Advanced Process Control
AS	Automation System
CAL	Client Access License
COM	Component Object Model
CPU	Central Processor Unit
DA	Data Access
DCOM	Distributed Component Object Model
DLL	Dynamic Link Library
ES	Engineering system
HDA	Historical Data Access
HMI	Human Machine Interface
HW	Hardware
LRPC	Lightweight Remote Procedure Call
OLE	Object Linking and Embedding
OLE DB	Object Linking and Embedding for Data Base
OPC	previously: OLE for Process Control, now: Openness, Productivity and Collaboration
OPC UA	OPC Unified Architecture

Abbreviation/acronym	Description
OS	Operator Station (PCS 7 OS) or Operating System
PC	Personal Computer
PCS 7	Process Control System 7
PH	Process Historian
RPC	Remote Procedure Call
SFC	Sequential Function Chart
SP	Service pack
SPOSA	Single Point of System Access
SQL	Structured Query Language
WinCC	Windows Control Center

Index

D

O