



**OPC Bridging Transfers Data between
Industrial Automation Systems**
OPC Training Institute

OPC Training Institute
Tel: 1-780-784-4444 | Fax: 1-780-784-4445
Web: www.opcti.com | Email: info@opcti.com

Copyright © 2009 OPC Training Institute (OPCTI). All rights reserved. The information contained in this document is proprietary to OPCTI. No part of this document may be reproduced, stored in a retrieval system, translated, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without prior written permission from OPCTI.

Introduction

Integrators frequently use OPC technology to connect one Industrial Automation system (PLC, DCS, SCADA, HVAC, etc) with another so data can be shared between the two systems. Because OPC technology is based on the Client/Server architecture, the challenge is that two OPC Servers cannot communicate with each other directly. A variety of vendors provide an intermediate software solution, generically called an “OPC Bridge,” to facilitate this sort of communication. This whitepaper discusses the concept of the OPC Bridge, the solution architecture, required software components, and various features to help Integrators differentiate between different OPC Bridge products.

OPC Overview

OPC is an industrial communication standard that enables manufacturers to use data to optimize production, make operation decisions quickly and generate reports. OPC enables plants to automate the transfer of data from a control system (PLC, DCS, analyzer, etc) to an industrial software application (HMI, Historian, Production system, Management system, etc). OPC is typically found in Level 3 networks and higher. Thus, OPC transfers process control data between the Control (Level 2) network and the Operations/Manufacturing (Level 3) network. It also exchanges data between the Operations/Manufacturing network and the Business (Level 4) network. In essence, OPC is the Modbus of the new century. It is not a replacement for low-level communication standards such as 4-20mA, Hart, Profibus, or Foundation Fieldbus. Rather, organizations use OPC in high-level communication.

Note: OPC is no longer an acronym. When OPC first released in 1996 it was an acronym for OLE for Process Control, and was restricted to the Windows operating system. OPC is now available on other operating systems and enjoys significant adoption outside of process control. So, the original name (OLE for Process Control) is no longer appropriate and OPC changed from an acronym to a word. Thus, OPC no longer stands for anything (OPC is just OPC).

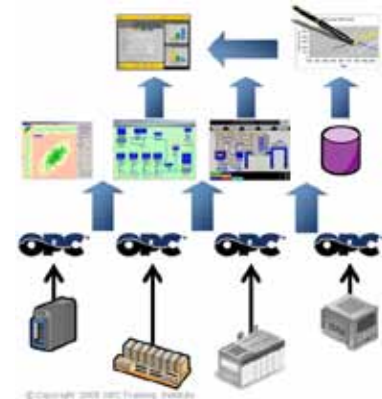


Image 1: OPC communication enables applications to interoperate and simplifies system architecture.

Connectivity Challenge

Successful automation of industrial systems often needs to connect various disparate systems, or islands of automation, together. The following are some common examples of system interconnectivity:

- Send data from a PLC to a DCS
- Connect disparate PLCs together on a single network
- DCS “Console Replacement”
- Connect an Emergency Shutdown System (ESD) with a DCS
- Send data from one plant area/unit to another for feed forward or feedback control
- Bring Heating Ventilation and Air Conditioning (HVAC) data into an Energy Management System (EMS)
- Augment an existing SCADA (Supervisory Control and Data Acquisition) system with a new data source

In the past, plants would achieve this system connectivity using hardwire (4-20mA) connections. But these were often extremely expensive, especially if the plant needed to transfer large quantities of data. Vendors then came up with less expensive options such as serial interfaces or various proprietary networks. However, the proprietary nature of these solutions kept installation/maintenance costs high and often was unable to connect all systems. Fortunately, the emergence of OPC provides a significant improvement to these solutions.

Since 1996, OPC has become the de facto standard for Integrators to connect industrial automation systems. OPC Servers are available for every major control system vendor and the list of available OPC software continues to increase on a monthly basis. With OPC providing the communication backbone, connectivity is easier to establish, provided that Integrators keep a few key OPC architectural concepts in mind.

Bridging Concepts

It is not possible for two OPC Servers to communicate with each other “out of the box” unless the vendor added non-OPC Server capabilities to the OPC Server software. OPC is based on Client/Server architecture. Therefore, an OPC Client application (HMI, Historian, Production system, Management system, etc) makes requests to exchange data with an OPC Server. The OPC Server responds to these requests. But, note that an OPC Server can only respond to requests; the OPC Server cannot make requests. In other words, OPC Servers can only do what they are told. So, two OPC Servers cannot exchange data with each other directly because each waits for a request from the other, and since OPC Servers don’t make requests, no data transfer will take place.



© Copyright 2009 OPC Training Institute

Image 2: The only way for two OPC Servers to communicate is to have an OPC Client application provide a bridge between the two servers, thus becoming an “OPC Bridge.”

This is true for any Client/Server based system architecture in general and OPC specifically.

The only way for two OPC Servers to communicate is to have an OPC Client application provide a bridge between the two servers, thus becoming an “OPC Bridge.” The OPC Bridge connects two or more OPC Servers together. The OPC Bridge requests data from one OPC Server, and writes the data to another OPC server. All OPC Bridge products follow this concept. Integrators must take three steps to setup the data transfer:

- a) Setup the OPC Bridge to connect to the OPC servers. The OPC Servers to which the bridge connects are preset by the Integrator.
- b) Setup the OPC Bridge to read data from one OPC Server, and write it to another OPC Server. The Integrator preselects the specific data to transfer.
- c) Setup the OPC Bridge to automatically execute the above configuration.

After the initial setup, most OPC Bridge applications are set to execute automatically with little to no human intervention. This enables data to transfer on a regular basis, just as if there were a hard-wire interface between the two systems.

Connecting to OPC Servers

The first step to setting up an OPC Bridge is connecting to the OPC Servers. OPC Bridge products typically enable Integrators with the following ways to connect to OPC Servers:

- a) Graphical User Interface (GUI) enables Integrators to graphically browse the local PC and other PCs on the network.
- b) Manual Configuration enables Integrators to get quick access to known OPC resources.

Most OPC Bridge applications provide both methods above and enable Integrators to select the most efficient way to establish the OPC Server connection. While a GUI is certainly easier for beginners to understand, the Manual Configuration method is preferred by experienced users.

Note: Establishing the first connection to OPC Servers is usually the most difficult step when using OPC, especially if the OPC Server resides on a remote PC.



Image 3: Cogent OPC DataHub enables Integrators to connect to local and remote OPC Servers. This OPC Bridge software provides a useful mixture of a GUI and manual configuration to help beginners and advanced users alike. Many options are easily accessible as checkmarks, which provides flexibility for custom projects.

Refer to section titled “Additional References” available on the OPC Training Institute website for troubleshooting information.

Mapping OPC Items

After making the initial connection to the OPC Servers, Integrators must select the source and destination of each data item. OPC Bridge applications typically enable Integrators to accomplish this as follows:

- a) Graphical interface enables Integrators to graphically select each source item and the destination of the data.
- b) Importing data files enables Integrators to create a configuration file using an application such as Microsoft Excel and import the file into the bridge for mass configuration.



Image 4: Kepware LinkMaster provides an intuitive drag-and-drop GUI that enables Integrators to quickly map data from its source to destination. Integrators can also make use of an import file for mass configuration.

😊 It may seem the graphical configuration method is an efficient way to configure the OPC Bridge; but, consider that most data transfers move hundreds and even thousands of items from one OPC Server to another. Setting up each data item individually is both time-consuming and error prone. Therefore, if you must move a lot of data, I recommend selecting an OPC Bridge that enables you to create standard configuration files, such as a Comma Separated Value (CSV) file, with a mass-configuration tool such as Microsoft Excel.

OPC Bridge Execution

Once the Integrator sets up the OPC Bridge, they must automate its execution. This automation comes in a few different flavors (listed in increasing desirability):

- a) Standard Windows application with explicit startup: The OPC Bridge starts only when someone explicitly selects it (i.e. double-click). This startup mode may be convenient for testing purposes, but not for production because this method requires manual intervention to initiate the operation of the OPC Bridge.
- b) Standard Windows application with automatic startup: The OPC Bridge starts when a specific user logs on to Windows. While this method does not require manual intervention, it does require a specific logon account.



Image 5: Cyberlogic Crosslink executes as a Windows Service. Crosslink’s configuration enables users to bypass DCOMCNFG to set the Server’s execution either as System, or a specific User Account.

- c) Windows Service: A Windows Service is a specialized application that is setup to execute with the “System” identity. In other words, it executes as the Operating System itself. This is the most desirable method to run the OPC Bridge. This application can automatically start as soon as Windows boots up and does not require a specific user to log on to the PC. The OPC Bridge is then able to immediately begin transmitting data with minimum interruption. Execution as a Windows Service is desirable even if the project requires specific User Accounts (with a User Name and Password combination) to connect to remote PCs.

👉 Most OPC Bridge applications can execute in all of the modes listed above, however, when selecting an OPC Bridge, I recommend that you choose a product that can run as a Windows Service.

Key OPC Bridge Considerations

Performance

The performance of OPC Bridge applications varies from vendor to vendor. However, most can easily out-perform even the fastest Automation networks (i.e. the connection between a PLC and PC, also known as ISA Level 2 networks). Specifically, connections to a DCS typically provide data in the range of about 100-500 points per second. Even connections to a PLC rarely exceed 2,000 points per second. However, OPC can transfer data at over 10,000 points per second.

👉 While an OPC Bridge can technically transfer data at over 10,000 points per second, its performance is usually limited by its connection to its various data sources. To put the performance of the OPC Bridge in the right perspective, I recommend you find out the speed of data from the original data source. This will help you to make more sense out of comments from various OPC vendors.

Multithreaded operation

It is important for an OPC Bridge to continue operations even when an acceptable fault in the system occurs. Consider an OPC Bridge that transfers values from two data sources (i.e. two OPC Servers) to a single destination (i.e. a third OPC Server). If one of the two data sources (i.e. one of the OPC Servers) stops operating or is delayed, it may still be necessary for the OPC Bridge to continue transferring information from the good data source to the destination.

The ability to wait for one operation to complete while the other continues requires a multithreaded design for the OPC Bridge. Unfortunately, most Windows applications are single threaded; in other words, they only do one task at a time.

🧐 When selecting an OPC Bridge, ask your vendor whether or not their OPC Bridge uses a multithreaded design when connecting to OPC Servers. This will enable your system to withstand faults without compromising the entire health and performance of the system.

Cache reads upon reconnection

OPC Client applications (such as an OPC Bridge) receive data updates from OPC Servers whenever the OPC Server detects a change in the data source. OPC Servers transmit data that changes at various rates: Some readings, like pressures and flows, change at high rates (many times per second), while other readings, like Setpoints, change at a very low rate (once per week, or never).

For example, the Setpoint for room temperature is (typically) set once and left alone. Suppose that an OPC Bridge disconnects from an OPC Server that has the room temperature Setpoint, and suppose the OPC Bridge then reconnects. In this case, the Setpoint does not change, so the OPC Server will not report a change in value. If the OPC Bridge does not read the value and quality of the Setpoint when it reconnects to the OPC Server, the OPC Bridge will never report the true value and quality of the temperature Setpoint. This is because the OPC Server sends values to the OPC Bridge when the values change, and the Setpoint does not change, so the OPC Server does not send an update.

🧐 When an OPC Bridge reconnects to an OPC Server, it should immediately perform a Cache Read from the OPC Server. This will ensure that the OPC Server will quickly pass all its available values. This transfer will take place quickly because a Cache Read tells the OPC Server that the OPC Bridge requires values that are already in the memory of the OPC Server. A Cache Read explicitly tells the OPC Server that the OPC Client (i.e. the OPC Bridge) only requires values that are already in the Servers address space (i.e. in the OPC Server's memory). This will ensure that no data ever displays incorrectly.

Bad Quality data


OPC uses a Quality identifier to inform users of the validity of the data in each OPC Item (or point/reading). For example, suppose an OPC Item contains the value of a flow reading. The value of the Flow reading could be questionable if any of the following occur:

- There is no reading at all
- Reading is out of range
- PLC disconnects from flow meter
- OPC Server disconnects from the PLC
- Etc.

Thus, in addition to providing a value for the flow itself, OPC also provides an associated Quality to describe the validity of the data, or a description of the reason the data is questionable. This helps Users troubleshoot the system.


The OPC Bridge may also be required to pass this value along. In this case, the OPC Bridge may be required to write the data as well as the Quality of the data. So, in addition to the standard OPC Server Quality values, the OPC Bridge must also inform the OPC Server at the destination of additional troubleshooting information such as:

- The OPC Bridge itself is disconnected from the source OPC Server
- The OPC Bridge is unable to transmit data
- The OPC Bridge received a bad calculation
- The OPC Bridge is overloaded
- Etc.

 I recommend selecting an OPC Bridge that enables the Integrator to select how the OPC Bridge passes OPC Quality information. The OPC Bridge should also pass along additional OPC Quality values in case the OPC Bridge itself has a problem with the data.

Calculations

Integrators often use OPC Bridges to perform calculations and transformations on incoming data. For example, an OPC Bridge could take a 4-20mA reading from a PLC, and transform it to a voltage that is between 0 to 240 volts.

 I recommend selecting an OPC Bridge that enables you to perform calculations on incoming data. In case of an error, the OPC Bridge should notify the destination OPC Server of the problem (typically this would be done using the OPC Quality identifier).

Available OPC Bridge Products

The following is a list of the most popular OPC Bridge software products on the market today. I encourage you to send me information on missing OPC Bridge products. I will modify this list as I become aware of additional OPC Bridges. Most of these products are available on the OPCTI website for a free download.

Company	Product Name	Website
CAS	DataPorter	www.cas.eu
Cogent	OPC DataHub	www.cogent.ca
Cyberlogic	OPC Crosslink	www.cyberlogic.com
Emerson	OPC Mirror	www.emerson.com
Honeywell	OPC Integrator	www.honeywell.com
Iconics	DataWorX32	www.iconics.com
Kepware	LinkMaster	www.kepware.com
Northern Dynamic	OPC Gateway	www.nordyn.com
Open Automation Software	OPC Route.NET	www.opcsystems.net
SAE Automation	OpcDbGateway	www.saeautom.sk

Additional References

The OPC Training Institute has various other whitepapers to help you troubleshoot and learn more about OPC. These are available on the OPCTI website at www.opcti.com. If these are of interest, you can also consider attending one of OPCTI's many hands-on OPC courses where you get the opportunity to benefit from experts' context, learn new concepts, and immediately apply them in live connectivity scenarios and troubleshooting activities.

OPC & DCOM: 5 Things You Need to Know

This whitepaper discusses the five steps to a simple and effective strategy to establish reliable DCOM communication. In addition, the whitepaper covers troubleshooting tips to identify common OPC and DCOM problems, their symptoms, causes, and how to solve them. This will help integrators set up reliable and secure OPC connections.

OPC & DCOM Troubleshooting: Quick Start Guide

OPC is powerful industrial communication standard. However, OPC relies on having DCOM work properly. Luckily, DCOM problems can usually be overcome with relatively simple configuration changes as documented in this whitepaper.



The OPC Training Institute's website is full of helpful whitepapers and additional resources to help you solve common OPC configuration problems and learn more about OPC.

Cannot browse for OPC Servers on remote PC

Find out the reasons an OPC application is unable to browse for OPC Servers on a remote PC, and how to overcome them. The cause of the problem is typically poor configuration. However, it can also be caused by limitations in the specific software in use. This whitepaper explains the problem, its causes and symptoms. The whitepaper also details a structured and practical step-by-step approach to determine the exact cause and solution.

OPC Error: Failure to obtain a CLSID

Find out how to overcome the specific OPC error of "Failure to obtain a CLSID". The cause of the problem is typically poor configuration. However, it can also be caused by limitations in the specific software in use. This whitepaper explains the problem, its causes and symptoms. The whitepaper also details a structured and practical step-by-step approach to determine the exact cause and solution.

0x80040202 DCOM Error

0x80040202 error appears in the OPC Client application when it fails to receive a callback from the OPC Server. Find out about the various circumstances that can trigger this error.

0x80070005 DCOM Error

DCOM Error 0x80070005 appears in the OPC Client application when it succeeds in launching an OPC Server or OpcEnum, but fails to receive a reply from either of the applications. This error could be caused under several conditions.

0x800706BA DCOM Error

The 0x800706BA DCOM error appears in the OPC Client application when the OPC Client "believes" that it has a live connection to the OPC Server, but truly does not. This can happen under several conditions.

OPC Bridging Transfers Data between Industrial Automation Systems

Integrators frequently use OPC technology to connect one Industrial Automation system (PLC, DCS, SCADA, HVAC, etc) with another so data can be shared between the two systems. Because OPC technology is based on the Client/Server architecture, the challenge is that two OPC Servers cannot communicate with each other directly. A variety of vendors provide an intermediate software solution, generically called an "OPC Bridge," to facilitate this sort of communication. This whitepaper discusses the concept of the OPC Bridge, the solution architecture, required software components, and various features to help Integrators differentiate between different OPC Bridge products.