16055 Space Center Blvd.
Suite 450
Houston, TX 77062
Phone: 281-286-6017
Fax: 281-286-6902

AutoSol's OPCMessenger White Paper

The maturity of low cost computing systems and the acceptance of these technologies into mainstream automation products have introduced computing standards within the automation industry. However, communication between different vendor components in automation systems, and between automation systems and business systems, continues to be problematic. These issues are particularly acute in SCADA systems that address applications which are widely distributed over large geographically areas. The traditional problems of time and distance are compounded by many factors, such as the incompatibility of legacy communication protocols and the maintainability of distributed databases and control systems. Today's corporate strategy of growth through acquisitions often necessitates the consolidation of equipment and systems communicating in various protocols, both current and legacy. Consolidation is difficult however in an atmosphere of reduced budgets and restricted capital expenditures which is typical following most corporate acquisitions. As a result, IT and Automation departments are restricted from replacing incompatible components. Today there is a simple, economical solution, which is OPCMessenger.

As with computing standards in the past, the inclusion of standards evolving from the Internet into automation systems has the potential to change industrial automation. Products that merge new standards with legacy communication protocols and methods, in addition to industrial automation standards for uninterrupted service and scheduling, will eliminate proprietary and compatibility barriers providing options to upgrade legacy systems. The OPCMessenger addresses these issues by bridging on-line transports that are standards in the automation industries with transports that have been developed in the IT and Internet industries.

Product Description

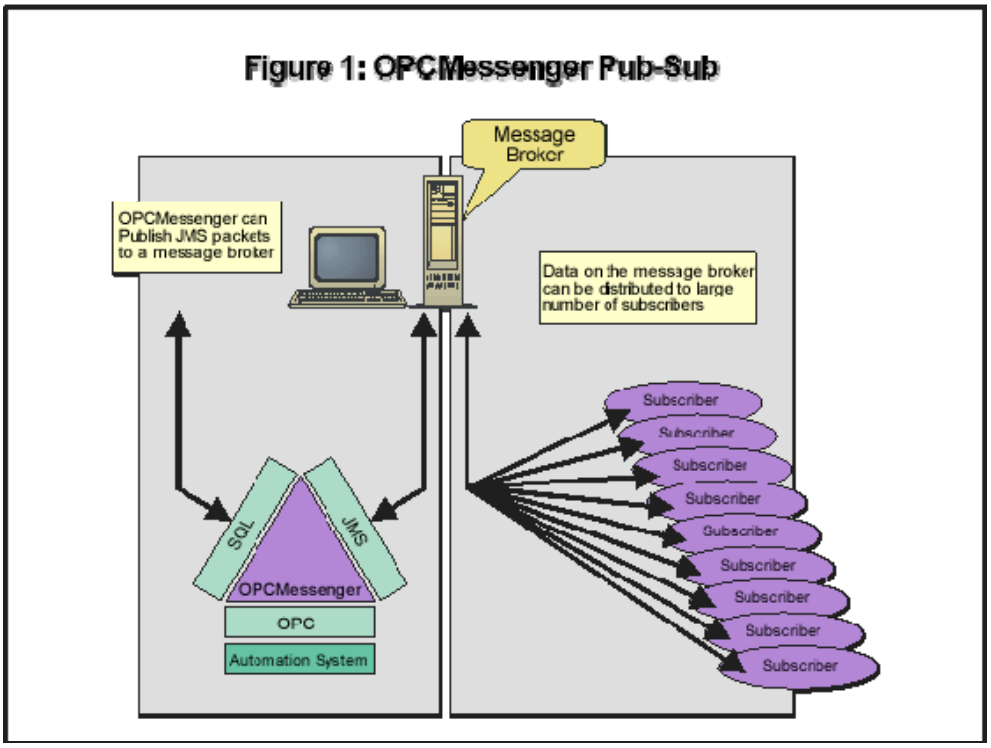
Automation Solution's OPCMessenger employs standards from the industrial automation industries and standards from the IT and Internet industries to provide a data delivery tool that can overcome traditional barriers to the gathering and distribution of data. Standards that OPCMessenger supports include;

- Uninterrupted service through redundant systems and communication paths,
- Data processing and data delivery that can be triggered on a schedule or on event,
- Support for industrial on-line protocols such as OPC (OLE for Process Control), and

Support for IT and Internet transports including Publish-Subscribe technologies or SQL.

THE PROBLEM: Traditionally, SCADA systems had a many-to-few approach in which data delivery was focused on a few recipients, users or systems. SCADA technologies are not appropriate to deliver information to thousands of users.

THE SOLUTION "OPCMessenger Pub-Sub": Bridging an industrial transport such as OPC and Publish-Subscribe transports such as those provided by the JMS standard enables a many-to-many data delivery model as illustrated in Figure 1.

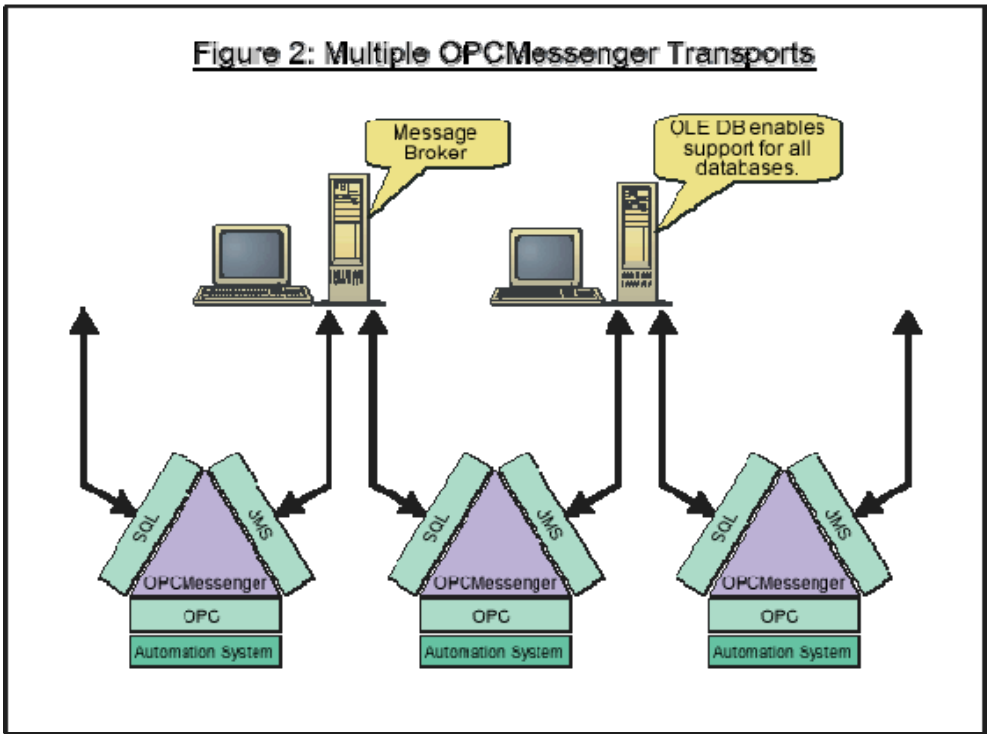


THE PROBLEM: Synchronizing geographically distributed automation systems and databases manufactured by different vendors, having different protocols, for the purpose of enhanced data security, to provide operational back-up, or to share data between corporations is not adequately supported in traditional SCADA systems.

THE SOLUTION "Multiple OPCMessenger Transports: Data delivery through multiple OPCMessenger installations (Figure 2) can synchronize widely distributed automation systems, databases, users or any combination including:

- OPCMessenger acting as a Redundancy Server can be inserted between an OPC Client such as a HMI and an OPC Server on two different systems to synchronize the systems and switch communication paths to enable one system to back-up the other.
- Remote automation systems utilizing either an OPC Server or an OPC Client interface can be synchronized by communicating through a JMS message broker via the Internet.
- Remote databases can be synchronized by communicating through a JMS message broker via the Internet.

An automation system that has either an OPC Server or an OPC Client interface and a remote database can be synchronized by communicating through a JMS message broker via the Internet.



Additionally, data that traverses OPCMessenger can be further processed for control, data acquisition, event detection and notification, and data management purposes by the OPCMessenger Application Objects. The Application Objects provide a development and runtime environment in which the user can develop and run processing blocks. Once a processing block has been developed, the bock can be scheduled for execution in the runtime environment on a periodic of event basis.

Base Product

The base OPCMessenger has an OPC Client and an OPC Server interface and can be enhanced through Add-in modules to include IT and Internet transports. The OPC Client-Server interface in the base product enables OPCMessenger to interface to any automation system that supports OPC. Additionally, through these OPC interfaces OPCMessenger can be configured as a Redundancy Server to provide uninterruptible data acquisition to an OPC client application.

When used as a Redundancy Server the OPCMessenger is inserted between an OPC Client such as an HMI, and external OPC Servers to provide a “hot back-up” scheme as shown in Figures 3 and 4.

Figure 3 illustrates a "hot back-up" strategy in which there is a Primary system and a Back-up system that is constantly synchronized with the Primary under normal circumstance. Should the Primary system fail however, OPCMessenger can switch the communication paths to allow the Back-up system to assume the Primary's responsibilities.

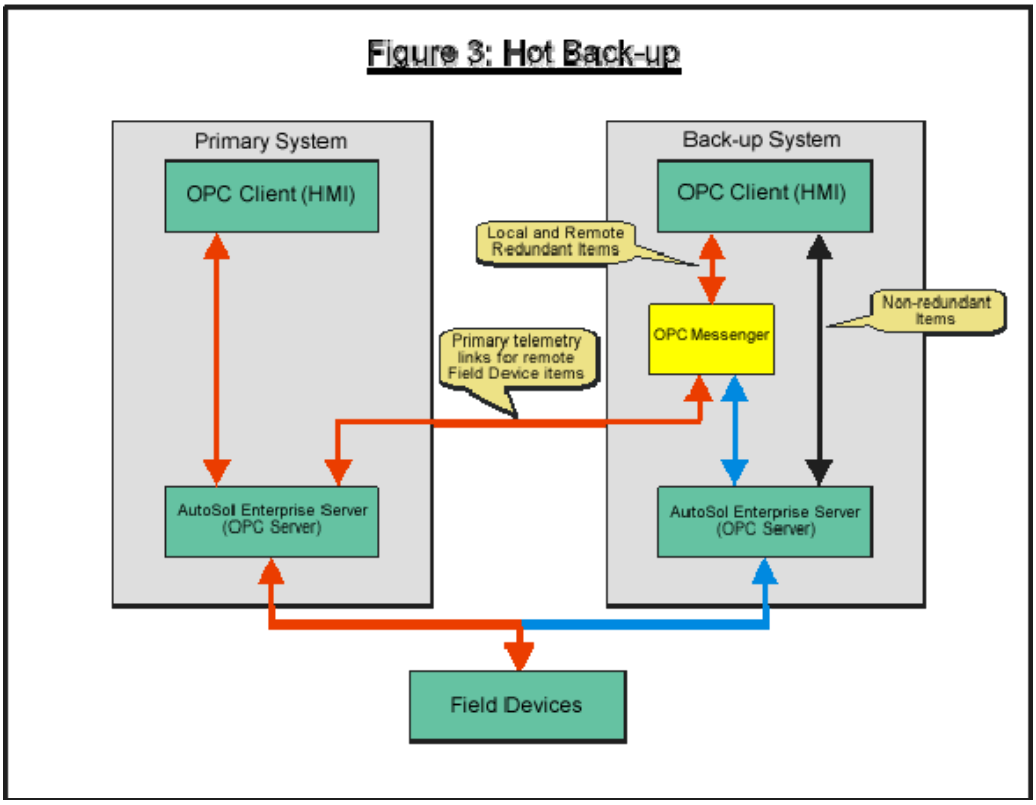
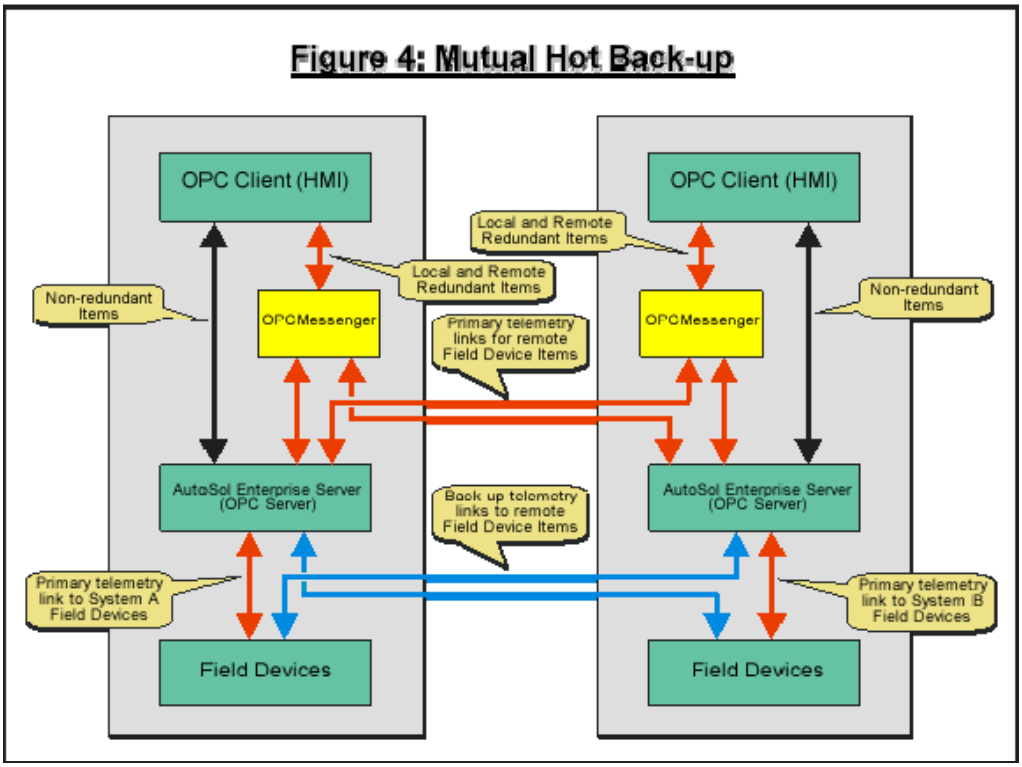


Figure 4 illustrates a "mutual hot back-up" system in which two normally autonomous SCADA systems back-up each other. Under normal circumstances each SCADA system sources the data necessary for its primary mission through its own OPC Server. Additionally, each SCADA system sources the data for its back-up mission through its companion's OPC Server. The latter connection insures that the two systems will be always be synchronized. If failure of one of the systems occurs, the OPCMessenger on the companion system will automatically switch communication paths to receive all data through its own OPC Server.



Ordinarily, the addition of a software layer to support redundancy adds complexity to a system which impacts system maintenance. However, OPCMessenger has an Auto-configuration feature that eliminates the additional database configuration that is required by many redundancy schemes.

Product Add-ins

The OPCMessenger is extensible to address new transports through SonicMQ JMS, Websphere MQ and SQL Add-in modules. In addition, an Application Object Add-in module can provide a processing environment within OPCMessenger for data management, event detection and response, process control, and data acquisition purposes. These Add-in modules can be purchased as needed to expand the product.

SonicMQ JMS Add-in

The SonicMQ JMS Add-in allows data acquired through other OPCMessenger interfaces to be written to, “Published”, or read from “Subscribed” from a SonicMQ message broker. The message broker can be remotely located some where on an Intranet or the Internet. The data that is published can be acquired through other OPCMessenger interfaces such as the OPC Client, OPC Server, or other Add-in interfaces. The data obtained by subscribing to a message broker can also be delivered to other systems through the OPCMessenger interfaces.

The JMS transport is more appropriate than OPC for delivery of data over Intranets and the Internet which can experience long latencies and where data recipients are not always connected. In addition, JMS packets can generally pass through corporate firewalls that would typically block OPC (OPC is based on Microsoft's DCOM technologies.) messages.

AutoSol MQ Add-in

The AutoSol MQ Add-in was developed by AutoSol as a low-cost replacement Message Broker for small applications that don't have the same requirements as a large scale corporate distributed network. Using the ASMQ (AutoSol Message Queue data can still be published and subscribed from the message broker. Data can be handle quickly and efficiently. Additionally, features (such as re-connect timeouts) have been added that suit the needs of our customer's particular industries.

SQL Add-in

The SQL Add-in extends the Publish-Subscribe paradigm to SQL databases. Through the Add-in module data can be written or published to specific database fields/columns on an event or periodic basis. In addition, the Add-in can subscribe to receive data from a database field/column periodically. The SQL Add-in supports all leading database products through OLE DB. In addition, multiple database products can be supported simultaneously from one OPCMessenger installation.

Application Objects Add-in

Data that is acquired through any transport can be processed through the OPCMessenger Application Objects Add-in for retransmission through another transport. The Application Objects Add-in provides a development and runtime environment in which processing blocks can be developed and scheduled for execution. The processing blocks can be developed for process control, data acquisition, event detection, event notification and data management purposes. The Application Objects basis processing unit is a Function Block that can be scheduled for processing on a periodic basis or on an event in OPCMessenger. The functionality of a Function Block is developed in a Visual Basic Scripting Language.