Foxboro Evo™ Process Automation System

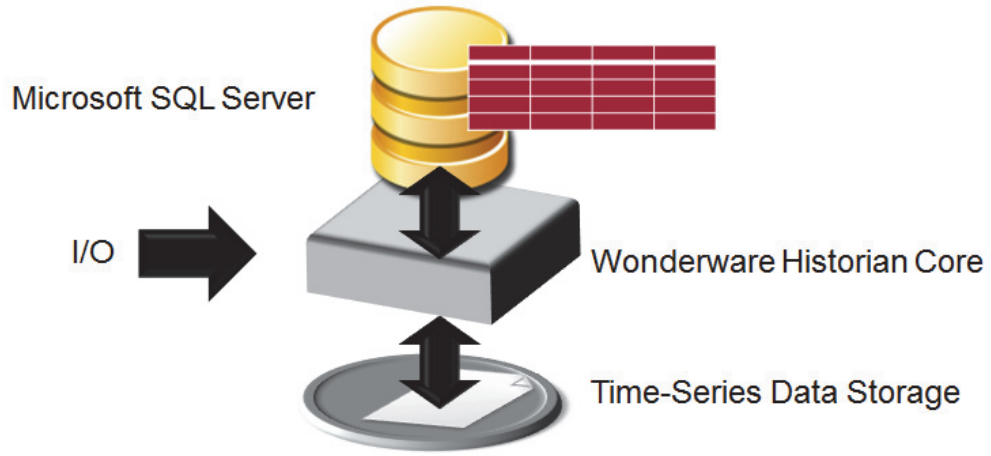
Product Specifications

Foxboro®

by Schneider Electric

PSS 31S-10WWHIST

Wonderware® Historian



The Wonderware® Historian is an integrated database that captures and stores real time data for future analysis and display. Data may be retrieved and displayed in many ways.

OVERVIEW

Wonderware Historian is designed to collect a wide variety of plant data, at full resolution and very high data rates, ensuring that decision-makers at all levels have the data they need to drive vital productivity improvement initiatives. The Wonderware Historian is hundreds of times faster than standard database systems and saves data in a small fraction of the space. Advanced data retrieval modes enable plant personnel to quickly generate the detailed, focused information needed to accelerate the decision-making process and provide access to the right information when a problem is identified or an opportunity is uncovered.

The Wonderware Historian is made up of specialized subsystems, which work together to manage data as it is acquired or generated, stored, and retrieved, as follows.

- Configuration Subsystem
- Data Acquisition Subsystem
- Data Storage Subsystem

- Data Retrieval Subsystem
- Event Subsystem
- Replication Subsystem

FEATURES

The Wonderware Historian is a real-time relational database that stores plant data. The historian acquires and stores process data at a full or specified resolution and provides real-time and historical plant data together with configuration, event, summary, and associated production data to client applications on the desktop. The historian combines the power and flexibility of Microsoft SQL Server® software with the high speed acquisition and efficient data compression characteristics of a real-time system.

Wonderware Historian is a component of the ArchestrA System Platform, and is also available as a stand-alone product.

Wonderware Historian:

- Acquires and stores plant data from high speed Wonderware I/O Servers, DAServers, ArchestrA-based systems and other devices.
- Compresses data for storage efficiency.
- Responds to SQL requests to retrieve plant data.
- Consolidates and summarizes data based on user defined criteria.
- Optionally sends raw or consolidated data to a second “tier-2” Historian.
- Uses commercial off-the-shelf software (Microsoft SQL Server) and high-performance data storage files, enabling open access to information.

BENEFITS

Wonderware Historian offers substantial productivity and analysis capabilities for process engineering and management personnel.

- Wonderware Historian is simple to configure. Using Wonderware Historian in a System Platform based application is as simple as checking a box during configuration. Integrating an InTouch® application tag database is a matter of a few mouse clicks.
- Wonderware Historian permits scalability from extremely small databases (32 points or less), to databases with up to 150,000 data points.
- Wonderware Historian is configurable in a two-tiered architecture, enabling multiple smaller tier-1 Historians to aggregate and consolidate information to a single tier-2 system.
- A highly reliable, disaster-proof system is easy to configure using replication services from tier-1 systems to tier-2 servers.
- Wonderware Historian, based on Microsoft SQL Server, can be configured to comply with enterprise-level security policies.
- Wonderware Historian Client as an available analysis toolset enables detailed drill-down analysis and reports to be constructed using Microsoft® Office components Word and Excel®.
- Wonderware Historian data can be accessed using Wonderware Information Server, a web-based portal.
- Using the Wonderware System Management Console, Wonderware Historian architectures can be easily designed and configured.

SYSTEM FUNCTIONALITY OVERVIEW

Configuration of a Wonderware Historian system involves defining information about elements that make up the Wonderware Historian, such as tag definitions, I/O Server definitions, and storage locations for historical data files. Configuration data is relatively static and does not change frequently during normal plant operation. The configuration subsystem stores and manages configuration data.

Configuration is performed mainly using the Management Console.

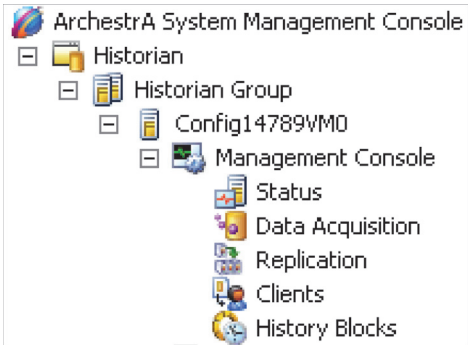


Figure 1. System Management Console

The Wonderware Historian Data Acquisition subsystem is designed for high-speed acquisition of data, acquiring and storing process data many times faster than a traditional relational database. Wonderware Application Server, DAServers, and I/O Servers are the main sources of plant data. Servers that use the SuiteLink or ArchestrA MX protocols can provide time and quality stamping at the server level. Data acquired from supported devices can propagate time stamping information from the data source.

The Wonderware Historian storage subsystem saves plant data from various sources to disk. The storage subsystem stores data for analog, analog summary, discrete, state summary, string, and system tags in sets of files on disk called history blocks.

Historical data can be retrieved by sending SQL queries through the Wonderware Historian OLE DB provider, which is part of the data retrieval subsystem. The Wonderware Historian data retrieval subsystem receives these SQL queries from clients, locates the requested data, performs necessary processing, and then returns the results. For configuration and event data, retrieval is made possible by normal SQL queries, because these types of data are stored in standard SQL Server database tables. Historical data, is retrieved from history blocks and then sent to clients as if it were stored in SQL Server tables.

The Wonderware Historian event subsystem detects events that have occurred in history and can execute an associated action upon detection. At a basic level, anything that can be determined by examining stored data can be used as an event. The event subsystem is configured to periodically check to see if an event occurred.

Data from one Wonderware Historian can be replicated to one or more other Wonderware Historians, creating a “tiered” relationship between the historians. Wonderware Historians can be configured in a variety of tiered architectures. In a common configuration, data from multiple individual historians (called tier-1 historians) is fed into a single centralized historian (called a tier-2 historian). A typical tiered architecture is shown in Figure 2.

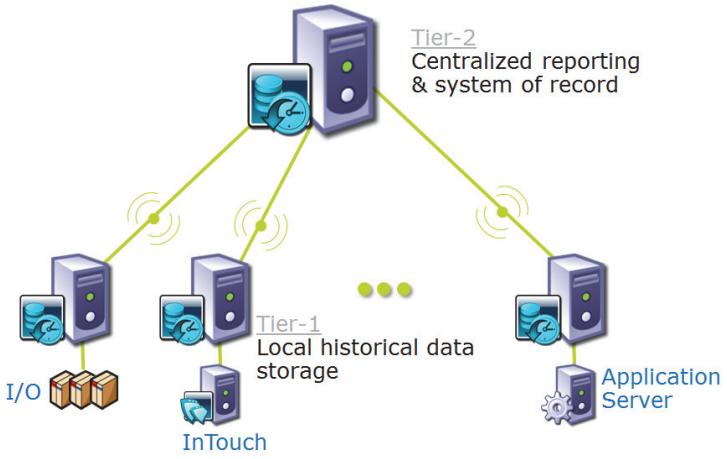


Figure 2. Typical Tiered Architecture

CONFIGURING WONDERWARE HISTORIAN

There are several subcomponents to the configuration subsystem. These include the runtime database, a SQL Server database that stores all configuration information; the configuration and management tools, consisting of the System Management Console (SMC) snap-in and the Historian Import/Export utility, shown in Figure 3.

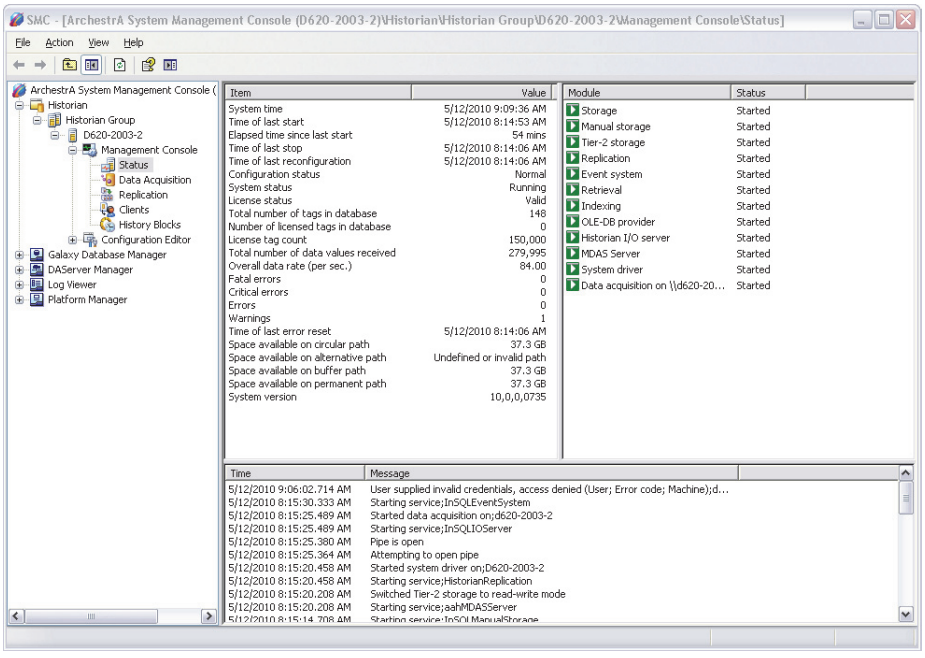


Figure 3. Configuring the Historian in the SMC

The Configuration Editor enables most of the Wonderware Historian configuration to be done in one place. The figure below shows the main menu tree of the Configuration Editor in the SMC.

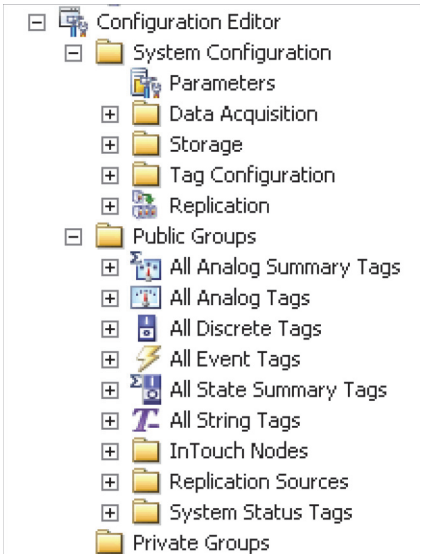


Figure 4. Configuration Editor Elements

The Wonderware Historian is shipped with two pre-configured databases: the Runtime and Holding databases.

The Runtime database is the online database against which the Wonderware Historian runs. The tables within the Runtime database store all configuration information, such as:

- System configuration.
- Data acquisition information.
- Tag definitions.
- InTouch integration information.
- System namespaces and grouping information.
- Event configuration information.
- User-entered annotations.

The Holding database temporarily stores topic and configuration data imported into Wonderware Historian from an InTouch node. When you import configuration data from an InTouch application, the data is first mapped to table structures in the Holding database. Then, the data is moved into the Runtime database. It is important never to attempt to manually modify entities within this database.

A Windows service, aahCfgSvc.exe, is an internal process that, as a part of the configuration subsystem, handles all status and configuration information throughout the system. This process accepts configuration changes and updates the Runtime database. Thus, this configuration service is the only component that interacts with the configuration store.

The Wonderware Historian supports dynamic configuration; that is, you can modify the configuration of tags and other objects in the historian database while the system is running. The historian automatically detects and applies the modifications to its internal run-time state, in most cases without requiring the system to be restarted. In addition, clients do not experience interruptions due to configuration changes. A system restart may be required, however:

- When you change the main historization path in the system, a parameter that is rarely modified after installation, or
- When you modify the DataImportPath system parameter.

DATA ACQUISITION

This section describes some of the most important components of the Data Acquisition subsystem. For full details, refer to the product documentation.

Data Servers	Wonderware-compatible software application that reads values from PLCs and other factory devices and forwards the real-time data to Wonderware applications.
IDAS Service	A process that accepts real-time data from one or more I/O Servers and forwards it to a single Wonderware Historian.
MDAS Service	A process that can accept non-I/O Server data and send it to the historian to be historized. Data is passed to MDAS through a COM interface. MDAS is used by Wonderware Application Server, the Wonderware Historian OLE DB provider, the event subsystem, and custom client applications.
InTouch History Importer	A utility to import data from one or more Wonderware InTouch history files (.lgh)
System Driver Service	An internal process that monitors the entire historian system and reports the status with a set of system tags. The system driver also sends data values to storage for the current date and time, as well as for pre-defined “heartbeat” tags, such as a discrete pulse.

The manual data acquisition service (MDAS) accepts data (real-time data, as well as inserts and updates) from its host and forwards it to the Wonderware Historian storage subsystem. For example, Wonderware Application Server uses MDAS to send history data to the historian. Replicated data from a tier-1 historian is sent to the historian using MDAS.

The storage subsystem merges data acquired from MDAS to existing historized data. All data can be accessed from the History, WideHistory, and Live extension tables.

MDAS is implemented in two ways within the system: one as a client-side Windows DLL and one as a Windows service. The DLL version of MDAS uses DCOM and file shares to send data to the historian. For both the MDAS and Wonderware Historian computers, make sure that DCOM is enabled (not blocked) and that TCP/UDP port 135 is accessible. The port may not be accessible if DCOM is disabled on either of the computers or if there is a firewall between the two computers that blocks the port. For information on enabling DCOM communication through a firewall, see your Microsoft Windows operating system documentation.

Configuration of data acquisition sources is done within the SMC. Acquisition from an ArchestrA system is configured on an ApplicationEngine within the ArchestrA IDE.

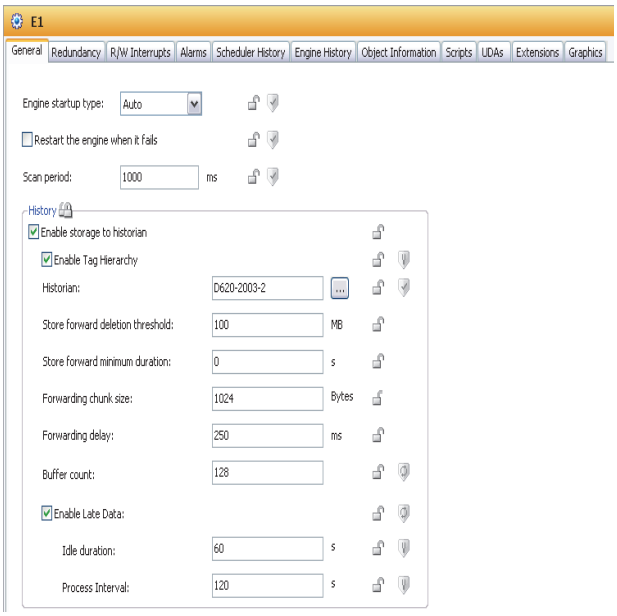


Figure 5. Configuring Historization in the IDE

In Figure 5, the ApplicationEngine E1 is configured using the "Enable Storage to Historian" check box to send data values to the historian on node D620-2003-2. Individual data points on ApplicationObjects running on this engine would be individually configured to send data values to the Historian, as shown in Figure 6.

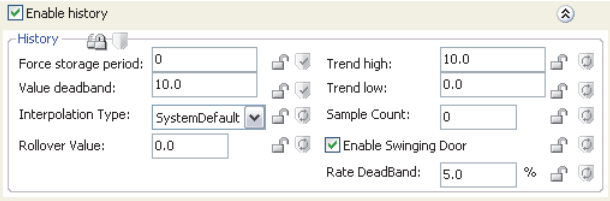


Figure 6. Configuring Storage on an Object Attribute

DATA STORAGE

The Wonderware Historian storage subsystem saves plant data from various sources to disk. The storage subsystem stores data for analog, analog summary, discrete, state summary, string, and system tags in sets of files on disk called history blocks.

Historical data can be retrieved by sending SQL queries through the Wonderware Historian OLE DB provider, which is part of the data retrieval subsystem. At retrieval, the historized tag data is presented as if it resided in SQL Server tables.

There are a number of components to the Data Storage subsystem. Some of the most important are discussed here. For full details refer to the product documentation.

Real-time Data Storage Service (aahStoreSvc.exe)	An internal process that stores real-time data to disk. This process runs as a Windows service.
Manual Data Storage Service (aahManStSvc.exe)	An internal process that processes non-real-time data and stores it to disk. This process runs as a Windows service. This process is also called "alternate" storage.
Active Image	A portion of system memory that temporarily holds all real-time data while the storage subsystem stores the actual values to disk.
History Blocks	A set of folders and files on the disk that contain historical data in compressed format.
Tier-2 Storage Process (aahStorageEngine.exe)	A secondary storage process that handles replication data on a tier-2 historian. This is not a Windows service.

Data saved to the Wonderware Historian belongs in one of the following categories: real-time data, “late” data, and “old” data. Each type of data has a separate set of characteristics and is handled differently by the historian. These characteristics are:

- Time sequential data. The data acquired by the historian can be in either time sequential order or in any order. For time sequential data, each consecutive data value received has a timestamp that is later than the previously received value. Data coming from an I/O Server is typically time sequential. Blocks of data that are imported into the system do not necessarily follow each other in time and would be an example of non-time sequential data.
- Relationship to the system-wide real-time data “window.” The real-time window is the maximum delay, relative to current time of the server, for which data is considered real-time by the system. The real-time window is the maximum delay, relative to current time of the server, in which data is considered real-time by the system. The real-time window can range from -30 seconds to +999 milliseconds of the current server time.
- “Future” data. If the incoming data has a timestamp that is in the future, relative to the server time, it will be handled differently based on what data category it falls in.

If necessary, incoming data timestamps are converted to Coordinated Universal Time (UTC) before storing the data.

The Real-time Data Storage Service (aahStoreSvc.exe) and the Manual Data Storage Service (aahManStSvc.exe) work together to store all of data to disk, organize the data in such a way that, upon retrieval, the data is as seamless and integrated as possible.

For flexibility based on data conditions, there are a number of types of storage modes available:

- No data values are stored. This is useful if legacy tag information is in the database, but is no longer needed except for reporting.
- All data values are stored (forced storage). Forced storage is useful, for example, if you have an I/O Server that collects data by exception from the plant floor device (instead of using a polling interval). You can allow the I/O Server to filter the values by exception, instead of the Wonderware Historian.
- Only changed data values are stored (delta storage). The Delta storage mode stores data based on a change in a value. Delta storage writes a historical record only if the current value changes from the previous value. Delta storage is also called “storage by exception.” Various options are available with delta storage, such as value, time, and rate of change (swinging door) deadbands.
- Only data values that occur at a specified time interval are stored (cyclic storage). Cyclic storage is the storing of analog data based on a time interval. Cyclic storage writes a record to history at the specified interval, only if a data changes during the time interval.

DATA RETRIEVAL

The Wonderware Historian data retrieval subsystem receives SQL queries from clients, locates the requested data, performs necessary processing, and then returns the results.

For configuration and event data, retrieval is made possible by normal SQL queries, because these types of data are stored in standard SQL Server database tables. Historical data, however, must be retrieved from history blocks and then sent to clients as if it is stored in SQL Server tables.

To accomplish retrieval from both data repositories, the retrieval subsystem includes:

- An implementation of a SQL Server data provider, which determines whether the requested data is saved in normal SQL Server tables or in history blocks.
- A retrieval service, which is responsible for extracting the requested data from the history blocks and presenting to the Wonderware Historian OLE DB provider as "virtual" history tables.
- A set of SQL Server extensions, which are implemented as columns in the history tables. You can use these extensions to specify the nature of the row set that is returned, such as the number of rows returned, the resolution of the data, or the retrieval mode.

The following table describes some of the components of the data retrieval subsystem.

Runtime database	The SQL Server database in which configuration and event data are stored.
History Blocks	Files in which plant history data is stored. In the context of Microsoft SQL Server, history blocks are considered a non-local data source.
Retrieval Service (aahRetSvc.exe)	A process that retrieves data from the history blocks and presents it as data sets. This process runs as a Windows service.
Manual Data Acquisition Service (MDAS)	Component that allows data retrieval, data insertions, and configuration functions, such as tag creation.
Wonderware Historian OLE DB Provider	A SQL Server software component used to query data in history blocks. The Wonderware Historian OLE DB provider can expose history data to client applications as if it were formatted as normal SQL Server tables.
Wonderware Historian I/O Server (aahIOSvrSvc.exe)	An internal process that allows clients to access current tag values from the active image using the SuiteLink or DDE protocols.

Some of the main features of the data retrieval subsystem are:

- All tag types can be included in the same query when retrieving from the History table. Any combination of tags can be submitted in a single query.
- Both fixed length and variable length strings are supported.
- All internal time computation and manipulation is done using a resolution of 100ns (0.1µs). The resolution exposed in queries depends on the version of SQL Server used.
- All times are handled internally as universal coordinated time (UTC). Conversions to and from local time are handled going in and out of retrieval so the external interface is local time.

EVENT SUBSYSTEM

Plant events range from startups and shutdowns, through trips and shift changes, to batch events and operator actions.

You can use the Wonderware Historian event subsystem to detect events and associate actions when they are detected. At a basic level, anything that can be determined by examining stored data can be used as an event. The event subsystem can be configured to periodically check to see if an event occurred.

The event subsystem performs the following basic functions:

- Detects when events occur by comparing sets of criteria against historical data in the database.
- Optionally logs event records to a dedicated SQL server table (EventHistory).
- Optionally triggers a configured action each time an event has been detected.

The following table describes some of the components of the event subsystem.

Runtime database	Stores event definition information and all data generated by the event subsystem, such as records of event detections, data summaries, and data snapshots
Configuration Editor	Part of the System Management Console. Used to specify event definitions and possible actions.
Event System Service (aahEventSvc.exe)	An internal process that coordinates event detection and action functions. This process runs as a Windows service. Using the System Management Console, you can configure the event service to automatically start and stop at the same time as the Wonderware Historian.

REPLICATION SUBSYSTEM

You can set up Wonderware Historians in a variety of tiered configurations. In a common configuration, data from multiple individual historians (called tier-1 historians) is fed into a single centralized historian (called a tier-2 historian). Another configuration is to have multiple tier-1 historians that feed information to multiple tier-2 historians in a many-to-many relationship, shown in Figure 7.

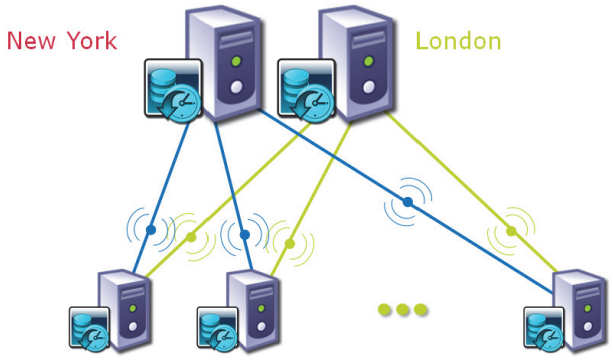


Figure 7. Multi-site Replication Configuration

Data from a tier-1 historian is replicated to a tier-2 historian using tags in the same way that information is collected by an individual Wonderware historian, namely using MDAS.

The tier for a tag is determined by where it comes from.

- Values for tier-1 tags are gathered directly from an IDAS or MDAS source.
- Values for tier-2 tags come from another Wonderware Historian server using MDAS.

A historian can act as a tier-1 and a tier-2 historian simultaneously.

There are two types of replication: simple replication and summary replication. Summary replication provides periodic summaries of high resolution data, while simple replication retains the original data resolution.

When a tag is configured for simple replication, all values stored in the tier 1 historian are replicated to the tier 2 server. Analog, discrete, and string tags can be configured for simple replication. Replicated tags of a tier-2 historian cannot be configured for further replication.

Summary replication involves analyzing and storing statistical information about the tag value at the tier-1 historian. This occurs at an interval you specify, called the calculation cycle. The result of the calculation is sent to the tier-2 historian to be stored with the timestamp of the cycle. Tier-2 tags are not dependent on the “real-time window” that usually applies to tier-1 tags.

Analog summary replication produces summary statistics for analog tags. The statistics relate only to the recorded interval. Statistics available are:

- Time-weighted average
- Standard deviation
- Integral
- First value in a period with timestamp
- Last value in a period with timestamp
- Minimum value in a period with timestamp
- Maximum value in a period with timestamp
- Start time of summary period
- End time of summary period
- OPC Quality
- Percentage of values with Good quality
- Value

Each real-time summary has a specified schedule, by which the summary is calculated and sent to the tier-2 historian to be stored with the timestamp of the cycle.

There are two types of replication schedules: Periodic and Custom (user defined).

SYSTEM CONFIGURATION AND PERFORMANCE

Configuring a server system for Wonderware Historian is straightforward since standard hardware and computer operating system software are used.

However, because the Wonderware Historian is a high-performance relational database, it is important to size your system to handle the level of input that you expect to store.

It is highly recommended that you run the historian on a dedicated computer. For example:

- Do not use the historian computer as a domain controller, mail server, or an Internet server.
- Do not use the historian computer as a workstation.
- Do not use the historian computer for InTouch HMI software, InControl, or other Wonderware products.

The minimum hardware and software requirements for the Wonderware Historian are based on the tag count and the anticipated data throughput rate. These requirements are divided into four levels, which are outlined in this section.

The recommended memory configuration for SQL Server is to clamp its memory consumption to 50 percent of the amount of physical memory installed on the server or 512MB, whichever is larger. The recommended Windows virtual memory setting is twice the amount of physical RAM installed on the server. You can also refer to the Microsoft website for updated installation requirements for different SQL Server versions.

For tag counts less than 30,000, the data throughput rate is assumed to be equal to the tag count. For tag counts greater than or equal to 30,000, the data throughput rate is assumed to be 30,000 values per second. The storage sub-system will support a burst rate of 60,000 updates per second for up to 1 second. This is the guaranteed throughput that the system can handle, but throughput rates substantially higher are possible, depending on hardware configuration.

If you are running the Wonderware Historian on a virtual server, the historian must have an adequate CPU, adequate network memory, and disk I/O resources at all times. Overloading the virtual server leads to unpredictable behavior.

The Windows Server Standard or Enterprise Editions R2 SP1 (64-bit) is the supported operating systems.

The following are the supported SQL Server versions (32-bit versions only):

- Microsoft SQL Server Standard, Enterprise 32-bit
- Microsoft Server R2 SP1, SP2, Standard, Enterprise, 32- and 64-bit
- Microsoft SQL Server SP3 Standard Edition

Disk Space requirements need to take into consideration not just the installation requirements for the Historian, but also the historical data storage needs (which may be on a separate storage unit from the installation).

- 300 MB of free disk space to install the Wonderware Historian
- Appropriate space for history block storage. For more information, see the Installation documentation on the Wonderware Historian product CD.

The following are guidelines for computer server sizing. As technologies change rapidly in this field, these are guidelines only but should suffice for most typical installations. Three different system configuration levels are listed below, as suggested base platforms depending on the tag count.

A level 1 server can handle a load of about 5,000 tags. For example, 2,600 analog, 2,200 discrete, 300 string, and 20 non-I/O Server (manual) tags.

Recommended Processor	Dual core CPU
Minimum RAM	2 GB
Recommended RAM	4 GB
Other	100 Mbps Network

A level 2 server can handle a load of about 65,000 tags. For example, 40,000 analog, 20,000 discrete, 300 string, and 5,000 non-I/O Server (manual) tags.

Recommended Processor	Quad core CPU
Minimum RAM	4 GB
Recommended RAM	6 GB
Other	1 Gbps Network

A level 3 server can handle a load of about 130,000 tags. For example, 70,000 analog, 50,000 discrete, 300 string, and 10,000 non-I/O Server (manual) tags.

Recommended Processor	Dual Quad core CPUs
Minimum RAM	6 GB
Recommended RAM	10 GB
Other	1 Gbps Network

Requirements for remote IDAS system are not as demanding typically as a server system because it will not be performing all of the same functions (for example, processing SQL Server transactions), but it should be powerful enough to handle the tag load that you expect.

The amount of free disk space required depends on whether or not you will have store-and-forward enabled for the IDAS. If store-and-forward is enabled, you need to make sure that the disk space on the remote IDAS computer is sufficient to store cached data if the network connection to the historian fails. Estimate the disk space requirements for a remote IDAS the same as for the historian.

More detailed system sizing examples and performance indications are presented in the Wonderware Historian Installation Guide, available on the Historian CD.

NETWORKING RECOMMENDATIONS

The Wonderware Historian is a highly configurable package that can be set up in many different ways depending on your needs.

The historian can use any protocol currently supported by Microsoft SQL Server. You can use the default Microsoft SQL Server protocol (named pipes) with TCP/IP. TCP/IP is required if SuiteLink™ is used. To change the default network protocol used by Microsoft SQL Server to something other than named pipes, configure the client network access using the SQL Server Client Network Utility.

Generally, it is recommended that you split the process and IS networks to ensure that the process network does not become overloaded. Figure 8 shows one possible network architecture where the historian is the link between the process network and the business LAN/WAN.

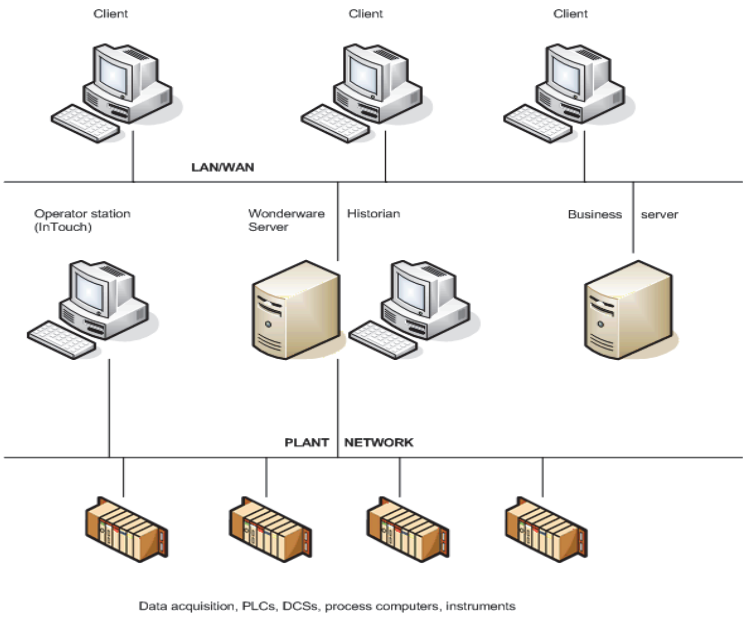


Figure 8. Process and IT Network Architecture with Wonderware Historian

In the architecture outlined in Figure 8, two network cards are installed on a Historian server computer and configured to segment the IS network from the process network.

CLIENT ACCESS

This section discusses client side connections to the Historian. This includes internal connections and is essential for understanding the implications on operating system client activity, since certain versions of the operating system and SQL Server impose connection limits or workload limits. For example Windows XP Professional rejects all connections above the 10-connection limit.

The Wonderware Historian and its clients consume both Windows operating system connections and SQL Server connections in the following ways:

- Wonderware Historian: When the historian itself is running without the event subsystem, it uses six database connections and zero Windows connections.
- System Management Console: Each open System Management Console consumes one database connection, and each remote System Management Console also consumes a Windows connection.
- Event System: Each different time interval for event tags uses a database connection and zero Windows connections. For example, if there are 15 event tags with time interval of 30 minutes, and 10 event tags with an interval of 60 minutes, that consumes two connections. The event subsystem uses zero Windows connections.
- Local IDAS: Consumes no connections.
- Remote IDAS: Each remote IDAS uses one Windows connection but no database connections.

- Wonderware Application Server Platform: A platform configured to historize data consumes a Windows connection and a database connection.
- Wonderware Application Server Engine: Each Engine configured to historize data will consume a database connection.
- Historian Client applications and controls: Each Historian Client application or control consumes a database connection for each specified server, and each remote node consumes a Windows connection.

In general, all clients should connect to the Wonderware Historian using the default Microsoft SQL Server connection. Usually, this means using the name of the computer on which the historian is running as the server name when logging on.

LICENSING

With one exception, the Wonderware Historian requires a valid license to run. Use the ArchestrA License Utility to manage licenses and associated feature lines. The historian allows functionality based on the presence of a valid license file and/or feature lines.

If a valid license file cannot be found, or if the file does not contain the appropriate feature lines, the historian is considered to be unlicensed. If no license is found, the historian will start up and run for an unlimited period of time, but is restricted to storage and retrieval of system tags and not more than 32 user defined tags.

Several functional aspects of the Historian can be independently licensed and configured. Unless otherwise noted, all aspects of historian licensing are dynamic. That is, you can make licensing changes during runtime, and the system will continue to run interrupted.

The following features can be restricted depending on license installed.

- Tag count: The Historian_Tagcount feature line specifies the number of tags for which the Wonderware Historian acquires and stores data. The tag count feature line does not restrict the number of tag definitions you can create in the database, just the tags for which data is acquired and stored. System tags and event tags are not included in the tag count.
- Server OS: The Historian_ServerOS feature line controls whether the installed version of the Wonderware Historian is licensed to run on a specific Microsoft server operating system.
- Remote IDAS: The Historian_RemoteIDASCount feature line controls the maximum number of remote IDASs that can be used by the historian. The remote IDAS count feature line does not restrict the number of remote IDAS definitions you can create in the database.
- Historical data modification: The Historian_ModifyHistoryData feature line controls whether history data modifications are allowed. This controls whether you can modify history data via SQL queries (inserts or updates) and CSV file imports (both normal CSV imports and “fast load” CSV imports).
- History Duration: If present, then the Historian_HistoryDuration feature line controls the maximum number of days in history, starting with the current day, for which data can be retrieved from the Wonderware Historian. For example, if the history duration is 50, you can only retrieve data that was stored during the last 50 days. If this feature line is 0, then there is no limit on retrieving data. If this feature line is missing (or no license is present), the default is seven days.

- System Processors: The Historian_Processors feature line controls the maximum number of processors (CPUs) allowed in the Wonderware Historian computer. This feature line has no impact on the operation of remote IDASs or other remote clients of the historian.

OTHER REFERENCES

There are a number of documents which will assist you in gaining more detailed knowledge of Wonderware Historian, its functionality, configuration and capabilities. The following documentation can be found on the product CD.

- Wonderware Historian Administration Guide
- Wonderware Historian Installation Guide
- Wonderware Historian Concepts Guide
- Wonderware Historian Database Reference
- Wonderware License Utility Guide

Foxboro®
by Schneider Electric

Invensys Systems, Inc
10900 Equity Drive
Houston, TX 77041
United States of America
http://www.invensys.com

Global Customer Support
Inside U.S.: 1-866-746-6477
Outside U.S.: 1-508-549-2424
Website: https://support.ips.invensys.com

MB 031 0914