Plant Applications 8.2

Server Management

Proprietary Notice

The information contained in this publication is believed to be accurate and reliable. However, General Electric Company assumes no responsibilities for any errors, omissions or inaccuracies. Information contained in the publication is subject to change without notice.

No part of this publication may be reproduced in any form, or stored in a database or retrieval system, or transmitted or distributed in any form by any means, electronic, mechanical photocopying, recording or otherwise, without the prior written permission of General Electric Company. Information contained herein is subject to change without notice.

© 2021, General Electric Company. All rights reserved.

Trademark Notices

GE, the GE Monogram, and Predix are either registered trademarks or trademarks of General Electric Company.

Microsoft® is a registered trademark of Microsoft Corporation, in the United States and/or other countries.

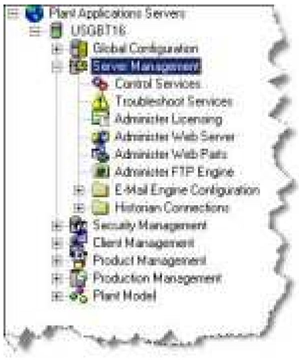
All other trademarks are the property of their respective owners.

We want to hear from you. If you have any comments, questions, or suggestions about our documentation, send them to the following email address:

doc@ge.com

Table of Contents

Server Management Node



Server Management is where you can manage a variety of server functions. When you expand the Server Management node, you can do the following:

- From **Control Services**, you stop and start the Proficy Server services, debug a Proficy Server service or Plant Applications model.
- From **Troubleshoot Services**, you view valuable information about the Proficy Server, such as current activity and pending tasks.
- From **Administer Licensing**, you can find out which modules and interfaces you are licensed to use and the number of licensed users.
- From **Administer Web Server**, you open the Web Server Administrator where you configure Web Server users and Web reports.
- From **Administer Web Parts**, you open the Web Part Administrator where you configure Web Parts.
- From **Administer FTP Engine**, you configure the Proficy Server FTP Engine service, which monitors specific directories for files which it then sends to a specified host or gets from a specified host. File Transfer Protocol (ftp) is used for scheduled interfaces and for transferring reports between the Report Engine and Web Server.
- From **E-Mail Engine Configuration**, you create email groups and users to be automatically notified of events for the Proficy Server Services, for Alarms, and other system events and triggers.
- From **Historian Connections**, you add and manage historians.

Plant Applications Control Services

Plant Applications Manager and Plant Applications Control

Plant Applications Services are managed through the Plant Applications Control Services node, which can be accessed through the Plant Applications Administrator or directly from the Proficy server. The default location for the Plant Applications Control (ProficyControl.exe) is C:\Program Files\Proficy\Proficy Server\Server on the Proficy server. This control is used by an administrator to start, stop, reload and debug a service.

After an install, the Proficy Server Manager service is set to start up automatically. The Proficy Server Manager starts all of the other Proficy Server services that are designated to be watched by the Proficy Server Manager. Services can also be stopped and started through the Plant Applications Control. If a service is stopped using the Plant Applications Control the Proficy Server Manager will not restart it.

The Plant Applications Control Manager should be used to manage Proficy Server services instead of using Microsoft's Services through the Control Panel. The Proficy Server Manager acts as a watch dog and will try to restart a service if it is not running. If cluster servers are used, GE Intelligent Platforms should be contacted for more information.

Service Log On Rights and Startup Type

Proficy Server services should be set to log on as a Local System account, which is the default used on the installation. This is accomplished by accessing Microsoft's Services through the Control Panel. Within the Services window, select the service and right click on it to select the Properties. The Startup Type can be set on the General tab and the Log On rights can be set using the Log On tab.

Specific Plant Applications services are required to operate under a Windows User account. For example, the Plant Applications Report Engine services needs to be logged in as a Windows User account that allows Excel to run without any dialogs appearing. This User Account should also have printers set up to allow for automatic printing. The Plant Applications PDF Generator service requires a specific log in as well.

Functions of the Proficy Services

The following is a list of services and functions that run on Proficy Server. The MS SQL Server Service must be running to start Plant Applications. The Proficy Manager will be set to start automatically as part of the installation. The remainder of the Proficy Server services will be started up by the Proficy Server Manager.

The Proficy Server Manager, Database Manager and Router make up the core service structure of Plant Applications. These services create the path to and from SQL Server for all Proficy Server services. They also manage buffering to ensure all data transfers are successful.

Proficy Server Manager

The Proficy Server Manager handles the automatic starting and stopping of all the Proficy Server Services. When the Proficy Server Manager is started, it starts all of the other Proficy Server Services. The Proficy Server Manager acts as a watch dog to keep all other Proficy Server Services up and running. If a Proficy Server Service were to stop unexpectedly, the Proficy Server Manager would restart the service.

The Proficy Server Manager can be configured to monitor other services by adding a record to the CXS_Service table. For more information, see the topic Using Proficy Server Manager to Monitor Other Services.

The recovery option must be set to **Restart the Service** on all failures.

Proficy Server Database Manager

The Database Manager handles most of the writing to the Plant Applications SQL database. It processes writes and confirmations.

Proficy Server Gateway

The Proficy Server Gateway controls the interaction of the Proficy Server with distributed clients running Plant Applications. Should client connections be lost, the Client Management Services will buffer outbound data so that when a connection is re-established no data is lost. Likewise buffering of data is performed on the client side such that data entered at the client is not lost due to connection loss to the server. It is the communication link between the Plant Applications Client and the Proficy Server. It is also the communication link between EAS applications and the Proficy Server.

Proficy Server Alarm Manager

The Proficy Server Alarm Manager generates alarms for all variables that have been configured for alarming. It checks the data and the alarm limits and compares them to see if values are outside the alarm or SPC limits.

Proficy Server Calculation Manager

The Proficy Server Calculation Manager generates results for all variables defined with a data source of CalculationMgr. These calculations can be of the types Equation, VBScript, or Stored Procedures. Plant Applications calculations can be set up to have dependency variables which for calculations to be re-fired when dependent data changes.

Proficy Server Email Engine

The Email Manager Service communicates to and from a MAPI client such as Microsoft Outlook. The service can be used, for example, to send emails to a system administrator for priority alarms or to clients with a Certificate of Analysis attachment when an order is complete. This service must run using an account with an email profile.

Proficy Server Event Manager

The Event Manager creates the various types of Plant Applications events such as Production Events, Product Change Events, Downtime Events, Waste Events, Genealogy Events, Schedule Change Events, Import Events, and Export Events. The Event Manager watches for different types of data to change and creates events based on that data. Events are created based on Event Models, which describe what data to watch, and how to interpret that data to produce an event. For example, an event model may call for the Event Manager to watch a set of tags in the historian to capture when a batch was started or completed, capture or create the batch ID, and capture or create the product code of the batch.

Proficy Server FTP Engine

The FTP Engine allows files to be passed to and from the Proficy Server. It monitors specific directories on the Proficy Server and FTP Servers for files and handles the sending and getting of files from a specified host. Typically this is utilized for file passing interfaces.

Proficy Server Reader

The Proficy Server Reader retrieves data and calculates values for all time-based variables that have a datasource type of Historian.

Proficy Server Stubber

The Proficy Server Stubber creates columns for time based AutoLog sheets, stubs out null tests based on the test frequency settings for each variable, and replicates values for repeating variables.

Proficy Server Summary Manager

The Proficy Server Summary Manager retrieves data and does calculations on all event-based variables that have a datasource type of Historian, based on the Event Type trigger. The Summary Manager also writes specifications to the Historian and generates all product run summarization data.

Proficy Server Writer

The Proficy Server Writer writes values back to the Historian.

Plant Applications Report Engines

The Report Engines are typically located on a dedicated Report Engine computer, or on the Web Server computer. Together with the Proficy Server Scheduler and PDF Generator services, these services allow for the creation of scheduled and ad-hoc reports.

Once the Report Engine services have been configured with a local NT user account using the Microsoft Services window through the Control Panel, the services can then be controlled through the Plant Applications Control Manager.

Multiple instances of the Report Engine service are typically configured to support parallel generation of reports.

Plant Applications Scheduler

Configured through the Plant Applications Web Administrator, this service will trigger the running of reports on the Report Engines. It allows reports to be scheduled on a regular interval, e.g. once a day or month etc., or with immediate priority. The starting and stopping of the Proficy Server Scheduler service should be done through the Plant Applications Control Manager.

This service resides on the Web Server computer with the Plant Applications Web Administrator.

Plant Applications PDF Generator

Located on the Report Engine computer, this service must be running to create a **PDF** version of a report when required. A distiller is installed during Plant Applications installation.

Once the PDF Generator service has been configured with a local NT user account using the Microsoft Services window through the Control Panel, the services can then be controlled through the Plant Applications Control Manager.

Plant Applications License Manager

This service determines if incoming client connections are valid based on the available licenses.

The recovery option must be set to **Restart the Service** on all failures.

Proficy Server Print Server

This service controls the printing of existing Web reports.

Proficy Server Remote Data Service

This service is used when there is an OPC Historian. It handles the API calls between the Proficy Server and the remote server.

Proficy Client Manager

The Proficy Client Manager is used by pre-4.0 Plant Applications clients. If a connection between Proficy Server and a client is lost, Proficy Client Manager will buffer data so that when the connection is re-established no data is lost.

Proficy Server Config Manager

The Proficy Server Config Manager provides message subscription information for the Proficy services. In other words, it tells the router the type of message the service is interested in.

Proficy Server Data Generator

The Proficy Server Data Generator is used for testing.

Proficy Server Router

The Proficy Server applications are built on an asynchronous messaging protocol based on TCP/IP. The Router service is the central hub that coordinates the movement of data between the Plant Applications services and guarantees the delivery of data to its destination, regardless of the status of individual clients or services.

Server Log Files

Dynamic log files, *.log, are generated for each service and are stored under the C:\Proficy\Logfiles directory. They are useful for looking up configuration and system errors. These should be checked regularly and configuration errors should be fixed.

Whenever a service starts, it creates a new log file. If a service is reloaded, data is written to the existing log file. The log files become larger as the service runs and will grow indefinitely. Old log files can be deleted any time. The active log files can be deleted if the service is stopped. The log files provide the best troubleshooting information for the Proficy Server Services. Special parameters for debugging and logging, control the detail of information stored in the files.

Buffer files will also be generated if a service is having problems or is blocked by another SQL server process. There are two types of buffer files. The Retry, *.rty, are generated when buffering occurs with the Historian. The *.hdr and *.msq files are generated when there is buffering between Proficy Server services. These files are located in C:\Proficy\BufferFiles\SERVICENAME.

Static show files, *.shw, hold the current configuration for a particular service. These show files are rewritten whenever a service re-loads which occurs at least every six hours. These files are helpful in troubleshooting configuration problems with event detection models and also with tags, which are configured to read and write from Historians.

Buffering

Buffering exists between each, and every, component of Plant Applications to ensure there is no loss of data if a component is offline. This also enables independence of the services where one does not rely upon another for proper operation. The buffering scheme also ensures that data is processed in the correct order. Plant Applications data is buffered when a service is offline and the buffered information is written to a binary file (e.g. Reader In, Reader Out, Writer In, Writer Out). When the service comes back online, the buffered data is transferred first in order of time sequence/accumulation and then the service returns to normal operations.

There is a 15 minite buffer zone on the Plant Applications Client. If the Client loses connection with the Gateway service, the data is held on the Client workstation and gateway memory, but after 15 minutes the port is dropped and all data written on the Client is lost. It is important to see the Green light on lower right portion of Client screen to ensure there is an active connection.

When there is a communication slowdown, data is maintained in RAM.

Proficy Server Troubleshooting Details

Using Proficy Server Log Files

Each of the Plant Applications services has a log file. The location of all the log files on the server is set by the site parameter **LogFilePath** (the default path is C:\Program Files\Proficy\Proficy Server\LogFiles). The log files report reload times, execution messages and errors and, when debug is turned on, debugging information. Log files should be check periodically and any error messages should be resolved. Every time a service is restarted, a new log file is created.

CalculationMgr Service

The following commonly used debug information is available in the CalculationMgr and can be used to verify that the calculation is being executed at the right time and with the right arguments:

- Triggering variable
- Calculation timestamp
- Actual execution start time and end time
- Passed argument values

- Result values

DatabaseMgr Service

The DatabaseMgr will show the messages that it receives. This is useful when using result sets and you need to verify that the message format is correct and is being properly sent to the DatabaseMgr.

EventMgr Service

The EventMgr log file will show the times it wakes up to check the historian for data and any times that it fires an event model because of it.

Proficy Server Show Files

Each of the Plant Applications services has a show file (.shw). The show files are always located in the same directory as the log files. The show files contain the configuration information acquired by the Plant Applications service when it was last reloaded. The running configuration can be verified here. If changes have been made in the Administrator, the show file can be checked to see if they have been picked up by the Plant Applications service. They are also a handy place to copy configuration information from when submitting issues to Plant Applications Support (the more information the better!).

Proficy Server Buffer Files

Each of the Plant Applications services has a buffer file. The location of the buffer files is determined by the Site Parameter BufferFilePath. The buffer files stored queued actions for the Plant Applications service to execute. If missing, the files and directories located in the BufferFilePath are automatically created.

Deletion of the buffer files can be a potentially helpful option when troubleshooting a problem as it will effectively reset the server. The Plant Applications services must be shut down for the buffer files to be deleted. By removing the buffer files, you are removing any queued actions so when the services are restarted there is no historical processing and Plant Applications will only process new events. It must be noted that, for the same reason, it will also result in some lost data as any recently queued actions will be lost.

Plant Applications Administrator Data Verification

Event Data: Event data can be verified through the Plant Applications Client displays. The displays are all based off the actual event data so any and all events present in the tables will be reflected in the Client displays.

The event data can also be retrieved through the Excel Add-In. This method is preferable as it easier to do analysis in Excel and check out the event data against other event and variable data.

Variable Data: Variable data is harder to fully verify in the Plant Applications Client displays due to the nature of the Client's data retrieval methodology. In the Autolog display, the columns are analogous to windows into the Tests table. The Client will search the Tests table for any data that matches the timestamp of the column in the Autolog display. As such, there may be additional data in the database for a particular variable that is not shown in the Autolog display.

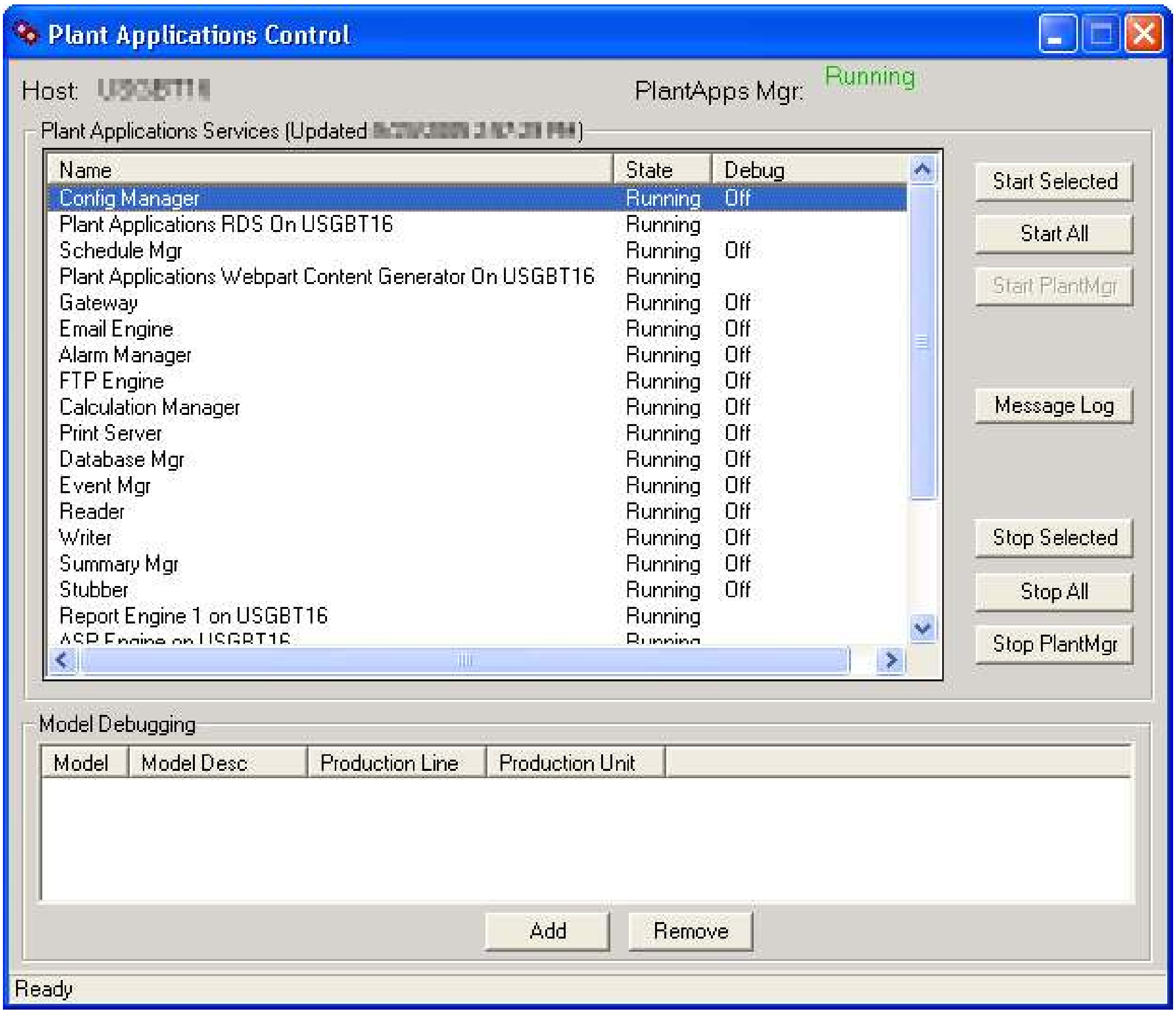
The Plant Applications Administrator has the capability to show all of the data associated with a particular variable. The View Variable Data feature is essential for verifying the data shown in the Plant Applications Client. For example, if there is missing or a problem with data in the Client, the View Variable Data feature should be used to verify the data in the database before checking anything else. Often the problem may be due to a configuration problem and the problem with the data is due to a timestamp mismatch or extra values in the database.

The variable data can also be retrieved through the Excel Add-In. This method is preferable as it easier to do analysis in Excel and check out the event data against other event and variable data.

To Retrieve Variable Data in the Administrator:

1. Expand the **Plant Model** folder and then drill down to the desired variable.
2. Right-click on the variable and click **View Variable Data**.
3. If necessary, modify the filter to retrieve data for the desired time frame.

Plant Applications Control Dialog Box



Host: [redacted] PlantApps Mgr: Running

Plant Applications Services (Updated [redacted])

Name	State	Debug
Config Manager	Running	Off
Plant Applications RDS On USGBT16	Running	
Schedule Mgr	Running	Off
Plant Applications Webpart Content Generator On USGBT16	Running	
Gateway	Running	Off
Email Engine	Running	Off
Alarm Manager	Running	Off
FTP Engine	Running	Off
Calculation Manager	Running	Off
Print Server	Running	Off
Database Mgr	Running	Off
Event Mgr	Running	Off
Reader	Running	Off
Writer	Running	Off
Summary Mgr	Running	Off
Stubber	Running	Off
Report Engine 1 on USGBT16	Running	
ASP Engine on USGBT16	Running	

Model	Model Desc	Production Line	Production Unit

Service Debugging

Plant Applications has a debug facility where additional execution information can be written to the Plant Applications log files. This is an extremely valuable tool for debugging calculations and diagnosing other execution issues.

Debug mode should be used sparingly as the log files can grow to an immense size very quickly. Do not leave debug mode on for an extended period of time or you may find yourself unexpectedly running out of disk space.

To turn on Debug mode:

1. In the Plant Applications Administrator tree, expand **Server Management**.
2. Double-click **Control Services**. The **Plant Applications Control** dialog box appears.

3. Right-click on the desired service, then select **Debug** and then choose the debug level. There are 3 levels of debugging, each of which writes progressively more information.

- The first level, **Low**, will write minimal information.
- The next level, **Medium**, will provide more information to most likely solve the issue
- The **High** level should only be used by GE Intelligent Platforms as it produces stack traces which will only be understood by GE Intelligent Platforms.

Determining Services Start Order

If the Proficy Server is having difficulty starting, it might be because one of the services has a lot of work to perform and is using up most of the CPU and preventing other services from starting.

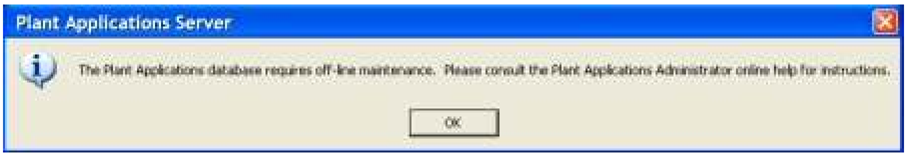
To prevent this, you can change the order in which the services start and force the service that is using most of the CPU to start last. To change the start order, you will need to modify the Start_Order of the desired service in the CXS_Services table in the Plant Applications database (default name, GBDB). Start_Order is used by PlantAppsMgr to determine the start order of each service.

> *Another column to be aware of is Start_Check_Time. Start_Check_Time is the maximum length of time (in seconds) that PlantAppsMgr will wait for a service to start before attempting to start the next service. If PlantAppsMgr detects that a service has started before the Start_Check_Time, PlantAppsMgr will attempt to start the next service, even if the Start_Check_Time has not expired. In other words, PlantAppsMgr will not wait for the Start_Check_Time to expire before it starts the next service.*

Server Maintenance

During an upgrade, Plant Applications may determine that tables in your database may require re-indexing, which will help improve performance. If this is the case, the Server Management icon will be yellow, instead of gray, and a new item will be added under Server Management: **Server Maintenance**.

During installation, a message box may also be displayed informing you that server maintenance needs to be performed.



Plant Applications Server

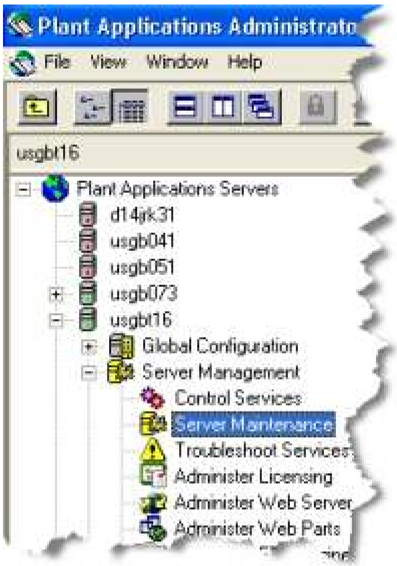
The Plant Applications database requires off-line maintenance. Please consult the Plant Applications Administrator online help for instructions.

> *This maintenance must be performed during a scheduled down time because the Plant Applications Manager service must be stopped.*

To perform server maintenance

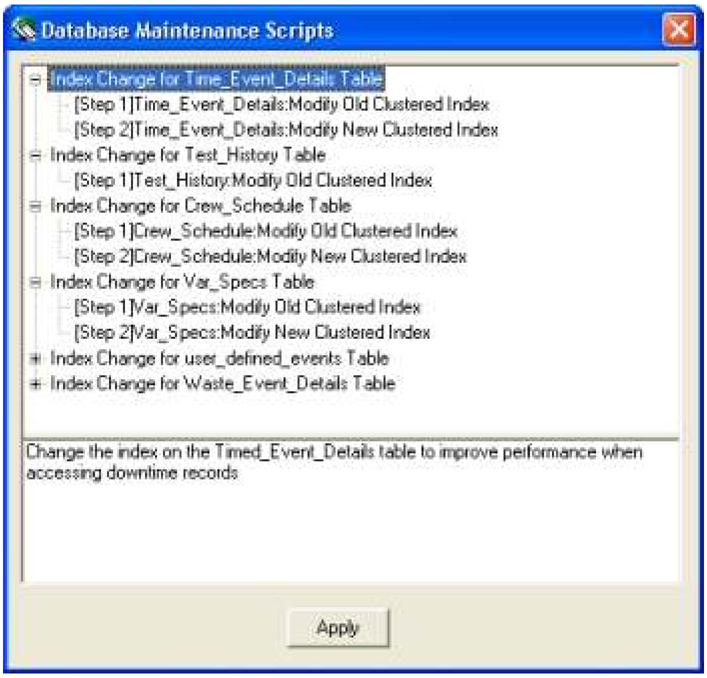
1. Open the Plant Applications Administrator.
2. Expand Server Management.
3. Double-click Server Maintenance.

> *Server Maintenance will not be in the tree if maintenance is either not necessary or it has been completed.*



4. A message box will inform you that the **Plant Applications Manager** service must be stopped before you apply the maintenance scripts. If necessary, stop the Plant Applications Manager service. Click **OK**.

5. In the **Database Maintenance Scripts** dialog box, select the table that you want to re-index and click **Apply**.


Database Maintenance Scripts
Index Change for Time_Event_Details Table
[Step 1]Time_Event_Details:Modify Old Clustered Index
[Step 2]Time_Event_Details:Modify New Clustered Index
Index Change for Test_History Table
[Step 1]Test_History:Modify Old Clustered Index
Index Change for Crew_Schedule Table
[Step 1]Crew_Schedule:Modify Old Clustered Index
[Step 2]Crew_Schedule:Modify New Clustered Index
Index Change for Var_Specs Table
[Step 1]Var_Specs:Modify Old Clustered Index
[Step 2]Var_Specs:Modify New Clustered Index
Index Change for user_defined_events Table
Index Change for Waste_Event_Details Table
Apply

Change the index on the Timed_Event_Details table to improve performance when accessing downtime records

You can select only one table at a time.

6. A message box will inform you that the process could take a long time. Click **Yes** to continue.

7. Repeat steps 5 and 6 for each table you want to re-index.

8. After all the tables have been re-indexed, click the **Close** ☒button on the **Database Maintenance Scripts** dialog box.

Site Parameters

Editing EndOfDayHour Site Parameter

The **EndOfDayHour** site parameter is one of two site parameters that are used to define the hour that your production day starts and ends. The **EndOfDayHour** site parameter determines the hour that your production day will start and end. The **EndOfDayMinute** site parameter determines the minute of the hour that your production day will start and end. For example, The default value of the **EndOfDayHour** site parameter is **7** and the default value of the **EndOfDayMinute** is zero, which means that your production day will start at 7:00 a.m. and will end the following day at 7:00 a.m.

Tell me about the **EndOfDayMinute** site parameter.

To edit this site parameter:

1. Start the Plant Applications Administrator.
2. Open the **Global Settings** folder
3. Right-click on **Site Parameters** and select **Edit Site Parameters** from the pop-up menu.
4. Scroll down and double-click **EndOfDayHour**.
5. In the **Edit Parameter** dialog box, type a new value in the **Value** box.
6. Click **OK** to save your changes and close the **Edit Parameter** dialog box.
7. Close the **Site Parameters** dialog box.

Editing EndOfDayMinute Site Parameter

The **EndOfDayMinute** site parameter is one of two site parameters that are used to define when your production day starts and ends. The **EndOfDayMinute** site parameter determines the minutes of the hour when your production day will start and end. The default value is **0**, which means that your production day will start at the hour defined by the **EndOfDayHour** site parameter and zero minutes. For example, if the **EndOfDayHour** site parameter value is **7** and the **EndOfDayMinute** site parameter value is 35, then your production day is defined as starting at 7:35:00 a.m. and ending at 7:35:00 a.m. on the next day.

Tell me about the **EndOfDayHour** site parameter.

To edit this site parameter:

1. Start the Plant Applications Administrator.
2. Open the **Global Settings** folder
3. Right-click on **Site Parameters** and select **Edit Site Parameters** from the pop-up menu.
4. Scroll down and double-click **EndOfDayHour**.
5. In the **Edit Parameter** dialog box, type a new value in the **Value** box.
6. Click **OK** to save your changes and close the **Edit Parameter** dialog box.
7. Close the **Site Parameters** dialog box.

EngineLoggingLevel

***IMPORTANT!** This site parameter should be changed only by someone with advanced knowledge of Plant Applications or when instructed by Customer Support.*

This site parameter determines the level of error messages that will be recorded by the report engine. The number of days that error messages are kept are determined by the site parameter, **LogfilePurgeDays**.

Error Message Levels

Level 0: Level 0 messages are messages that are always recorded. Typical messages included information about services starting and stopping; problems with the system, such as lack of disk space; or initialization information. If you select level 0, no other messages will be logged.

Level 1: Level 1 messages are critical messages and generate an email. Typically, a level 1 message means that the engine will shut down. If you select level 1, all level 1 and level 0 messages will be recorded.

Level 2: Level 2 messages record known, specific errors that have occurred within a report. If you select level 2, all messages with a level 2 and below (2, 1, 0) will be recorded.

Level 3: Level 3 messages record general errors within the report engine. If you select level 3, all messages with a level 3 and below (3, 2, 1, and 0) will be recorded.

Level 4: Level 4 messages record all general activity. If you select level 4, all messages with a level 4 and below (4, 3, 2, 1, and 0) will be recorded.

1. To change this site parameter:
2. Log in to the Plant Applications Administrator.

   *You must have **Administrator access** to the Administrator security group to change site parameters.*

3. Open the Global Configuration folder.
4. Right-click on the Site Parameters folder and select Edit Site Parameters from the pop-up menu.
5. In the Site Parameters dialog box, double-click on EngineLoggingLevel.
6. In the Edit Parameter dialog box, in the Value field, enter the maximum level of error messages you want to record.
7. Click the OK button to save your changes and close the dialog box.
8. Click the **Close**☒ button to close the Site Parameters dialog box.
9. Refresh the server. How?

To view the Report Engine log files:

1. Log in to the Plant Applications Administrator

   You must have **Manager access** or greater to the Administrator security group to start the Web Administrator.

2. Open the **Server Management** folder.
3. Right-click on the **Web Server** folder and select **Web Administrator** from the pop-up menu.
4. In the **Plant Applications Server Login** dialog box, enter your Plant Applications user name and password and click **OK**.
5. Open the **Web Server** folder, and then open the **Engines** folder.
6. Select an engine to view its log file.

Troubleshoot Services

Using Proficy Server Manager to Monitor Other Services

Server Management

The Proficy Server Manager service can be used to monitor and restart other services that have stopped. This is sometimes referred to was "watchdog" functionality.

To monitor other services

1. Insert a record into the CXS_Service table.

   Fields in the CSX_Service table that are required for the Proficy Server Manager to monitor a particular service are:

- **service_id:** Must be greater than 100 to prevent the record from being deleted when upgrading Plant Applications.
- **Is_Active:** Is it an active service (1 = Yes, 0 = No).
- **Start_Order:** Order in which Proficy Server Manager attempts to startup the services on reboot.
- **Monitor_interval:** default is always 0 (Can Proficy Server Manager restart the service).
- **Restart_Non_Responding:** If the service stops responding, can the Proficy Server Manager stop and restart the service? Special logic exists in some of our services, such as Proficy Server Gateway and Proficy Server EventMgr, which says after 17 minutes of not responding stop the service and restart it automatically. Default always 0 (only Proficy Server Gateway and Proficy Server EventMgr will ever have a value of 1).
- **Stop_Check_time:** Always default to 15.
- **Restart_wait_Time:** Amount of time (in seconds) Proficy Server Manager waits before attempting to restart the service. The default is 3 seconds.
- **Auto_Start:** Should Proficy Server Manager automatically start the service (1 = Yes, 0 = No)
- **Auto_Stop:** Should Proficy Server Manager automatically stop the service (1 = Yes, 0 = No)
- **Monitor_Service:** Should Proficy Server Manager monitor the service (1 = Yes, 0 = No)
- **Service_Desc**, **Service_Display:** Description of the service
- **Proficy_Service_Name:** Name of the .EXE that Proficy Server Manager will be monitoring.
- **Node_Name:** Name of server where the .EXE resides.

The following sample code demonstrates how to insert a record into the CXS_Service table.

```
SET IDENTITY_INSERT cxs_service ON
insert into cxs_service
(service_id,start_order,is_active,time_stamp,should_reload_timestamp,monito
r_interval,
restart_non_responding,stop_check_time,start_Check_time,restart_wait_time,r
eload_flag,
listener_port,Auto_start,Auto_stop,monitor_service,Service_desc,Service_dis
play,
NTService_name,Proficy_Service_name,Non_Responding_kill_script,Listener_add
ress,Node_name)
values(101,5,1,null,null,null,0,5,5,0,null,null,1,1,0,'Router','Router',NUL
L,'PRRouter',null,null,'9LYNQ91')
SET IDENTITY_INSERT cxs_service OFF
```

2. Restart Proficy Server Manager to pick up the changes.
3. Verify changes were picked up in the Proficy Server Manager's .SHW file (PlantAppsMgr.shw). By default, the .SHW file is located at: C:\Program Files\Proficy\Proficy Server\LogFiles

Plant Applications Administrator Troubleshooting Tools

The Plant Applications Administrator has a section for Server Troubleshooting. It is an integrated front end and much of the data is available through other tools as well.

To Access Server Troubleshooting Tools

1. In the Plant Applications Administrator tree, expand **Server Management**.
2. Right-click **Troubleshoot Services** and click **Troubleshoot Services**. The **Current Activity** dialog box opens.

Current Activity Tab

The Current Activity display gives a snapshot of the current processes running in SQL and indicates the specific SQL statement being executed.

This is a very useful facility for determining potential causes of high SQL Server utilization. If a particular SQL statement continues to appear after many refreshes then it likely is causing the SQL Server utilization. This high utilization is undesirable because it is preventing other Plant Applications SQL processes from executing. Most often the offending statement is a custom calculation, event model or report query. The Application Name tells you what kind of query it is (i.e., CalcMgr = calculation, EventMgr = event model, etc.) while the Job Description shows you the name of the stored procedure or the contents of the query.

- **Remote Clients:** The Host Name is useful in instances where the statement is being run from a remote client other than the server. When rolling out a Plant Applications system it is very helpful to maintain a list of all computers (ie. Computer name, owner and contact number) that will be connecting to the server so in situations where there is a runaway query on a remote computer, the administrator can contact the owner to further diagnose the issue.
- **Kill:** If there is a query using all of the SQL Server CPU, an extreme measure is to execute the KILL command from SQL Server Management Studio. Using the SPID from the Current Activity tab, execute 'KILL <Spid>' in Management Studio to end the process. This would generally be used to kill a SQL process executing from a remote client connection as any server based processes can be aborted by shutting down the relevant service.
- **Alternate Data Source:** The information contained in the Current Activity display is also available in Management Studio by running the SQL stored procedure spSupport_ShowRunning. This is handy to know as Management Studio has the option of returning the results directly to a file, instead of to the display, so in situations where you want to capture a record of the activity on the server you can easily generate a report without having to copy and paste.

Pending Tasks Tab

Plant Applications's execution is task-based. When an event occurs in the system the subsequent actions are defined as tasks and queued up in the PendingTasks table. This tab allows you to dynamically view the contents of that table and monitor Plant Applications's progress.

- **Number of Queued Tasks:** The data in the Pending Tasks tab has several things to offer. Foremost among them are number of tasks queued up and the rate at which they are cleared out. This is an indication of Plant Applications' performance. To properly judge this, the administrator must get an appreciation for the normal level of activity on the server (for example, are there normally 30 tasks waiting to be processed or 300). If there are a lot more tasks queued than normal then that's an indication that something is preventing Plant Applications from processing effectively. If the number of tasks continues to increase and no tasks are being then something may be totally blocking Plant Applications from executing.
- **Recalculation and Service Reloads:** High SQL Server usage is often due to the reevaluation of Plant Applications variables due to a historical data change or a service

reload. This can be diagnosed by checking for historical data being reprocessed through Pending Tasks. Generally, the old data would appear in relatively sequential order. Service reloads can be verified by checking the log files. Historical data changes can be verified by checking the test history for the suspected variable.

- **Execution Loops and Continuously Repeated Tasks:** Additional items that may contribute to high SQL Server usage are Plant Applications execution loops. By default, it is not possible to use a variable to trigger itself in a calculation. However, it is possible to use a variable to trigger another calculated variable and then have that variable trigger the original one. While, this is obviously dependent on the values constantly changing there are situations where during the creation of custom calculation stored procedures that an execution loop could be inadvertently created.

  One way to troubleshoot this is to check for continuously repeated tasks. If the same tasks (distinguished by Task, Unit, TimeStamp and Actual Id) repetitively show up and get cleared out, then it is possible there is an execution loop of some sort. Ultimately, to verify the presence of an execution loop, check the history of the suspect events and/or variables to check for repetitively changing values.

NOTE: *There is a user parameter, Allow Circular Calculations, that can be enabled on the on the CalculationMgr system parameter. For more information on enabling system (user) parameters, see Plant Applications System Parameters.*

Audit Trail Tab

The Audit Trail captures all configuration changes made through the Administrator. Often the first course of action in any troubleshooting situation should be to first try and determine what has changed recently because it is likely the changes that are causing the issue. The Audit Trail provides that facility along with who made those changes. It should be noted though that this only captures actions performed in the Administrator.

The Audit Trail gives the following information:

- Which configuration stored procedure was executed and the parameters used
- Which user made the change
- When the change was made

Diagnostics Tab

All diagnostic statistics are valid since the last start of the service. The Total columns are not actual but are calculated as the Average Number. Some of the statistics may be skewed on servers that have multiple CPUs.

The settings for the statistics are controlled by the following Plant Applications Manager User Parameters:

- **StatSaveInterval -** How often the statistics are saved in minutes. The default = 5. Some statistics will not show up in the display until this amount of time has elapsed after the last services restart.

IMPORTANT: *If these are set too low, system performance could be affected.*

- **StatHistoryTimeRange** - How many minutes the statuses are kept (default =60).

The **Blocking** diagnostics option is intended to show blocked SQL Server processes. This data is also available in the Current Activity display. Blocked processes are processes that have stalled while waiting to acquire a lock on a table. They stall because another process has already acquired the same lock and it too has stalled. This is a complex situation that requires some in depth analysis of SQL code to resolve. The same information is available in SQL Server Management Studio from the stored procedure spSupport_ShowBlocking.

By default, all processes are shown in this display. Blocked processed are identified in the Blocked column by SPId and application name (i.e. "53 / Plant Applications CalculationMgr").

Generally, when blocking occurs, two processes will be simultaneously blocked. A simple way to resolve the issue is to stop one of the processes. As such, if the DatabaseMgr and the CalculationMgr are both blocked, shutting down and restarting one of the services will resolve the blocking.

Blocking is mostly due to poor coding of custom stored procedures. As such, there are a few simple rules one can use to avoid blocking:

- Always use result sets to make modifications to the database: Most blocking tends to occur when a custom stored procedure attempts (ie. through a calculation or event mode) to directly update or delete records in the database and it conflicts with the DatabaseMgr service. The Plant Applications DatabaseMgr service is responsible for making updates to the database so by using result sets the DatabaseMgr service will make the changes for you and situations where both processes are attempting to access the same record locks will be avoided.
- Avoid the use of SQL transactions: If you have to use them, make the SQL transaction as short as possible. SQL transactions help to ensure database integrity by performing all the actions at once. As such, if the required lock cannot be acquired, the whole processes stops until it’s released. This is a leading cause of blocking.
- Update/Delete records using the record id: It is better to determine the record id through a select statement and then use that in the update/delete than it is to use your selection criteria directly in the update/delete statement. If SQL Server does not have a useful indexed reference to find the desired record, it will use a table scan. This means that for the duration of the update every row in the table will be locked.
- Avoid the use of cursors: To preserve data integrity a cursor will by default lock the selected record set and prevent changes to it by other connections. Cursors are inefficient and generally perform poorly so they can hold unnecessary locks for a long period of time. If a cursor is deemed necessary, consider making it a STATIC or INSENSITIVE cursor. The behavior of these cursors is different in that they implicitly take a snapshot of the desired data and thereby freeing it up for modification by other processes. This can also be achieved explicitly by selecting the record set into a temporary table and using the temporary table instead of the database table in the cursor.

The **CPU Statistics** option shows CPU load of each service along with a total. This can help in diagnosing problems with a particular service. This information is also available from the Windows Task Manager.

Under the Plant Applications architecture all communication between the server and clients is conducted through the Router service (called Message Bus in earlier versions). The **Message Statistics** option exposes the actions of the Router service and basically shows whom everyone is talking to. It reports a variety of statistics regarding the message communication between all the clients and services and the Router service. The following key fields are available:

- **Processed:** The number of messages successfully sent. It continues to rise when the connection is alive.
- **In:** The number of messages waiting to be sent. If this continues to consecutively rise then there might be a problem with the communication.
- **Out:** The number of messages sent but not yet processed. This should generally stay at 0. If it continues to consecutively rise then there might be a problem with the communication.
- **MaxIn:** The most messages to have ever been queued up.

The **Model Statistics** option shows Unit and Event detection stats. This is especially helpful for tracking performance of generic models as it is easy to track model performance and watch for excessive processing.

The following key fields are available:

- **Percent:** The amount of clock time (not CPU) taken as a % of the total since the service started for a specific model.
- **Runs:** The number of times the model detected an event and executed its model logic.

Generic models that fire a custom stored procedure have a potential for high CPU usage if careful attention is not paid to the performance of the stored procedure itself and the number of times it's executed. If possible, analog historian tags used to trigger generic models should have a deadband applied to them so that they don't needlessly fire the attached stored procedure repeatedly.

The **Stored Procedures** diagnostic option shows the performance of all stored procedures running on the server. This is a very useful feature for diagnosing high SQL Server CPU usage. The last tab, % of Services Time, will not appear until the StatSaveInterval time has elapsed after the last service restart.

The following key fields are available:

- **Percent:** Percentage of clock time stored procedure use.
- **Runs:** Total times run.
- **Last n (min):** Total runs in the last n minutes.

Log Files Tab

All logging activity is written to the database. This allows it to be searchable for strings for each individual service and also allows filtering by a start and end time. The tables are automatically purged by the system. The data presented is the same as that contained in the log files with the exception of the debugging information, which is written only to the log files.

Unknown Historian Tags Tab

This tab identifies historian tags that have been defined in the configuration but which do not exist in the Historian. It checks all variable tags as well as all tags used in event models. Unknown historian tags should be resolved immediately as it can affect the performance of the EventMgr and Reader services.

Plant Applications Service Heartbeat

There are a number of Plant Applications Site Parameters that the Administrator should be aware of when setting up a Plant Applications Server and troubleshooting issues with the Proficy Server.

Site Parameter	Default	Description
IsAlive	0	When the parameter is set to 1 it immediately activates a heartbeat signal for all the services. In each service's log file the text "Alive..." is written once per minute. If the frequency is significantly greater than a minute or not being written at all then there is a problem with the service and a restart should be attempted. This flag should not be left on for a long period of time as it will generate large log files.

Administer FTP Engine

Setting Up the Proficy Server FTP Engine

The Proficy Server FTP Engine service monitors specific directories for files which it then sends to a specified host or gets from a specified host. File Transfer Protocol (ftp) is used for scheduled interfaces and for transferring reports between the Report Engine and Web Server. When configuring events, the Import/Export models are for transferring event-based data. Be careful to use the proper slash / (web) or \ (files) for file locations.

The FTP Engine allows files to be passed to and from the Proficy Server. It monitors specific directories for files and handles the sending and getting of files from a specified host. Typically, this is utilized for file passing interfaces.

To Set Up an FTP Job

1. In the Plant Applications Administrator tree, expand **Server Management**.
2. Double-click **Administer FTP Engine**. The **Configure FTP** dialog box appears.
3. On the **FTP Servers** tab, click **New** to create a new FTP Job.
4. On the **Configure FTP Job** tab, configure an FTP job for each set of files to be transferred in or out. Define the following parameters for each FTP Job.

Parameters	Definitions
Engine	Server where the FTP Engine is running. Multiple FTP Engines can be running at one site.
Description	Description of this FTP Job.
Direction	Put or Get.
Transfer Type	ASCII or Binary.
Interval	Number of minutes to wait between checking for files.
Host Name	Name of the FTP host to get or put files.
Operating System	Windows NT, OpenVMS, or UNIX.
Host Username	FTP Username used to log in to the FTP Host.
Host Password	FTP Password used to log in to the FTP Host.
Mask	Indicates the filenames to be retrieved (e.g. *.dat)
Local Path	Path on the FTP Engine machine where files are put or sent from.
Remote Path	Path relative to the FTP root of the Host where files will be retrieved from or placed. A / indicates the FTP root of the host.
Email Success	Email Group to send success messages to.
Email Warning	Email Group to send warning messages to.
Email Failure	Email Group to send failure messages to.

5. Click **Save** when completed.

The entry is now available on the **FTP Server**s tab for editing.

The Plant Applications Email Engine

Plant Applications comes with an email engine that can automatically notify users of events for the Plant Applications services, for alarms (variables, products, equipment), and other system events and triggers. This can speed up the support process by quickly notifying system administrators and manufacturing personnel of problems. It allows custom alerts to be sent out as well.

The email engine uses SMTP as the delivery mechanism for email messages. When the email engine has been configured and set to Active, the Email_Messages table is checked once per minute to see if there are messages to send, and sends them if they exist. This means that specialized monitoring routines can be created using custom events or calculation variable stored procedures. These routines would create the custom email messages that are to be sent out by populating the Email_Messages table.

SMTP, Microsoft Exchange, and IIS

In Exchange 2000 and Exchange 2003, the Simple Mail Transfer Protocol (SMTP) connector replaces the Internet Mail Service in earlier versions of Exchange for mixed mode environments. However it should be noted that in pure Exchange 200x environments a connector is not a requirement for internet mail flow. Exchange 2000 and Exchange 2003 work differently than Exchange Server 5.5. SMTP is an add-on to Exchange Server 5.5 through Internet Mail Service. SMTP is native to Exchange 2000 and Exchange 2003. Everything is SMTP-based. The default SMTP virtual server (by itself) can handle all Internet traffic (inbound and outbound).

The Simple Mail Transfer Protocol (SMTP) service provided by IIS is a simple component for delivering outgoing email messages. Delivery of a message is initiated by transferring the message to a designated SMTP server. Based on the domain name of the recipient email address, the SMTP server initiates communications with a Domain Name System (DNS) server, which looks up and then returns the host name of the destination SMTP server for that domain.

Next, the originating SMTP server communicates with the destination SMTP server directly through Transmission Control Protocol/Internet Protocol (TCP/IP) on port 25. If the user name of the recipient email address matches one of the authorized user accounts on the destination server, the original email message is transferred to that server, waiting for the recipient to pick up the message through a client program.

In the case where the originating SMTP server cannot communicate directly with the destination server, the SMTP service can transfer messages through one or more intermediate relay SMTP servers. A relay server receives the original message and then delivers it to the destination server, or redirects it to another relay server. This process is repeated until the message is delivered or a designated timeout period passes.

Configuring the Email Engine

The Email Engine is configured and activated by setting the EmailEngine system user parameters in the Plant Applications Administrator.:

- InternalSMTPServer
- InternalSMTPSignature
- ExternalSMTPServer
- FromEmailAddress

The Email Engine monitors the Email_Messages table and sends e-mail messages to the SMTP server specified by the parameter, InternalSMTPServer.

If the InternalSMTPSignature is left blank, it will always try to send e-mail only to the InternalSMTPServer. If the SMTPSignature mask (i.e. company.com) is set, then that is used to figure out the actual internal SMTP server that is used.

System User Parameters

InternalSMTPServer: This is the node name of your company mail server (i.e. company.com). The Email Engine will send all e-mail to the SMTP server and this will act as the mail server for the Email Engine.

InternalSMTPSignature: This parameter is used by the e-mail engine to decide which SMTP Server to send the e-mail to. If your e-mail is john.doe@company.com, then an ideal value for this parameter would be company.com. If this parameter is left blank then the Email Engine will always try to send e-mail through the SMTP server specified by the InternalSMTPServer parameter.

ExternalSMTPServer: The node name of an SMTP Server that allows relaying. This is specifically used for relaying messages to addresses outside of the internal company e-mail addresses. This machine will most likely not be the same as your company e-mail server. It should be a machine that is not exposed outside your company firewall. This can be any pc or server and required an SMTP Server to be set up. In Windows XP or in Windows Server, Internet Information Services can be used to set up a Default SMTP Virtual Server for this purpose. If an ExternalSMTP server is set, then if an address is not found on the Internal SMTP server, the message is routed to the ExternalSMTPServer. The ExternalSMTPServer is essentially an SMTP e-mail server that is a relaying e-mail server and is used to send e-mails to addresses that are not in company mail server. This parameter is not required to have a value unless this feature of relaying is needed.

FromEmailAddress: Reply address for any e-mails sent out by the e-mail engine. Value must not be blank, the default value of xx@yy.zz will work. Otherwise provide a valid e-mail address here to enable the reply feature.

Email Recipients and Email Groups also need to be set up in the Administrator under **Server Management > E-Mail Engine Configuration** to complete the configuration for the Email Engine**.**

To Customize the EMailEngine Service for a Site:

1. Display the System User Parameters for the Email Engine service.
2. Expand **Security Management** and then expand **System Users**.
3. Right-click on **EmailEngine** and select **Edit EmailEngine Parameters**. The **User Parameters - [EmailEngine]** dialog box appears.
4. Set the parameters **InternalSMTPServer**, **InternalSMTPSignature**, **ExternalSMTPServer**, and **FromEmailAddress** to the site-specific values required.

Using IIS to Set Up an External SMTP Server

To Set Up an External SMTP Server Using IIS with Windows XP

1. Under **Administrative Tools** in the menu or in Control Panel, select **Internet Information Services**.
2. Select the **Default SMTP Virtual Server** entry under the computer name.
3. Right click and select **Properties**. Select the Access tab and under the Relay restrictions section select **Relay**.
4. In the Relay Restrictions window select the button **All except the list below** option.
5. Make sure the check box at the bottom is checked on the entry **Allow all computers which successfully authenticate to relay, regardless of the list above**.

Troubleshooting the E-Mail Engine

To Test and Troubleshoot the Plant Applications Email Engine

1. Start the Email Engine Service from the **Control Panel – Services** if it is not already running. See Service Debugging for more information.
2. Verify the Email Engine service is running by checking the EmailEngine-xx.Log file in the Proficy\Logfiles directory. If there is a log-on failure, see Troubleshooting the E-Mail Engine .
3. Sending a test message. Retrieve the EG_Id from the Email_Groups table for one of the e-mail groups. This value can be retrieved from the EmailEngine.Shw file in the Proficy\Logfiles directory or by running the following SQL Query against the GBDB database.
4. Select * from EMail_Groups. Use this query to find the EG_Id

- Insert a test message into the Email_Messages table using the query inside the brackets. [Insert into Email_Messages (EG_Id, Submitted_On,EM_Subject,EM_Content) Values (51, GetDate(),'Test Emails','Testing e-mail engine...')]

5. Check for the message to be processed correctly by running the following query from inside the brackets.

- [Select EM_Processed,* from Email_Messages]

  If the EM_Processed field is changed to a 1, after waiting 1 minute for the message to be picked up, then the Email is getting sent out and the Plant Applications Email Engine should be fully set up.

  If the EM_Processed field stays a 0 then the Email is not getting sent out. If it is not getting sent out, then check the EmailEngine-xx.Log file in the Proficy\Logfiles directory.

Exporting E-Mail Engine Configuration

To export the e-mail engine configuration

1. In the Plant Applications Administrator, expand Server Management.
2. Right-click E-Mail Engine Configuration and select Export Configuration. The Configuration Export dialog box appears.
3. Select one or more of the following:

- **E-Mail Recipient:** Select this option to create a worksheet containing the configured e-mail recipients description, e-mail address, and the e-mail group(s) to which they belong.
- **E-Mail Messages:** Select this option to create a worksheet containing all the e-mail messages, the message number, the email group to which it is assigned, and the assigned severity.

4. Click **OK**. Excel creates one worksheet for each of the selected options.

Importing E-Mail Engine Configuration

After you export the e-mail engine configuration, you can edit the information or add information and import the changes into a Plant Applications server.

- **Email Recipients** worksheet: You can edit the E-Mail Group Description and a **new** email group is created along with the email recipient and recipient email address. If you edit the email recipient, a new email recipient is created and made a member of the email group.
- **Email Messages** worksheet: Email messages with message IDs less than zero are reserved and only the **Subject**, **Text**, and **Email Group** can be updated. Inserts and deletes are not allowed and any changes to the **Severity** are ignored. You can add a new message and delete or update a user-created message.
  - To insert (add) a new email message, type the letter **I** in the **Selected** column.
  - To update an email message, type the letter **U** in the **Selected** column.
  - To delete an email message, type the letter **D** in the **Selected** column.

To import e-mail engine configurations

1. Ensure the worksheet is open and:

- you have typed an **I**, **D**, or **U** to select the email message configurations you want to import.
- you have typed an **X** to select the email recipient configurations you want to import.

4. In the Plant Applications Administrator, expand **Server Management**.

5. Right-click **E-Mail Engine Configuration** and select **Import Configuration**. The selected configurations will be imported.
6. Click **OK** and refresh the server.

E-Mail Groups

E-mail groups are configured in the Administrator for inputting email addresses (recipients) and assigning groups of recipients to receive the messages. When e-mail messages are sent out, they are sent to members of a group, not selectively to recipients. One or more e-mail recipients can belong to an email group. Custom e-mail messages are sent out through a user-defined e-mail group.

The following options are available by right-clicking on an e-mail group.

- Create a new e-mail group
- Rename an e-mail group
- Add or remove group members (e-mail recipients)
- Delete an e-mail group

Server Notification Groups

By default, three Server Notification groups are included within the system and have been created for automatic sending of messages to the members of the groups, if a particular event occurs on the server. The three Server Notification groups are:

- Critical
- Warning
- Informational

Proficy Service	E-mail Group	Event
Proficy Server Manager	Critical	Service restart
Proficy Server FTP Manager	Critical	FTP transfer fails
Proficy Server FTP Manager	Warning	PostAction error occurs (for example, the move of the transferred file)
Proficy Server FTP Manager	Informational	Each PostAction

Creating E-Mail Groups

By default, Plant Applications provides three e-mail groups. You also can create user-defined e-mail groups and assign e-mail recipients to these groups.

To create a user-defined e-mail group

1. In the Administrator, expand **Server Management** and expand **E-Mail Engine Configuration**.
2. Right-click **E-Mail Groups** and select **Add New Email Group**. A blank new entry field displays in the E-Mail Groups list.
3. Type the name of the group and press ENTER.

Specifying Group Membership

All the members of an e-mail group will receive e-mail messages sent to the group. You cannot send system e-mail messages to individual recipients.

To specify group members

1. In the Administrator, expand **Server Management**.
2. Expand **E-Mail Engine Configuration** and **E-Mail Groups**.
3. Right-click the group and select **Edit Membership**. The **Edit Email User Membership** dialog box appears.
4. Do the following:
    a. To add members to the group, select the e-mail recipient(s) from the **Non-Members** list and click the **Add Selected** button.
    b. To remove members from the group, select the e-mail recipient(s) from the **Members** list and click the **Delete Selected** button.
5. Click the **Close** button.

Renaming an E-Mail Group

To rename an email group

1. In the Administrator, expand **Server Management**.
2. Expand **E-Mail Engine Configuration** and **E-Mail Groups**.
3. Right-click the group you want to rename and select **Rename <*e-mail group*>**. The group name becomes an editable field.
4. Type the new name of the group and press **ENTER**.

Deleting E-Mail Groups

To delete an e-mail group

1. In the Administrator, expand **Server Management**.
2. Expand **E-Mail Engine Configuration** and **E-Mail Groups**.
3. Right-click the group you want to delete and select **Delete <*e-mail group*>**. The **Confirm Delete** dialog box appears.
4. Click **OK** to delete the e-mail group.

E-Mail Recipients

E-mail recipients determine the e-mail address to which e-mail messages are sent. E-mail recipients must be a member of a group. E-mail messages cannot be sent to an individual e-mail address. An e-mail recipient can be a member of more than one e-mail group.

Creating an E-Mail Recipient

To create an e-mail recipient

1. In the Administrator, expand **Server Management** and expand **E-Mail Engine Configuration**.
2. Right-click **E-Mail Recipients** and select **Add New Email Recipient**. A blank new entry field displays in the E-Mail Recipients list.
3. Type the name of the recipient and press ENTER.

After you create the e-mail recipient, you must edit the recipient's properties to specify the recipient's e-mail address.

Specifying an E-mail Address

After creating an e-mail recipient, you must enter the recipient's e-mail address where e-mail messages are to be sent.

To specify an e-mail address

1. In the Administrator, expand **Server Management** and expand **E-Mail Engine Configuration**.
2. Expand **E-Mail Recipients**.
3. Right-click the recipient and select **Recipient Properties**. The **Email Recipient Address** dialog box appears.
4. In the **Email Address** box, type the e-mail address for the recipient.
5. Click 💾 to save your changes.
6. Click the **Close** button.

Deleting an E-Mail Recipient

To delete an e-mail recipient

1. In the Administrator, expand **Server Management** and expand **E-Mail Engine Configuration**.
2. Expand **E-Mail Recipients**.
3. Right-click the recipient and select **Delete Recipient**. The **Confirm Delete** dialog box appears.
4. Click **OK** to delete the recipient. The recipient will be deleted from the E-Mail Recipients folder and any e-mail groups.

Renaming an E-Mail Recipient

To rename an e-mail recipient

1. In the Administrator, expand **Server Management** and expand **E-Mail Engine Configuration**.
2. Expand **E-Mail Recipients**.
3. Right-click the recipient and select **Rename Recipient.** The recipient's name becomes editable.
4. Type the recipient's new name and press ENTER.

Email Messages

To view and edit email message: In the Plant Applications Administrator, expand the Server Manager tree, expand **Server Management**, and then expand **E-Mail Engine Configuration**.

Plant Applications comes installed with default email messages that can be sent when a specific error has occurred to a specific email group with an assigned severity. Each email message is identified with a unique number, which is used by certain Plant Applications stored procedures to generate email notifications.

There are default user-defined properties available for you to use with your custom stored procedures. The values of the properties can be determined by you. You can also create your own user-defined properties. This feature is provided as an easy way for you to manipulate the values of the properties. How you use them is completely up to you. For a list of the default user defined properties, see User-defined Properties.

To view email messages

1. In the Server Manager tree, expand **Server Management** and then expand **E-Mail Engine Configuration**.
2. Right-click **Email Messages** and click **Edit Email Messages**. The **Email Messages** dialog box appears.

*You can also view the **Email Messages** dialog box by double-clicking on **Email Messages**.*

Historian Overview

By default, Plant Applications creates a local historian, which uses the Proficy Server as an historian. In addition to the local historian, you can now access a number of different historians including generic OLE for Process Control (OPC) based historians, such as Intellution OPC EDA, and SQL Server based historians including Industrial SQL (now known as Wonderware Historian). In addition, the Remote Data Service allows you access historian data without installing any historian client access tools on the client machine.

Each historian you create will need to be configured through a predefined set of options that are specific to the source that it is collecting from. As the user you need to have enough knowledge about the target historian that allows you to intelligently set these options. For OPC servers, the server vendor will need to supply values for some of the required options.

You must have Administrator access to the Administrator security group to configure historians.

For more information, click the following links:

- Adding an OPC Historian
- Adding a SQL Server Historian
- Adding an InSQL Historian
- Adding an AspenTech InfoPlus.21 Historian
- Adding an OSI PI Historian
- Adding a GE Proficy Historian
- Adding an Intellution Advanced Historian

Setting Up an Historian (Basic Steps)

Historians are set up and managed through the Plant Applications Administrator. By default, Plant Applications comes with an Historian called **Historian (Local)**, which provides a connection to the database installed by Plant Applications to act as a data source.

TIP: A Local historian enables you to use Plant Applications variables as sources for other variables. For example, imagine that you create a variable to store the Production Amount for each generated Production Event and then you create another variable to store the daily Production Amount. Using the local historian, you could search for the variable with the Production Amount for each Production Event. This would be the input tag for the daily Production Amount tag.

In general, to add or edit an Historian:

1. Select the appropriate server.
2. Click **Server Management**.
3. Right-click **Historian Connections** and click **Add New Historian** from the pop-up menu.
4. Type the actual name or alias of the new Historian and press **Enter**.
5. Right-click the new Historian and select **Edit <historian name> Properties** from the pop-up menu.
6. In the **Edit Historian** dialog box, do the following:
    a. In the **Server Name** box, type the actual name of the server
    b. If necessary, in the **Username** box, type the user name required for connecting to the server.
    c. If necessary, in the **Password** box, type the password required for connecting to the server.
    d. Select the correct operating system from the **Operating System** drop-down list.
    e. Select an appropriate GE Proficy - Historian (based on the version) from the drop-down list as the **Historian Type**. For example:

Historian Type	Supported Versions
GE Proficy Historian 1.x - 2.x	Historian versions 1.x and 2.x
GE Proficy Historian 3.x - 5.x	Historian versions 3.x to 5.x
GE Proficy Historian 6.x and Greater	Historian versions 6.0 SP1 to 8.x

Follow the detailed instructions in other topics in *Plant Applications Help* to connect to the other "standard" historians.

Importing Tags

Follow the instructions in *Proficy Historian Help* to add the Excel Add-in for Proficy Historian. In Excel, use the **Import Tags** feature under **Administration** from the Historian Add-In menu at the Add-Ins tab.

Testing the Historian

To test connectivity, click the **Test** button in the Edit Historian screen where the connection is set up.

To test the data acquisition from the historian, create a Plant Applications variable that is time based and put in an input tag from the historian for this variable. Check the Reader log file for errors. Build an AutoLog time-based log sheet to view the data for this variable.

Raw Historian Data Analysis

The Raw Data feature of the Plant Applications Excel Add-In is a key tool for diagnosing issues with data retrieved from an historian. Refer to "Raw Historian Data Analysis in the Troubleshooting and Maintenance section of this help system as well as "Excel Add-In," also in this help.

Remote Data Service

The Proficy Server Remote Data Service (RDS) allows Plant Applications client components, including the Administrator, to access historians without installing the API on the local computer. By consolidating the historian connections, this facility provides a single point of configuration and software installation, making it easier to deploy the Plant Applications software.

In regard to OPC servers, the RDS eliminates the need for DCOM configuration by providing a point-to-point connection dedicated to Plant Application software between the Proficy Server and the OPC server. An RDS installation is required on OPC servers that are to be treated as a historian. The RDS install is located as part of the Plant Applications Client install.

Other situations that might require an RDS to be installed on another machine include instances when the historian API cannot be installed on the Proficy Server. For example when an API will not run on the latest version of Windows, you may need to install the RDS on an older version of Windows and then install the API there.

Adding a GE Proficy Historian

To add a GE Proficy Historian:

1. In the Plant Applications Administrator tree, expand the appropriate server.
2. Click Server Management.
3. Right-click Historian Connections and click Add New Historian from the pop-up menu.
4. Type the name of the new historian and press Enter.

You can type either the actual name of the server or use an alias.

5. Right-click the new Historian and select **Edit <historian name> Properties** from the pop-up menu. The **Edit Historian** dialog box appears.
6. Do the following:
    a. In the **Server Name** box, type the actual name of the server.
    b. If necessary, in the **Username** box, type the user name required for connecting to the server.

*If this historian uses Windows Authentication, you will need to add the domain to the front of the user name in this format: **mydomain\username**.*

    c. If necessary, in the **Password** box, type the password required for connecting to the server.
    d. Select **Windows NT** as the operating system.
    e. Select **GE Proficy - Historian** as the historian type.

   f. Double-click the **Remote Data Source** option to set up for remote use of an OPC server with an historian.

Generally you will not need to check the Remote Data Source check box as this type of historian can be configured to run through the Proficy Server.

7. Click the **Save** icon or close the **Edit Historian** dialog box and you will be prompted to save your changes.
8. After editing any of these settings, you must reload the following Plant Applications services: Reader, Writer, Event Manager, and Summary Manager. You will prompted to reload these four historian connect services upon exiting.
9. You should verify that your new historian is working and connected to the Plant Application Server. Click the **Test** button and, as an option, add a specific tag to test.

Adding an Intellution Advanced Historian

To add an Intellution Advanced Historian:

1. Select the appropriate server.
2. Click Server Management.
3. Right-click Historian Connections and click Add New Historian from the pop-up menu.
4. Type the name of the new Historian and press Enter.

You can type either the actual name of the server or use an alias.

5. Right-click the new Historian and select **Edit <historian name> Properties** from the pop-up menu.
6. In the **Edit Historian** dialog box, do the following:
   a. In the **Server Name** box, type the actual name of the server
   b. If necessary, in the **Username** box, type the user name required for connecting to the server.

*If this historian uses Windows Authentication, you will need to add the domain to the front of the user name in this format: **mydomain\username**.*

   c. If necessary, in the **Password** box, type the password required for connecting to the server.
   d. Select **Windows NT** as the operating system.
   e. Select **Intellution Advanced Historian** as the historian type.

Generally you will not need to check the Remote Data Source check box as this type of historian can be configured to run through the Plant Applications Server.

   f. Double-click the **Remote Port** option to

7. Click the **Save** icon or close the **Edit Historian** dialog box and you will be prompted to save your changes.
8. After editing any of these settings, you must reload the following Plant Applications services: Reader, Writer, Event Manager, and Summary Manager. You will prompted to reload these four historian connect services upon exiting.
9. You should verify that your new historian is working and connected to the Plant Application Server.

Adding an OPC Historian

Plant Applications can connect to any OPC server, including OPC Historical Data Access (HDA) and OPC Data Access (DA). Any OPC server that Plant Applications will be connected to must comply with all OPC standards.

> ***NOTE:*** *The vendor of the OPC server must supply values for the options listed in step 6.g.*

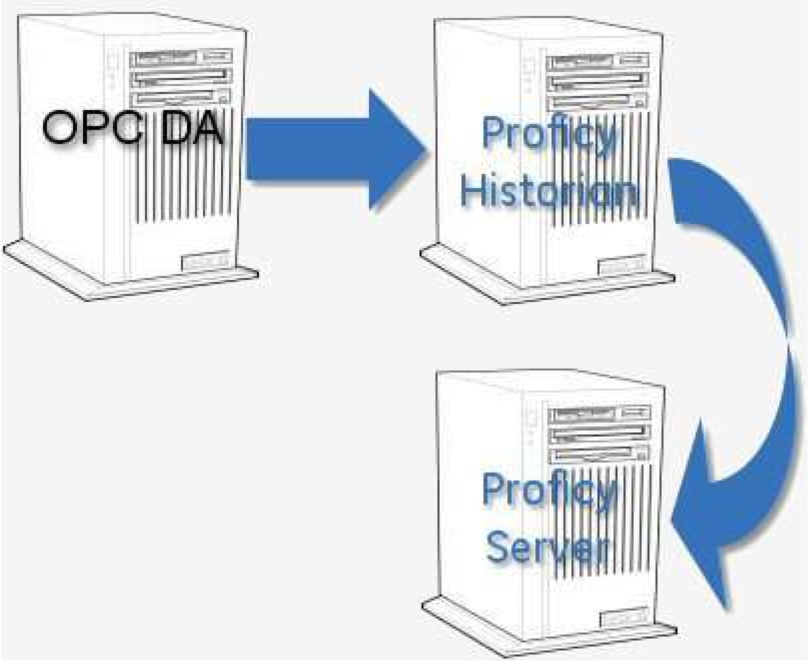
Capturing Historical Data

Keep in mind that OPC DA servers provide access to only real-time data. In other words, there are no historical values. OPC DA is only useful for:

- Reads with a sampling type of Last Good Value or Closest Good Value (see the topic Variable Property Definitions for definitions of Last Good Value and Closest Good Value)
- Writes only

What this means is that, with an OPC DA server, if the Event Manager or Reader or Summary Manager services stop for any reason, there is no way to read triggers or acquire values that occurred while these services were stopped. For example, if you have configured Reader or Summary Manager to read Last Good Value on time- or event-based intervals, the value will be missed if the services are down, even if the services are reloaded. In addition, the Summary Manager service cannot aggregate (for example, SUM or AVG) any data over a specified time period, event run, or product run. Therefore, OPC DA is not supported for use by the Event Manager or Summary Manager because it has no “historical” context.

To retrieve historical values, you could use an OPC HDA server or a better solution would be to install Proficy Historian and write your OPC DA data to it, which would then historize your data for you, eliminating the need for an OPC HDA compliant server.



Readable tags

When Plant Applications browses tags on an OPC server, it requests all tags that are **Readable**. According to the OPC DA 2.05 specification, **Readable** tags should also include tags that are **Readable and Writable**. However, some OPC servers return only tags that are **Readable** and exclude tags that are **Readable and Writable**. For these servers, you must manually enter tags that are **Readable and Writable**.

NOTE: *You must have Administrator access to the Administrator security group to configure historians.*

To add a new OPC historian

1. In the Plant Applications Administrator, select the appropriate server.
2. Click Server Management.
3. Right-click Historian Connections and click Add New Historian. An editable field appears.
4. Type the name of the new historian and press Enter. You can type either the actual name of the server or use an alias.
5. Right-click the new historian and select Edit <historian name> Properties. The Edit Historian dialog box appears.
6. Do the following:

    a. In the **Server Name** box, type the name of the server.

    b. If necessary, in the **Username** box, type the user name required for connecting to the server.

NOTE: *If this historian uses Windows Authentication, you will need to add the domain to the front of the user name in this format:* ***mydomain\username****.*

    c. If necessary, in the **Password** box, type the password required for connecting to the server.

    d. Select **Windows** for the operating system.

    e. Select **OPC** as the historian type.

    f. Select the Remote Data Source check box. Note that all OPC historians require that a Plant Applications Remote Data Service is running on the same computer.

    g. Set the OPC options. Most of these options will be supplied by your OPC server vendor and must match the values they supply. Those that are optional allow for some site customization.

    To change the value of an option, double-click the option.

- **BrowseMethod:** vendor supplied
- **Disable OPCDA Write Limit:** If set to True, then an old tag value can be written to the OPC-DA server. If set to False, then an old tag value cannot be written to the server.
- **Prod Id:** vendor supplied (e.g. Intellution.OPCEDA)
- **Read Mode:** vendor supplied
- **Read Windows (Seconds):** Historical reads are prevented from coming from Plant Applications by blocking the read if the requested timestamp is before the last read time or if it’s before the read window. The read window is offset from current time. See also Start Time Setback.
- **Remote Port:** optionally vendor supplied
- **ServerExpectsUCT:** vendor supplied
- **Start Time Setback (Secs):** Sets a base time when the OPC server is initially connected in order to determine the timestamp of tag values in OPC-DA, This window is used for reads/writes. For writes, the first write is allowed to go as long as the write timestamp is within this time window, after that, the time of the last write is know; therefore, writes aren't allowed before that time.
- **Tag Filter:** optional, can be used to filter specific tags
- **Threading Model:** vendor supplied

- **Write Method:** vendor supplied

7. Click the **Save** icon or close the **Edit Historian** dialog box and you will be prompted to save your changes.
8. After editing any of these settings, you must reload the following Plant Applications services: Reader, Writer, Event Manager, and Summary Manager. You will prompted to reload these four historian connect services upon exiting.
9. Verify that your new historian is working and connected to the Plant Application Server.

Connecting to Plant Applications with an OPC Client

You can configure the Plant Applications server so that it can communicate with an OPC client. There are two tasks you need to perform in order to use the Plant Applications server as an OPC server. First, you will need to register the **PROPCSVR.DLL** file, and then you will need to create a site user that has a valid Windows NT account. This is the Windows NT account that you will use to log in to the computer where the OPC client is installed.

To register the PROPCSVR.DLL file

1. On the server where the Plant Applications server is installed, locate the **PROPCSVR.DLL** file. Typically, it is located at **C:\Program Files\Proficy\Plant Applications\Server**
2. Click **Start** and then click **Run**. The **Run** dialog box appears.
3. In the **Open** box, type:

```
Regsvr32 "r;C:\Program Files\Proficy\Plant
Applications\Server\PROPCSVR.DLL"
```

The path "C:\Program Files\Proficy\Plant Applications\Server\PROPCSVR.DLL" is the location of PROPCSVR.DLL on a typical installation of the Proficy Server. Be sure to use the path of your PROPCSVR.DLL.

4. Click **OK**.

To create a site user

1. Start the Plant Applications Administrator.
2. Open the **Security Management** folder.
3. Right-click on the **Site Users** folder and select **New User**.
4. Type the name of the site user and press **ENTER**.
5. Right-click on the site user and select **Edit <*user name*> Properties**. The **Edit User** dialog box appears.
6. Do the following:
   a. **optional**: In the **Description** box, type a brief description to help identify the user or provide additional information. You are limited to 255 characters.
   b. **optional**: In the **Password** box, type a password. You are limited to 30 characters.
   c. In the **Windows Info** box, enter the Windows NT account that you want to use to log in to the computer where the OPC client is installed. You can click the **Browse** ... button to select an available Windows NT account.
   d. **optional**: In the **Default View** box, select a default view for this user. The selected view will be automatically opened when this user logs in to the Plant Applications Client application.
   e. **optional**: The **Active** option is selected by default and must be checked in order for this user to have access to Plant Applications.

f. **optional**: Select the **Role-Based** option if you want this site user to participate in role-based security.

g. **optional**: Select the **Mixed-Mode** option to allow this site user to use either their Plant Applications user name (the name you entered in step 4) or their Windows NT account name and password. If this option is not selected, the site user must use their Windows NT account to log in to Plant Applications

5. Click the **Save** button to save your changes.

Adding a Wonderware (InSQL) Historian

Similar to the SQL Server historian, the IndustrialSQL (now known as Wonderware) historian is a more predictable configuration because the tables are predefined and consistent. You must manually import our standard set of stored procedures.

To add an Industrial SQL Historian:

1. Select the appropriate server.
2. Click Server Management.
3. Right-click Historian Connections and click Add New Historian from the pop-up menu.
4. Type the name of the new Historian and press Enter. You can type either the actual name of the server or use an alias.
5. Right-click the new Historian and select Edit <historian name> Properties from the pop-up menu.
6. In the **Edit Historian** dialog box, do the following:

a. In the **Server Name** box, type the actual name of the server

b. If necessary, in the **Username** box, type the user name required for connecting to the server.

If this historian uses Windows Authentication, you will need to add the domain to the front of the user name in this format: ***mydomain\username****.*

c. If necessary, in the **Password** box, type the password required for connecting to the server.

d. From the **Historian Type** drop-down list, select **IndustrialSQL** as the historian type.

Generally you will not need to check the Remote Data Source check box because this type of historian can be configured to run through the Proficy Server.

e. Set the options. To change the value of an option, double-click the option.

BrowseSQL: This string allows the tag browse to work. The where clause of this string accepts the value supplied in the browse window; the value it is compared to must be a string and the 'r;like ?TagId' is required. Here is an example of this string, note the required columns here. In this example, 'r;Float' is a required column, but is always null in this particular implementation.

```
Select convert(varchar(20),lRSDataPointId) as TagID,
 sDescription,
 sEngUnits,
 'Float'
 from rsdatapoint
 where convert(varchar(20),lRSDataPointId)
 like ?TagId
```

Connection String: This is the ADO connection string used to make a database connection. This string might look something like this:

```
DRIVER={SQL Server};
Server=192.168.0.53;Database=BizWare;UID=BizWareUser;PWD=
```

UID is for the user ID and **PWD** is for the password. This named user must have permissions to execute the strings you are about to supply as values for the remaining options.

DeleteSQL: This string allows the deletion of data for a given timestamp and tag id.

```
Delete from rsdatapointsample
  where tTimeStamp = ?TimeStamp
    and lRSDataPointId = convert(int,?TagId)
    and lMillisecond is null
```

ExecDirectFlag: Some SQL servers may not support parameter passing mechanisms. This allows for the PR2SQL.dll to put together the SQL statement with the parameters already parsed, instead of using the normal prepared statement method.

InsertSQL: This string allows the insertion of data for a specific tag, with a specific timestamp, valid value, and data quality status. This string is very specific to the target SQL Server and therefore is a good example of the use of a stored procedure and the passing of named and parameters.

```
exec spProficy_InsertData ?TagId, ?TimeStamp, ?Value, ?Status
```

TimeSQL: This provides Plant Applications the current time while performing functions like storing data. This string must return a result set with the first column named CurTime. This string might look like:

```
Select getdate() as CurTime
```

7. Click the **Save** icon or close the **Edit Historian** dialog box and you will be prompted to save your changes.
8. You must reload the following Proficy services: Proficy Server Reader, Proficy Server Writer, Proficy Server EventMgr, and Proficy Server Summary Mgr. You will be prompted to reload these four historian connect services upon exiting.
9. To add the InSQL stored procedures, go to C:\Program Files\Proficy\Proficy\Proficy Database\Util on the Proficy server and run the **Scripts.exe**. The **Script Utility dialog box** appears. Do the following:
    a. In the **Server Name** box, type the name of the InSQL historian server.
    b. In the **Database Name** box, type the name of the InSQL historian database.
    c. In the **User Name** box, type the user name for the InSQL historian database.
    d. In the **Password** box, type the password for the user name used in the preceding step.
    e. Under **Scripts Path**, navigate to C:\Program Files\Proficy\Proficy Server\Support\InSQL.
    f. Click **Run**.
10. Verify that your new historian is working and connected to the Plant Applications server.

Adding an AspenTech InfoPlus.21 Historian

Plant Applications can easily connect to your AspenTech InfoPlus.21 historian. You will need to know the user name and password of an account that is a member of the **Administrators** group on the InfoPlus.21 server and that has Read/Write access to the following InfoPlus.21 definition records (tables):

- IP_DISCRETEDEF
- IP_ANALOGDEF
- IP_TEXTDEF

The following Proficy Server services will need to be configured to log in using this account:

- Proficy Server Remote Data Service
- Proficy Server Event Manager
- Proficy Server Calculation Manager
- Proficy Server Reader
- Proficy Server Writer

If you do not select the **Remote Data Source** option when you configure the InfoPlus.21 historian connection, you must install the InfoPlus.212 API on the Proficy Server.

IP_ARCHIVING tag field

When you configure your InfoPlus.21 tags, you must set the IP_Archiving tag field to **ON** so that the InfoPlus.21 database passes the data to the history repository.

Please see the documentation that came with your InfoPlus.21 software for more information.

To add an AspenTech InfoPlus.21 Historian:

1. In the Plant Application Administrator tree, expand the desire server.
2. Click Server Management.
3. Right-click Historian Connections and click Add New Historian.
4. In the Historian Connections node, type the name of the new historian and press Enter.

   You can type either the actual name of the server or use an alias.
5. Right-click the new historian and click **Edit <historian name> Properties**.

   The **Edit Historian** dialog box appears.
6. Do the following:
   a. In the **Server Name** box, type the name of the AspenTech server.
   b. In the **Username** box, type the user name required for connecting to the server.

      If this historian uses Windows Authentication, you will need to add the domain to the front of the user name in this format: ***mydomain\username****.*

      The user that you enter must have a Windows account with Read/Write privileges on the AspenTech server
   c. In the **Password** box, type the password required for connecting to the server.
   d. From the **Operating System** drop-down list, select the operating system of the server where the AspenTech historian is installed.
   e. From the Historian Type drop-down list, select AspenTech InfoPlus.21.
   f. Select Remote Data Source.

*If you do not select **Remote Data Source**, the InfoPlus.21 API must be installed on the Proficy Server.*

7. Click the 💾**Save** icon or close the **Edit Historian** dialog box and you will be prompted to save your changes.
8. After editing any of these settings, you must reload the following Proficy Server services: Proficy Server Reader, Proficy Server Writer, Proficy Server Event Manager, and Proficy Server Summary Manager. You will be prompted to reload these four services upon exiting.

Adding a SQL Server Historian

Plant Applications can connect to and use SQL-based database as an historian.

If you are using IndustrialSQL you should configure the historian as that type, not a SQL Server historian. See Adding a Wonderware (InSQL) Historian.

Configuring a SQL-based historian requires a high level of expertise in SQL and the Transact-SQL language. Many of the option strings return result sets require specific and positional column headers in order for the Proficy Server to obtain and in some cases update the appropriate data. Additionally, some of these strings require named parameters like ?TagId; these allow the Proficy Server to pass data to the SQL to be executed.

In the examples below the carriage returns and line feeds are intended to make the string readable; they are not actually allowed in the string. As of version 4.0, 1000 characters are allowed. If you need more than 1000 characters, or you simply want to keep your logic in stored procedures, you may supply stored procedure names as the strings. Depending upon the version of SQL you are using, you may need to add additional commands to execute a stored procedure. For example, under SQL Server, you would need to place an 'exec' statement before the name and pass the named parameters as parameters to the stored procedure.

To setup a SQL Server historian connection

1. Select the appropriate server.
2. Click Server Management.
3. Right-click Historian Connections and click Add New Historian from the pop-up menu.
4. Type the name of the new Historian and press Enter.

You can type either the actual name of the server or use an alias.

5. Right-click the new Historian and select **Edit <historian name> Properties** from the pop-up menu.
6. In the **Edit Historian** dialog box, do the following:
    a. In the **Server Name** box, type the actual name of the server
    b. If necessary, in the **Username** box, type the user name required for connecting to the server.

*If this historian uses Windows Authentication, you will need to add the domain to the front of the user name in this format: **mydomain\username**.*

    c. If necessary, in the **Password** box, type the password required for connecting to the server.
    d. Select **SQL Server** as the historian type.
    e. Generally you will not need to check the Remote Data Source check box as this type of historian can be configured to run through the Proficy Server.
    f. Set the options:

To change the value of an option, double-click the option.

BrowseSQL: This string allows the tag browse to work. The where clause of this string accepts the value supplied in the browse window; the value it is compared to must be a string and the 'r;like ?TagId' is required. Here is an example of this string, note the required columns here. In this example, 'r;Float' is a require column but is always null in this particular implementation.

```
Select convert(varchar(20),lRSDataPointId) as TagID,
 sDescription,
 sEngUnits,
 'Float'
 from rsdatapoint
 where convert(varchar(20),lRSDataPointId)
 like ?TagId
```

Connection String: This is the ADO connection string used to make a database connection. This string might look something like this:

```
DRIVER={SQL Server};
Server=192.168.0.53;Database=BizWare;UID=BizWareUser;PWD=
```

UID is for the user ID and **PWD** is for the password. This named user must have permissions to execute the strings you are about to supply as values for the remaining options.

DeleteSQL: This string allows the deletion of data for a given timestamp and tag id.

```
Delete from rsdatapointsample
  where tTimeStamp = ?TimeStamp
    and lRSDataPointId = convert(int,?TagId)
    and lMillisecond is null
```

ExecDirectFlag: Some SQL servers may not support parameter passing mechanisms. This allows for the PR2SQL.dll to put together the SQL statement with the parameters already parsed, instead of using the normal prepared statement method.

InsertSQL: This string allows the insertion of data for a specific tag, with a specific timestamp, valid value, and data quality status. This string is very specific to the target SQL Server and therefore is a good example of the use of a stored procedure and the passing of named and parameters.

```
exec spProficy_InsertData ?TagId, ?TimeStamp, ?Value, ?Status
```

ReadAfterSQL: This string reads data for a given tag id after a supplied timestamp.

```
Set RowCount ?NumValues;
Select tTimeStamp,
  value = case when sValue is null then
convert(varchar(20),dValue)
               else sValue end,
  status = case when dQuality = 0 then 'Good'
               else 'Bad' end
  from rsdatapointsample
  where lRSDataPointId = convert(int,?TagId)
    and tTimeStamp > ?TimeStamp
```

```
      order by tTimeStamp
```

ReadBeforeSQL: This string reads data for a given tag id before a supplied timestamp.

```
    Set RowCount ?NumValues;
    Select tTimeStamp,
      value = case when sValue is null then
    convert(varchar(20),dValue)
                   else sValue end,
      status = case when dQuality = 0 then 'Good'
                   else 'Bad' end
      from rsdatapointsample
      where lRSDataPointId = convert(int,?TagId)
        and tTimeStamp < ?TimeStamp
      order by tTimeStamp desc
```

ReadBetweenSQL: This string allows data to be read between two supplied timestamps for a given tag id.

```
    Set RowCount 0;
    Select tTimeStamp,
      value = case when sValue is null then
    convert(varchar(20),dValue)
              else sValue end,
      status = case when dQuality = 0 then 'Good'
              else 'Bad' end
      from rsdatapointsample
      where lRSDataPointId = convert(int,?TagId)
          and tTimeStamp > ?StartTime
          and tTimeStamp < ?EndTime
      order by tTimeStamp
```

TimeSQL: This provides Plant Applications the current time while performing functions like storing data. This string must return a result set with the first column named CurTime. This string might look like:

```
Select getdate() as CurTime
```

7. Click the **Save** icon or close the **Edit Historian** dialog box and you will be prompted to save your changes.
8. After editing any of these settings, you must reload the following Plant Applications services: Reader, Writer, Event Manager, and Summary Manager. You will prompted to reload these four historian connect services upon exiting.
9. You should verify that your new historian is working and connected to the Plant Application Server.

Adding an OSI PI Historian

Proficy Server can still access the OSI PI historian. The two most important settings are the operating system and the remote port.

***NOTE:** Plant Applications cannot overwrite historical values of OSI PI string tags.*

To add an OSI PI Historian:

1. Select the appropriate server.
2. Click Server Management.
3. Right-click Historian Connections and click Add New Historian from the pop-up menu.
4. Type the name of the new Historian and press Enter.

You can type either the actual name of the server or use an alias.

5. Right-click the new Historian and select **Edit <historian name> Properties** from the pop-up menu.
6. In the **Edit Historian** dialog box, do the following:
   a. In the **Server Name** box, type the actual name of the server
   b. If necessary, in the **Username** box, type the user name required for connecting to the server.

If this historian uses Windows Authentication, you will need to add the domain to the front of the user name in this format: ***mydomain\username****.*

   c. If necessary, in the **Password** box, type the password required for connecting to the server.
   d. Select the proper operating system. For OSI PI, the port used to access the server is determined by the operating system.
   e. Select OSI PI as the historian type

Generally you will not need to check the Remote Data Source check box as this type of historian can be configured to run through the Plant Applications Server.

7. Click the **Save** icon or close the **Edit Historian** dialog box and you will be prompted to save your changes.
8. After editing any of these settings, you must reload the following Plant Applications services: Reader, Writer, Event Manager, and Summary Manager. You will be prompted to reload these four historian connect services upon exiting.
9. You should verify that your new historian is working and connected to the Plant Application Server.

Exporting Historian Configurations

When you export historian configurations, an Excel worksheet is automatically created containing the configurations for all historians, except for the default Historian (Local).

To export historian configurations

1. In the Plant Applications Administrator, expand **Server Management**.
2. Right-click **Historian Connections** and select **Export Historians**. The Configuration Export dialog box appears.
3. Ensure **All Historian Data** is selected and click **OK**. Excel automatically creates a worksheet containing the historian configurations.

Importing Historian Configurations

After exporting historian configurations, you can edit the configurations and import the changes into a Plant Applications server. If you edit the historian Alias, a new historian will be created. Editing any of the other columns will change the configuration of the historian.

To import historian configurations

1. Ensure the worksheet with the historian configurations is open and the configurations you want to import are selected.
2. In the Plant Applications Administrator, expand **Server Management**.
3. Right-click **Historian Connections** and select **Import Historians**. The selected historian configurations will be imported.
4. Click **OK** and refresh the server.

Connecting to Honeywell PHD

To connect to a Honeywell PHD data source

1. In the Plant Applications Administrator, click **Server Management**.
2. Right-click **Historian Connections** and click **Add New Historian**. An editable field appears under Historian Connections.
3. Type the name of the new Historian and press **Enter**.

You can type either the actual name of the server or use an alias.

4. Right-click the new Historian and select **Edit <historian name> Properties**. The **Edit Historian** dialog box appears.
    a. In the **Server Name** box, type the actual name of the server
    b. If necessary, in the **Username** box, type the user name required for connecting to the server.

*If this historian uses Windows Authentication, you will need to add the domain to the front of the user name in this format: **mydomain\username**.*

    c. If necessary, in the **Password** box, type the password required for connecting to the server.
    d. Select **Windows** for the operating system.
    e. Select **OPC** as the historian type.
    f. Select **Remote Data Source** and **Active**. For more information, see Remote Data Service.

***NOTE:** All OPC historians require that a Plant Applications Remote Data Service is running on the same box.*

10. Set the OPC options. Double-click the OPC options to edit the values.

***NOTE:** Most of these options will be supplied by your OPC server vendor and must match the values they supply. Those that are optional allow for some site customization.*

- **BrowseMethod** – Hierarchical
- **Prod Id** – OPC.PHDServerDA.1
- **Read Mode** – Cache
- **Read Windows (Seconds)** – 0
- **ServerExpectsUCT** – False
- **Start Time Setback (Secs)** – 0
- **Tag Filter** – 0
- **Threading Model** – Multi Threading

- **Write Method** – Real-time

11. Install the Remote Data Service (part of the client install) on the machine where the OPC server is running.
12. Set the RDS service to run under a domain account which has privileges on the network.
13. Set up DCOM for the RDSessionMgr and RDSHost applications. To do this:
    a. Start > Run > DCOMCNFG.
    b. In the list of applications, find the RDSession and RDSHost entries.
    c. Right-click to view the properties.
    d. On the Security tab, set up the Access and Launch permissions such that the Interactive, System, Network accounts have full control and also the account which is being used to run the RDS service.
14. Ensure PR2OPC.dll is in the System32 directory.

Testing

Configure a time-based historian variable using a valid OPC historian tag. The variable's data source must be historian. Set the interval to 5 minutes. After configuration has been completed stop and restart the RDS, Reader and Stubber services to pick up the changes.

Go to the latest Reader log file and to verify the loading of the historian configuration. The service should attempt to read the Historian tag configured. If there's a failure, an error will be returned.