

Human Machine Interface (HMI) & Supervisory Control and Data Acquisition (SCADA) Specification

*Version 8
August, 2004*

Table of Contents

1. PREFACE.....	5
2. REQUIREMENT NUMBERS	5
3. SOFTWARE ARCHITECTURE.....	6
3.1. GENERAL DESIGN FEATURES	6
3.2. DATA CONNECTIVITY	7
3.3. NETWORKING	7
3.4. MISCELLANEOUS	8
4. SYSTEM HARDWARE	9
4.1. SYSTEM UNIT	9
4.2. COLOR MONITOR	9
4.3. KEYBOARD AND MOUSE	9
4.4. PRINTER.....	9
5. GRAPHIC DISPLAY SOFTWARE MODULE	10
5.1. OBJECT CREATION TOOLS	11
5.1.1. Common Requirements	11
5.1.2. Line, Polygon.....	11
5.1.3. Rectangle, Square	12
5.1.4. Ellipse, Circle.....	12
5.1.5. Arc, Pie, Chord	12
5.1.6. Text	12
5.1.7. Images	13
5.1.8. Metafile	14
5.1.9. ActiveX Object.....	14
5.1.10. OLE Object.....	14
5.2. DRAWING UTILITY TOOLS	14
5.3. ANIMATION TOOLS.....	16
5.3.1. Dynamic Connections.....	16
5.3.2. Size Dynamic	16
5.3.3. Flash Dynamic	17
5.3.4. Digital Color Dynamic	17
5.3.5. Analog Color Dynamic.....	18
5.3.6. Hide Dynamic	18
5.3.7. Disable Dynamic	18
5.3.8. Rotation Dynamic.....	18
5.3.9. Dial Dynamic	19
5.3.10. Location Dynamic	19
5.3.11. Slider Dynamic	19
5.3.12. Animator Dynamic.....	20
5.3.13. Digital Selector Dynamic	20
5.3.14. Analog Selector Dynamic	20
5.3.15. Value Dynamic	21
5.3.16. Data Entry Dynamic	21
5.3.17. State Field Dynamic.....	22
5.3.18. State Field Data Entry Dynamic	22
5.3.19. Time/Date Dynamic	23

5.3.20. Display Switch Dynamic	23
5.3.21. Button Dynamic.....	23
5.3.22. Pick Dynamic.....	25
5.4. ALARM/TREND/DATA-MINING TIE-IN.....	25
5.5. SYMBOL STORAGE LIBRARY	25
5.6. EXPRESSIONS	26
5.7. LAYERS.....	26
5.8. SCRIPTING.....	27
5.8.1. Visual Basic for Applications	27
5.8.2. JScript.....	28
5.8.3. VBScript.....	28
5.9. GRAPHICAL HISTORICAL REPLAY FUNCTION.....	29
5.9.1. Graphical Historical Replay Tool.....	29
5.9.2. Unified Data Configuration Tool	29
5.10. OTHER REQUIRED FEATURES	29
6. TRENDING AND DATA LOGGING SOFTWARE MODULE	31
6.1. HISTORICAL DATA LOGGING	31
6.2. HISTORICAL DATA REPORTING	32
6.3. DATA TRENDING/CHARTING	33
6.3.1. Configuration	33
6.3.2. Runtime	35
6.3.3. Replay.....	36
7. ALARM MONITORING MODULE.....	37
7.1. ALARM SERVER.....	37
7.1.1. Digital Alarms	38
7.1.2. Analog Limit Alarms	38
7.1.3. Deviation Alarms	38
7.1.4. Rate of Change Alarms	38
7.1.5. Alarm Toggle Count.....	39
7.1.6. Time in Alarm	39
7.2. ALARM LOGGER	39
7.3. LIVE/ACTIVE ALARM VIEWING.....	40
7.3.1. Configuration	40
7.3.2. Runtime	42
7.3.3. Replay.....	42
7.4. HISTORICAL ALARM REPORTING.....	43
8. SCRIPTING LANGUAGE MODULE.....	45
9. SECURITY.....	46
10. PROJECT MANAGEMENT	48
11. WEB INTERNET AND INTRANET CONNECTIVITY.....	49
12. OPC DATA.....	50
13. DATA MINING.....	50
13.1. DATABASE CONNECTIVITY	50
13.2. DATA MINING ACTIVE X CONTROL TOOL	51
14. ALIASING.....	51
15. LANGUAGE SWITCHING	52

16. REDUNDANCY.....	52
17. TRAINING.....	53
18. TECHNICAL SUPPORT	53
19. SOFTWARE UPGRADES	54
20. SOFTWARE MANUFACTURER QUALIFICATIONS.....	54

1. Preface

The purpose of this document is to provide users with a detailed specification to include in various proposal and bid documents. This is a GENERAL HMI & SCADA SPECIFICATION for defining most functions that are typically used on industrial automation systems.

Although many projects may not require all the features described in this document or may require additional special features, this document should still be included as a whole in any proposal or bid without modification.

This document has been carefully compiled to ensure that the specifications will provide a system that has all the features herein. Modifying the document may provide loopholes for vendors that are unable to provide full functionality or create a specification that no vendor can meet, nullifying the specification and again opening up the possibility for vendors to bypass the specifications.

2. Requirement Numbers

Requirements in this document are individually identified with numbers to the left of the paragraph in the form: *RX.Y*, where *X* matches the section number and *Y* is a unique number identifying a specific requirement.

It uses the automatic list numbering feature used in Microsoft Word. Note that changing the order of the paragraphs, or adding/deleting requirements will change the numbers assigned to the rest of requirements below the change (later on in the document). So, once a document goes out for bid, care must be taken to keep changes from affecting these numbers. If new requirements must be added, it is suggested that they be placed at the end of the section, such that the *Y* numbers preceding the change are not affected.

These requirement numbers should be referenced by the bidders, identifying any non-compliance with this requirement specification.

3. Software Architecture

3.1. General Design Features

The HMI/SCADA package shall:

- R3.1 Be 32-bit software capable of running Windows 98, and Windows NT, and Windows 2000 and Windows XP. The system must provide options for the additional operating systems of Windows CE and Embedded systems.
- R3.2 Support and take advantage of multiple processors on the same machine (symmetric multiprocessor design). For example, adding another Pentium chip to the computer should distribute the load of the HMI/SCADA software across both chips, thereby increasing performance, etc.
- R3.3 Support and take advantage of multiple threads within the CPU. This multi-threading, among other benefits, aids in multi-tasking for CPU optimization, dependencies, etc.
- R3.4 Have a Modular architecture based on Microsoft's Distributed Internet Applications (DNA) architecture and plug & play features. It must not be one large monolithic software product.
- R3.5 Permit the user the flexibility to purchase only the modules and components necessary for the application. For example, if only alarming is required on one station, then only the alarming module should have to be purchased (and not the graphics, trending, etc. for that node).
- R3.6 Offer separate stand-alone modules for:
 - Dynamic Graphic Displays
 - Real-Time Trending and Historical Data Logging
 - Alarm Management
 - Project Level Scripting
 - Security
 - Real-time OPC Servers
 - OPC Redundancy
 - OPC Data Bridging
 - Screen Management
 - Process Control
 - Project Management
 - Visual Control Replay
- R3.7 Have a scalable architecture such that the user can start with a small application and later grow the application and database to any size by upgrading the license.
- R3.8 Embrace and integrate Microsoft Standards including:
 - Microsoft Access
 - Microsoft SQL Server
 - Microsoft Data Engine (MSDE)
 - Open Database Connectivity (ODBC)
 - ADO, OLE-DB for Database Interface
 - Object Linking and Embedding (OLE)
 - ActiveX Technologies

- OLE for Process Control (OPC)
- Visual Basic for Applications (VBA)

3.2. Data Connectivity

- R3.9 Connectivity to Data Sources shall be through a Universal Data Browser that provides the ability to specify the tag or point name using point and click selection with the mouse.
- R3.10 The Universal Data Browser shall serve as a single tag browser across all types of OPC tags (real-time and historical), global aliases, local aliases, language aliases, and local variables. Data Sources shall be divided into tabs and sub menus including:
- OPC Data Access including specification versions 1.0, 2.05 and 3.0 and OPC XML DA.
 - My Computer
 - Network
 - Internet
 - Favorites
 - Database Access
 - Any Databases available
 - Global Aliases
 - Any Aliases available
 - Graphics Displays
 - Any Graphical displays available
- R3.11 The Universal Data Browser tool shall be persistent and resizable.
- R3.12 The Universal Data Browser shall be multi-threaded

3.3. Networking

The HMI/SCADA package shall:

- R3.13 Implement Client / Server Distributed Networking
- R3.14 Support industry standard network protocols such as TCP/IP, NetBEUI using COM/DCOM.
- R3.15 Support OPC over IP and/or SOAP/XML, configurable on a per-node basis.
- R3.16 Operate on a network with multiple protocol stacks.
- R3.17 Plug and Play with standard routers, network Intranets and the Internet.
- R3.18 Offer a Web based Browser Plug-in so that Alarming, Graphics, Trends can be viewed from an Internet Explorer station. Also, it must permit interaction (e.g., changing set point, switching screens, etc.) from these Internet Explorer stations.
- R3.19 Permit users to perform any function in the SCADA Software from any node on the Network.
- R3.20 Be configurable as an independent distributed database regardless of the number of nodes in the system.
- R3.21 Allow configuration changes to the distributed database from any node on the system.
- R3.22 NOT require a file server. However, the HMI/SCADA software shall be capable of storing and managing its database on a redundant file server.
- R3.23 Support up to 256 workstations (nodes) or greater simultaneously. Expansion of the system shall be possible with the addition of non-proprietary PC based off the shelf hardware.

- R3.24 Connect to a large number of I/O Device Interfaces via OPC Servers across the network.
- R3.25 Offer a "working directory" feature, whereby multiple nodes can point to one common workstation for accessing and working with the project displays.

3.4. Miscellaneous

The HMI/SCADA package shall:

- R3.26 NOT require a hardware "dongle" key for licensing (like those plugged into the parallel port). Software-based licensing is all that should be required.
- R3.27 Provide an option for a hardware "dongle" key for licensing (like those plugged into the parallel port) as an alternative to any software-based licensing.
- R3.28 Allow such licensing to reside on each individual node in the system, or be served up from one central server (whereby all other nodes point back to that one station for its licensing).
- R3.29 Support Tool Tips, dockable toolbars, Context sensitive help and right click mouse function for easy context menu commands.
- R3.30 Be supplied as a complete package. No additional software should be required to configure or run the features of the system.

4. System Hardware

4.1. System Unit

The system shall consist of stand-alone personal computers, IBM PC compatible, with the following configuration as a minimum:

- R4.1 The CPU shall be an Intel Pentium or equivalent running at 400 MHz or faster. A minimum of a Pentium III running at 950 MHz or better is recommended for large implementations.
- R4.2 A minimum of 64MB RAM memory shall be supplied, a minimum 512MB is recommended for large implementations.
- R4.3 A Hard Disk Drive with a capacity of 6 GB or larger, with 11ms or less access time.
- R4.4 An 8X CD-ROM Drive (for software installation, etc.)
- R4.5 At least one Floppy Disk Drive: high density 1.44Mb 3.5-inch
- R4.6 A printer port must be included.
- R4.7 All Communication Port Hardware necessary to interface to the field devices shall be included.
- R4.8 Communication cards and interconnecting hardware shall be provided to connect the SCADA/HMI computers to a Local Area Network.
- R4.9 The Operating System already installed shall be Microsoft Windows NT 4.0, 2000, XP or Windows Server 2003.

4.2. Color Monitor

A high resolution SVGA Color Monitor shall be furnished with:

- R4.10 A 21-inch diagonal screen size.
- R4.11 Video resolution at least 1024x768.
- R4.12 A minimum of True 16bit colors supported.
- R4.13 Energy-Star Compliant.

4.3. Keyboard and Mouse

- R4.14 A full 80-101 key ASCII keyboard shall be furnished.
- R4.15 The keyboard shall be the standard QWERTY style, including cursor control keys, numeric keypad, and at least 10 special function keys.
- R4.16 A standard Microsoft mouse, or equivalent, shall be provided for point-and-click system configuration, color display creation, and runtime function selection.

4.4. Printer

- R4.17 A color ink jet printer suitable for documenting the graphic displays shall be provided.

5. Graphic Display Software Module

The graphical display module shall:

- R5.1 Allow configuration of highly detailed screens with animation.
- R5.2 Permit using the mouse for object creation, editing, and placing on the screen.
- R5.3 Show object properties for customization with either menu choices, right-clicking, pop-up menus, or double-click shortcuts.
- R5.4 Offer floating, dockable tool bars with drawing and animation tools for building the display.
- R5.5 Provide a grid that can be displayed on screen to assist in aligning objects precisely.
- R5.6 Permit re-sizing of the grid in both the X and Y direction (independently) in one-pixel increments
- R5.7 Allow "Snap to Grid" functionality to be turned on/off during configuration.
- R5.8 Not require compilation of the displays before use; a simple "save" is all that is needed for use.
- R5.9 Permit having multiple instances of the development environment open at the same time within the same PC.
- R5.10 Permit drag-n-drop of static and dynamic objects between the multiple instances of the development system for moving and/or copying of parts of the display.
- R5.11 Offer Password Protection of displays created.
- R5.12 The system shall allow multiple graphic displays to be viewed on the screen simultaneously.
- R5.13 All graphic displays shall be able to be configured as either "Fixed scale", or as "Scalable" which would allow automatic scaling of the display so as to maintain a full view of the graphics regardless of the window size and/or screen resolution.
- R5.14 Graphical displays shall have a maximum display size of at least 30,000 by 30,000 pixels.
- R5.15 Graphic displays shall have the ability to turn off their Menu and/or Title bar during runtime.
- R5.16 Graphic displays shall have the ability to make their window non-re-sizable during runtime.
- R5.17 An option shall be provided to enable a display (on a per-display basis) to "Always Be on Top".
- R5.18 An option shall be provided to permit one color (as selected by the user from a color pallet) to be "Transparent", enabling whatever is behind the active display to show through. Clicking on the transparent region will transfer such clicks/mouse-movements to the underlying window.
- R5.19 The system shall support networked "Portable Displays", whereby completed graphical displays can be copied to any other node without the need to edit and/or re-establish dynamic connections.
- R5.20 An option shall be provided to show tool tips during runtime (and configuration mode) that can be customized to show any combination of:
 - Tag used (for dynamic objects)
 - Process Value (for dynamic objects)
 - User defined text for help/description
 - Object name

5.1. Object Creation Tools

Standard object editing tools for creating and modifying graphic objects shall consist of at a minimum the requirements outlined below.

5.1.1. Common Requirements

The following is a list of requirements common to several "object" types, including lines, polygons, rectangles, ellipses, circles, arcs, chords, pies, and text. The graphics package shall offer:

- R5.21 Configurable line thickness
- R5.22 Configurable line style (solid, dash, dot-dash, etc.)
- R5.23 Configurable line color (selection from a palette of not less than 256 colors)
- R5.24 Toggle the "Fill" property of an object (so it's either solid, or transparent)
- R5.25 Configurable Fill Color of the object (selection from a palette of not less than 256 colors)
- R5.26 Configurable Fill Pattern of the object (selection from a palette of not less than 30 different styles; e.g., stripes, dots, waves, bricks, dashes, etc.)
- R5.27 A Gradient Fill (instead of a solid color, or a pattern) for an object, permitting the following:
 - Style of the gradient (horizontal, vertical, square)
 - Configurable number of detail steps used to create the gradient shades while blending the colors (e.g., 5 would be a "course" gradient, while 200 would be "very smooth")
 - Configurable Color of the fill (selection from a palette of not less than 256 colors)
 - Configurable direction, both X and Y, of the light source used for the gradient creation
- R5.28 Configurable Shadow and Shadow Color of an object (selection from a palette of not less than 256 colors)
- R5.29 Static rotation of the drawn object
- R5.30 Re-sizing of the object once drawn by using standard "resize handles" employed in common graphics packages.
- R5.31 Constraining the re-sizing of the object to the ratio it was originally drawn (so it maintains its shape during the resize edit) by holding down a modifier key (for example, while pressing the standard Shift key while dragging the mouse).
- R5.32 An option to copy just the "visible" properties of an object, and then pasting those appearance properties to other objects. This facilitates quickly making objects having a similar look and feel (for example, all having the same line thickness, gradient fills, shadow color, etc.).

5.1.2. Line, Polygon

The graphics package shall have a "Line Object" creation tool offering:

- R5.33 Lines drawn to any length, any angle using the mouse to define start/end points
- R5.34 Polygon creation with joined "multi-segmented" lines
- R5.35 Add, delete, and edit the anchor points of multi-segment lines
- R5.36 Creating lines (or segments of polygons) which "snap" to 45-degree increments by holding down a modifier key (for example, while pressing the standard Shift key while dragging the mouse).
- R5.37 "Joining" of individual lines into one polygon (which can be then filled, etc) when grouping individual segments into an object. This "merging" of segmented lines into a "polyline" must be at the user's choice.

5.1.3. Rectangle, Square

The graphics package shall have a "Rectangle Object" creation tool offering:

- R5.38 Rectangles drawn to any length using the mouse to define corner locations
- R5.39 Creating perfect squares by holding down a modifier key (for example, while pressing the standard Shift key while dragging the mouse).
- R5.40 Rounded corners which can be applied to any rectangle
- R5.41 Independent configuration of the horizontal and vertical curve extent used in rounded corners (to be configured graphically using the mouse).
- R5.42 A rectangle object shall be able to be converted to a "poly-line" equivalent (i.e., converted to a series of four segmented lines), so that these rectangles can be modified and/or merged with other poly-lines to form complex objects which can enjoy the Fill properties.

5.1.4. Ellipse, Circle

The graphics package shall have an "Ellipse Object" creation tool offering:

- R5.43 Ellipses drawn to any length using the mouse to define boundary locations
- R5.44 Creating perfect circles by holding down a modifier key (for example, while pressing the standard Shift key while dragging the mouse).
- R5.45 An ellipse object shall be able to be converted to a "poly-line" equivalent (i.e., converted from a curve definition to a series of segmented lines which approximate the curve), so that these ellipses can be modified and/or merged with other poly-lines to form complex objects which can enjoy the Fill properties.

5.1.5. Arc, Pie, Chord

The graphics package shall have an "Arc Object" creation tool offering:

- R5.46 Arcs drawn (i.e., a portion of an ellipse) using the mouse to define its length.
- R5.47 Creating arcs of a perfect circle by holding down a modifier key (for example, while pressing the standard Shift key while dragging the mouse).
- R5.48 The angles used for the arc are to be in at least single-degree increments, defined by dragging the start/end points using the mouse
- R5.49 Joining the endpoints of the arc will create an object that can have all of the Fill properties already described.
- R5.50 The endpoints shall have the option to be able to be combined by two methods:
 - Chord - connecting a straight line from each end point
 - Pie - drawing radii from each end point to the center of the ellipse, circle
- R5.51 An arc object shall be able to be converted to a "poly-line" equivalent (i.e., converted from a curve definition to a series of segmented lines which approximate the curve), so that these arcs can be modified and/or merged with other poly-lines to form complex objects which can enjoy the Fill properties.

5.1.6. Text

The graphics package shall have a "Text Object" creation tool offering:

- R5.52 Font family selection from the standard installed Windows True Type font list on a per-object basis.
- R5.53 Configurable Font Size on a per-object basis.
- R5.54 Configurable Font Style (Bold, Italic, Underline, Bold Italic) on a per-object basis

- R5.55 Multi-line text which remains as one object (i.e., it is NOT acceptable to have paragraphs always broken up into individual objects per line used ... it must remain as a multi-line paragraph which can be manipulated as a whole).
- R5.56 Text Alignment within the boundaries including Left-Align, Center-Align, Right-Align
- R5.57 Text size and extents can be stretched to any height or width using the mouse (thereby distorting the shape of the font).
- R5.58 An option to turn off additional stretching of the text (so the font maintains the shape drawn). Subsequent stretching of the object shall merely change the rectangle area used by the text object.
- R5.59 Applying a background color to the text, thereby creating a "Filled" text object (the Fill going to the extent of the rectangle defined by the area of the text).

5.1.7. Images

The graphics package shall have an "Image Object" creation tool offering:

- R5.60 Ability to import various types of images including:
- Windows bitmaps (*.BMP)
 - CompuServe GIF (*.GIF)
 - JPEG (*.JPG;*.JPEG)
 - Adobe Photoshop Files (*.PSD)
 - C64 Koala Graphics (*.KOA)
 - Dr. Halo (*.CUT)
 - Icon (*.ICO)
 - IFF Interleaved Bitmap (*.IFF)
 - Kodak PhotoCD (*.PCD)
 - Liner Bitmap Graphics (*.LBM)
 - Multiple Network Graphics (*.MNG)
 - Zsoft Paintbrush (*.PCX)
 - Portable Network Graphics (*.PNG)
 - Portable Network Media (*.PBM;*.PGM;*.PPM)
 - Sun Raster Images (*.RAS)
 - Truevision Targa (*.TGA;*.TARGA)
 - Tag Image File Format (*.TIF;*.TIFF)
 - Wireless Bitmap (*.WBMP)
 - Microsoft Visio Files (*.EMF)
- R5.61 Image size and extents can be stretched to any height or width using the mouse (thereby distorting the shape of the image).
- R5.62 An option to reset a distorted image back to its original dimensions.
- R5.63 Select a "transparent color" for an image (e.g. selecting blue would eliminate the display of that shade of blue within the image, thereby revealing any object behind the Image, making an "irregular shaped" image).
- R5.64 Support compression of simultaneous images for optimal storage and performance.

5.1.8. Metafile

The graphics package shall have a "Metafile Object" creation tool (Metafiles are Vector-Based graphics, as opposed to pixel-based Images) offering:

- R5.65 Ability to import Windows Metafiles (*.WMF) images into the display as an object
- R5.66 Metafile size and extents can be stretched to any height or width using the mouse (thereby distorting the shape of the Metafile).
- R5.67 The individual vectors defined in the Metafile can be converted to symbols native to the Graphics package (e.g., a clipart metafile of a house would be converted into the individual rectangles and lines used to draw the doors, windows, roof, etc). These converted objects can then be manipulated individually as any natively drawn graphic object would.

5.1.9. ActiveX Object

The graphics package shall have an "ActiveX Object" creation tool offering:

- R5.68 Ability to insert and run multiple ActiveX Controls from third party vendors.
- R5.69 Bring up the ActiveX Property Inspector for the embedded controls so that it can be customized for each instance.
- R5.70 Provide links to the scripting environment to directly manipulate the ActiveX Control and capture any Events it may trigger.

5.1.10. OLE Object

The graphics package shall have an "OLE Object" creation tool offering:

- R5.71 Ability to embed multiple OLE Documents from third party vendors (some examples include: Excel Spreadsheets, Word Documents, Wave Files, Movies, etc.)
- R5.72 Support for Drag-n-Drop of OLE Documents directly from the Windows File Explorer
- R5.73 Editing of the embedded OLE Object by changing the menus and tools shown in the Graphics package to those of the embedded object. (e.g., double-clicking on an embedded Excel worksheet would change the menus and tools to that of Excel, all while remaining in the Graphics package).

5.2. Drawing Utility Tools

Standard drawing utility tools for manipulating graphic objects shall consist of at a minimum:

- R5.74 A Color Palette supporting at least 256 colors, including standard colors with the ability to add custom colors to the palette. Custom colors shall be defined using the standard Windows color palette tools, and support "millions of colors" choices).
- R5.75 A Color Eyedropper tool whereby a color can be extracted from an object (e.g. an image) and placed as a custom color in the palette.
- R5.76 Group/Ungroup Tool to define if selected objects are grouped into one symbol or not.
- R5.77 Grouped symbols must still be able to be edited individually without destruction of the grouping or any dynamics associated with that group. For example, it should be possible to edit a pump symbol having color and pick dynamics on the pump as a whole (changing perhaps one of the individual line or arc objects used in drawing the pump) without first having to ungroup, and thereby losing the dynamics. All must be preserved.
- R5.78 Bring-to-Front/Send-To-Back Tools to move the Z-order of any object(s) forward or backward.
- R5.79 Delete Tool to delete selected object(s).

- R5.80 Alignment Tools to line up multiple selected objects based on:
- Tops
 - Left Sides
 - Right Sides
 - Bottoms
 - Middles in the Vertical Direction
 - Centers in the Horizontal Direction
 - Evenly Distributed Across, with boundaries defined by left-most and right-most selected objects
 - Evenly Distributed Down, with boundaries defined by top-most and bottom-most selected objects
- R5.81 Make Same Height Tool to make all selected objects the same vertical length as one of the selected objects.
- R5.82 Make Same Width Tool to make all selected objects the same horizontal size as one of the selected objects.
- R5.83 Make Same Size Tool to make all selected objects the same size (both height and width) as one of the selected objects.
- R5.84 Make Same Font Tool to make all selected text objects the same font as a master text object.
- R5.85 Search/Replace Tool to allow search and replace editing of:
- dynamic connection tag names (described later)
 - filenames used in various "Pick Action" dynamics (described later)
 - text used in Text Object labels
- R5.86 The "Scope" of the Search/Replace shall be user selectable as:
- Only those objects selected
 - The Entire Display
 - Multiple Displays as selected by the user (so an entire project can be updated at once)
- R5.87 The Search/Replace tool shall provide the ability to create a report of which objects were affected by the Search/Replace.
- R5.88 Zoom options for finer control of drawing and editing of objects, including:
- Predefined percentages for easy access (e.g., 50%, 75%, 100%, 200%)
 - Custom Percentage (user enters a number between 10% and 1000%)
 - "Rubber band" or "Box" Zoom (whereby the user defines a "rectangle" defining the area to zoom to by dragging the mouse).
 - Show the Entire Display
 - Zoom to the Selected Objects
 - Return to the "home" view, the default used for the display.
 - Zoom in/out around the mouse pointer by rolling the center Mouse Wheel (for mice so equipped)
 - Zoom shall be selection sensitive such that if an object is selected, the display automatically recenters the screen to the object. This also shall also apply to multi-selection display objects.
- R5.89 Auto-Panning feature, so that when objects are drawn/moved beyond the zoomed section of the display, the area auto-scrolls in the direction of the mouse drag.

5.3. Animation Tools

Standard dynamic/animation features, which can be added to the various graphic objects, shall consist of at a minimum the requirements outlined below.

- R5.90 Animation connections shall be constructed using an intuitive Point and Click interface or "Browser" which would support local and network, point configuration without additional user configuration.
- R5.91 Multiple dynamics of differing type can be applied to the same object (e.g., a size dynamic, and a color dynamic both added to a rectangle).
- R5.92 Multiple dynamics of the SAME type can be applied to objects (e.g., one size dynamic tied to one variable, and another size dynamic tied to another variable both added to a rectangle).
- R5.93 Only applied dynamics associated with the object shall be listed.
- R5.94 The option shall be available to only copy dynamic connections from one object to others.
- R5.95 A Format Painter (similar to MS Office Format Painter) shall be provided for transferring object dynamics and automation features from one object to other objects.

5.3.1. Dynamic Connections

- R5.96 Dynamic connections shall include the ability to browse for tags, eliminating the need to type the OPC Tag name.
- R5.97 Wherever a tag name can be used, an "Expression" shall also be permitted (Expressions are outlined elsewhere in this specification).
- R5.98 Wherever a tag name can be used, a "Local Variable" shall also be permitted. These local variables are unique for each display file, and shall NOT count against any "tag count" limit imposed by the license.
- R5.99 Wherever a tag name can be used, an "Alias" shall also be permitted (Aliases are outlined elsewhere in this specification).
- R5.100 Wherever a tag name can be used, a "Language Alias" shall also be permitted. The Language Aliases shall allow the user to substitute various language phrases based on a substitution table enabling an operator to read the item in his native language.
- R5.101 A most recently used tag name list shall allow quick selection of often-used tags.
- R5.102 Should a dynamic connection fail in getting a value from I/O, the dynamic must indicate such failure visually on the screen as follows:
 - Visual Dynamics on graphic objects (size, color, flash, location, etc.) must change the color of the bound object (color user-definable)
 - Data Values shown numerically must have an option to show a user-defined character (question marks or asterisks, for example) instead of the last known value.
- R5.103 Dynamics can be applied to either individual objects, or a group of objects.

5.3.2. Size Dynamic

- R5.104 A Size Dynamic shall be able to be applied to Line, Polygon, Arc, Ellipse, Circle, Rectangle, Image, Text, and Metafile objects, as well as grouped combinations of these objects.
- R5.105 The object shall grow in size in one of two methods (selectable on a per-object basis):
 - Scale (the entire object grows/shrinks relative to the analog signal it's connected to)
 - Clip or Reveal (the dimensions of the overall object remain the same, but the object is revealed relative to the analog signal it's connected to)

R5.106 The range of the analog connection through which the object can grow shall be configurable by the user.

R5.107 The starting and ending size, as a percentage, shall be configurable by the user.

R5.108 The size of the object(s) shall be able to be configured to change in the following directions:

- To the Left
- To the Right
- To the Top
- To the Bottom
- To any combination of 2 directions simultaneously (e.g., left and right growing from the center, top and right growing from the lower left corner, etc.)
- To any combination of 3 directions simultaneously (e.g., left, right, and top, etc.)
- In all four directions simultaneously
- To the Left or Right, with a "bias" in the middle (starting in the middle and either going to the left or to the right, but not both at the same time)
- To the Top or Bottom with a "bias" in the middle (starting in the middle and either going to the top or bottom, but not both at the same time)

5.3.3. Flash Dynamic

R5.109 A Flash Dynamic shall be able to be applied to Line, Polygon, Arc, Ellipse, Circle, Rectangle, Image, Text, and Metafile objects, as well as a grouped combination of these objects.

R5.110 The object shall flash based on the status of a Digital Signal (or the Boolean result of a logical expression).

R5.111 The user shall be able to select whether it flashes on the "True" or on the "False" condition.

R5.112 The object will flash using one of two methods (selected by the user):

- Hiding the object
- Changing the Object's Colors (either it's Line color, or Fill color, or both)

R5.113 The rate at which the object will flash shall be configurable by the user on a per object basis (range at least 50ms - 2 minutes)

5.3.4. Digital Color Dynamic

R5.114 A Digital Color Dynamic shall be able to be applied to Line, Polygon, Arc, Ellipse, Circle, Rectangle, and Text objects, as well as grouped combinations of these objects.

R5.115 The object shall change its color (either its line color, or Fill color, or both) based on the status of a Digital Signal (or the Boolean result of a logical expression).

R5.116 The user shall be able to select whether it changes color on the "True" or on the "False" condition.

R5.117 More than one Color Dynamic shall be permitted to be configured per object, each with its own independent Boolean expression to control the color dynamic. For example, one digital signal would change the object's fill color to red, while a different digital signal would change the fill color to yellow and the line color to black, while a third digital expression evaluates an analog signal to go above a threshold to change the fill color to green.

R5.118 If more than one digital color dynamic is assigned to the same object, the user shall be able to configure their order of precedence.

5.3.5. Analog Color Dynamic

- R5.119 An Analog Color Dynamic shall be able to be applied to Line, Polygon, Arc, Ellipse, Circle, Rectangle, and Text objects, as well as grouped combinations of these objects.
- R5.120 The object shall change its color (either its line color, or Fill color, or both) based on the value of an Analog Signal as it goes through its range.
- R5.121 The range of the analog connection through which the object will change color shall be configurable by the user.
- R5.122 The user shall be able to configure the Start color (corresponding to the low range) and the End Color (corresponding to the high range). As the Analog signal goes through this range, the object will change its color from the Start to End colors.
- R5.123 The user shall be able to configure if the object shall revert to the default color assigned to the object if the analog signal falls below or goes above the specified range.

5.3.6. Hide Dynamic

- R5.124 A Hide Dynamic shall be able to be applied to Line, Polygon, Arc, Ellipse, Circle, Rectangle, Image, Text, and Metafile objects, as well as grouped combinations of these objects.
- R5.125 The object shall hide based on the status of a Digital Signal (or the Boolean result of a logical expression).
- R5.126 The user shall be able to select whether it hides on the "True" or on the "False" condition.

5.3.7. Disable Dynamic

- R5.127 A Disable Dynamic shall be able to be applied to Line, Polygon, Arc, Ellipse, Circle, Rectangle, Image, Text, and Metafile objects, as well as grouped combinations of these objects.
- R5.128 The object shall become based on the status of a Digital Signal (or the Boolean result of a logical expression).
- R5.129 The user shall be able to select whether it becomes disabled on the "True" or on the "False" condition.
- R5.130 If this Disable Dynamic becomes active for any type of user data-entry dynamic (like data entry fields, sliders, dials, check boxes, radio buttons, push buttons, etc.), then that dynamic will no longer accept input from the user while it remains disabled.
- R5.131 The fill and line colors for objects that become disabled shall change to reflect that the object is now disabled.
- R5.132 The default "Disabled colors" for all objects on the display shall be light gray for Fill, and dark gray for Line, but these shall be user configurable.

5.3.8. Rotation Dynamic

- R5.133 A Rotation Dynamic shall be able to be applied to Line, Polygon, Arc, Ellipse, Circle, Rectangle, and Image objects, as well as grouped combinations of these objects.
- R5.134 The object shall rotate based on the value of an Analog Signal as it goes through its range.
- R5.135 The range of the analog connection through which the object will rotate color shall be configurable by the user.
- R5.136 The starting angle and ending angle shall be configurable by the user, in one-degree increments.
- R5.137 The pivot point around which the object rotates shall be configurable by the user.
- R5.138 Whether the object rotates clockwise or counterclockwise shall be configurable by the user.

5.3.9. Dial Dynamic

R5.139 A Dial Dynamic shall be able to be applied to Line, Polygon, Arc, Ellipse, Circle, Rectangle, and Image objects, as well as grouped combinations of these objects.

R5.140 The requirements outlined for the Rotation Dynamic shall also apply to the Dial Dynamic.

R5.141 The user will be able to click and drag the object, rotating it around the pivot point defined through the angles defined. As this rotation occurs, it will write values to the connected Analog tag, with the low range corresponding to the start angle, and the high range corresponding to the end angle.

R5.142 The user shall be able to select when the analog values are updated, either:

- As quickly as possible while the "dial" action is taking place (continuous updates)
- Only when the user releases the mouse click.

R5.143 The user shall be able to configure whether the rotation of the dial is smooth through the range, or uses just a number of discrete steps (detents) through the range.

5.3.10. Location Dynamic

R5.144 A Location Dynamic shall be able to be applied to Line, Polygon, Arc, Ellipse, Circle, Rectangle, Image, Text, and Metafile objects, as well as grouped combinations of these objects.

R5.145 The object change its location (move across the screen) based on the value of an Analog Signal as it goes through its range.

R5.146 The range of the analog connection through which the object will move shall be configurable by the user.

R5.147 The location dynamic shall follow one of the following paths:

- Horizontal
- Vertical
- Diagonal
- Multi-line Path

R5.148 The starting point and ending point defining the location extent shall be configurable by the user graphically on the screen by dragging with the mouse.

R5.149 Should a Multi-line Path be selected, the user shall be able to define, edit, and remove inflection points (nodes) in the line graphically on the screen by clicking/dragging with the mouse.

5.3.11. Slider Dynamic

R5.150 A Slider Dynamic shall be able to be applied to Line, Polygon, Arc, Ellipse, Circle, Rectangle, Image, Text, and Metafile objects, as well as grouped combinations of these objects.

R5.151 The requirements outlined for the Location Dynamic shall also apply to the Slider Dynamic, with the exception of Multi-line Paths (not required).

R5.152 The user will be able to click and drag the object, sliding it along the defined path defined by the start and end point. As this slide/translation occurs, it will write values to the connected Analog tag, with the low range corresponding to the start point, and the high range corresponding to the end point.

R5.153 The user shall be able to select when the analog values are updated, either:

- As quickly as possible while the "slider" action is taking place (continuous updates).

- Only when the user releases the mouse click. There shall be an optional configurable pop-up a dialog box for users to confirm data entry upon release of the slider, prior to downloading a value (i.e., and "are you sure you want to change Tag1 to 100" type of message with OK/Cancel buttons). The confirm dialogue box must include functionality for customizable text.

R5.154 The user shall be able to configure whether the location of the slider is smooth through the range, or uses just a number of discrete steps (detents) through the range.

5.3.12. Animator Dynamic

R5.155 An Animator Dynamic shall be able to be applied to Line, Polygon, Arc, Ellipse, Circle, Rectangle, Image, Text, and Metafile objects, as well as grouped combinations of these objects, when more than one such object is selected.

R5.156 The Animator Dynamic will cycle through all of the images selected at a user-specified "frame rate" (range at least 50ms to 2 minutes), thereby creating a picture slide-show animation.

R5.157 The objects shall become animated based on the status of a Digital Signal (or the Boolean result of a logical expression).

R5.158 The user shall be able to select whether the animation occurs on the "True" or on the "False" condition.

R5.159 When not animated, the user shall be able to configure what is shown, selecting either:

- All objects in the animation become invisible
- The first object in the animation sequence is shown
- The last frame animated/shown remains shown.

R5.160 The user shall be able to change the order in which the objects are shown (re-sequence the animation).

5.3.13. Digital Selector Dynamic

R5.161 A Digital Selector Dynamic shall be able to be applied to Line, Polygon, Arc, Ellipse, Circle, Rectangle, Image, Text, and Metafile objects, as well as grouped combinations of these objects, when more than one such object is selected.

R5.162 Only one of the objects within the grouping shall be shown at a time, with each object in the group having its visibility tied to the status of a Digital Signal (or the Boolean result of a logical expression). For example, one digital signal would make just object one be shown, while a different digital signal would make the just second object shown, while perhaps a third digital expression evaluates an analog signal to go above a threshold to make just the third object shown.

R5.163 The user shall be able to select whether each object hides on the "True" or on the "False" condition of its digital connection.

R5.164 The user shall be able to configure the order of precedence (in case two conditions occur simultaneously, the higher priority object would be shown).

R5.165 The user shall have access to the properties of each of the objects in the dynamic so changes can be easily made.

5.3.14. Analog Selector Dynamic

R5.166 An Analog Selector Dynamic shall be able to be applied to Line, Polygon, Arc, Ellipse, Circle, Rectangle, Image, Text, and Metafile objects, as well as grouped combinations of these objects, when more than one such object is selected.

R5.167 Only one of the objects within the grouping shall be shown at a time, with each object in the group having its visibility tied to a portion of a range of an analog signal.

R5.168 The user shall be able to configure how the range is broken up (by percentage) and assigned to the objects selected for the dynamic. The default will evenly divide the range into the number of objects selected.

R5.169 The user shall be able to configure the order in which the objects are shown.

R5.170 The user shall have access to the properties of each of the objects in the dynamic so changes can be easily made.

5.3.15. Value Dynamic

R5.171 A Value Dynamic shall be able to be configured, with its "visible" properties matching those of Text objects described earlier.

R5.172 The data connection shall support (and be configurable as) reading the following types:

- Byte
- Word
- Double Word
- Short Integer
- Long Integer
- Floating Point
- Double Precision Floating Point
- Boolean
- String

R5.173 The data connection shall support displaying the data as any one of the above mentioned types, as well as Hexadecimal, Octal, or Binary.

R5.174 For Floating Point and Double Precision data types, the user shall be able to configure how many decimal places are shown.

R5.175 For the String data type, the user shall be able to configure the number of characters displayed.

R5.176 For the Boolean data type, the user shall be able to configure whether a simple 1/0 is shown, or two different words (e.g., stop/start, off/on, open/closed, etc.)

R5.177 The user shall be able to specify whether leading zeros are shown (e.g., 23 vs. 0023).

R5.178 The user shall be able to specify whether thousands-separators are shown (e.g., 1000000.00 vs. 1,000,000.00)

R5.179 The thousands -separator, if shown, will conform to the local symbol (e.g., comma or period for US, Europe, etc.).

R5.180 The user shall be able to specify a text string to show for the units of the data being shown, and whether it is shown to the right or to the left of the number.

R5.181 The text string for units shall be able to be tied to a global alias or a language string (instead of just a constant text string).

5.3.16. Data Entry Dynamic

R5.182 A Data Entry Dynamic shall be able to be configured, with its "visible" properties matching those of Text objects described earlier.

R5.183 The requirements for Value Dynamic shall apply to the Data Entry Dynamic as well.

R5.184 The user shall be able to override the High and Low ranges used to define the values allowed to be entered into the field.

- R5.185 The user shall have the option to specify an Initial Value.
- R5.186 The user shall have the option to pop-up a dialog box confirming the user's data entry prior to downloading the Value (i.e., and "are you sure you want to change Tag1 to 100" type of message with OK/Cancel buttons). The confirm dialogue box must include functionality for customizable text.
- R5.187 The user shall have the option to pop-up a keypad for the data entry (thereby facilitating touch screen installations or other such pointing devices without requiring an actual keyboard).
- R5.188 The pop-up keypad shall be a numeric pad for all number data types, and automatically switch to being a QWERTY keyboard for String data types.
- R5.189 The pop-up keypad shall be user configurable for scalable sizing.
- R5.190 During runtime, the user shall be able to increase/decrease the value by pressing the up/down arrow keys, respectively.
- R5.191 The user shall be able to hit the ESC key, or simply clicking onto another object without first pressing the Enter key as the confirmation of the entry. Doing so shall cause the data entry field to revert to the value it contained prior to the user typing in values.
- R5.192 Confirm dialogue boxes must include functionality for customizable descriptive text.

5.3.17. State Field Dynamic

- R5.193 A State Field Dynamic shall be able to be configured, with its "visible" properties matching those of Text objects described earlier.
- R5.194 The text displayed shall be based on the value of the Analog Signal it is tied to.
- R5.195 The User shall be able to build a table with the state definitions. For example, a value of 0 corresponds to the string "Auto", a value of 5 corresponds to the string "Manual Mode", a value of 15 corresponds to the string "Clean Cycle Beginning", and so on.
- R5.196 A "default" string shall be able to be defined, which is shown in the event that the analog signal has a value not defined in the string definition table.
- R5.197 The maximum number of states defined per dynamic shall be 100 at a minimum.
- R5.198 The state table definitions shall be able to be exported to a standard ASCII text file so it can be edited by Excel, Word, etc. A file so edited shall be able to be re-imported back into the State Field Dynamic.

5.3.18. State Field Data Entry Dynamic

- R5.199 The State Field Dynamic described earlier shall have the option to be turned into a data entry field.
- R5.200 When clicking on such a field, the user shall be presented with a drop-down list of the states defined in the dynamic.
- R5.201 Selection of an entry in the list shall be by:
- Mouse
 - Up/down arrow keys
 - Typing in the first letter of an entry
- R5.202 Once selected, the corresponding numerical value shall be downloaded to the Analog Tag connected to the dynamic.

5.3.19. Time/Date Dynamic

R5.203 A Time/Date Dynamic shall be able to be configured, with its "visible" properties matching those of Text objects described earlier.

R5.204 It shall be configurable to show the current system date/time in the following manner:

- Current Date Only
- Current Time Only
- Current Date and Time (in that order)
- Current Time and Date (in that order)

R5.205 The Time format shall be able to be configured as follows:

- Any combination of Hours, Minutes, Seconds (e.g., just the hour, or just the minutes, or just seconds, or hour and minute but not seconds, all three, etc.)
- Military Time (24hour) or AM/PM format
- AM/PM indicator shown or not shown

R5.206 The Date format shall be able to be configured as follows:

- Any combination of Day, Month, Year (e.g., just the month, just the day, just the year, all three, etc.), and in any order (e.g., month/day/year, or day/month/year, or year/month/day, etc.).
- The separator character between fields shall be user defined
- The Day shall either be numeric, abbreviated string, or full name (e.g., 06, or Wed, or Wednesday)
- The Month shall either be numeric, abbreviated string, or full name (e.g., 12, or Dec, or December)
- The Year shall be configured to show either the last two numbers in the year, or the entire 4 numbers (e.g., 01, or 2001)

5.3.20. Display Switch Dynamic

R5.207 A Display Button Dynamic shall be able to be configured to, when clicked during runtime, either:

- Load in a whole new graphic display file
- Pop-up a separate window containing a display file (child window)
- Drag/Drop load the associated display file into a graphic ActiveX Viewer control

R5.208 A miniature "thumbnail" view of the associated display graphic shall be shown on the button.

R5.209 The user shall be able to configure the size of this image, choosing from at least 5 different sizes.

R5.210 An optional text label shall be shown (on either the top or bottom of the button), to help the user identify the display.

R5.211 Pop-Up Windows (child windows) must have the capacity to inherit the variables and alias values of the Parent display.

5.3.21. Button Dynamic

R5.212 A Button Dynamic shall be able to be configured, with its "visible" properties matching those of Text objects described earlier. Further, this function must allow an operator to click on a button to expose a data entry box where a data tag value may be set.

R5.213 The Button shall be able to be configured to, when clicked during runtime, either:

- Load in a whole new graphic display file. Parameters include filename with or without a path, and optional alias definitions.
- Pop-up a separate window containing a display file. Parameters include filename with or without a path, optional alias definitions, pop-up window position (centered to parent, last saved default position, or specific top, left, width, height parameters), and whether it is "Modal" or not.
- Drag/Drop load an associated display file into a graphic ActiveX Viewer control. Parameters include filename with or without a path, and optional alias definitions.
- Go Back (to the Previous Display), or Go Forward (similar to the forward/back arrows in the Internet Explorer).
- Launch a third-party application. Parameters include any EXE file name (or .BAT or .COM file) and optional command line strings for it.
- Launch a document (e.g., *.DOC, *.TXT, *.XLS, *.MDB, etc.).
- Launch a link to a web page (e.g., *.HTM, *.HTML, etc.).
- Close the current window
- Download a Value to an analog or digital tag. Parameters include whether it should be initiated via a right/middle/left mouse click, and whether the action should take place on the down-click, up-click, and/or while-held-down (in which case it is repeated at a configurable interval, range at least 50ms to 2 minutes). Additional configurable parameters must include the value(s) used to write, and the analog or digital tag to write to.
- Toggle a Value of an analog or digital tag between two values. Parameters include whether it should be initiated via a right/middle/left mouse click, and whether the action should take place on the down-click, up-click, and/or while-held-down (in which case it is repeated at a configurable interval, range at least 50ms to 2 minutes). Additional configurable parameters must include the two values to toggle between, and the analog or digital tag to write to.
- Run a Script. Parameters include whether it should be initiated via a right/middle/left mouse click, and whether the action should take place on the down-click or up-click and also includes the name of the script to invoke.
- Set Aliases in the current Display. Parameters include the alias definitions, which can be directly defined, imported from a text file, or directly loaded during runtime from the text file.
- Select a Global Alias Theme for the entire project.
- Select which spoken language is used (for multilingual installations).
- Manipulate the layers in the graphic by hiding them, showing them, or toggling their state. Parameters include whether it should be initiated via a right/middle/left mouse click, and whether the action should take place on the down-click or up-click and also includes the name of the Layer Name it should act upon.
- Pop-up a menu, which itself then contains an array of such buttons, each being able to take on the actions described in this list.
- Trigger a database update in a data mining ActiveX control that is connected to databases through ADO (OLEDB, Access, JET, TEXT or CSV) or a data mining Server.
- Trigger a report.

R5.214 A shortcut key shall be able to be configured for each Button, using a combination of Shift, Alt, and Control keys (or none) along with other keys on the keyboard.

R5.215 There shall be an optional configurable pop-up a dialog box for users to confirm data entry prior to toggling or downloading a value (i.e., and "are you sure you want to change Tag1 to 100" type of message with OK/Cancel buttons). The confirm dialogue box must include functionality for customizable text.

5.3.22. Pick Dynamic

- R5.216 A Pick Dynamic shall be able to be applied to Line, Polygon, Arc, Ellipse, Circle, Rectangle, Image, Text, and Metafile objects, as well as a grouped combination of these objects.
- R5.217 This dynamic shall offer the same capabilities as the Button Dynamic, thereby turning any object into a "button". The visible characteristics of the button would not be the text object described for normal buttons, but rather the visual graphic of the object(s) it is applied to.

5.4. Alarm/Trend/Data-Mining Tie-In

- R5.218 It shall be possible to include embedded Trend Views (both Real-time and Historical) from the Trending Module of the HMI/SCADA package.
- R5.219 It shall be possible to include embedded Alarm Views (both Live and Historical) from the Alarming Module of the HMI/SCADA package.
- R5.220 It shall be possible to include embedded Data-Mining Views from the Data-Mining features of the HMI/SCADA package.
- R5.221 More than one of each (Trend, Data-Mining and/or Alarm) can be included in any given display.
- R5.222 The user shall be able to configure the placement (Top, Left) and size (Width, Height) of each Alarm/Trend/Data-Mining window.
- R5.223 All of the features enjoyed by such windows with their respective Module shall also be available when embedded within the Graphics Module.
- R5.224 Icons to insert Trend, Data-Mining and/or Alarm views to any given display shall be through an ActiveX which is accessible through the main menu tool bar.

5.5. Symbol Storage Library

- R5.225 The graphics module shall include a symbol storage capability that supports both static and dynamic symbols.
- R5.226 The system shall be supplied with a standard set of graphic symbols in the library.
- R5.227 Symbols in the library shall be able to be dragged into a graphic display for use.
- R5.228 User created Symbols shall be able to be dragged into the library for storage
- R5.229 The library shall include the ability to create multiple categories in which to store the symbols. This will facilitate management of many symbols.
- R5.230 A preview of the stored symbols shall be provided to allow the user to view a symbol before it is selected from the library.
- R5.231 It shall be possible to make the preview thumbnail large (for seeing more detail), or small (for fitting more of them on a single page).
- R5.232 The symbol library shall be able to "Float" above the Graphics workspace, thus enabling the drag-n-drop of symbols with minimal screen switching.
- R5.233 The symbol library shall also be able to "dock" to any side of the graphics development environment as a toolbar.
- R5.234 The ability to lock a symbol category with a password shall be provided.

R5.235 Symbols shall have the ability to be linked together (by some shared keyword or the like). Objects sharing the same keyword shall be able to be updated on a global basis (whereby user selects the scope of the changes via a display list). This permits objects to have their look-n-feel visual properties, as well as their associated dynamics, updated throughout the project.

5.6. Expressions

R5.236 A built-in Expression editor shall be provided by the Graphics Package.

R5.237 Expressions shall be able to be used anywhere a Tag Name can be used.

R5.238 Functions provided by the Expression editor shall include:

- Open and Closed Parenthesis (for defining order of operation)
- Addition, Subtraction, Multiplication, Division, Modulus Division
- Relational expressions of greater than, less than, equal to, not equal to, less than or equal, greater than or equal
- Logical AND, Logical OR, Logical NOT operations
- Bit-wise manipulation including AND, OR, NOT, XOR, Bit Shift Left, Bit Shift Right, and Bit Test
- Mathematical functions for:
 - Conditional Function: if(BooleanCondition,ValueIfTrue,ValueIfFalse) Condition
 - Arithmetic Functions: Sine, Cosine, Tangent, Arc-sine, Arc-cosine, Arc-tangent, Square Root, Raised to Power of, Logarithm, Natural Logarithm, Exponential, Absolute Value, Integer Ceiling, Integer Floor, Minimum and Maximum
 - String functions: Wildcard String Compare, String Length, Substring Extraction, Left Substring, Right Substring, String Concatenation, String Search, Trim Left and Right, Trim Left, Trim Right, Lower Case Cast, Upper Case Cast and Type Conversion
 - Constant Functions: Math Constant PI, Math Constant E, Hexadecimal Constant, Octal Constant, Binary Constant
 - Conditional If, Then, Else statement
 - OPC Quality Testing (e.g., 192 - Quality GOOD) of values of OPC Tags
 - Ability to check for Division by Zero

R5.239 Direct access shall be provided to connect to OPC Tags.

R5.240 Numeric and String constants shall be able to be used in the expressions.

R5.241 . Wherever a tag name can be used, an "Alias" shall also be permitted (Aliases are outlined elsewhere in this specification).

R5.242 The Editor shall parse the entered expression to check for syntax errors.

R5.243 "White space" (i.e., tabs, spaces, CR/LF) shall be ignored, permitting better formatting for readability of the expressions.

R5.244 A list of the last 50 expressions used shall be available (e.g., perhaps via a drop-down list) to aid in expression development and re-use.

5.7. Layers

R5.245 The Graphics Package shall offer Layering

R5.246 The user shall be able to create and name at least 32 layers.

- R5.247 The user shall be able to change the order of the layers.
- R5.248 Layers above and/or below the currently active layer during configuration mode shall be able to be turned on/off to assist in display creation. A toolbar shall be available for easy access to this feature during configuration.
- R5.249 The ability to duplicate an active layer during configuration shall be provided.
- R5.250 The visibility of a layer during runtime shall be able to be tied to the status of a Digital Signal (or the Boolean result of a logical expression).
- R5.251 The visibility of a layer shall have the ability to be tied to the zoom factor used during runtime. For example, a layer which is only visible for zoom factors of 300% and greater would not be shown under normal view, but would rather only appear once the user has zoomed into a section of the display.
- R5.252 Dynamics on a hidden layer shall have the choice of being configured to either remain on scan (so their values are cached and ready to be shown when the layer is once again shown), or taken off scan (to reduce I/O overhead, etc.).
- R5.253 The visibility of a layer shall be easily tied to a Security Group or Level in the HMI software.

5.8. Scripting

- R5.254 The Graphics Package shall allow the creation of buttons and pick actions that, when activated (clicked-on by the user), will run a script.
- R5.255 The scripts will be “associated” with the object containing the pick action, such that if it is copied from one screen to another, or stored in the Symbol Library, the scripts will be stored along with the object.
- R5.256 Scripts can be disabled on start/load.
- R5.257 The user shall have a choice, on a per button/pick-action basis, of which type of script will be used, choosing from:
- Visual Basic for Applications
 - JScript
 - VBScript

Each of these three scripting choices is further explained in the subsections that follow.

5.8.1. Visual Basic for Applications

- R5.258 The Graphics Package shall provide Microsoft Visual Basic for Applications (VBA) as a scripting language
- R5.259 The graphic display shall fire off the following VBA Events during runtime (so such events can trigger a script):
- Data Entry (or Dial or Slider) Activated (to capture a user about to enter data into a field)
 - Data Entry (or Dial or Slider) Value Entered (to capture a user who has just entered data into a field)
 - Data Entry (or Dial or Slider) Deactivated (to capture a user leaving a data field)
 - Display Load/Unload (when the display is first called up, or exited)
 - Animation Ready for entire display (when all of the dynamics are ready for animation)
 - Animation Ready for just a specific layer
 - Animation Stop for entire display
 - Animation Stop for a specific layer

- Runtime Start/Stop
- Mouse Movement (optionally also turn this off, for better performance)
- Keyboard Key Press (optionally also turn this off, for better performance)
- Mouse Click (right, middle, and/or left) (optionally also turn this off, for better performance)

R5.260 The graphic display shall expose the following OLE Automation methods:

- Create any of the visual objects
- Add any of the dynamics to an object
- Manipulate the display (open, close, save, print, exit, change its size, etc.)
- Change the zoom percentage using various methods (percent, show whole view, fit to window, prompt user, etc.)
- Manipulate the layers in the display
- Close all associated pop-up windows with one command
- Manipulate the aliases used by the display

R5.261 The various properties and methods so exposed will make use of the VBA "Intellisense" help, so that context sensitive pop-up tool-tip help shall be available while creating the scripts.

R5.262 The graphic display shall also provide the ability to access its objects through OLE Automation in order to facilitate VBA programmers to gain access to their properties and methods.

R5.263 VBA Routines for creating/modifying symbols and/or their dynamics shall be able to be invoked during Configuration Mode. This will facilitate the ability to create "Wizards" for assisting in creating the displays.

R5.264 Any VBA Script associated with an object's pick-action or button dynamic shall be "Tied" to that object. Therefore, when the object is dragged into the Symbol Library, all associated VBA Scripting will go along with it. The same is true for copy-paste or drag-drop between multiple instances of the graphics package.

R5.265 There will be a tool to clean out any unused VBA Modules.

5.8.2. JScript

R5.266 The Graphics Package shall provide JScript as a scripting language.

R5.267 A JScript Editor will be provided in the Graphics Package, so script creation can be done within the application.

R5.268 The JScript Editor shall provide the ability to import and export scripts.

R5.269 Users will be able to define their own "keywords" and colors used in the JScript Editor.

R5.270 Graphic Displays with JScripts will be able to run over the Web in the Web Interface provided.

5.8.3. VBScript

R5.271 The Graphics Package shall provide VBScript as a scripting language

R5.272 A VBScript Editor will be provided in the Graphics Package, so script creation can be done within the application.

R5.273 The VBScript Editor shall provide the ability to import and export scripts.

R5.274 Users will be able to define their own "keywords" and colors used in the VBScript Editor.

R5.275 Graphic Displays with VBScripts will be able to run over the Web in the Web Interface provided.

5.9. Graphical Historical Replay Function

5.9.1. Graphical Historical Replay Tool

- R5.276 The Graphics package shall have available (as option) the capability to replay historical data in both graphical displays and trend displays.
- R5.277 The graphical historical replay tool shall be compatible with HDA (OPC Historical Data Access) databases.
- R5.278 The graphical historical replay tool shall replay historical alarms and events through the alarm viewer.
- R5.279 The graphical historical replay tool shall have “media player” style controls for the following playback functions:
- 4.1.1.Play – Replay of data at standard speed
 - 4.1.2.Fast Forward – Incremental speed increases for replay of data
 - 4.1.3.Slow Motion – Incremental decrease (from standard speed) for replay of data
 - 4.1.4.Reverse – replay of data in reverse direction
 - 4.1.5.Pause – freezes display of replayed data
 - 4.1.6.Single Step – from pause mode, allows incremental replay of data
- R5.280 The graphical historical replay tool shall have the capability for user selectable skins for various player control styles.
- R5.281 Both full and micro displays shall be available according to user preference.
- R5.282 The graphical historical replay tool shall include a built-in search utility to locate historical logged data
- R5.283 The graphical historical replay tool shall feature a display indicating replay status. The status parameters include; replay mode, replay speed, replay data time (H/M/S format), replay data date.
- R5.284 Date/Time displays in graphics used in graphical historical replay shall reflect the replay time graphical historical replay is active.

5.9.2. Unified Data Configuration Tool

- R5.285 To compliment the graphical historical replay tool, there shall be a graphical tool for configuration of the logging and alarm server functions
- R5.286 The Unified Data Configuration tool shall allow the selection of OPC tags from previously configured graphical display screens or by using an included unified data browser.
- R5.287 The Unified Data Configuration tool shall automatically generate the conversion from a real-time OPC DA tag to a historical OPC HDA tag for interpretation by the Graphical Historical Replay Tool.
- R5.288 The Unified Data Configuration tool shall facilitate configuration of both data logging configurator and alarm and event server configurator databases and generate the respective configuration files.

5.10. Other Required Features

- R5.289 The display should support the ability to enter the Runtime Environment from the Menu bar of the configurator.

- R5.290 The display should support the ability to enter the Runtime Environment from the selection of a shortcut by adding the phrase “ - runtime” to the end of the Target path.
- R5.291 During a runtime session, Security Login can be required with Security Groups and Levels determine how much accessibility operators have during Runtime.
- R5.292 During a runtime session, the user shall have the ability to select a display file from the Menu bar, based on Security Group or Level.
- R5.293 The display Menu bar shall be user configurable to be hidden based on Security Group or level during a runtime session.
- R5.294 During a runtime session, the graphical interface shall maintaining a history of the last 50 display files (similar to Internet Explorer’s back/forward feature) allowing the user to navigate through the display file history via “Display Forward” or “Display Backward” commands.
- R5.295 The display file history feature (forward/back commands) shall remember the initial alias settings specified when a display was opened (see description of the aliasing feature), and will reset those aliases when you go back to that display in the file history.
- R5.296 During a runtime session, the user shall have the ability to View display data statistics from the Menu bar, based on Security Group or Level.
- R5.297 During runtime, the user shall be able to move among the various data entry dynamics (sliders, buttons, dials, data entry fields, state field entries, pick actions) by either pointing to it with the mouse (or touch screen, etc.), or by tabbing to the field to enter data.
- R5.298 The order in which the objects are tabbed to shall be configurable by the user.
- R5.299 The mouse cursor shall change whenever the user hovers over an object with a dynamic for user input. An optional "focus rectangle" can be shown, whose colors can be configured.
- R5.300 The graphics editor shall have capability to create/edit templates and apply these on various displays. These templates can be saved without any related VBA code.
- R5.301 The graphics module shall offer a Print Preview option to show what the printout will look like prior to sending it on to the printer.
- R5.302 The graphics module shall offer a Print feature to specify on a per display basis if it is to be printed with a white background in order to save ink.
- R5.303 The graphics module shall offer several print regions, selectable from: current zoom level, a default/home view, all objects in the display, and un-zoomed to minimum size.
- R5.304 The graphics editor shall maintain a list of the last 9 displays that were edited (making it easier to switch between them).

6. Trending and Data Logging Software Module

The Trending and Data Logging Module shall:

- R6.1 Be based on OPC specifications by the OPC Foundation, being compliant not only with the OPC-Data Access spec as a client, but also OPC-Historical Data Access (HDA) as a server (as of this writing, the latest HDA specification is Version 1.2).
- R6.2 Be of modular design, with separate components for:
 - Logging Configuration
 - Data Logging to a Database
 - Data Trending/Charting
 - Historical Reporting

6.1. Historical Data Logging

The Data Logging Module shall be able to be configured to:

- R6.3 Log OPC Data and/or "Expressions" (Expressions are outlined earlier in this specification).
- R6.4 Collect the data at a user-defined frequency, ranging from 10ms to once every 24 hours.
- R6.5 First cache a user-defined amount of such collected data into RAM before writing the information to disk. This "block writing" to disk will permit disk-write optimizations, data compression, filtering, etc.
- R6.6 Define when the block writing to disk occurs, choosing from:
 - Time Interval (range at least 100ms to 24 hours)
 - Specific Time (e.g., every day at 8:30am)
 - Number of samples collected (range 1 to 10,000)
 - Condition/Event tied to an OPC Tag and/or an Expression (e.g., flush the buffer when a new part is made)
- R6.7 Apply optional data compression filters on the data as it is written to disk as follows:
 - The Maximum value
 - The Minimum value
 - The Average value
 - The Standard Deviation
 - A Running Totalization
 - The Running Maximum value (the maximum value over the entire logging period)
 - The Running Minimum value (the minimum value over the entire logging period)
 - The Running Average value (using an exponentially weighted moving average filter, giving more "weight" to newer samples)
 - The Moving Maximum value (the maximum value over a user-defined "sliding window" of time)
 - The Moving Minimum value (the minimum value over a user-defined "sliding window" of time)
 - The Moving Average value (the average value over a user-defined "sliding window" of time)
- R6.8 Use either the user-defined logging interval or a user-defined calculation period (range 100ms to 24 hours) for the above mentioned data compression filters.

- R6.9 Offer the choice of logging data only when it changes by a user-defined percentage (e.g., 1.3% of the value), or by a user-defined absolute threshold (e.g., 4 PSI).
- R6.10 Start the logging process (collecting to RAM, applying filters, writing to disk, etc.) based on:
- Time Interval (range at least 100ms to 24 hours)
 - Specific Time (e.g., every day at 8:30am)
 - Condition/Event tied to an OPC Tag and/or an Expression (e.g., begin data monitoring/logging whenever the machine is turned on.)
 - As soon as the Trending Software Module starts up
- R6.11 Stop the logging process (collecting to RAM, applying filters, writing to disk, etc.) based on different set of conditions, choosing from:
- Time Interval (range at least 100ms to 24 hours)
 - Specific Time (e.g., every day at 8:30am)
 - Condition/Event tied to an OPC Tag and/or an Expression (e.g., begin data monitoring/logging whenever the machine is turned on.)
 - Whenever the Trending Software Module exits
- R6.12 Log the data to the following open databases:
- Microsoft Access
 - Microsoft SQL Server
 - Microsoft Data Engine (MSDE)
 - Oracle 8.05, 8.10, and 9i
- R6.13 Log data to more than one of the above-mentioned databases at a time.
- R6.14 Store the database on either the local machine, or on a remote file server, or other such networked computer.
- R6.15 Define more than one group of data, at minimum permitting a dozen such data groups, each with its own logging start/stop times, calculation periods, filters, frequencies, and so on.
- R6.16 Contain more than 500 signals per group.
- R6.17 The storage of the configuration data itself shall also be stored in an open database (e.g., an MS Access *.MDB file).
- R6.18 The timestamps used for the logged data shall have the option of using Universal Time Code (UTC), as well as have an option for correcting for Daylight Savings Time.

6.2. Historical Data Reporting

The Historical Data Reporting component shall:

- R6.19 Extract Data out of the historical database(s) logged by the Data Logging Module for use in generating reports.
- R6.20 Allow selection of just a sub-set of the tags logged for inclusion in the report(s)
- R6.21 Include either the actual data logged, or apply one of the filters offered in the logging module to the already data logged (post-collection filtering). For reference, these include: Maximum, Minimum, Average, Standard Deviation, Totalization, Running Maximum, Running Minimum, Running Average, Moving Maximum, Moving Minimum, and Moving Average.
- R6.22 If a filter is selected, allow definition of time-interval for applying the filter (e.g., enter Days, Hours, Minutes, Seconds, and Milliseconds for the processing of the data).
- R6.23 Allow the report target name to be either a constant name, or tied to an OPC point.

- R6.24 Store the extracted data into a report in one of the following formats:
- Another Database Table, with the user identifying the Data Source Name (DSN)
 - An ASCII Text File, with user-defined file path and number of data value decimal places
 - An Excel File, with user-defined starting data row/column and optionally worksheet name.
- R6.25 Run a new report:
- Immediately (On-demand)
 - At a specific scheduled time (e.g., run the report only once at 9:30 on January 31st)
 - On a scheduled periodic basis (e.g., hourly, daily, on a specific day each week, monthly, and at which time during the selected interval)
 - Whenever a digital event occurs (connected to either a digital OPC Tag, or an Expression)
- R6.26 Configure multiple such reports, each with their own sets of parameters, and all running simultaneously
- R6.27 In runtime, the Report Module shall show the status of each report configured (is it enabled, when was it last executed, when is the next scheduled report, etc.)

6.3. Data Trending/Charting

- R6.28 The Data Trending/Charting component shall be an ActiveX Control, capable of being embedded within a graphic display file or other appropriate real-time ActiveX Container.
- R6.29 A separate Trend "container" application shall be provided to contain the ActiveX (so that it can run "stand-alone" without the need to depend on being embedded within the graphics container). It shall also support embedding third party ActiveX Controls. It shall offer a multi document environment with a VBA Scripting interface.
- R6.30 The Trend ActiveX shall be capable of saving and re-loading various configuration files to define its operation. Such files can be stored on the local hard disk, on the network, or be a URL web path reference. Details of what can be configured are described below.

6.3.1. Configuration

The Data Trending/Charting component shall comply with the following configuration requirements:

- R6.31 Multiple pens shall be able to be defined per Window (range at least 1 to 32 pens)
- R6.32 Each pen shall have the following parameters be able to be configured independently
- Color (supporting at least 256 colors)
 - Line style (solid, dotted, dashed, dot-dash)
 - Solid Line Thickness
 - Description (either as a constant string or tied to a global alias).
 - Units (either as a constant string or tied to a global alias).
 - Auto-scale setting
 - High and Low Range (either as a constant value or tied to an OPC Tag or Expression, or to a global alias).
 - Alarm-limit lines for High, Low, HighHigh, and LowLow (either as a constant value or tied to an OPC Tag or Expression, or to a global alias).
 - Whether the alarm limit lines should be plotted along with the pen line
 - Connection of points method (straight line between samples, or step/stair plot connection)
- R6.33 Pen connections shall be able to be tied to:

- Live OPC Tags
 - Live Expressions
 - Data collected in a RAM buffer (user-defined timeline) for instant plotting
 - Historical Data from the databases logged to by the Data Logger Module.
- R6.34 Both Live and Historical data shall be able to be plotted in the same trend window at the same time.
- R6.35 Historical pens shall be able to be turned into an "Ideal Trend Pen", allowing the historical data from log files to be anchored on the screen, overlapped by current real-time data.
- R6.36 The trend viewer shall be configurable to allow unlimited stacked plots within the same viewer.
- R6.37 Trend window shall support multiple ranges displayed at the same time.
- R6.38 The number of samples plotted shall be in the range of at least 6 to 6000.
- R6.39 Offer a user-defined plotting interval/rate (range 10ms to 24 hours per data point)
- R6.40 It shall be possible to create trend display windows in the following formats
- Time (Horizontal): Data values are plotted horizontally against time. Time is on the X-axis and data range is on the Y-axis. The data scale shall be configured to be either linear or logarithmic
 - Time (Vertical): Strip Chart Recording. Data values are plotted vertically against time. Time is on the Y-axis and data range is on the X-axis.
 - Circular Charts: Data values are plotted around a circular grid over time
 - X vs. Y: Data values are trended against a specified data pen, rather than time. The data range is on the Y-axis, with one of the pens having its values defining the X axis
 - Bar Plots: Each data value gets a bar distributed along the X-axis, with its value giving the bar height on the Y-axis.
- R6.41 Options shall be provided to show in an area adjacent to the plot:
- Ranges
 - Time stamp for each pen used, including optional milliseconds (Note: historical pens will have differing times!)
 - Date stamp (Note: historical pens will have differing dates!)
 - Pen data values
 - Pen data engineering units
 - Pen signal name (showing OPC name or expression used)
 - Pen description (user-defined name)
 - OPC Quality
 - User configurable major and minor tick marks on both X and Y axis
 - Configurable chart titles and labels on the X and Y axis
- The user shall be able to specify a title for the trend view, defining either a constant string or tying it to a global alias, a language alias or OPC tag.
- R6.42 The area used for the plot shall permit configuration of:
- Background Color and/or image.
 - Background images types supported shall include; bmp, jpg, jpeg and png.
 - Printing of the active background image shall be configurable.

- Grid (Independent X and Y Grids with properties of: Visibility, Color, thickness, line style, and the number of them used)
- Font (Independently set for detail list, ranges, timeline, and trend title)
- Pen Markers

R6.43 The order in which the pens appear in the list shall be changeable.

R6.44 The user shall be able to define if each pen is plotted in the same range window, or if multiple "stacked" plots (with pens being plotted in sub-region windows) are desired. If stacked plots are used, then the assignment of each pen to a given sub-region shall be user selectable.

R6.45 The trend viewer shall be configurable to allow customizable background printing.
The trend viewer shall be configurable to allow printing the viewer without any background, thereby saving ink.

6.3.2. Runtime

The Data Trending/Charting component shall comply with the following requirements while it is in "runtime" mode plotting the data:

R6.46 The plot shall be able to be "frozen" with the current data snapshot on the screen (with a button click or VBA Method). During this time, the data is still being collected for plotting in the background, ready to be shown once "un-frozen".

R6.47 While frozen, the user shall be able to drag the mouse across the plot, with the data in the "detail table" section (if shown) updating to reflect the data under the mouse cursor.

R6.48 While frozen, if the user hovers over a particular data point, a pop-up window shall show the details of that data point.

R6.49 While frozen, the user shall be able to scroll forward/backward in time, seeing data that may have been scrolled out of the view window.

R6.50 While frozen, the user shall be able to zoom in (with the data re-scaled automatically to fill the defined trend area) by the following methods:

- The mouse dragging a zoom box around a window of data
- A zoom button (each click zooms in a user-defined percentage)
- VBA Method

R6.51 Multiple zoom levels shall be permitted (i.e., zoom in further into an already zoomed plot)

R6.52 The trend shall support on-line configuration of the following:

- Pen parameters including: color, width, style, range, engineering units, description, number formatting, and alarm line placement
- Turning the Visibility of a pen on/off (if invisible, it is still collecting the data, just not showing it on the screen during that time, aiding in "de-cluttering" of the plot)
- Number of grids used for the X-axis and Y-axis

R6.53 The trend window shall have capability of adding new pens on the fly, while it is plotting existing data, by the following methods:

- A button click
- Drag-n-drop a data point from the Graphic screen directly into the plot area
- VBA Method

R6.54 It shall be possible to delete existing pens during runtime.

R6.55 The trending system shall support the realignment of historical data pens in a window (forward and/or backward in time specifying years, months, days, hours, minutes, seconds for the shift). Time shifting shall be accomplished by:

- Time entry into a pop-up dialog window
 - Dragging the pen with the mouse
 - VBA Method
- R6.56 It shall be possible to print out a trend plot.
- R6.57 When viewing a Historical Pen, the user shall be able to attach notes to a specific data value collected. Such notes shall be stored in the database along with the data value itself.
- R6.58 A mechanism for viewing any existing notes already stored with the data values in the database shall be provided.
- R6.59 A single dockable toolbar shall be provided for access to the following Runtime actions:
- Edit Pens
 - Edit Trend
 - Edit Period
 - Freeze Trend
 - Trend Statistics
 - Zoom
 - Paging Forward and Back
 - Set Time
 - Show Comments
 - Paint Trend
 - Save Trend
 - Dock Toolbar
- R6.60 The user shall be able to obtain statistics of the data shown in the window (either the live view, or a frozen zoomed sub-window). Included for each pen shall be:
- Minimum Value
 - Maximum Value
 - Average Value
 - Standard Deviation
 - Number of Samples
- R6.61 The statistics shall be able to be saved to a text file.

6.3.3. Replay

- R6.62 With optional replay module, it shall be possible to visually replay Data Trending/Charting within a graphical display.

7. Alarm Monitoring Module

The Alarm Module shall:

- R7.1 Be based on OPC specifications by the OPC Foundation, being compliant not only with the OPC-Data Access spec as a client, but also OPC-Alarms & Events (as minimum the alarm module shall comply with OPC A&E Specification version 1.1), being both a client and a server.
- R7.2 Be of modular design, with separate components for:
 - Alarm Configuration
 - Alarm Server
 - Alarm Logging to a Database and Printer
 - Live "Active" Alarm Viewing
 - Historical Alarm Reporting

7.1. Alarm Server

The Alarm Server Module shall be able to be configured to:

- R7.3 Monitor OPC Data and/or "Expressions" (Expressions are outlined earlier in this specification) for alarm conditions.
- R7.4 Contain help instructions on a per-alarm basis to assist operators in solving the alarms when viewing them.
- R7.5 Associate a default graphic display with each alarm. That way, when alarms are viewed in runtime, an operator could quickly jump to the associated display to view details on the alarm.
- R7.6 Enable monitoring of each individual alarm based on an OPC Data digital tag or the result of an expression (to aid in eliminating nuisance alarms, etc.) If the result is "true", then the alarm is monitored; if "false", then the alarm is ignored and won't even be generated.
- R7.7 Offer grouping the alarms so configured into categories and sub-categories (also known as Alarm Areas).
- R7.8 Have a delay time specified, on a per-alarm basis, that an alarm must be active before it is considered a true alarm. This delay can be defined as either a constant (i.e., 10 seconds), or tied to an OPC Tag or Expression.
- R7.9 Associate one or more (up to 10) OPC Tags / Expressions with each alarm as "supplemental" information to the alarm. For example, one could tie a "Batch ID" to a specific alarm. Then, when the alarm becomes active, a permanent record of the Batch ID is also stored along with the alarm information.
- R7.10 Run locally or over the network, supplying alarms and events to multiple alarm clients on multiple machines.
- R7.11 The storage of the alarm configuration data itself shall be stored in an open database (e.g., an MS Access *.MDB file). This database shall be accessible over the Internet for configuration, diagnostics and trouble shooting.
- R7.12 Send individual alarm acknowledgement bits to a PLC via the OPC server
- R7.13 Provide the following alarm types:
 - Digital
 - Analog Limit
 - Deviation

- Rate of Change
- Alarm Toggle Count
- Time in Alarm

7.1.1. Digital Alarms

R7.14 The alarm server shall generate an alarm when a digital signal goes into alarm.

R7.15 The user shall be able to configure:

- 4.1.7. Whether the Alarm State is 0 or 1, being either a constant value or tied to an OPC Tag and/or expression.
- 4.1.8. The Priority of the alarm, being either a constant value or tied to an OPC Tag and/or expression.
- 4.1.9. The message to display when the alarm occurs. This message can be either entered directly as a string constant, or tied to a global alias.
- 4.1.10. The message to display when the alarm returns to normal
- 4.1.11. Whether the alarm requires acknowledgment from the operator

7.1.2. Analog Limit Alarms

R7.16 The alarm server shall monitor analog signals, generating an alarm when it passes a limit threshold.

R7.17 The user shall be able to configure:

- 4.1.12. Individual alarm thresholds for four different alarm states: LowLow, Low, High, HighHigh
- 4.1.13. The alarm thresholds, being either a constant value or tied to an OPC Tag and/or expression.
- 4.1.14. A deadband through which the alarm must change before a change in state occurs, being either a constant value or tied to an OPC Tag and/or expression.
- 4.1.15. The Priority of each alarm state, being either a constant value or tied to an OPC Tag and/or expression.
- 4.1.16. Separate messages to display when each of the alarm states occur. These messages can be either entered directly as string constants, or tied to a global alias.
- 4.1.17. The message to display when the alarm returns to normal. This message can be either entered directly as a string constant, or tied to a global alias.
- 4.1.18. Whether each of the alarm states require acknowledgment from the operator

7.1.3. Deviation Alarms

R7.18 The alarm server shall be able to monitor two analog signals, comparing them and generating an alarm when they deviate from one another.

R7.19 The user shall be able to configure the same parameters for Deviation alarms as those available for Limit Alarms.

7.1.4. Rate of Change Alarms

R7.20 The alarm server shall monitor analog signals, generating an alarm when it changes faster than a defined rate.

R7.21 The user shall be able to configure:

- 4.1.19. The Rate of Change Limit, being either a constant value or tied to an OPC Tag and/or expression.

- 4.1.20. The Priority of the alarm, being either a constant value or tied to an OPC Tag and/or expression.
- 4.1.21. The message to display when the alarm occurs. This message can be either entered directly as a string constant, or tied to a global alias.
- 4.1.22. The message to display when the alarm returns to normal. This message can be either entered directly as a string constant, or tied to a global alias.
- 4.1.23. Whether the alarm requires acknowledgment from the operator

7.1.5. Alarm Toggle Count

R7.22 The Alarm Server shall be able to count the times a point goes into and out of an alarm state.

7.1.6. Time in Alarm

R7.23 The Alarm Server shall provide monitoring and display capabilities for the amount of time each alarm has been active.

7.2. Alarm Logger

The Alarm Logger Module shall be able to be configured to:

- R7.24 Print alarms/events to a printer (either local or on a network).
- R7.25 Print alarms/events to an optional backup printer (either local or on a network).
- R7.26 Log alarms to the following types of open databases:
 - Microsoft Access
 - Microsoft SQL Server
 - Microsoft Data Engine (MSDE)
 - Oracle 8.05, 8.10, and 9i
- R7.27 Log the alarms/events to a user-defined table name within the specified database
- R7.28 Select which fields/parameters of the alarms get logged/printed, selecting from:
 - Date/Time
 - Tag Name
 - Priority
 - Alarm Area
 - Alarm Message
 - Alarm Type/Category
 - OPC Quality
 - Node the Alarm was generated on
 - Acknowledgment Status
 - An Operator-entered Acknowledgment comment
- R7.29 Log the Time Stamp to a precision of 1 millisecond (time stamp as supplied by the OPC Data)
- R7.30 Filter which alarms are logged and/or printed. This alarm filtering shall be permitted on a per-alarm basis.

- R7.31 The Alarm Logger shall provide the capability to log up to ten associate up to 10 OPC Tags or Expressions with each alarm as "supplemental" information to the alarm. For example, one could tie a "Batch ID" to a specific alarm. Then, when the alarm becomes active, a permanent record of the Batch ID is also stored along with the alarm info.

7.3. Live/Active Alarm Viewing

- R7.32 The Live/Active Alarm Viewer component shall be an ActiveX Control, capable of being embedded within a graphic display file or other appropriate real-time ActiveX Container.
- R7.33 A separate Alarm "container" application shall be provided to contain the ActiveX (so that it can run "stand-alone" without the need to depend on being embedded within the graphics container). It shall also support embedding third party ActiveX Controls. It shall offer a multi document environment with a VBA Scripting interface.
- R7.34 The Alarm ActiveX shall be capable of saving and re-loading various configuration files to define its operation. Details of what can be configured are described below.
- R7.35 The Alarm ActiveX shall have the optional capability of interfacing with the security server to force a user login prior to acknowledgement of critical alarms.

7.3.1. Configuration

The Alarm Viewer ActiveX component shall comply with the following configuration requirements:

- R7.36 View window dimensions (height, width, top, left) shall be defined by simply clicking/dragging with the mouse.
- R7.37 Background color of the alarm window shall be defined from at least 256 colors
- R7.38 Lines (grids) separating the rows and/or columns shall be defined as to their color, width, line style, and visibility (shown or not).
- R7.39 Number of rows visible shall be user defined (range at least 2 to 50)
- R7.40 Each row shall contain one alarm message
- R7.41 A "multi-line" row shall be permitted, with the user specifying the number of lines (height) of each row (range at least 1-9 lines per alarm row)
- R7.42 Different Font Styles (font name, size, bold, italic, underline, color) shall be defined for the following 5 categories:
- Active alarms not yet acknowledged
 - Active alarms which are acknowledged
 - Alarms which have returned to their normal state, but which have not yet been acknowledged
 - Simple "System" Events
 - Tracking Events for following operator actions
- R7.43 Within each of the above five categories, additional font styles shall be able to be defined for different priorities. For example, unacknowledged alarms are white text on red background, Arial size 12; in addition, any of those which are also over 500 in priority should also be Bold, Underline, and overriding the size to 14).
- R7.44 Columns visible in the alarm window shall be configurable to include any number of the following, or user defined names::
- Date/Time
 - Tag Name
 - Priority

- Alarm Area
 - Alarm Message
 - Alarm Type/Category
 - OPC Quality
 - Node the Alarm was generated on
 - Acknowledgment Status
 - An Operator-entered Acknowledgment comment
 - Help Information
 - Time the Alarm has been Active
 - Additional Associated Values (e.g., other live data values) related to the alarm
 - Alarm Server Name
- R7.45 An overriding font style shall be able to be defined for a specific column. For example, the Help Text column might always be black text on a white background.
- R7.46 Column Text Alignment shall be user-defined as either Center, Left, or Right, on a per-column basis.
- R7.47 The Alarm View shall be sorted by a user-selected column, either in ascending or descending order.
- R7.48 Additional sub-sorting by another column shall be permitted. For example, first sort by alarm priority (highest at the top), then by area (alphabetical order), then by Tag name.
- R7.49 A column shall be able to be designated as the source for pop-up tooltips during runtime, and that column therefore be made invisible. For example, the Help column could be used so operators get context sensitive pop-up assistance while hovering over an alarm line.
- R7.50 The Alarm Viewer shall be able to be configured to get its alarms from either a local alarm server, or one on the network.
- R7.51 The Alarm Viewer shall be able subscribe to more than one such alarm server.
- R7.52 Filters shall be able to be defined to select which types of alarms should be shown in a particular alarm view. For example, show only alarms with a priority greater than 100, or show only unacknowledged alarms, or only operator tracking events, etc.
- R7.53 Alarm Filtering shall be configurable based on:
- Events vs. Alarms
 - Acknowledgment Status
 - Priority
 - Tag Name
 - Alarm Area
 - Alarm Type (limit, deviation, rate of change, digital)
 - Alarm Sub-Type (highhigh, hi, low, lowlow)
 - OPC Quality
 - Date/Time
 - Node
- R7.54 Multiple Alarm Filters shall be able to be configured for selection by the operator during runtime operation.

7.3.2. Runtime

The Alarm Viewer shall be an ActiveX component and shall comply with the following requirements while it is in "runtime" mode showing live/active alarms:

- R7.55 If more alarms exist than there are rows visible, then a scroll bar shall be provided such that operators can scroll through the listing.
- R7.56 If the operator scrolls down to another part of the listing and a new alarm comes into the system, a visual indicator shall be given (so the operator knows the view has changed beyond what he's looking at).
- R7.57 The user shall be able to select just one alarm in the list, or window a region to select multiple alarm lines at the same time.
- R7.58 The alarm selection will not affect the color of the alarm line, so that operators can still use color as an alarm indicator.
- R7.59 Alarms acknowledged at any one client shall be reflected on all other subscribed alarm clients on the network
- R7.60 When an alarm is acknowledged, the operator shall have the opportunity to enter in a comment to be stored in the log database. The capability shall be available to provide a pre-configured list of comments to select from.
- R7.61 Operators shall have access to viewing the "Operator Comments", and append additional comments to the database.
- R7.62 Acknowledgment of alarms shall be by:
 - Individual alarm, as selected by user
 - A group of alarms, selected by the user
 - All alarms belonging to the same Area
 - All filtered alarms
 - All alarms currently visible in the alarm view window
 - All alarms in the system
 - By alarm severity level
 - Optional setting -"Returned to Normal"
- R7.63 Rows and/or Columns shall be able to be resized, if such option was configured to be available to the operator.
- R7.64 Which column is sorted by shall be able to be changed during runtime.
- R7.65 By right-clicking on an alarm, the operator may select to jump to an associated display for that alarm (if one was configured for it).
- R7.66 More than one Alarm ActiveX Viewer shall be permitted, each with its own set of configuration parameters (e.g., subscriptions, filters, styles, etc.)
- R7.67 Individual alarms, configurable on a per alarm basis, shall be capable of acknowledgement on return to normal condition.
- R7.68 An optional pop-up message shall be available to alert the operator if they are attempting to acknowledge a previously acknowledged alarm.

7.3.3. Replay

- R7.69 With optional replay module, it shall be possible to visually replay the alarms states within a graphical display.

7.4. Historical Alarm Reporting

- R7.70 The Historical Alarm Report component shall be an ActiveX Control, capable of being embedded within a graphic display file or other appropriate real-time ActiveX Container.
- R7.71 Alarms shall be gathered from the Alarm Log Database, as created by the Alarm Logger component. This shall be from either the local machine, a networked PC, or via a URL web site.
- R7.72 These historical alarms shall be presented in one of for main report styles:
- Tabular view
 - Bar Chart view
 - Pie Chart view
 - Crystal Report view
- R7.73 The user shall be able to switch between the various report styles while viewing them.
- R7.74 The Font Style (font name, size, bold, italic, underline, color) used in the reports and the on-screen displays shall be user-defined.
- R7.75 Which fields (columns) are shown in the table shall be selected from those logged to the alarm database and the columns shall allow for user configuration, including renaming.
- R7.76 The user shall be able to sort the information based on any of the fields in the report, choosing from both Ascending and Descending sorts.
- R7.77 An option for “auto-refreshing” the information from the database shall be provided, with the user specifying the frequency of the report updates.
- R7.78 An Auto-Print feature shall be provided, with the report being printed whenever it is refreshed with new data.
- R7.79 The appearance of the Bar/Pie charts shall be configurable as to the size, plot area, optional title, and optional legend location.
- R7.80 For Bar and Pie Charts, the sections plotted shall be a summary of the number of alarms per either:
- Node
 - Alarm Area
 - Tag (Data Source)
 - Priority (Severity)
 - Date/Time
 - Alarm Type
- R7.81 Links to Crystal Reports shall be provided, permitting the user to create customized reports for alarm details and/or alarm summaries.
- R7.82 The configuration of the Crystal Reports shall not require the user to actually have Crystal Reports on the machine, but rather the software will provide a convenient mechanism whereby users can adjust such basic report items as:
- Report Title (including font selection, color, position, caption, etc.)
 - Fields to include in the report (these may be different than those used for the on-screen charts)
 - Field headers (including font selection, color, divider styles, etc.)
 - Paper Orientation (Portrait vs. Landscape)
 - Printout Margins (top, left, right, bottom)

- Alarm Filters (so only a sub-set of the records are used in the report)
- Record Sorting

R7.83 The ability to print the report during runtime shall be provided with a simple push of a button.

8. Scripting Language Module

- R8.1 A stand-alone project level Scripting Module shall be provided.
- R8.2 It shall use the industry standard Microsoft Visual Basic for Applications (VBA), thereby enjoying such required features as:
- Modular Code
 - On-line Debugger with breakpoints, instant data view, stepping in and through code, variable trapping watch points, and more.
 - Rich set of 100's of core commands (e.g., if, then, else, elseif, shell, case, do while, loop until, next, on error goto, instr, len, mid, right, left, print, math operators, concatenation, gosub, goto, exit sub, for each, dim, get, put, open, filecopy, close, mkdir, resume, and many more).
 - Multi-dimensional arrays
 - Direct interface to Microsoft Office products
 - Direct manipulation of ActiveX Properties and Methods
 - Capture of ActiveX Events
 - User-defined Forms for operator interface design
 - On-line IntelliSense tool-tip Help
 - A plethora of tutorials, training, and expertise in the marketplace
- R8.3 It shall be possible to configure multiple scripts, each with its own unique "trigger" to start its execution. These shall be based on:
- Time
 - Alarm Condition
 - OPC Data Event, based on the status of a Digital Signal or the Boolean result of a logical expression (Expressions are covered earlier in this document).
- R8.4 For Time triggered scripts, it shall be possible to base these on:
- Simple Interval (e.g., every 10 seconds)
 - Absolute Time/Date (e.g., July 1st at 10am)
 - Re-occurring Pattern (e.g., the 1st Wednesday of every month)
- R8.5 For Alarm triggered scripts, it shall be possible to base these specific filters, as described under the Alarm section of this document. For example, one script runs whenever an alarm in a specific area occurs, while another script runs whenever a HighHigh alarm with priority greater than 500 occurs)
- R8.6 For OPC Data triggered scripts, they shall be able to be configured only the first time the condition occurs, or every time the condition occurs, or continuously whenever the condition is true and gets updated changes in the underlying OPC Data Values.
- R8.7 An interface shall be provided to monitor the execution of the configured scripts, showing which ones are active, time of last run, and the ability to manually trigger a specific script.

9. Security

- R9.1 The software shall provide a Security component for restricting access to different areas of the system.
- R9.2 It shall be possible to configure different sets of individual users (range at least 1 to 1000 users) as well as categorizing those users (e.g., engineers, operators, supervisors, etc.) into groups (range at least 1 to 100 groups).
- R9.3 It shall be possible to assign a person to more than one group (e.g., Fred is both an Engineer and a Supervisor).
- R9.4 The names and groups shall be able to be optionally obtained from a Domain Server (NT4, 2000, XP) or the local PC's definitions.
- R9.5 If the names/groups are obtained from the Windows system, then a synchronization period shall be able to be specified (to check for new user names, etc.)
- R9.6 An option shall be provided so a user's information from logging into the Windows operating system (NT4, 2000, XP) can be automatically used within the HMI/SCADA system as well, thus only requiring the user to log in once.
- R9.7 For each individual user, and for each group, it shall be possible to define:
- Name
 - Alternate "Full Name" (e.g., username may be "Joe", but Full Name would be "Joseph R. Smith")
 - Optional Description
 - Password, including such parameters as minimum length required, time allowed before it must be changed, uniqueness over time, and password complexity).
 - Time they're permitted to log in, specifying hours on a per-day basis (e.g., log in only on Mondays and Wednesdays from 8am to 5pm).
 - Data Points which can and cannot be written to
 - Files which can and cannot be accessed
 - Workstations which can be used to log in from
 - A custom section to define security items not anticipated by the software, accessible from the VBA environment for customized security, or from other programming environments through OLE Automation calls.
- R9.8 Besides the per-user/group restrictions on writing to OPC Data Points, an additional list of OPC Data points shall be able to be identified as "Critical" points. Then, whenever a user attempts to modify the value of one of these points via the HMI interface, they will be immediately prompted to log in and re-verify themselves before the write operation is permitted.
- R9.9 It shall be possible to configure an Auto-Logout period, whereby the user is automatically logged out of the system after a specified amount of time has elapsed.
- R9.10 Should Auto-Logout be in effect, there shall be an early warning message prior to the logout action (time remaining configurable by the user).
- R9.11 It shall be possible to optionally require a password for logging out early (so someone doesn't log another user out against their wishes).
- R9.12 A utility shall be provided to show which users are logged into the system (both on the local machine, and on networked machines).
- R9.13 When the user is prompted for Username/Password for login, support for "Touchscreens" shall be provided (for example, offering a pop-up keyboard for entering the information).

- R9.14 If a user fails to successfully log in, it shall be possible to bring up a Graphics display offering user-configured options (e.g., re-attempt to login, go to a “home” screen, go back to previous display, etc.)
- R9.15 If a user fails to successfully log in after a configurable number of times, it shall be possible to lock that account out until either an administrator clears it, or after a preset amount of time has elapsed.
- R9.16 All log-in and log-out activity (name, time, node, etc.) shall be monitored and posted such that it can be logged to a database and/or printer. Such “Audit Trail” information shall be able to be viewed and reports generated by other components in the system.
- R9.17 Restrictions on software module interaction shall be configured on a per-user and per-group basis. Items to lock out include:
- Exiting the applications
 - Printing
 - Entering Configuration Mode
 - Switching Languages
 - Starting/Stopping of Alarm and/or Data Logging
 - Acknowledgement of Alarms
 - Alarm Filtering
 - Changing zoom factors in graphics and trends
 - Modifying and Executing Trend Reports
 - Adding, Editing and/or Deleting Trend Pens
 - Changing columns visible in Alarm and Trend viewers
 - Running a Script
 - Accessing the Windows Desktop
 - Executing each of the Pick Actions in the Graphics Module (configurable per type)
 - Opening other Graphics Displays

10. Project Management

A Project Management package shall be included that integrates all of the applications outlined in this specification into a single format or container, embedded in Microsoft Management Console. This will provide the capability to:

- R10.1 Be able to manage more than one project at a time.
- R10.2 Permit association of not only documents/files of the type provided by the HMI/SCADA software, but also integrate other file types as well (e.g., CAD drawings, images, text files, etc.)
- R10.3 Pack/Zip up all of the files required for a project into a single, compressed file format for easy deployment from one machine to another.
- R10.4 Unpack/Unzip a project back into the proper locations on the same or other PC.
- R10.5 Tie into Microsoft SourceSafe (directly from within the GUI for the Project Manager), permitting "versioning" of the each of the files used in the project.
- R10.6 Permit "batching" of the SourceSafe process (i.e., check in/out entire project at once)
- R10.7 Provide the ability to create a new project from an existing SourceSafe project.
- R10.8 Manage the publishing of Trend and Alarm configurations as well as entire projects to the Web (Intranet/Internet).
- R10.9 Generate reports on the resources used for each project (e.g., listing the files, types, etc.). Such reports shall be able to be created on either the entire project, or just the files associated with a particular application used within the project (e.g., Graphics Module)
- R10.10 The reports shall list all parts of the project, giving details as to when each file was created, modified, its size, as well as giving an summary of entire project disk space required.
- R10.11 Have an option to list all OPC Tag names used in a project, and flagging any points that cannot be verified.
- R10.12 Have optional project summary information field for such items as:
 - Project Name
 - Project Version Number
 - Author Name(s)
 - Last Modified By
 - Comments
- R10.13 Permit locking a packed project via a password (so it cannot be unpacked without proper authorization).
- R10.14 Serve as a location from where users can launch each application associated with a project.
- R10.15 Be able to "Start/Activate" the entire project (putting the various configured components into "runtime" mode, for example).
- R10.16 Offer links to tools for trouble-shooting a project (e.g., network connection diagnostics, OPC Server diagnostics, etc.)
- R10.17 Provide a utility to globally search and/or replace text throughout a project and all of its sub-documents including individual files.
- R10.18 Support Application configurations in Microsoft Access and SQL.

11. WEB Internet and Intranet Connectivity

- R11.1 A stand-alone Web Server software module shall be provided so the HMI/SCADA software can be accessed over the Internet.
- R11.2 The Web Server module shall run under Windows NT, embedded NT, Windows 2000, Windows XP and Windows Server 2003.
- R11.3 The Web Server module shall run on a Windows CE platform, supporting such processors as Hitachi, MIPS, Intel Strong Arm, and Intel XScale.
- R11.4 A user shall only require Internet Explorer to be installed on their machine (Thin Client) to access the system, with any "Plug-ins" automatically installed for them by the Server when they first access a graphic, trend, or alarm screen.
- R11.5 A mechanism shall be provided to deliver the software and data updates through firewalls.
- R11.6 The Web Server module shall manage the number of concurrent users of the Internet connection.
- R11.7 It shall be possible to have "floating" web licenses, whereby if one user logs off, that license is freed up so that another user can log in.
- R11.8 The same security system employed by the HMI/SCADA software shall also manage users connecting via the web.
- R11.9 It shall be possible to utilize the product's language switching feature to switch between written languages through the Web interface.
- R11.10 The graphics files shall be represented over the web in the same manner as shown on a "standard" HMI/SCADA workstation (i.e., close-approximations via HTML is NOT acceptable: the graphics should look and interact in the same fashion as a regular station). The scope of this shall include:
- Visual objects drawn
 - Visual Dynamics tied to these objects
 - User interactions (data entry fields, sliders, dials, etc.)
- R11.11 Anywhere a file path name can be used within the graphics software, a URL shall also be able to be specified. For example, the Working Directory can be either "C:\project1" or www.mywebserver.com, so a Display Load operation can go directly to a web server for the display file.
- R11.12 Alarms on an IE web station shall look and provide user interaction in the same fashion as a standard workstation. The scope of this shall include:
- Visual alarm summary
 - Alarm Viewer features such as pop-up help, jumping to associated displays, etc.
 - Acknowledgment of Alarms (resulting in the same actions as if the alarm were acknowledged on any other workstation in the system)
 - The alarm configuration database shall be accessible over the Internet for configuration, diagnostics and trouble shooting.
- R11.13 Trend View windows on an IE web station shall look and interact the same as a standard workstation.
- R11.14 It shall cost less per station to run via the Web than a standard workstation.
- R11.15 A "standard" workstation user shall also be able to connect to a networked system of other "standard" workstations via the Internet.
- R11.16 The Web client must support a native single-click feature to launch a graphical Historical Replay function.

12. OPC Data

- R12.1 The system will support OPC Data Access specifications version 1.0, 2.05 and 3.0 and OPC XML DA servers.
- R12.2 The system will support OPC Alarm and Events Specification Version 1.2
- R12.3 The system will support OPC Historical Data Access Specification Version 1.2
- R12.4 When configuring connections to OPC Data, the OPC Tag Browser feature in the OPC Specification shall be used.
- R12.5 The OPC Browser shall list all registered OPC Data Servers when making a connection to I/O data, and OPC Alarm and Event Servers when configuring alarms, and OPC Historical Data Access servers when configuring trending applications.
- R12.6 It shall be possible to browse OPC Servers installed on the local machine as well as those installed on any node visible on the network.
- R12.7 A list of OPC Items recently used shall be maintained, facilitating faster connections to common data items.
- R12.8 On-line OPC Data Configuration changes (e.g., adding a new tag on-line) shall be supported by the HMI/SCADA package.
- R12.9 OPC Data Ranges shall be read in and used to configure limits on graphic dynamics, trend limits, data entry field ranges, and so on.
- R12.10 OPC Quality Flags shall be used in graphic dynamics (non-good quality indicated by change in color used for objects), data logging, and alarm presentation.
- R12.11 A component for bridging/mapping OPC Data from one server to another shall be provided (e.g., map an AB data point to a Modbus register).
- 4.1.24. This capability shall run as an NT Service
- 4.1.25. Configuration shall be stored in a database for easy access
- R12.12 Connections to a database (e.g., MS Access, SQL Server) shall be able to be made via OPC.

13. Data Mining

Data Mining capability allows for the connection of data to and from various sources between and into other locations or modules.

13.1. Database Connectivity

- R13.1 The HMI/SCADA product shall allow connectivity between databases and the graphic display software module.
- R13.2 Read and write capability to (and from) the following list of spreadsheets and databases
- MS Access
 - MS SQL
 - Oracle 8.05, 8.10, and 9i
 - MSDE
 - MS Excel

- R13.3 The HMI/SCADA product shall include a database connection wizard tool and will support database object browsing.
- R13.4 The HMI/SCADA product shall incorporate a configurator tool for the data mining server.
- R13.5 The HMI/SCADA product shall support the ability to create connections to the database with the use of the Universal Data Browser (in Section 3.2). The database connections shall not be dependent upon pre-configuration connections or tags at the server level.
- R13.6 Support for indexing into tables via aliasing shall be included
- R13.7 Capability shall be included for easy access database tables for:
- single field read/write
 - column read/write
 - row read/write
 - two-dimensional array read/write
 - Simple SQL Commands
- R13.8 Capability shall be provided to:
- Access stored procedures
 - Execute complex SQL commands

13.2. Data Mining ActiveX Control Tool

- R13.9 The HMI/SCADA product shall include an ActiveX control tool to provide data mining functionality.
- R13.10 The data mining ActiveX control shall provide the capability to view whole tables, rows or columns from any open database.
- R13.11 The data mining ActiveX control shall connect to databases through ADO (OLEDB, ODBC, Access, JET, TEXT or CSV) or a data mining server.
- R13.12 The data mining ActiveX control shall support global aliasing.
- R13.13 This ActiveX will support automatic updates and refresh

14. Aliasing

Instead of connecting graphic dynamics, trend data, alarms, etc. directly to OPC Data tags, there should be a method of creating what some call “data pointers”, “indirect tags”, “data placeholders” or, as is designated in this document “aliases”.

- R14.1 The Graphics Module shall provide its own set of “local” aliases, which can be resolved within the graphic itself.
- R14.2 The HMI/SCADA system as a whole shall provide “project-level” aliasing, which will be designated in this document as “Global Aliasing”.
- R14.3 The global alias editor shall permit creating user-defined names for the aliases, permitting thousands of aliases per project.
- R14.4 It shall be possible to define “themes” for global aliasing. (For example, themes could relate to a product being made, a batch ID, a building number, etc.)
- R14.5 Each global alias will be tied to a theme, with each value for the alias being tied to a specific value/resolution of the theme. Therefore, simply changing the theme will change all of the associated aliases.

- R14.6 When setting a theme, it shall be possible to define the “scope” of the alias changes to the entire project, just the current application, or just the current document within that application.
- R14.7 It shall be possible to import/export the alias definitions from/to a text file (e.g., comma separated variable CSV file).
- R14.8 The configuration of the global aliases and themes shall be stored in an open database (e.g., Microsoft Access).

15. Language Switching

- R15.1 A native function for language-aliasing supported by a language configurator shall be provided by the HMI/SCADA software for multilingual language switching with automatic range and value scaling, as well as engineering unit conversions, formatting, and font selection. Language aliasing must apply to graphics displays, alarms and historical trends. Configurations data must be stored in an open database, such as MS-Access or MS-SQL Server.
- R15.2 Language Switching shall allow the user to configure the whole display or a specified object/symbol to switch.
- R15.3 Language Switching shall be available through Pick action-Load display, Pick action-Drag/drop load display, Pick action-Popup window, Pick action-Embedded window, Pick action-Set aliases, Pick action-Alias dialog, Tab Load Display, launching the display from the command-line and during a runtime session are modifiable via OLE automation.
- R15.4 Wherever a tag name can be used, a "Language Alias" shall also be permitted.
- R15.5 Confirmation Dialogue boxes must include the capacity to be language sensitive allowing the description of an object to utilize Language Switching.
- R15.6 The language configurator must support Czech, Dutch, English, French, German, Italian, Russian, and Spanish as default languages.
- R15.7 It must be possible to add a new language to the configuration database based on a Microsoft locale identifier (LCID), a 32-bit value that uniquely defines a locale. A locale is a set of user preference information related to the user's language. The locale determines how dates, times, currencies, and numbers are formatted, how items are alphabetically sorted, and how strings are compared.
- R15.8 Throughout the graphics, trending and alarm applications, it must be possible to configure language aliases in data fields (e.g. process points and trend pens) with a simple browser methodology with a tree control to simply select any language alias by double-clicking the alias name.
- R15.9 It must be possible to define a custom dictionary. The custom dictionary must be exportable and importable to and from various applications in XML and CSV formats.

16. Redundancy

- R16.1 An option for OPC Data Redundancy shall be provided by the HMI/SCADA software.
- R16.2 It shall be possible to define which node on the network is the “Primary” server station.
- R16.3 It shall be possible to define one or more additional nodes to serve as the “Backup” station.
- R16.4 Settings for determining when a node goes offline shall include:
- Frequency for server “alive” checks

- Response failure time required for server to be considered down
 - Reconnect attempt frequency
- R16.5 When the Primary node goes off line, the Backup station shall automatically take over serving the live OPC Data.
- R16.6 When the Primary node comes back on-line, there shall be an option to automatically switch back to that server, or keep the “Backup” server as the “new Primary”.
- R16.7 There shall be an option to provide switch-back confirmation.
- R16.8 The status of which node is the active server shall be provided to the Graphics package (for use in dynamic connections for operator indication).
- R16.9 It shall be possible to log redundancy switchover events.

17. Training

- R17.1 The software shall be easy to learn, such that the only prerequisites for attending training are familiarity with personal computers, and knowledge of the fundamentals of industrial/commercial measurement and control.
- R17.2 The software supplier shall offer training courses on at least a monthly basis.
- R17.3 There shall be both basic and advanced classes offered.
- R17.4 Professional trainers experienced in HMI/SCADA shall conduct the courses.
- R17.5 Training courses shall be optionally available on-site.
- R17.6 Any on-site training course curriculum shall be customized to the needs of the user if requested.
- R17.7 The training classes shall be “hands-on”, such that students can work with the software while instruction is taking place (e.g., learning labs to verify understanding of the taught material).
- R17.8 There shall be software-based tutorials included with the HMI/SCADA software (including, but not limited to, such things as template files, training displays, project examples, help files, and so on).

18. Technical Support

- R18.1 Technical support shall be available from a Factory Regional Office or local Representative/Distributor of the software supplier.
- R18.2 Factory technical support shall be available from technicians/engineers familiar with the software and typical industrial applications.
- R18.3 Technical support shall be available via:
- Telephone
 - Email
 - Fax
 - Web
- R18.4 Technical support shall be available from at least 8:00 a.m. to 7:00 p.m. EST weekdays, with optional support available 24 hours a day, 7 days a week.

- R18.5 The software vendor shall provide 24-hour availability of Internet ftp and World Wide Web (www) services.
- R18.6 Technical support shall be offered for free for 90 days from product shipment, which may be extended to one year if a product maintenance plan is purchased.
- R18.7 A utility for trouble-shooting and tracing the HMI/SCADA system shall be provided.

19. Software Upgrades

- R19.1 The vendor shall offer an Annual Software Maintenance Service for a nominal fee.
- R19.2 This service shall provide upgrades to the software for release of new features as they occur.
- R19.3 The vendor must have a record of providing a continuous upgrade path for the product.

20. Software Manufacturer Qualifications

- R20.1 As evidence of stability and market presence, the software manufacturer must have an installed software base in excess of 125,000 systems.
- R20.2 The software manufacturer must be an incorporated, established company, in business for at least EIGHTEEN (18) years.
- R20.3 Multiple user references must be provided.