

BID # FY 2016-2017-005
SUPERVISORY CONTROL AND DATA ACQUISITION
(SCADA) SOFTWARE

EXHIBIT A
TECHNICAL SPECIFICATIONS

Background:

We currently have two SCADA systems, one which is primarily used for our lift stations (DATAFLOW) and one that is primarily used for the Water Treatment Plant (GE-IFIX). Both systems are outdated.

Objective:

Have a single non-proprietary SCADA system for the Water Treatment Plant and our Lift Stations, etc.

A. General Requirements

1. SCADA software shall be commercially available off-the shelf and shall be non-proprietary, such that independent systems integrators are able to provide configuration and maintenance services as required.
2. Software shall be a Client/Server architecture. No Microsoft Client Access Licenses (CAL) or Terminal Services shall be required for full installation (thick) or browser-based (thin) clients.
3. Software shall offer options for unlimited, concurrent Thin Client connections for connectivity of common mobile device operating systems.
4. Software shall support running as a Microsoft Windows® service.
5. Software shall support running in a virtualized server environment.
6. Software shall support installation on a Storage Area Network (SAN) or Network-Attached Storage (NAS).
7. Software licenses shall be upgradable for an annual fee such that the client is able to download and install the current version of the product.
8. Software shall be tag-based and have an integrated development environment for creation of all aspects of the application.
9. Software shall be compatible with commercially available, off-the shelf PC hardware running Microsoft Windows client and server operating systems currently available at the time of installation.
10. Software shall not require dedicated server-level PC hardware for any individual system components.
11. Software shall support any computer running a thick copy of the software performing as both an application server and a user interface. Software shall support automatic server failover to an unlimited number of servers.
12. Software shall include the following integrated components available as standard components or for an optional add-on price. These components shall not require 3rd party software. However, the system shall allow 3rd party components to be used if required.
 - I. Online application development environment with version control.
 - II. I/O drivers for a wide selection of communication protocols.
 - III. Alarms management and alarms/events history.
 - IV. Fully-integrated historian.
 - V. Real-time and historical data trending and tabular views.
 - VI. Report generation.
 - VII. Security management.
 - VIII. Operator logbook.
 - IX. Support for networked applications.

- X. Support for server redundancy.
- XI. An object oriented scripting language with debugging tools.
- XII. Browser-based thin clients for PCs and Mobile devices.
- XIII. Alarm notification to off-site operators (email, text message and voice.)
- XIV. Interfaces for 3rd party software programs to access data (real-time and historical) and alarms via ODBC, OPC, or SNMP.
- 13. Software shall automatically compensate for deploying the same application simultaneously on a variety of monitor resolutions, while maintaining the aspect ratios of all displays.
- 14. Software shall protect against file corruption in the event of an unexpected loss of power or hardware failure. Software shall support an automatic, orderly shutdown when power levels drop to a user defined set point.
- 15. Software shall support automatic startup, upon computer restart, to full operation without user intervention.
- 16. Software shall provide a mechanism to backup and restore the entire application configuration.
- 17. Software shall include an integrated security system supporting an unlimited number of user accounts, roles and privileges. System users with appropriate account privileges shall be capable of changing the application configuration without requiring the software supplier's assistance. No lockout mechanisms or passwords shall be withheld from the final customer.
- 18. Integrated software help manuals shall be provided to assist operators and maintenance personnel with operational and configuration tasks.
- B. **Server Redundancy and Load Balancing**
 - 1. Redundancy
 - I. A minimum of two levels of redundancy for all application services shall be supported. Software shall support automatic failover from primary server to backup server(s). No manual intervention shall be required. This functionality shall be supported in both standard and virtual server environments.
 - II. Software must not require each redundant server to use a second network card to monitor the availability status of the primary server.
 - III. All servers shall be aware of which server is in control of each software process. No two servers shall perform the same function at the same time (e.g. I/O communications to a specific device, incrementing a totalizer.)
 - 2. Load balancing
 - I. Software shall support the assignment of specific services (e.g. alarms management, alarm notification) to specific computers.
 - II. Software shall support distribution of historical data storage to an unlimited number of computers.
 - III. Software shall automatically redirect incoming thin client connections to the server with the lowest number of active thin client connections.
 - IV. Software shall support redundant networks and shall be able to use these for load distribution when both are available.
- C. **Thick (full-installation) Client User Interface**
 - 1. Software licensing shall not limit the number of application display pages allowed.
 - 2. Software shall support the following navigation methods:
 - I. A menu for navigating from one display to another. Menu shall be configurable to allow logical grouping of displays.
 - II. Tiled view of any number of displays concurrently, all tiles showing real-time data or pictures of displays (user-selectable). Selecting one of the tiles will switch to the full-screen view of that display.
 - III. Mouse-over selectable hot box for navigating to a specific display.
 - IV. Button for navigating to a specific display.
 - V. Browser-like forward and reverse buttons to scroll through previously viewed displays.

- VI. A list of previously viewed displays with the option to navigate to a selection from the list.
- VII. A mapped view of assets with the option to navigate to an application display by selecting any asset from the map.
- 3. Application displays shall be event-driven, in that data will be delivered to client computers by the server immediately upon receipt. Client computers will not poll the server for new data.
- 4. Means shall be provided to allow the operator to print graphical displays.
- 5. Software shall support flagging tags as 'questionable data' (i.e. not commissioned or value is in question.) These values will continue to display the incoming values. These flags shall be removable by users with sufficient privileges.
- D. **Thin (browser-based) Clients**
 - 1. Software shall offer an optional zero-footprint thin client for use with HTML5-compatible browsers running on operating system variants, including
 - I. Windows
 - II. Linux
 - III. Apple
 - IV. Android
 - 2. Server for thin clients shall be an integral feature of the SCADA software. Use of the thin client shall not require a 3rd party Internet server software (e.g. Microsoft IIS, Apache).
 - 3. System shall support two levels of server redundancy for thin client connectivity, with automatic failover and client load sharing.
 - 4. System shall support cyber security measures including Firewalls, Virtual Private Networks (VPN) and Secure Socket Layer (SSL.)
 - 5. Thin client connections shall be concurrent. Tools shall be provided to monitor client connectivity and to disconnect users or switch them to alternate servers on demand.
 - 6. The thin client shall share the same security accounts as the rest of the SCADA Software. A separate privilege shall be required for browser client access. Revocation of this privilege will immediately terminate the user's client connection.
 - 7. Thin Client displays shall be generated automatically, requiring no additional configuration.
 - 8. On-line configuration changes shall be deployed immediately to all thin client interfaces without requiring the client interface to be restarted or refreshed.
 - 9. The Thin Client user interface shall be offered in two variants, the choice of which to use being user-selectable.
 - I. Graphical (preferred for large user interfaces) – A user experience mirroring (graphically and operationally) that of the Thick Client.
 - II. Text-based (preferred for mobile phone interfaces) – Simplified lists of monitored values, with support for control actions, alarms management, trending and mapped asset view.
- E. **Historian**
 - 1. Historian shall support logging of all the SCADA system data, including real-time, historical, transactions, alarms and events, regardless of the number of tags in the system. For example, a 25,000 tag system shall support historical data storage for 25,000 tags, plus alarms and events.
 - 2. Software shall include an integrated Historian and historical data storage at no additional cost, but may optionally use [Oracle,] [MySQL,] [Microsoft SQL Server,] [SQLite] as the database.
 - 3. The Historian and its historical data storage shall not require dedicated server computers, however, dedicated servers may be used if preferred by the customer.
 - 4. Historian shall be capable of logging up to 4,000 values per second.
 - 5. A synchronization scheme shall be included such that a copy of all historical data storage resides on all designated computers. Data shall be synchronized in real-time.

6. Any historical database that has been offline must be automatically resynchronized with the historian holding the most recent data. Software shall be capable of synchronizing 100,000+ values per second between databases. This process shall occur in parallel with runtime SCADA processes and shall be designed to minimize interference.
7. Distributed Historian architecture shall be supported. For example, two plants (A and B) may be responsible for historical storage for locally collected data, while a central facility (C) may be responsible for redundant historical storage for both A's and B's data. This architecture shall be scalable to support the addition of future plants.
8. Historian shall support SQL queries of logged historical data. Queries of historical data may be for raw data or summary (value at beginning, average, max, min, and delta) over a period.

F. Transaction Data

1. System shall provide a means to reliably collect and log (to historical data storage) transaction data from a field device. Transaction data is defined as a data set of related values stored with the same timestamp.
2. The transaction data capture process must include error checking.

G. Historical Data Display (trend plots and tabular)

1. Software shall display historical and real-time data in both plot and tabular format.
2. Software shall allow users to generate ad-hoc plots of historical data by clicking on each of the values to be trended. Selected groups shall be recordable for future recall.
3. Software shall support ad-hoc and preconfigured trend plots of real-time and historical data as integrated elements of graphical process displays. Historical and real-time plotted values shall be shown in a continuous, uninterrupted, scrolling fashion. The display shall support:
 - I. An unlimited number of pens (i.e. tag values), including both analog and digital values. Name and description of each tag shall be provided.
 - II. Analog pens as either layered or as stacked individual plots.
 - III. Time frame selectable from one second to five years.
 - IV. Scaling for each pen as high/low values or graduated divisions.
 - V. Value of each tag at the date/time of the mouse location.
 - VI. Statistical data, including average, minimum and maximum values, for each plot.
 - VII. Annotating trends
 - VIII. Stop/pause scrolling.
 - IX. Zoom in/out on the time (x) and value (y) axis'.
 - X. Pan/Scroll along the time axis or select a particular date to display.
 - XI. Move analog tag plots vertically (in the value (y) axis), either individually or as a group.
 - XII. Print displayed plot data.
 - XIII. Annotate a particular point in time on the plot.
 - XIV. Display alarm set points that are associated with the tag(s) as continuous marker lines across the plot.
 - XV. Export plotted data to comma separated value (.csv) file or directly to a database, for use by 3rd party data analysis software.
4. For tabular data, means shall be provided for the following;
 - I. An unlimited number of pens (i.e. tag values), including both analog and digital values. Name and description of each tag shall be provided.
 - II. Time frame selectable from one second to five years.
 - III. Stop/pause scrolling.
 - IV. Show raw data.

- V. Show summary data for a duration sampled at consistent periods (e.g. average value every 15 min for past 8 hrs.).
- VI. Export plotted data to comma separated value (.csv) file or directly to a database, for use by 3rd party data analysis software.

H. **Alarms and Events Management**

- 1. Software shall include alarms and events management tools consist with implementation of the ANSI/ISA 18.2-2009 Management of Alarm Systems for the Process Industries standard.
- 2. Software shall include a predefined alarms and events management interface. The interface shall provide the following operational tools.
 - I. Lists of Current, Unacknowledged, Disabled, Active, Configured Historical and Shelved alarms and events.
 - II. Tools for searching and filtering lists
 - III. Add a log-style note to any alarm
 - IV. Day and night view toggle
 - V. Acknowledgement of individual alarms or all visible alarms
 - VI. Font size adjustment for visually impaired users
 - VII. Representation of alarm priorities using shape, color and numeric priority level for enhanced operational awareness
 - VIII. Option to plot data associated with an alarm
 - IX. Option to open a process display where the alarm is currently shown
 - X. Alarm mute and silence
 - XI. Alarm shelving options for a defined period or indefinitely. Shelved alarms shall still be recorded to the alarms history but shall not annunciate or require acknowledgement.
 - XII. Alarm disable option
 - 3. Alarm acknowledgement shall immediately be propagated to all networked stations.
 - 4. Software shall allow alarms to be associated with functional areas, such that a user only has to deal with alarms in his/her functional area(s).
 - 5. Alarm occurrence, acknowledgement, clear, disable and shelve actions shall be recorded.
 - 6. Software shall provide user-configurable settings for deadband on analog alarms and delay on analog and digital alarms.
 - 7. Alarms and events records shall include;
 - I. Time/Date stamp.
 - II. The name and description of the alarm tag.
 - III. Priority.
 - IV. Status of Alarm (i.e. Active, Acknowledged, Cleared). Alarm Acknowledgement records shall include the name of the user who acknowledged the alarm.
 - V. The value of the associated tag at the time of alarm occurrence.
 - 8. Software shall support an unlimited number of alarm priorities and shall allow unique annunciation sounds and colors for each.
 - 9. Alarm annunciation shall be configurable to use alarm tones, text to speech descriptions, sound files or popup displays.
 - 10. Users must be notified, both visibly and audibly, of the occurrence of an alarm, regardless which display is presently being viewed.
 - I. Alarm System Auditing and Reporting
 - 1. Software shall support printing of alarms/events created over a range of dates/times.
 - 2. Software shall provide a means to identify frequently occurring alarms (e.g. nuisance alarms).
 - 3. Software shall provide a means to identify and analyze alarm flood conditions.

4. Software shall provide a summary of alarm percentages by priority, for comparison with generally accepted percentages as defined in the ANSI/ISA 18.2-2009 Management of Alarm Systems for the Process Industries standard.

J. **Alarm Notification System (for remote users)**

1. The Alarm Notification System shall support alarm notification via dial-out over voice modem (using text-to-speech), VoIP (either direct or via POTS to VoIP converter) SMS text message, email and alphanumeric pager. Alarm acknowledgement shall be supported during voice calls and via email and text message.
2. The Alarm Notification System shall allow status retrieval and alarm acknowledgement via dial-in over voice modem (using text-to-speech.)
3. The Alarm Notification System shall be an integrated feature of the SCADA systems and shall not require a separate tag database and security system.
4. Email messages shall support outgoing mail with transport layer security (e.g. Gmail, Yahoo Mail.)
5. The Alarm Notification System shall be capable of annunciating alarms to rosters of users with up to 30 contacts per roster. An unlimited number of rosters shall be supported. The system shall support associating rosters with functional areas, such that alarms in these functional areas will activate notification to the appropriate contacts.

K. **Mapping**

1. Software shall support common online mapping services (e.g. MapQuest, OSM, Google Maps) with zoom and pan features such that remote assets can be automatically mapped based on latitude and longitude coordinates.
2. Software shall allow for the creation of "sites" which are collections of I/O tags with a location that can be represented using Latitude and longitude coordinates.
3. Sites shall be user-definable to store and provide a configuration interface for any user-specified data.
4. Software shall include an integrated map display, wherein sites can be displayed at their latitude and longitude coordinates.
5. Software shall allow for user creation and selection of the icons used to represent sites on the map.
6. Software shall allow users to navigate the map including zoom and pan features.
7. Icons used to represent sites on the map shall be capable of displaying information about the site including current connection status and presence or absence of active alarms within the site.
8. Icons used to represent sites on the map shall include a built-in navigational link to either an automatically-generated page displaying all I/O tags within the site, or to a user-selected page.
9. Software shall provide a means to define and display pipes or other connections between sites.
10. Pipes or other connections between sites shall be user-configurable to include any relevant information about the connection.

L. **Security Management**

1. Software shall include a security system with privilege and role based user accounts. Level-based access shall not be acceptable.
2. Security system shall support an unlimited number of user accounts and roles. System shall allow creation of an unlimited number of additional security privileges where necessary.
3. User passwords must be configurable to require a minimum length, contain a combination of letters, numbers and special characters, and expire after a pre-set period. User passwords shall be stored in an encrypted format.
4. System shall allow changes to user accounts, roles and privileges while the application is running. Changes shall become effective immediately.

5. User logon and logout activity shall be recorded in the application event log. Disabling accounts after X failed attempts shall be supported.
6. System shall provide a mechanism to limit client access to specific IP addresses.
7. System shall support authentication of user accounts via a Windows domain and authorizing SCADA user roles from domain security groups.
8. System shall support the use of proximity cards/readers.
9. The integrated version control system shall allow the option of including or excluding security-related changes if returning the application to an earlier configuration state.

M. Report Generation

1. Authenticated operators shall, in the runtime environment, be able to produce simple reports including any analog, digital or calculated tag data from the historical database.
2. Data format options shall be as follows:
 - I. To screen.
 - II. To a comma separated value (.csv) file.
 - III. To a text file.
 - IV. To an ODBC-compliant database.
 - V. To any direct-connected or networked printer.
 - VI. To a 3rd party software (e.g. Microsoft Excel) template for advanced data analysis and formatting.
 - VII. To e-mail, if an external email server is provided.
3. Reports may be created for one-time use or saved for reuse.
4. The following reports should be included:
 - I. Analog Summary Report
 - II. Daily Snapshot Report
 - III. Daily Total Report
 - IV. Derived Flow Report
 - V. Detail Report
 - VI. Driver Communication Error Detail Report
 - VII. Driver Communication Summary Report
 - VIII. Hourly Snapshot Report
 - IX. Hourly Total Report
 - X. Rainfall Report
 - XI. Pump Activity Report
 - XII. Pump Discrepancy Report
 - XIII. Standard (raw data) Report
5. Report generation shall be invoked either on demand, by a monitored event, or on a scheduled basis.
6. For reports that are created on a scheduled basis, a mechanism shall be provided to allow operators to re-create the last scheduled report.

N. Electronic Operator Notebooks

1. System shall support operator logbooks for recording ad-hoc notes or predefined notes as threads, in that notes can have associated comments.
2. Notebooks shall be searchable by keyword, user account and time/date.
3. Notebooks shall be color-coded for easy identification.
4. System shall support creation of an unlimited number of notebooks for association with system elements, such as equipment and trended pen groups.
5. All notes shall be encrypted and include the name of the user's account and the time/date of creation. All notes entered into the notebook shall be immediately viewable from all clients and servers.

6. System shall support the option of requiring operator authentication as part of the note creation process.
7. System shall support printing of notes by range of dates/times.
8. System shall support the export of notes for a selected range of dates, to a format that can be easily viewed / printed from any workstation without the need for a (HTML)

O. Integrated Development Environment

1. An intuitive graphical development environment with drag and drop tools shall be used for configuration of application displays. A ribbon bar along the top of the display, as common to Microsoft office tools, is preferred.
2. Displays shall support zoom and pan actions.
3. A large library of static and animated graphics shall be provided with the software. Software shall support dragging and dropping additional graphics into the library.
4. Standard features shall be included for the following;
 - I. Analog and digital inputs and outputs.
 - II. Retentive counters. Values will persist if power is lost and subsequently restored or if failing over to a backup server.
 - III. Retentive totalizers. Values will persist if power is lost and subsequently restored or if failing over to a backup server.
 - IV. Historical calculations (e.g. average flow over last hour.)
5. Software shall include pre-built displays for standard user interfaces. The following pre-built displays shall be provided as a minimum;
 - I. Alarm display that can be filtered by name and includes current, unacknowledged, disabled, shelved and history.
 - II. Trending and tabular viewing of historical data.
 - III. Report creator.
 - IV. Operator notebook.
 - V. Site map.
 - VI. Site list.
6. Software shall allow calculations to be associated with each graphic object to facilitate movement, visibility, sizing and rotation.
7. Software shall allow multiple objects to be saved as a template graphic. The following template capabilities shall be supported;
 - I. A template may be associated with a tag structure.
 - II. Each new instance of the template will inherit the properties of the template, such that changes to the template will automatically update all instances created from it.
 - III. The template may have any number of parameters, including tags and text values, which can be used to animate objects within the template. Each new object created from the template may include different parameters.
 - IV. Templates may be imported from other projects.
 - V. Copy/paste/rename/delete for any template.
 - VI. Ungrouping of any instance of the template.
8. Software shall support the creation of template displays. The following capabilities shall be supported;
 - I. A template display may be associated with a tag structure.
 - II. Each new instance of the template will inherit the properties of the template, such that changes to the template will automatically update all instances created from it.

- III. The template may have any number of parameters, including tags and text values, which can be used to animate objects within the template. Each new object created from the template may include different parameters.
- IV. Templates may be imported from other projects.
- V. Copy/paste/rename/delete for any template.
- 9. There shall be no limit to the number of animated graphics that can be used to represent the same tag value.
- 10. Software shall support background bitmaps on graphical pages.
- 11. Users shall be able to select all or a subset of the graphics on any display and see a list of the associated tags. The list of tags shall be modifiable individually or using search and replace tools.

I/O Drivers (Field Device Communications)

- 1. Software shall support an unlimited number of field devices and different I/O drivers in the same application.
- 2. Software shall include support for communications over [Serial port] [TCP/IP] [UDP/IP]
- 3. Software shall include, at no additional cost, I/O drivers for:
 - I. Protocols - [Aquatrol] [BACNet] [Bristol BSAP] [Ethernet/IP (CIP)] [Allen Bradley DF1] [Koyo/Automation Direct DirectNet] [DNP3] [Omron FINS] [Foxboro] [Granville Phillips] [Omron Hostlink] [Internet Bristol Protocol] [IDEC] [IEC 60870-5-101] [Mitsubishi] [Modbus ACSII, Embedded, TCP, RTU] [Enron Modbus] [Landis & Gyr] [ODBC] [Dexter Fortson Associates OpenLink] [Siemens OP/PG] [Opto22 Optomux] [Campbell Scientific Pakbus, Loggernet and RTU] [QNet] [Fisher ROC, ROC Plus] [Serial] [SNMP Client] [GE SNP, SNPX] [Square D] [SRTP] [Stevens Water Serial] [Surflin] [Toshiba].
 - II. Manufacturers - [Acromag] [Rockwell/Allen Bradley] [Aquatrol] [BACNet] [Bailey] [Campbell Scientific] [Control Microsystems] [CSE Semaphore] [DAQ] [Data Flow Systems] [Dexter Fortson] [Enron] [Emerson Fisher] [Foxboro] [General Electric] [Granville Phillips] [Hewlett-Packard] [IDEC] [IEC] [Landis & Gyr] [Lantronix] [Mitsubishi] [Motorola] [NMEA] [OSI Soft] [Omron] [OPC] [Opto 22] [PLC Direct (Koyo)] [Power Measurements Ltd.] [Quindar] [Schneider (Bristol Babcock, Control Microsystems)] [Schlumberger] [Siemens] [Sixnet] [Square D] [Stevens Water] [Surflin] [Texas Instruments] [Toshiba] [Woodward] [Zetron]
 - III. Other - [DDE Client], [API], [OPC Client to support OPC Servers from 3rd party software providers].
- 4. Software shall support the development of additional I/O drivers where necessary.
- 5. Software shall support multiple communications protocols over a single communications port.
- 6. I/O drivers shall support redundant failover to one of more server computers. Software shall support redundant physical links to any field device, such as primary connectivity via Ethernet and redundant connectivity via serial port. Redundant links shall support similar or different protocols.
- 7. Software shall provide tools for polling telemetry devices (e.g. RTUs) directly. Software shall allow real-time tuning of each device's polling frequency without interrupting the polling cycle or restarting the application. To optimize I/O communications for telemetry applications, the polling order shall be configurable and polling shall be asynchronous (if permitted by the remote telemetry unit.)
- 8. Software shall be capable of pooling modems connected to one or more servers, for use in I/O communications.
- 9. Tools shall include methods for monitoring communication statistics and reporting errors for each I/O driver. Software shall support radio diagnostics monitoring for radio modems (e.g. Dataradio/Calamp, MDS.)
- 10. Software shall support writing to multiple output tags via a single write request. This shall allow writing a set of default values to a set group of field device registers.
- 11. Software shall support rewriting the last written value to an output.

Q. **Tag Database**

1. A browser shall be provided for tag creation, modification and deletion. The tag browser shall include a summary of all tags' current values.
2. A tag export/import utility shall be provided to allow bulk tag changes using Microsoft Office tools.
3. Software shall support the creation of template tag structures to represent a typical piece of equipment (e.g. a pump, an engine.)
4. Tag configuration shall support the use of expressions that can set configuration parameters based on developer-defined rules.
5. Template tag structures shall be copied and pasted to create any number of equipment instances having similar structures, but differing configuration.

R. **Configuration Management**

1. Software shall be capable of on-line configuration. That is, changes to most aspects of the application (e.g. tags, displays, calculations, reports, trends, server lists) can be deployed in real-time without recompiling or restarting the application or restarting computers.
2. Software shall be capable of offline configuration, such that changes to most aspects of the application can be imported and deployed without recompiling or restarting the application or restarting computers.
3. Software shall be capable of testing configuration changes to tags and displays in the runtime environment before changes are deployed.
4. Users shall be able to deploy a set of changes either automatically or manually, with the option to select and deploy specific changes.
5. Software shall allow multiple users to configure an application simultaneously.
6. Any shut down client shall automatically download newly deployed changes from the server when the client is restarted.
7. All application servers and clients shall automatically synchronize with the primary application server. No manual file duplication shall be required.

S. **Application Version Control**

1. The software shall have an integrated version control system that automatically logs application configuration changes to an encrypted repository. Versions shall be auto-numbered. No manual changes to the repository shall be permitted.
2. A chronologically ordered summary of versions shall display the time and date when the version was created, the user who created the version and any comments entered by the user when deploying the version.
3. It shall be possible to determine which version is currently deployed on each client and server station.
4. The version history shall allow review of all changes applied within each version.
5. The version control system shall allow the following version management methods.
 - I. Switch (aka rollback) to a previous version of the application.
 - II. Reverse changes applied during a specific version.
 - III. Merge changes made local to a specific workstation.
 - IV. Option to include or exclude security changes made between the current revision and the target revision when switching.

T. **Multi-Plant Application Support**

1. Software shall allow centralized configuration, security management and monitoring of multiple sub-applications within one large application. The sub-applications may be connected via either LAN or WAN.

2. Each of the sub-applications must be capable of running autonomously in the event of network unavailability.
3. Software shall allow users to be authorized for access to information from one or more sub-applications, or all sub-applications, of the large application.
4. Software shall support limiting the sub-applications that can be accessed from a specific SCADA node.

U. **Information Technology (IT) Tools**

1. Software shall include server-to-server and server-to-client IP link monitoring for the SCADA network. Link failure shall generate an alarm.
2. Software shall include tools for monitoring of historical data storage.
3. Software shall include tools for monitoring of any modems and SMS appliances used for sending alarm notifications to remote operators.
4. Software shall include an SNMP Agent option for integrating with Network Monitoring Software.
5. Software shall include tools to monitor computer resources (e.g. CPU, virtual memory, drive space) with the ability to generate an alarm on an out of range condition.

V. **Product Upgrades, Support and Training**

The software must have a history of allowing applications that have been deployed on current technology (operating systems and PC hardware) to be upgradable to new technology with minimal change to the application. It must have a history of allowing applications that have been deployed on one version of the product to be upgradable to a new version with minimal change to the application.

3. Software manufacturer shall offer product support via phone, email, user forum and remote access methods (e.g. Remote Desktop.)
4. Software training shall be available from the manufacturer via classroom courses and self-directed study (e.g. workbooks and tutorials online.)

W. **SCADA Software License Requirements per Computer**

(Repeat for each computer to be licensed.)

Computer #N

Tag Limit: 5K

License Types

Runtime License (SCADA node for runtime and/or server use)

Development Runtime License (SCADA node for application development, runtime and/or server use)

Additional Components

Alarm Notification (SMS, Email and phone notifications)

Concurrent Thin Client Connections, quantity: Unlimited

Connectivity Pack (ODBC, OPC Server, SOAP connections to Historian)

Server Configuration

[Primary][Backup] server for All Application Services.