3

SCADA Software

3.1 SCADA communication architectures

3.1.1 SCADA system

A supervisory control and data acquisition (SCADA) system means a system consisting of a number of remote terminal units (RTUs) collecting field data connected back to a master station via a communications system (Figure 3.1). The RTU acquires the data from the field devices and undertakes any required local control functions. This enables the RTU to do the local real-time control functions autonomously and it passes the supervisory information to the central control station. The master station displays the acquired data and also allows the operator to perform remote control tasks.

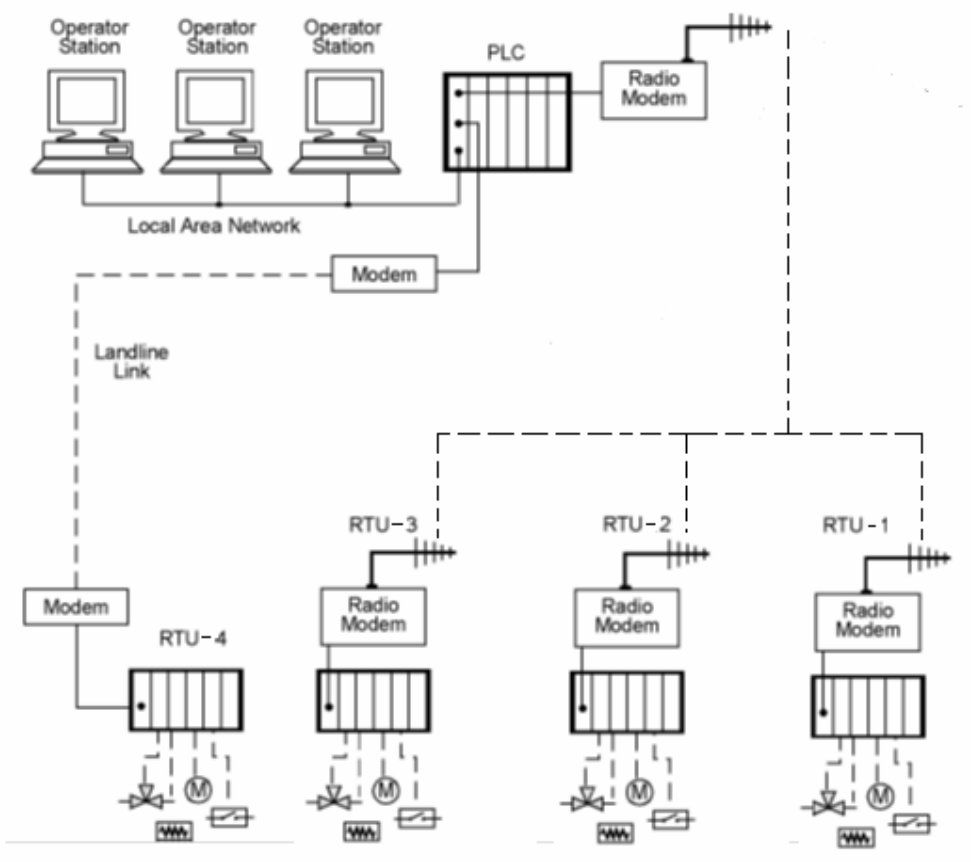

Figure 3.1
SCADA system

The communications system provides the pathway for communications between the master station and the remote sites. This communication system can be a radio, a telephone line, microwave, and possibly even satellite.

The master station (and submasters) gather data from the various RTUs and generally provide an operator interface for the display of information and the control of the remote sites. In large telemetry systems, submaster sites gather information from remote sites and act as concentrators relaying back to the master control station.

3.1.2 SCADA software

The SCADA software can be divided into two types: proprietary or open. Companies develop proprietary software to communicate with their hardware. These systems are sold as "turn-key" solutions. The main problem with these systems is the overwhelming reliance on the supplier of the system. Open software systems have gained popularity because of the interoperability they bring to the system. Interoperability is the ability to mix different manufacturers' equipment on the same system.

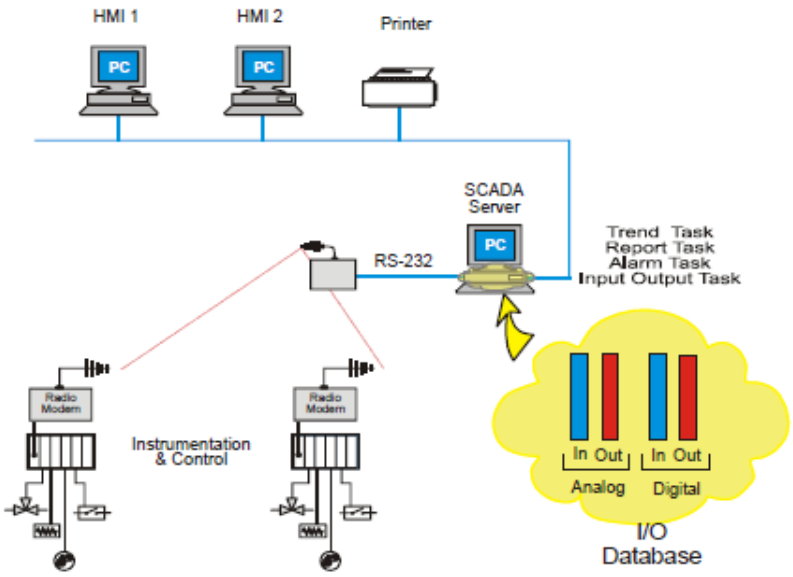

Figure 3.2
Typical SCADA system

Citect and **Wonderware** are just two of the many open software packages available in the market for SCADA systems. Some packages, including asset management, are now integrated within the SCADA system. The typical components of a SCADA system are indicated in Figure 3.2.

Key features of the SCADA software include:

- user interface
- graphics displays
- alarms
- trends
- RTU (and PLC) interface
- scalability
- access to data
- database
- networking
- fault tolerance and redundancy
- client/server distributed processing

3.1.3 SCADA hardware

A SCADA system consists of a number of RTUs collecting field data and sending that data back to a master station via a communications system. The master station displays the acquired data and also allows the operator to perform remote control tasks.

The accurate and timely data allow for the optimization of the plant operation and process. A further benefit is more efficient, reliable, and most importantly, safer operations. All these result in a lower cost of operation compared to earlier non-automated systems.

In a more complex SCADA system, there are essentially five levels or hierarchies as follows:

- field-level instrumentation and control devices
- marshalling terminals and RTUs
- communications system
- the master station(s)
- the commercial data processing department computer system

The RTU provides an interface to the field analog and digital sensors situated at each remote site.

The communications system provides the pathway for communications between the master station and the remote sites.

3.2 SCADA software blocks

While the performance and the efficiency of the SCADA package with the current plant is important, the package should be easily upgradeable to handle future requirements. The system must be easily modifiable as the requirements change, and expandable as the task grows – in other words, the system must use a scalable architecture.

There have been two main approaches that have been followed while designing the SCADA system in the past:

1. Centralized

Where a single computer or a mainframe performs all plant monitoring, all plant data are stored on one database, which resides on this computer (Figure 3.3). The disadvantages with this approach are as follows:

- initial upfront costs are fairly high for a small system
- a gradual (incremental) approach to plant upgrading is not really possible due to the fixed size of the system
- redundancy is expensive as the entire system must be duplicated
- the skills required of the maintenance staff in working with a mainframe type computer can be fairly high

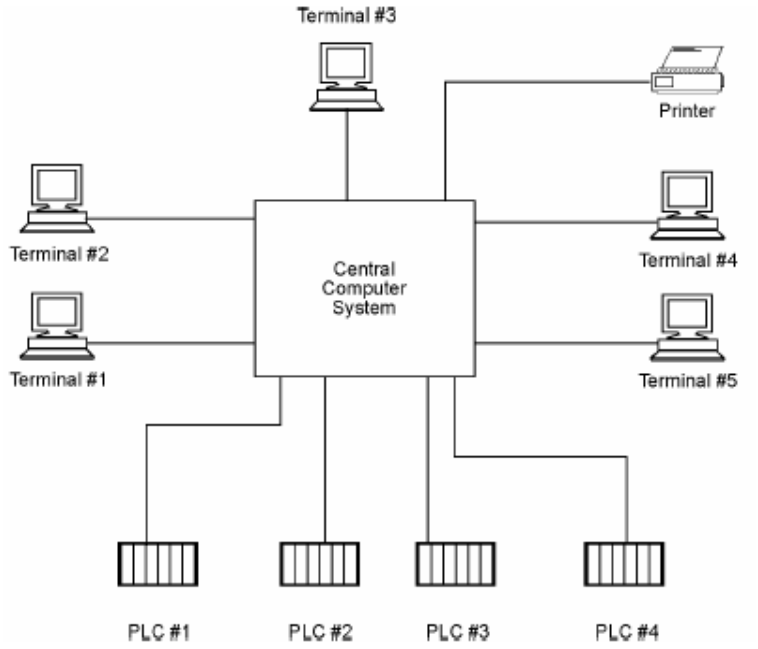

Figure 3.3
Centralized processing

2. Client–Server

A server node is a device that provides a service to other nodes on the network. A common example of this is a database program. A client on the other hand is a node that requests a service from a server. The words *client* and *server* refer to the program executing on a particular node.

A good example is a display system requiring display data. The display node (or client) requests the data from the control server. The control server then searches the database and returns the data requested; thus reducing the network overhead compared to the alternative approach of the display node having to do the database search itself.

A typical client–server implementation of a SCADA system is shown in Figure 3.4.

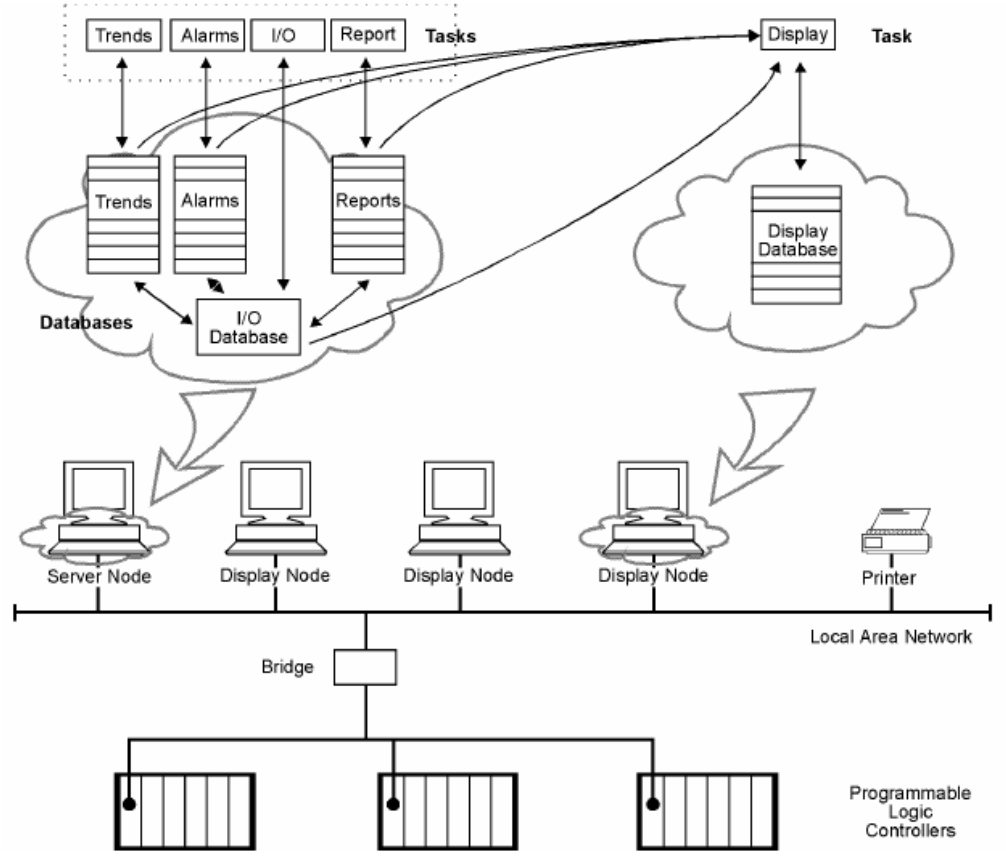

Figure 3.4
Client–server approach as applied to a SCADA system

There are typically five tasks in any SCADA system. Each of these tasks performs its own separate processing.

- **Input/output task –** This program is the interface between the control and the monitoring system and the plant floor.
- **Alarm task –** This manages all alarms by detecting digital alarm points and comparing the values of analog alarm points with alarm thresholds.
- **Trends task –** The trends task collects data to be monitored over time.
- **Reports task –** Reports are produced from plant data. These reports can be periodic, event triggered, or activated by the operator.
- **Display task –** This manages all data to be monitored by the operator and all control actions requested by the operator.

A large system with 30,000 points could have an architecture indicated as shown in Figure 3.5.

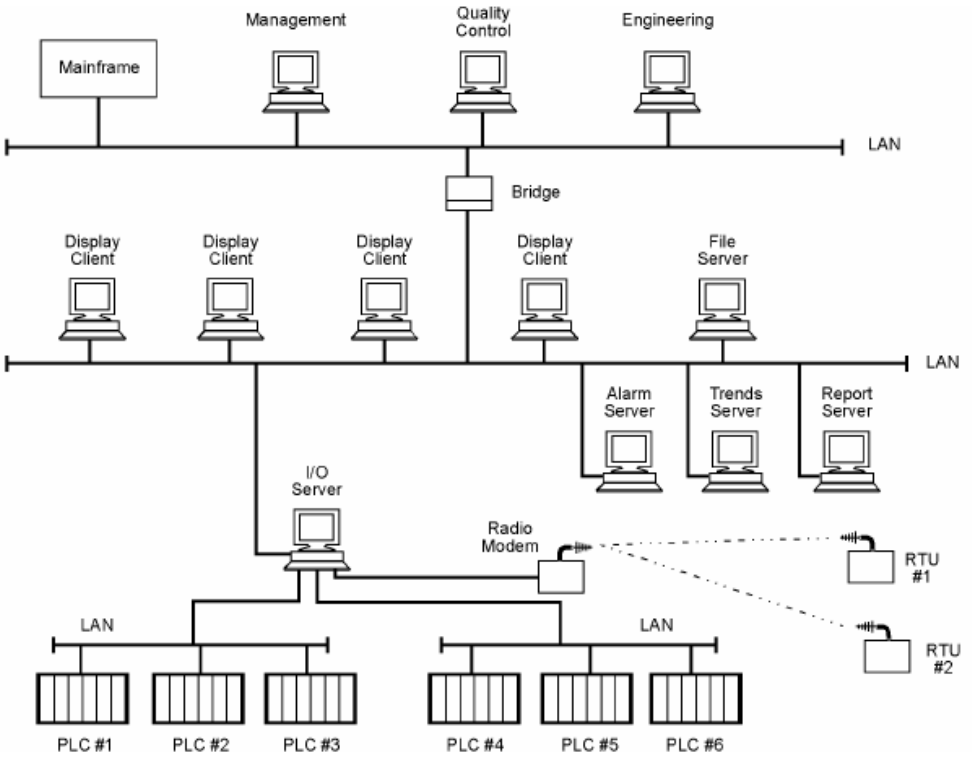

Figure 3.5
A large SCADA application

3.3 Human–machine interface

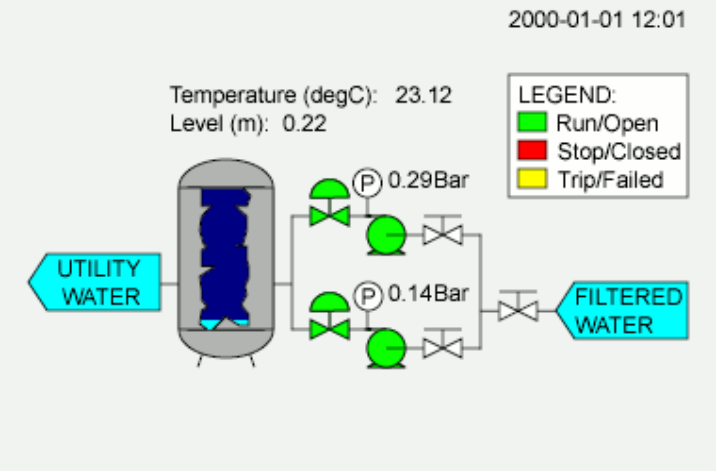

Figure 3.6
Typical HMI application

A human–machine interface (HMI) is an equipment, which presents process data to a human operator, and through which the human operator controls the process.

An HMI is usually linked to the SCADA system's databases and software programs, to provide trending, diagnostic data, and management information such as scheduled maintenance procedures, logistic information, detailed schematics for a particular sensor or machine, and expert-system troubleshooting guides.

The HMI system usually presents the information to the operating personnel graphically, in the form of a mimic diagram. This means that the operator can see a schematic representation of the plant being controlled. For example, the picture of a pump connected to a pipe can show the

operator that the pump is running and how much fluid it is pumping through the pipe at the moment. The operator can then switch the pump off. The HMI software will show the flow rate of the fluid in the pipe decreasing in real time. Mimic diagrams may consist of line graphics and schematic symbols to represent process elements, or may consist of digital photographs of the process equipment overlaid with animated symbols.

The HMI package for the SCADA system typically includes a drawing program that the operators or the system maintenance personnel use to change the way these points are represented in the interface. These representations can be as simple as an on-screen traffic light, which represents the state of an actual traffic light in the field, or as complex as a multi-projector display representing the position of all of the elevators in a skyscraper or all of the trains on a railway.

An important part of most SCADA implementations is alarm handling. The system monitors whether certain alarm conditions are satisfied, to determine when an alarm event has occurred. Once an alarm event has been detected, one or more actions are taken (such as the activation of one or more alarm indicators, and perhaps the generation of email or text messages so that the management or the remote SCADA operators are informed). In many cases, a SCADA operator may have to acknowledge the alarm event; this may deactivate some alarm indicators, whereas other indicators remain active until the alarm conditions are cleared. Alarm conditions can be explicit – for example, an alarm point is a digital status point that has either the value NORMAL or ALARM that is calculated by a formula based on the values in other analog and digital points – or implicit: the SCADA system might automatically monitor whether the value in an analog point lies outside high- and low-limit values associated with that point. Examples of alarm indicators include a siren, a pop-up box on a screen, or a colored or flashing area on a screen (that might act in a similar way to the "fuel tank empty" light in a car); in each case, the role of the alarm indicator is to draw the operator's attention to the part of the system "in alarm" so that appropriate action can be taken. In designing SCADA systems, care is needed in coping with a cascade of alarm events occurring in a short time, otherwise the underlying cause (which might not be the earliest event detected) may get lost in the noise. Unfortunately, when used as a noun, the word "alarm" is used rather loosely in the industry; thus, depending on the context, it might mean an alarm point, an alarm indicator, or an alarm event.

3.4 SCADA alarm management

Alarm systems are an integral part of the HMI. An alarm system consists of both hardware and software, including field signal sensors, transmitters, alarm generators and handlers, alarm processors, alarm displays, annunciator window panels, alarm recorders, and printers.

Alarm systems, whether hardwired or software configured, play an important role in monitoring and controlling industrial plants and equipment and are an integral part of the control system. Alarm systems indicate the abnormal conditions and problems of the plant and equipment to the operators, enabling them to take corrective action and bring the plant/equipment back to normal conditions. Alarm systems give signals to the operators in the form of audible sound, visual indications in different colors and/or continuous or blinking, text messages, and so on.

An alarm management system (AMS) is a new software application, which displays an overview of alarms from numerous plants with different SCADA systems. An AMS:

- gives an overview of all alarms for all your plants
- allows to choose specific alarms for forwarding to specific on-duty personnel
- collects and displays the alarms in one complete list
- any alarm can be forwarded by an SMS or an e-mail
- reduces the need for on-duty personnel outside normal working hours
- duty calendar makes sure that the right alarm is sent out to the right duty operator
- prints out duty calendars for validating duty schedules
- makes quick response possible to adverse plant conditions
- easy setup – online in minutes.

An organization running plants at various locations, each equipped with a different SCADA system, can now create a secure solution for alarm handling with AMS. During normal working hours, alarms can be acknowledged directly at the AMS console. Outside normal working hours, the AMS forwards alarms to operators with mobile phones, who then take over monitoring and

alarm acknowledgment. The AMS can also forward incoming alarms as e-mails. Organizing and setting up alarm-monitoring routines are easily done by creating duty calendars and operator profiles.

An example of a typical AMS solution is seen in Figure 3.7 where diverse SCADA systems "feed" their SMS alarm messages into the AMS server via a GSM modem or via a TCP/IP connection.

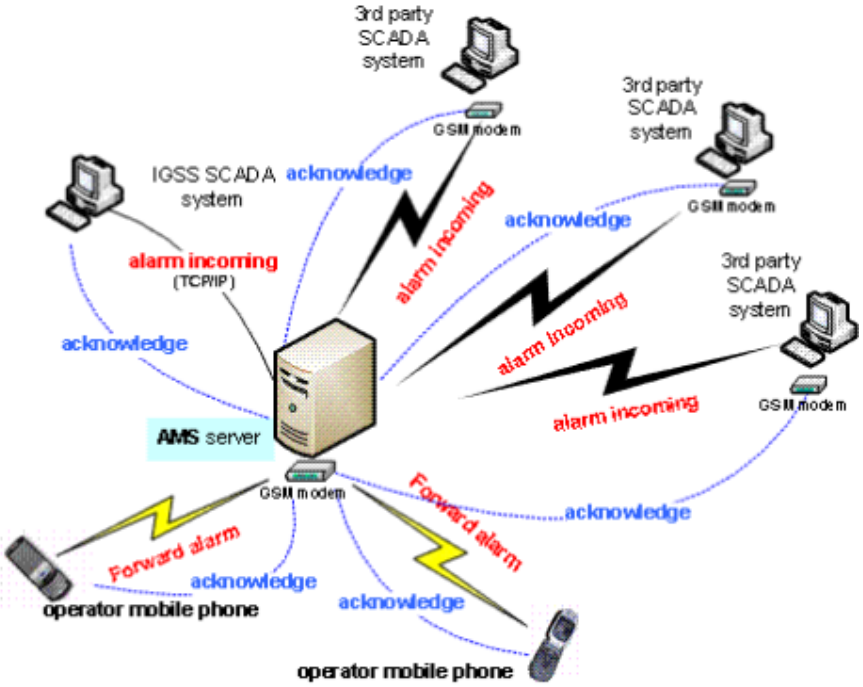

Figure 3.7
Typical AMS solution in SCADA

An alarm system needs to bring the following to the notice of the operator:

- problems that need operator attention
- process changes that require corrective action
- unsafe operating conditions before emergency shutdown of the plant
- hazardous conditions
- deviations from desired/normal conditions

An alarm system also assists an operator in:

- maintaining the plant, equipment, and processes within normal and safe operating conditions
- correcting dangerous and hazardous conditions that arise in the plant before emergency shutdown is initiated by the emergency shutdown system. This results in increased plant/equipment safety and also improves plant/equipment availability.
- recognizing a hazard well in advance and to take corrective action to avoid hazards
- better understanding complex processes and plant conditions and in identifying deviations from desired normal operating conditions.

While designing an effective alarm system, it is important that every alarm that is configured and presented to an operator should be useful and relevant to the operator. For each alarm, it is important to consider how important the alarm is and what should be its reliability. To determine the importance and reliability of an alarm, it is necessary to carry out qualitative and quantitative risk assessment to consider whether the alarm is safety related and whether it is to be implemented on an independent stand-alone system as opposed to the process control system.

Although it is a time-consuming process to generate a database for large number of alarms in a plant, it is important that design issues for each and every alarm are resolved and documented

before configuring the alarms. Following are some questions that need to be answered for each and every alarm and can be considered as minimum design documentation:

3.4.1 What is the purpose of the alarm?

- Whether the response is required for this alarm from the operator?
- In case the operator does not respond to the alarm, what can be the likely consequences?
- How much time is available for the operator to respond to the alarm?
- What will the effectiveness be of the operator response?

The above questions can be used as the basis for an alarm review at the later stage when the plant is operational and an improvement in alarm system is needed to improve the overall performance.

3.4.2 Management of alarm settings

- Procedures for making changes in the alarm settings. Who should be authorized to do the changes in the alarm settings?
- Generally, an alarm-shelving facility should be given to the operator. However, in some plants, the operators may not be authorized to shelve alarms due to various safety or other reasons. Special procedures may be made for shelving critical and important alarms.
- Does the alarm need testing? If yes, then how should the alarm be tested and how frequently? How should the alarm be maintained?

3.4.3 Alarm management: best practices

The best alarm management practices involve the following steps:

- Create and document an alarm philosophy: this process can be as valuable as the document itself. It helps standardize the configuration across multiple lines, especially where pipelines have been acquired rather than built by the operating company.
- Benchmark and performance audit: each of the KPIs defined in the alarm philosophy, which may match those of EEMUA 191 where they are applicable, are calculated and interpreted. To complete the audit, the alarms and events must be historized in a fashion that allows for alarm and event analysis.
- Rationalize alarms: clean up bad-acting tags, which can contribute up to 50% of the daily alarm load. Alarm rationalization involves a team of people from operations and engineering, together with an impartial facilitator to methodically review alarm settings on each alarmable SCADA tag.
- Investigate dynamic and state-based alarming: this is not a simple activity and takes a solid understanding of the operational and control philosophies of the facilities during all relevant states.
- Implement changes: based on the results of rationalization and investigation.
- Continuous improvement: alarm management has a life cycle and is not a one-time project. Continuous performance monitoring helps to identify new opportunities for improvement.
- Manage change and corporate culture: organizations successful in alarm management integrate the practices into the workflow to optimize performance over the long term.