REQUIREMENTS FOR OPEN WATER AND IRRIGATION SYSTEM SCADA SOFTWARE

Peter Clout[1]
Cathy Laughlin[1]
David Steinman[1]

ABSTRACT

A successful modern SCADA system will generally require some form of software to drive communications, control processing, and store data. This must not only operate as required on the first day of operation, but it has to be easily modified by the users to reflect changes that occur every day. These changes include expansion of the system as well as calibration changes and other SCADA changes to reflect changes in the network and variable field conditions. There are a multitude of SCADA software packages that exist today, generally developed for manufacturing, utility, or scientific applications. However, irrigation canals have some unusual and unique requirements which must be met. Any successful implementation of a SCADA system must address those specific considerations. We list and discuss these requirements and detail the adaptation of a standard SCADA system, Vsystem.

INTRODUCTION

Unique Needs

SCADA software for irrigation and river systems must perform a task fundamentally similar to process control software used in science, industry, and manufacturing. Many of the requirements are shared with other interests. However, there are some particular requirements of irrigation and river systems which must also be met. These requirements may be broadly classified as technical, regulatory, and financial.

Technical requirements are commonly given the most consideration. These requirements are the simple "nuts and bolts" of the software, including such things as scan rates, hardware scanners, data storage, graphical interfaces, development flexibility, and the all important consideration of reliability.

Regulatory considerations may be less obvious. Irrigation and river systems often cross political boundaries and may have extensive involvement with

---

[1] Vista Control Systems, Inc., 176 Central Park Square, Los Alamos, NM 87544-4031: clout@vista-control.com, laughlin@vista-control.com, steinman@vista-control.com

government(s). An irrigation project could violate local, state, federal, or even international laws and agreements as a result of operational problems. Also, some data must be shared with other agencies and jurisdictions, and thus flexibility to communicate data in differing formats is a necessity. Finally, reliability looms large as many people are dependant on the proper monitoring and control of irrigation and river systems. Failure could potentially pose public health and safety issues, up to and including loss of life.

Financial requirements generally receive a lot of consideration. Unfortunately, it is common to consider only the immediate bottom line, overlooking long term costs. The operation of many irrigation and river systems operations are completely or in part publicly funded. Long-term commitment of public funding may be uncertain. An up-front expenditure for software may be clearly defined, but long-term funding for future maintenance and development may not be possible, so it becomes a requirement that, by and large, they are capable of being maintained and operated by project staff: adding simplicity and ease-of maintenance to the requirements.

Vsystem

Vsystem was originally developed internally at Los Alamos National Laboratory and then commercialized by Vista Control Systems some 17 years ago. It has found applications in many industries from water and waste water to steel and aluminum production to petrochemical plant monitoring to test cells for truck transmission and aircraft engine development and testing. In addition Vsystem is used in rocket and satellite control rooms as well as for research machines very like the machine that was the origin of the software.

MRGCD

The Middle Rio Grande Conservancy District (MRGCD) delivers irrigation water to approximately 73,000 acres of farmland in central New Mexico. Water is diverted from the Rio Grande at 3 low-head dams and the outlet works of a USACE flood control reservoir. The MRGCD encompasses a 150 mile reach of the Rio Grande, 800 miles of canals, and 400 miles of drains. The system is entirely gravity-fed and was constructed locally (non-federal project) between 1928 and 1935. It was not built from a clean sheet of paper, but combined the works of 79 independent Acequia associations, most of which date to the 17th and 18th century.

During construction in the 1930's a large number of hydraulic and mechanical automatic control structures were incorporated. Unfortunately these fell into disuse and disrepair over the years. In 1996, under the all-too-common demands of endangered species, encroaching urbanization, and competing water users, MRGCD water efficiency received close examination. In response, MRGCD embarked upon a course of modernization involving measurement and automation. Eventually, the need became clear for a comprehensive software (SCADA) system capable of bringing data, hardware, and operators together.

Collaboration

MRGCD had experimented with, and investigated a number of commonly available software packages. Some of these were generic process automation and control software packages, and some were specifically tailored for surface water applications. However, none met all their requirements. At the same time, we (Vista Control Systems) learnt of some of the SCADA requirements for open water SCADA systems that were not being met with the other popular commercial packages. In response to an RFP from MRGCD we decided to make additions to Vsystem that would meet these requirements. With the help of David Gensler of the MRGCD in Albuquerque, we refined these and installed the first system there.

REQUIREMENTS AND SOLUTIONS

As previously suggested, the unique requirements of SCADA for irrigation and river systems can be broken down into the broad categories of flexibility, simplicity, and reliability.

Flexibility

The term SCADA means different things to different users, so when evaluating software, careful thought should be given to exactly what the software must accomplish.

Software for a SCADA system should be flexible. In fact, most software for these types of applications is really along the lines of a do-it-yourself kit. The differences are how much of the kit has been pre-assembled. Some packages are blank sheets of paper, just collections of tools that the user may use to construct a system. Others have large parts, or even the entire system pre-designed. In general, the more that is pre-built, the easier the system will be to install and configure, but the less flexible it will be through its life. On the other hand, a blank sheet of paper, with the right collection of tools, can be used to build a very comprehensive and effective system. There is a balance that must be struck for each user, and defining this balance is the key to success.

Communication protocol: What sort of hardware can be supported? All SCADA systems must somehow acquire information. In general, this implies that SCADA software must be capable of communicating with gadgets in the field. This may be one specific type of device, or it could be multiple types, with differing communication protocols. Information could also be sourced from digital sources, either local or remote databases, which also may require specific communication protocol. One of the first steps in planning the implementation of a SCADA system is to list the various types of devices or data which must be supported.

In the case of MRGCD, all hardware uses the Modbus protocol. MRGCD uses a proprietary feature of its RTU's to retrieve weather station data from Campbell Scientific data loggers, but, since this feature is still accessed through standard MODBUS commands, it is accommodated by Vsystem.

Output: Many SCADA systems will, at some point, receive data from, or provide data to, others. The "others" may be separate organizations, or perhaps different software clients within the organization. All of these possible exchanges should be listed, and the SCADA software should be evaluated to determine what format(s) are appropriate.

In the case of MRGCD, data is received from U.S. Geological Survey (USGS) and United States Army Corps of Engineers (USACE), and exported to the United States Bureau Of Reclamation (USBOR). This data exchange is accomplished via the SHEF text format, an old but familiar format used by the National Weather Service. Also, data is exported from Vsystem to a MS SQL database, both for permanent archiving, and to allow future data review and reporting flexibility.

Data Rate: A primary consideration of most systems is the rate at which data is collected. The Vsystem Modbus scanner as used in other applications typically functions at frequencies of 1-100 Hz. Water systems are unique in that inputs generally change much more slowly than in other industries, thus scan rates in the range of 0.001-0.0005Hz (every 15-30 minutes) are more typical. Although on the face of it, a relatively slow data rate should simplify the application, new challenges may be introduced. As an example, in other applications, much faster scan rates are required both to capture data to reflect the speed of the process under control and to give immediate feedback to the operator when changes are made. For this application, the speed of response to operator commands is very much faster than changes in flow and level when no changes are being commanded. The choice is either to have as fast a scan rate as the communication channel will allow to give occasionally required fast feedback (in larger systems, the time of a complete scan will not allow very fast feedback) or implement an entirely new mechanism solely for the immediate feedback.

In the case of MRGCD, data was to be collected from sites at regular, user defined, time intervals. Due to varying types of data, multiple intervals or polling groups, are specifiable. Also incorporated is the option to control the timestamp for a piece of data either as the actual instant it is recorded or the time at which the polling group is initiated. The Modbus scanner was modified to allow a set delay after start-up and initialization before the first scan. This added set delay to the first scan allows the scanner to be started at any time and the scans to be, for example, on the hour and half-hour. The user only has to define the time of the first scan and the appropriate scan interval to use this feature.

Data Type: Another consideration regarding data rate is whether data collection is time driven or event driven. Irrigation and surface water systems are typically time driven, so that the condition of the system is observed and recorded at regular intervals. However, event-driven data collection may be an important part of such a system too, particularly when automation is employed, so that alarm or other pre-identified extreme conditions are recorded accurately.

Operator Outputs: Some SCADA systems only collect and display or store data. But increasingly SCADA systems are living up to the “Control" in the acronym, and outputs to physical devices are part of the process. The type and timing of these outputs must be considered. Each case is likely to be unique.

In the case of MRGCD, extensive local automatic control of structures is performed. Setpoints can be supplied either remotely or from an operator on site. When done remotely, Vsystem is used to write setpoints and operating parameters via radio link to control gates. A series of control screens was created, using Vdraw, to display sites and parameters. When an Operator makes a change in any displayed parameter, the new value is written out to the RTU register over the radio link as soon as possible, only waiting for any other transaction to complete. So that the operator can see the response to the change, a button was added to the control screen to allow the operator to trigger a manual scan, putting a command into the Modbus scanner command channel. This scan command includes the unit number of the RTU so that all the registers of the RTU are scanned out of sequence. This was the most efficient way to implement this feature.

Data Conversion: A SCADA system requires some form of data conversion or processing. The simplest form is linear, such as scale and offset to convert raw electronic values to engineering units. This is an area where many SCADA packages provide pre-packaged routines. When considering SCADA software, the types of data conversion should be considered just as carefully as hardware interfaces. Just like proprietary communication protocol, inflexible conversion routines can severely limit success of the project.

A simple example of such a conversion is the calculation of stage and discharge from the measured data. An example equation is:

Stage = ((raw measurement*0.001408)+0.77)

While linear data conversion of the reading from the RTU to engineering units is included in Vsystem and the coefficients are easily modified, other conversion algorithms in the past have required a subroutine to be written to implement the algorithm. This subroutine would take its conversion coefficients from the channel so that while the coefficients are easy to change, an algorithm change requires code to be modified or new code written. Clearly, this is neither flexible nor desirable for open-water SCADA applications. We have added a third option for conversion definition, the capability to define the conversion equation as a typed equation. The above conversion would be typed in as:

invalue*0.001408+0.77

The equation can include values from any other channel and values of any field of any channel. More complicated statements can include conditional statements and logical tests. This could occur with an overshot gate, where discharge is calculated with one equation when the gate is raised, and acts as a sharp crested weir, and with a different equation when the gate is lowered and starts to behave as a broad-crested weir. An example is:

```
vsq {if (cenww.gatep > 1.5)  outvalue = (((abs(cenww.stage + 0.001)) ^ 1.5) *
   26.13) else  outvalue = (((abs(cenww.stage + 0.001)) ^ 1.75) * 15.5) end}
```

<u>Higher Level Conversions:</u> Meaningful engineering data is often derived from more than one input, although from a water engineer's standpoint, these conversions would be just another equation. From the software developer's point of view, these are a completely different type of animal. It is one thing to perform a calculation on a piece of information as it arrives, it is an entirely separate process to gather various pieces of information, process them, and generate a new data value. Due to timing and scan rates, many packages do not have this ability. If this feature will be needed in a SCADA application, the flexibility of the package should be carefully evaluated. As an example of a higher level conversion, discharge from a processed stage value, is:

cenww.disch = (((cenww.poolh - cenww.gatep)^1.5)* 33.76)

<u>"Pseudo RTU's":</u> Another feature, commonly desirable in the irrigation and water business is what we have termed a "pseudo-RTU". This is data describing the system that is not immediately derived from hardware registers but is either from another data source or is higher-level data that is the result of calculations using data from one or more RTUs. It is convenient if this data could appear to come from a non-existent RTU. We have added the capability to define this RTU and its data exactly as a real RTU is defined except only for a flag to indicate to the Modbus scanner that this RTU does not actually exist. Thus the registers of this pseudo RTU provide higher-level data that is actually data calculated from other, real, RTU measurements and other sources. This higher-level data can also be used in other calculations. An example, from MRGCD's application is the calculation of a Diversion rate from 3 separate input values:

ANGDV.Q = ((ALBCN.Q + ATFCN.Q) - ALGDR.Q)

The generation of the Angostura dam diversion rate originates as the raw A-D output from RTU's on 2 canals and a drain. These outputs are first resolved individually to stage, then to discharge, and finally combined to arrive at the diversion value. ANGDV.Q then becomes a variable in other equations and could be implemented as a control output. The use of a pseudo-RTU is not a requirement for an implementation, rather a convenience to users. The same result could be obtained with channels unassociated with any RTU.

User Interface: Standard Vdraw, the Vsystem GUI development tool, was used by MRGCD to develop the screens required for the operation of the system. Two examples from the application are shown in Figure 1 below.

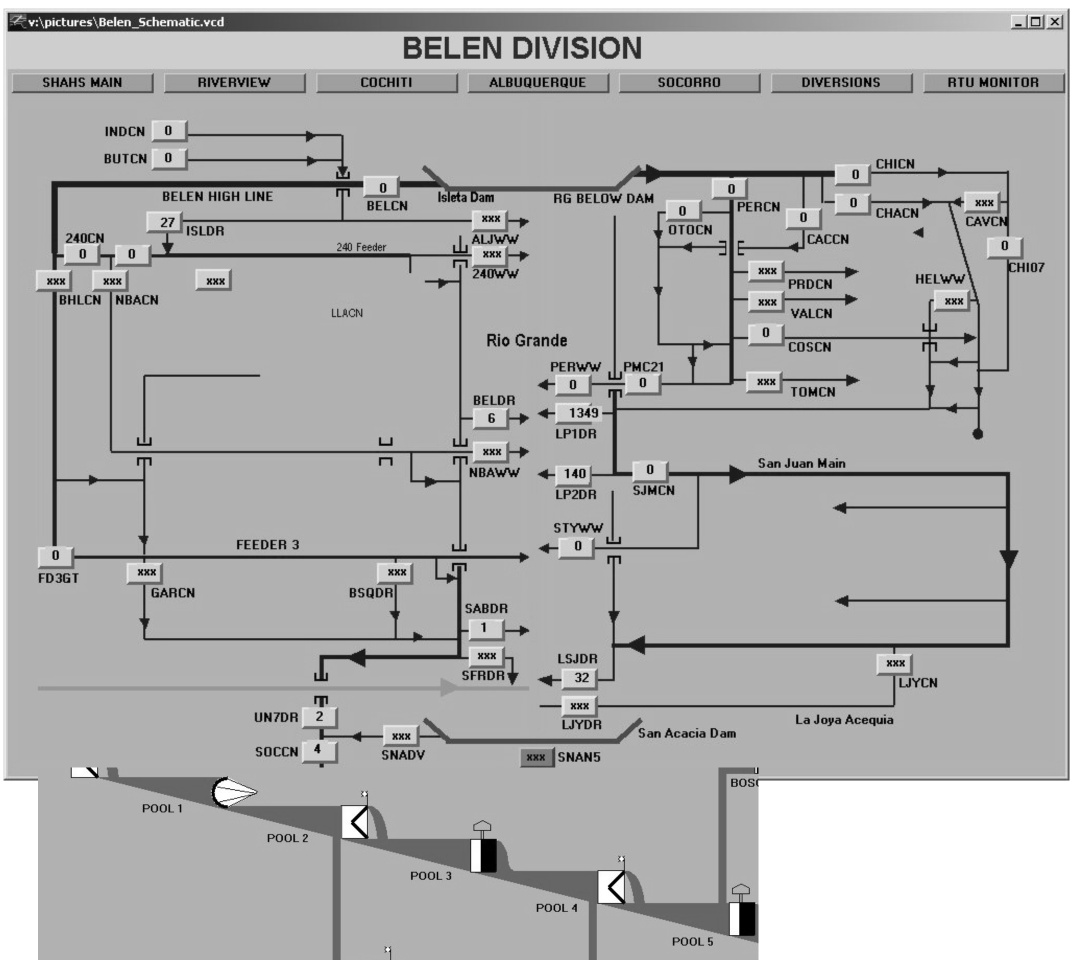

Figure 1. Two example operator screens form the MRGCD SCADA system.

Data Storage: For long-term analysis and reporting requirements (the regulatory environment), operational data should be stored to a permanent database. This may be a proprietary part of a SCADA package, or external to the software. Maximum flexibility, particularly with regards to data exchange, is more likely to be achieved with a common external database. In the case of MRGCD, there are 2 essentially identical but separate databases. Vsystem contains its own proprietary database (Vlogger). Within Vlogger is a permanent record of every piece of information generated by the SCADA system. External to Vsystem is a database built around Microsoft SQL. The operational data is first recorded to the Vsystem historian, Vlogger, and then, periodically, the data is written to records in Microsoft SQL-Server using the SQL interfaces, one SQL interface to Vlogger data and one to Microsoft SQL-Server. This Microsoft SQL-Server update utility operates on a user-defined schedule and automatically updates the Microsoft SQL-Server database with all the new data in the Vlogger archive.

The Microsoft SQL database tables are created automatically from the information in the Vsystem database. Data from each RTU is stored in a separate table. The columns are timestamp and each channel that was logged. Each time entry read from the Vlog archive file becomes a new row. Multiple Vlog archive files can be combined into a single Microsoft SQL database. If any RTU's or channels have been added to the Vsystem database, then new tables and columns are automatically added to the Microsoft SQL database.

Other Applications: It may be the case that a SCADA system will operate and interact with other software (ie: GIS) applications. As with hardware, all possible interactions should be considered, and compatibility evaluated. In the case of MRGCD, Vsystem will interact with a DSS model developed by Colorado State University to predict irrigation demand. Vsystem will provide inputs, and accept outputs from the DSS model so that it may be run from the Vsystem control screen, and projected canal flows from the DSS displayed alongside actual flows from Vsystem. Vsystem was designed from the beginning to easily incorporate computer models of the system being monitored and controlled.

Simplicity

Setup: When considering SCADA software, the user should consider the requirements for configuring the package. In large measure, these requirements will be a reflection of the background of the personnel using the product. Once again, irrigation and river systems seem unique. A large industrial setting, an oil refinery for example, may have dozens of trained computer personnel on staff, and thus building the SCADA package need not be particularly user friendly. An irrigation project may have only a few engineers (not computer specialists), and putting together the SCADA system will likely be only one of many unrelated tasks they must perform. So, the SCADA system must have a relatively straightforward and logical approach to configuration, which does not require intensive training. When planning SCADA software, one should consider carefully setup interface that is required without sacrificing necessary flexibility.

In the case of MRGCD a configuration interface, Vriver, was developed for Vsystem. Vriver is a wizard that interacts with the user in a way that is focused on the setting up of the software for the hardware configuration, polling groups/times, data processing, and data storage. Vriver was designed to be similar to the RTU configuration utility supplied by the RTU manufacturer, with which MRGCD was already familiar. The output of a set-up session is a set of files that then starts up and configures Vsystem to scan and record the data from the RTUs and when necessary write out new settings to RTUs. The capability still exists to write or edit setup files directly, without using the Vriver interface, so that a full range of flexibility is preserved.

The following screen shots in Figure 2 show the stages of setting up and editing the RTU configuration for Vsystem.

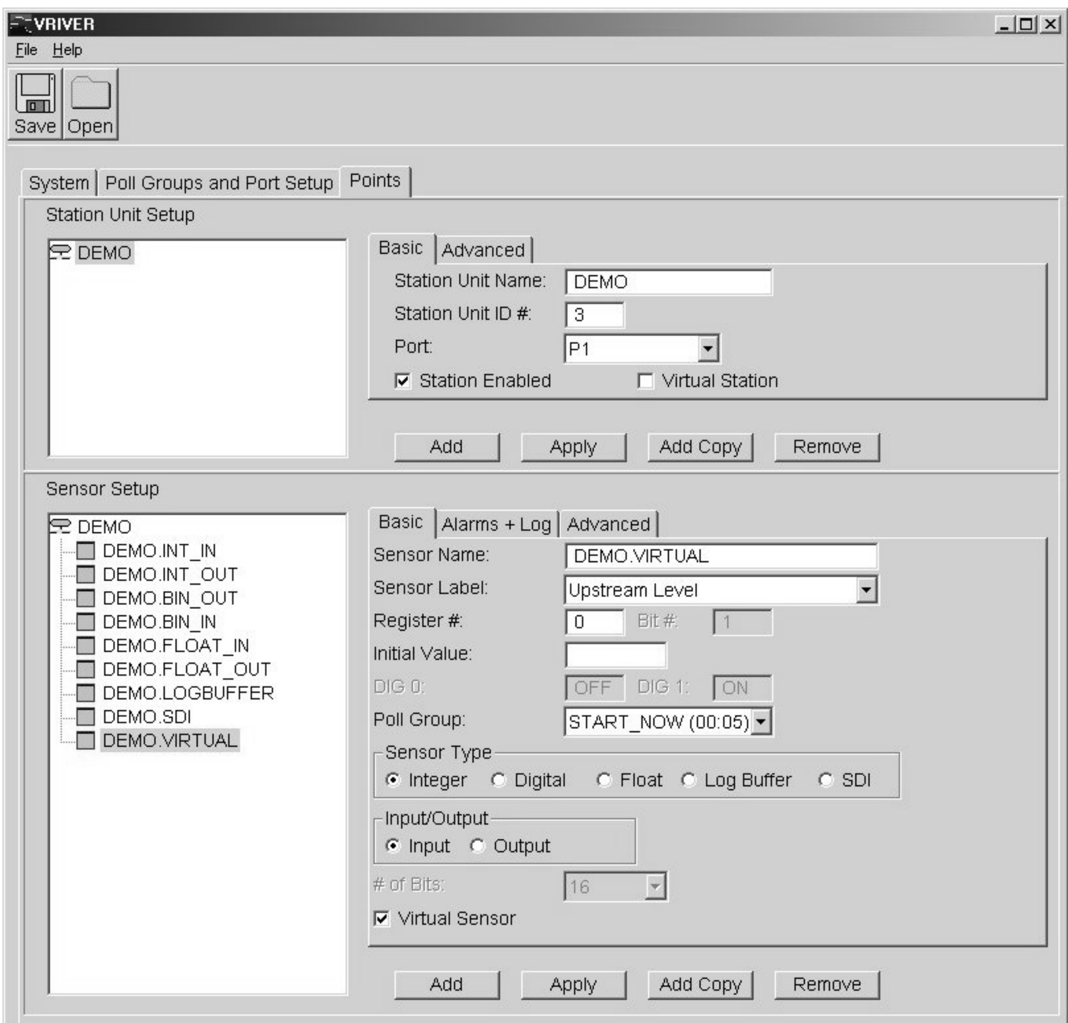

Figure 2. Screen showing some of the Vriver set-up functions.

Reliability

Vsystem in developed using traditional, proven, programming techniques and, being developed for several operating systems, it uses system services and graphics services at the lower and more dependable levels. Thus, Vsystem applications have been proven robust over years of operation. Data integrity is ensured by having two separate historian files maintained, Vlogger and MS SQL Server.

Radio communications are less reliable than solid, copper connections if only because of interference from other radio systems and changing atmospheric conditions. To monitor and improve radio reliability, it is important to keep track of the successful scans for each RTU and so the facility has been added to the Modbus scanner to count the number of successful scans as well as the total number of scans and maintain a channel for each RTU with the percentage of successful scans for that RTU.

CONCLUSION

A modern SCADA system requires software. A variety of products exist, though many are of a generic form and are designed for industrial and manufacturing environments. The needs of irrigation and rivers systems are unique, and these products may be adapted, with varying degrees of success.

When planning the implementation of SCADA software from any source, the specific needs of irrigation and river systems should be carefully considered. Flexibility is a highly valued feature, though it may come at the cost of some simplicity and ease of implementation. Conversely, effort expended upfront to build a good system will likely result in a much smaller burden for long term operation and maintenance costs. A system that is not flexible, regardless of how simple it is to implement, will probably turn out to be of dubious value. A high degree of reliability is also required.

Vista Controls has modified an existing SCADA product to meet the specific management needs of the Middle Rio Grande Conservancy District. The result has been a system which the MRGCD believes is a contributor to its water saving program. We have described the specifics of that application as an example of the unique needs of SCADA software for irrigation monitoring and control.

ACKNOWLEDGEMENTS

We very gratefully acknowledge the guidance and advice of David Gensler of the MRGCD for his technical help right through this project.