INDUSTRIAL AND COLLABORATIVE CONTROL SYSTEMS - A COMPLEMENTARY SYMBIOSIS -

M. Clausen, DESY, Hamburg, Germany

Abstract

Looking at today's control system one can find a wide variety of implementations. From pure industrial to collaborative control system (CCS) tool kits to home grown systems and any variation in-between. Decisions on the type of implementation should be driven by technical arguments Reality shows that financial and sociological reasons form the complete picture. Any decision has it's advantages and it's drawbacks. Reliability, good documentation and support are arguments for industrial controls. Financial arguments drive decisions towards collaborative tools. Keeping the hands on the source code and being able to solve problems on your own and faster than industry are the argument for home grown solutions or open source solutions. The experience of many years of operations shows that which solution is the primary one does not matter, there are always areas where at least part of the other implementations exist. As a result heterogeneous systems have to be maintained. The support for different protocols is essential. This paper describes our experience with industrial control systems, PLC controlled turn key systems, the CCS tool kit EPICS and the operability between all of them.

INTRODUCTION

Process controls in general started at DESY in the early 80^{th} with the installation of the cryogenic control system for the accelerator HERA (Hadron-Elektron-Ring-Anlage). A new technology was necessary because the existing hardware was not capable to handle standard process controls signals like 4 to 20mA input and output signals and the software was not designed to run PID control loops at a stable repetition rate of 0.1 seconds. In addition sequence programs were necessary to implement startup and shutdown procedures for the complex cryogenic processes like cold boxes and compete compressor streets.

Soon it was necessary to add interfaces to field buses and to add computing power to cryogenic controls. Since the installed D/3 system[1] only provided an documented serial connection on a multibus board, the decision was made to implement a DMA connection to VME and to emulate the multibus board's functionality. The necessary computing power for temperature conversions came from a Motorola MVME 167 CPU and the field bus adapter to the in house SEDAC field bus was running on an additional MVME 162. The operating system was VxWorks and the application was the EPICS toolkit.

Since this implementation was successful it was also implemented for the utility controls which were looking for a generic solution to supervise their distributed PLC's.

A SELECTION OF PROCESS CONTROL SYSTEMS AT DESY

DCS (D/3)

As a result of a market survey the D/3 system from GSE was selected for the HERA cryogenic plant. The decision was fortunate because of the DCS character of the D/3. The possibility to expand the system on the display- and on the I/O side helped to solve the increasing control demands for HERA. The limiting factor for the size of the system is not the total number of I/O but the traffic on the communication network. This traffic is determined by the total amount of archived data not by the data configured in the alarm system. The technical background of this limitation is the fact that archived data are polled from the display servers whereas the alarms are pushed to configured destinations like alarm-files, (printer) queues or displays.

SCADA Systems with DCS Features (Cube)

The fact that the D/3 system mentioned above had some hard coded limitations with respect to the Y2K problem was forcing us to look for an upgrade or a replacement of the existing system. As a result of a call for tender the company Orsi with their product Cube came into play [2]. The project included a complete replacement of the installed functionality. This included the D/3 as well as the integration of the DESY field bus SEDAC and the temperature conversion in VME. The project started promising. But soon technical and organizational problems were pushing the schedule to it's limits which were determined by the HERA shutdown scheduled at that time. The final acceptance test at the vendors site showed dramatic performance problems. Two factors could be identified as the cause of these problems. The first one was related to the under estimated CPU load of the 6^{th} grade polynomial temperature conversion running at 1 Hz. The second one was the additional CPU load caused by the complex functionality of the existing D/3 system. Here it was underestimated that each digital and analog input and output channel had it's own alarm limits in the D/3 system. In a SCADA like system as Cube the base functionality of a channel is to read the value and make it available to the system. Any additional functionality must be added. Last not least the load on the network for polling all the alarm limits – typically for a SCADA system – was also driving the network to it's limits.

Finally the contract with Orsi was cancelled and an upgrade of the D/3 system was the only possible solution. It was finally carried out in march 2003.

In any case it should be mentioned that the Cube approach had the advantage of a homogeneous configuration environment (for the Cube front end controllers) – compared with heterogeneous environments for 'pure' SCADA systems.

SCADA (PVSS-II)

The H1 experiment at the HERA accelerator decided to use PVSS-II for an upgrade of their slow control systems[3]. The existing systems were developed by several members of the H1 collaboration and were difficult to maintain. The decision to use PVSS as a replacement was driven by the results of an extensive survey carried out at CERN by the Joint Controls Project [4]. PVSS is a 'pure' Supervisory And Data Acquisition System (SCADA). It provides a set of drivers for several field buses and generic socket libraries to implement communication over TCP/IP. The core element is the so called event manager. It collects the data (mostly by polling) from the I/O devices and provides an event service to the attached management services like: control manager, database manager, user interface, API manager and the built in HTTP server. The PVSS scripting library allows to implement complex sequences as well as complex graphics. Compared with other SCADA systems PVSS comes with one basic feature: it provides a true object oriented API to the device's data.

One major disadvantage of SCADA systems is the fact that two databases, the one for the PLC and the one for the SCADA system must be maintained. Integrated environments try to overcome this restriction.

EPICS

EPICS has emerged at DESY from a problem solver to a fully integrated control system. Starting from the data collector and number cruncher for the cryogenic control system, EPICS made it's way to become the core application for the DESY utility group. In addition it is used wherever data is available through VME boards or by means of Industry Pack (IP) modules. For those cryogenic systems which are not controlled by the D/3 system EPICS is used with it's complete functionality. In total about 50 Input Output Controller (IOC) are operational processing about 25 thousand records.

1 EPICS as a SCADA System

The utility group (water, electrical power, compressed air, heating and air conditioning) is using a variety of PLC's spread out over the whole DESY site. EPICS is used to collect the data from these PLC's over Profibus (FMS and DP) and over Ethernet (Siemens H1 and TCP). The IOC's provide the interfaces to the buses and collect the data. The built in alarm checking of the EPICS records is used to store and forward alarm states to the alarm handler (alh) of the EPICS toolkit. In addition tools like the channel archiver and the graphic display (dm2k) are used. The default name resolution (by UDP broadcast) and the directory server (name server) are used to connect client and server applications over TCP. All of these are basically SCADA functions.

The textual representation of all configuration files (for the IOC, the graphic tool, the alarm handler and the archiver) provides a flexible configuration scheme. At DESY the utility group has developed a set of tools to create IOC databases and alarm configuration files from Oracle. This way the controls group provides the service to maintain the EPICS tools and the IOC's while the users can concentrate on the equipment being controlled.

2 EPICS as a DCS System

Besides the basic components of a SCADA system EPICS also provides a full flavoured Input Output Controller (IOC). The IOC provides all of the function a DCS system requires, such as: a standard set of properties implemented in each record, built in alarm checking processed during the execution of each record; control records like PID etc.; configuration tools for the processing engine. The flexible naming scheme and the default display and alarm properties for each record ease the connection between the operator tools and the IOC's. The flexible data acquisition supports the poll mode as well as the publish subscribe mode. The latter reduces the traffic drastically.

PLC's

PLC's provide nowadays the same rich functionality as it was known from stand alone control systems in the past. Besides the basic features like the periodic execution of a defined set of functions they also allow extensive communication over Ethernet including embedded http servers and different sets of communication programs. Besides the communication processors, display processors can be linked to PLC's to provide local displays which can be comprised as touch panels for operator intervention and value settings.

These kind of PLC's are attractive for turn key systems which are commissioned at the vendors site and later integrated into the customers control system.

Intelligent I/O

New developments in I/O devices allow to 'cluster' I/O in even smaller groups and connect theses clustered I/O channels directly to the control system. PLC's are not any more necessary for distributed I/O. Simple communication processors for any kind of field buses or for Ethernet allow an easy integration into the existing controls infrastructure. Little local engines can run IEC 61131 programs. The differences between PLC's and intelligent I/O subsystems fade away.

FUNCTIONALITY

The ever lasting question why control systems for accelerators and other highly specialized equipment are often home grown or at least developed in a collaboration but only in rare cases commercial shall not be answered here. We try to summarize here basic functionalities of different controls approaches.

Front-end Controller

One of the core elements of a control system is the front-end controller. PLC's can be used to implement most of the functions to control the equipment. The disadvantage is the complicated access to the controls properties. For instance all of the properties of a control loop like the P, I and D parameter, but also the alarm limits and other additional properties must be addressed individually in order to identify them in the communication protocol and last not least in the display-, alarm- and archive programs. In addition any kind of modifications of these embedded properties is difficult to track because two or more systems are involved. This might be one strong argument why control loops are mainly implemented on the IOC level rather than PLC's.

1 I/O and Control Loops

Complex control algorithms and control loops are the domain of DCS alike control systems. The support for sets of predefined display and controls properties is essential. If not already available (like in DCS systems) such sets of generic properties are typically specified throughout a complete control system (see namespaces).

2 Sequence/ State programs

Sequence programs can run on any processor in a control system. The runtime environment depends on the relevance of the code for the control system. Programs fulfilling watchdog functions have to run on the front-end processor directly. Sequence programs for complicated startup and shutdown procedures could be run on a workstation as well. The basic functionality of a state machine can be even implemented in IEC 61131. Code generators can produce 'C' code which can be compiled for the runtime environment.

3 Supported Hardware

The support for field buses and Ethernet based I/O is a basic functionality for SCADA type systems it is commercially available from any SCADA system on the market. The integration of specific hardware with specific drivers and data conversion is the hard part in a commercial environment. Open API's or scripting support sometimes help to integrate custom hardware. If these tools are not provided for the control system it is difficult – if not impossible - to integrate custom hardware.

New industrial standards like OPC allow the communication with OPC aware devices and the communication between control systems. One boundary condition for this kind of functionality is the underlying operating system. In the case of OPC it is bound to DCOM which is a Microsoft standard. UNIX based control systems have a hard time to get connected. Only control systems supporting multiple platforms can play a major role in a heterogeneous environments.

As a result the limited support for custom- or specialized hardware may give reason for the development of a new control system.

Display and Operation

Besides the front-end system the operator interfaces play a major role for the acceptance of a control system. SCADA tools come with a homogeneous look and feel throughout their set of tools. Toolkits implemented in a collaboration might vary because the individual tools were developed by different teams.

1 Graphic

Synoptic displays are the advertising sign for any control system. Commercial synoptic displays come with a rich functionality and lots of special features. Starting to make use of all these features one will find out that all individual properties of the graphic objects must be specified individually. Since SCADA systems must be generic they cannot foresee that an input channel does not only consist of a value but also consists of properties like display ranges and alarm values. Defining all of these properties again and again can be a pretty boring job. Some systems allow to generate prototypes of graphic objects. These prototype or template graphics are complex and need a specialist to generate them.

DCS or custom synoptic display programs can make use of the common set of properties each I/O point provides. This predefined naming scheme will fill in all standard property values and thus only require to enter the record – or device name into the configuration tool. A clear advantage for control systems with a notion of I/O objects rather than I/O points.

2 Alarming

Alarms are good candidates to distinguish between different control system architectures. Those systems which have I/O object implemented also provide alarm checking on the front-end computer. Those systems which only know about I/O points have to add alarm checking into the I/O processing. While the I/O object approach allows to implement alarm checking in the native programming language of the front-end system, I/O point oriented systems typically have to implement this functionality in their native scripting language. This is typically less efficient and error prone because all properties must be individually configured. This leads to a flood of properties. Not only the error states for each I/O point wind up to be individual I/O points but also the alarm limits and the alarm severity of each limit must be defined as I/O points if it is desired to be able to change their values during runtime.

Besides this impact on the configuration side the processing and forwarding of alarms makes the difference between SCADA and DCS systems. Since SCADA systems inherently do not 'know' about alarms, each alarm state must be polled either directly from the client application or in advanced cases from an event manager which will forward alarm states to the clients. In any case a lot of overhead for 'just' checking alarm limits. DCS system again have the advantage that clients can either register themselves for alarm states und thus get the information forwarded or are configured to send alarm changes to certain destinations spread around the control system. The latter case is only possible for systems which in total are configured with all the nodes taking part in the controls network.

3 Trending and Archiving

Trending has become an important business in control systems architectures. Trends are necessary to trace error conditions or for post mortem and performance analysis of the controlled plant. Besides some custom implementations which are capable to store the data of complete control objects, most of the trending tools archive scalar data. Additional features like conditional trending or correlation plots make up the difference between individual implementations.

4 Programming Interfaces

With respect to open programming interfaces PLC's and DCS systems have a common strategy. They are running reliably because there's no way to integrate custom code which could interfere with the internal processing. As a consequence the customer has to order 'specials' - which are extremely expensive – or forget about it and use the system as a black box.

Since SCADA systems by definition must be able to communicate with a variety of I/O subsystems they already have some built in API's which allow to integrate custom functionality.

Specially collaborative systems need a certain openness to fulfill all the requirements from various development groups. Programming interfaces on all levels like font-end I/O, front-end processing, networking etc. are mandatory. A clear advantage for this type of system.

5 Redundancy

If redundancy means the seamless switch which takes over all the states and all the values of the I/O and all states of all programs currently running, it is a domain of only a few DCS systems. Custom or CCS implementation do not provide this kind of functionality. Maybe because of the immense effort and the fact that it is only required in rare cases.

Besides processor redundancy, redundant networks or I/O subsystems are available for certain commercial DCS systems. Again – a domain which is not covered by SCADA or CCS implementations.

Advanced safety requirements may be covered by redundant PLC subsystems. These are for instance installed in (nuclear) power plants. Requirements for Personal Protection Systems (PPS) can sometimes only be fulfilled by redundant PLC's. In process controls redundant PLC's are only used in rare cases.

6 Namespace

The flat namespace of SCADA systems has already been described in the alarm section. Some SCADA systems (like PVSS-II) provide the notion of control objects or structured data which is a rare case. In all other cases so called field objects must be specified. These are objects which consist of a list of properties (implemented as I/O points) and a set of methods (implemented as macros or function calls). One of these approaches is the UniNified Industrial COntrol System (UNICOS) at CERN [5].

DCS systems and most of the custom/ collaborative systems are record – or device oriented. The difference being that typically one record is connected to a single I/O point and provides this way all sub features of a record implementation like individual engineering units, display- and alarm limits. The device oriented approach allows to connect several I/O points. The major difference being the fact that an object oriented device implementation provides methods and states for a device while (EPICS) records only serve a certain set of built in functions.

Naming hierarchies are not specific to a type of implementation. They are available for some systems of any kind. For sure hierarchical naming schemes are desirable.

IMPLEMENTATION STRATEGIES

After having shown all the possible controls approaches it is time to have a look at the implementation of control systems.

Starting from the I/O level one has to decide whether commercial solution are required, feasible or wanted. Special I/O does not always require custom solution for the font-end controller. Signals can be converted into standard signals but this does not apply for all kinds of signals. Resolution, repetition rates and signal levels might require custom developments which must be integrated into the overall control architecture. Even if the signals can not be connected to standard I/O interfaces it might be possible to develop I/O controllers which implement a field bus interface which allow the integration with commercial control systems. Once this level of integration is not possible custom front-end controllers like VME crates come into play.

Besides the decision whether special I/O requires dedicated custom solutions one has to decide who will do which part of the work? Does for instance the necessity of VME crates prohibit the delivery of a 'turn key' system built by industry? Or does a PLC based front-end system require a commercial SCADA system for high level controls?

Turn Key Systems

It is a clear trend in industry to deliver turn key systems. It allows a modular design of the whole system. Individual components can be subcontracted to several companies and tested locally. Once delivered to the construction site the primary acceptance tests have already been passed and the second phase, to integrate the subsystem into the global control system begins.

While the detailed specification of control loops etc. is now part of the subsystems contract, the customer has to specify clearly how much information of the subsystem must be made available, what the data structures will look like and which connection (field bus/ Ethernet) will be used.

Most turn key systems are delivered with PLC's. The construction of the Swiss Light Source (SLS) has shown that also a VME based I/O system running a CCS – in this case EPICS – can be successfully commissioned [6].

PLC Based Systems

PLC based systems are a consequence of the turn key ansatz. The next obvious approach might be to look besides commercial PLC's also for commercial SCADA systems. The advantage is clearly the same like for the PLC: stable software, no programming – only configuration, support and good documentation. At DESY we have successfully established a relation between the controls group which provides a CCS service based on EPICS and the utility group which uses the EPICS configuration tools to set up their control environment. The big advantage though being that the EPICS code can be adjusted to the special requirements from both sides.

Industrial Solutions

The difference between CCS solutions and commercial solutions is fading away as soon as industry starts to deliver and support collaborative control systems. At KEK a company was contracted to supply programmers for the KEK-B upgrade. These programmers were trained in writing drivers and application code for EPICS. As a result the KEK-B control system is a mixture of software developed partly by industry and partly in house. This is another example for an industrial involvement for a CCS implementation.

COST

The question: "Was is the total cost of ownership (TCO) of a PC?" has kept people busy since PC's exist. The answers vary to all extremes. The question what is the TCO of a control system might give similar results.

If you go commercial you have to pay for the initial licenses the implementation which is typically carried out by the supplier or by a subcontractor, and you pay for the on going software support which might or might not include the update license fee.

If you go for a collaborative approach, you might contract a company or implement everything on your own. A question of 'time and money' as industry says. You will have more freedom and flexibility for your implementations but also a steeper learning curve. You can rely on the collaboration to provide new features and versions or you can contribute yourself. A major difference calculating the long term costs for a control system.

At DESY one can roughly estimate that the (controls application)-support for a commercial approach – here D/3 - and the -support for a collaborative approach – here EPICS - is nearly the same. The software support and upgrade license fee is equivalent to one and a half FTE's – which is about the manpower necessary to support new hardware and to upgrade EPICS.

CONCLUSIONS

Depending on the size and the requirements for a controls project the combination of commercial solutions and solutions based on a collaborative approach is possible in any rate between 0 and 100 percent. This applies for all levels from implementation to long term support. Special requirements on safety issues or a lack of manpower might turn the scale commercial. The necessity to interface special hardware, special timing requirements, the 'having the code in my hands' argument or the initial costs for commercial solutions will turn the scale collaborative. As long as collaborative approaches like EPICS stay up to date and run as stable and robust as commercial solutions, both will keep their position in the controls world in a complementary symbiosis.

REFERENCES

[1] http://www.gses.com/d3.html
[2] A New HERA Cryogenic Control System - Requirements and Objectives -; Volkhard Klinger, DESY; ICALEPCS 1999, Trieste
[3] H1DCM - A Network Based Detector Control and Monitoring System for the H1 Experiment; G. Eckerlin, M. Hensel, S. Karstensen (DESY), S.K. Kotelnikov, I. Sheviakov (Lebedev Physical Institute); ICALEPCS 2001, San Jose
[4] The LHC Experiemnts' Joint Controls Project, JCOP; Presented on behalf of JCOP by D.R. Myers, CERN, Geneva, Switzerland. ; ICALEPCS 1999, Trieste
[5] http://lhc-cp.web.cern.ch/lhc-cp/Workshop/2002-03/Slides/Session3/1chs.pdf
[6] Purchasing Accelerator Subsystems as Turnkey Components; S. Hunt, PSI, Villigen, Switzerland; ICALEPCS 2001, San Jose