The Influence of Industrial Applications on a Control System Toolbox

Peter Clout
Vista Control Systems, Inc.
134B Eastgate Drive, Los Alamos, NM 87544, USA

Abstract

Vsystem is as an open, advanced software application toolbox for rapidly creating fast, efficient and cost-effective control and data-acquisition systems. Vsystem's modular architecture is designed for single computers, networked computers and workstations running under VAX/VMS or VAX/ELN. At the heart of Vsystem lies Vaccess, a user extendible real-time database and library of access routines. The application database provides the link to the hardware of the application and can be organized as one database or separate databases installed in different computers on the network. Vsystem has found application in charged-particle accelerator control, tokamak control, and industrial research, as well as its more recent industrial applications. This paper describes the broad features of Vsystem and the influence that recent industrial applications have had on the software.

I. Introduction

Developing control systems for physics research from basic components requires considerable effort and the acceptance of risk. Now, software products that can form the basis of an experimental physics control system are entering the product market. This paper describes one such product. Software products exist that were developed originally for factory automation but experience has shown that these products do not network well and are difficult to apply in a research environment, often costing more in total than a home-written system! Vista Control Systems has gone the other way—finding industrial markets for control system software originally developed for physics research. These new industrial applications have added to the flexibility of the Vsystem software in a way that benefits all users without threatening the flexibility and openness of Vsystem.

Vsystem consists of several modules: Vaccess, Vdraw, Vscript, Valarm, Vlogger, and a number of utilities.

II. Vsystem's Real-Time, Networked Database

The architecture of Vsystem requires that all components of the application, be they supplied with Vsystem or user-written, communicate only through the Vaccess database. A system's Vaccess database is defined by ASCII files and installed as global sections. The overall application database is usually made up from individual components distributed among the computers of the system. Collectively, the databases are a data model of the application, modelling both the actual connections and derived data.

The Vaccess database has many real-time features including change notification, alarm and warning checking, dynamic linking of hardware access and data conversion subroutines, and automatic I/O. There are many fields in each channel, or record, of the database that allow the modelling to be complete. Each channel is known by a name up to 40 characters long which is defined by the project. Database channels support most data types as either single valued or array channels. Table 1 lists the channel fields of Vaccess V2.2.

Channel Fields	Channel Name Current Value Current Value Significant Change Equipment Limits Clipping Enable Channel Data Type Read Only Enable, Constant Channel Channel Text Label Interest Count
Conversion Fields	Conversion Subroutine Name Conversion Enable Conversion Parameters (10) Built-in Linear Conversion Parameters (2)
Alarm Fields	Alarm and Warning Checking Enable Alarm and Warning Limits Alarm Label Alarm Type (Integer Channel)
Hardware Fields	Automatic I/O Enable I/O Type Hardware Type Hardware I/O Subroutine Name Hardware I/O Function Hardware Parameters (10) Survey Rates Automatic Survey Enable Input or Output
Display Fields	Display Limits Text Display Format Data Units

Table 1. Vsystem Database Fields

A. Database Access Routines

A library of access routines allows full access to the run-time database. Routines are included to search the database in various ways and to request and cancel change notification by wake-up

or interrupt routine (AST) execution with a parameter. Change notification can be requested on the change of *any field* in the database. The library of access routines handles the network in a manner transparent to the user. Network access can be synchronous, leading to simpler but slower code, or asynchronous, leading to code a little more complex but up to 15 times faster [1].

In physics data acquisition and control systems redundancy is quite rare and problems are fixed as necessary. In some industrial systems, spare sensors and sensor channels are maintained and when wiring changes are made these changes must be reflected in the software at the closest level possible to the hardware change. Access routines incorporating redundancy issues have been added to Vaccess for future release to all customers.

III. Creating Graphic Displays

It is fair to say that there is a broader range of people who will use industrial systems and that the operator interface has to reflect this. In industry there is often a strict categorization of jobs with the result that a system that requires the use of a keyboard to operate it is unacceptable. These considerations also added the requirement for run–time support for trackballs and the options necessary to restrict or completely eliminate the possibility of modification of the system at run–time. For another application, *flashing colors* were added as an option to the Vdraw color palette.

Another industrial testing project required complete, networked program control of the Vdraw windows displayed on a workstation and control of several of the tools in each window. Thus, the Vscript process (Vsystem scripting facility) which was sequencing the test could also determine the windows being displayed to an operator. Therefore, in this case the workstation or X–terminal has no requirement for a keyboard or mouse.

B. Complete Graphic Toolbox

Vdraw, under Xwindows, contains a complete set of drawing, editing and windowing tools. Users can create both prototype control screens and control screens ready for application. Vdraw features an intuitive, easy–to–use graphics toolbox, Figure 1, with which users can create data acquisition and control windows rapidly, without programming.

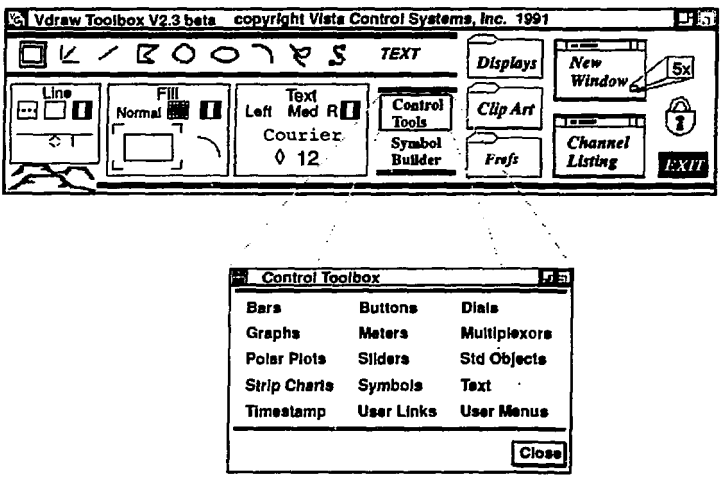

Figure 1. Vdraw's Graphics Toolbox and Control Tools Menu

In draw mode, both passive graphics and interactive graphics (known in Vsystem as *control tools*) are placed in the window. Control tools can display data from the Vaccess database, allow for changes of data in the Vaccess database, or both. The exact form of the control tool and the database connection(s) are defined with a Vdraw generated form. Vdraw also has full window editing capabilities.

When a window is placed in active mode, Vdraw connects the control objects to the Vaccess database and they immediately show the current state of the Vaccess database and hence the application.

C. Flexibility

Because in industrial systems the same window could be used in different applications—except for the actual configuration of some of the tools, such as menus—the definition of a menu can now be defined by a file rather than be hard-coded in the window.

D. Rapid Prototyping

Many industrial systems are implemented by a vendor for the client. With a toolbox like Vsystem, it is possible to develop a realistic demonstration as part of the sales effort. This has been done on several occasions with good success. An example done in-house is shown in Figure 2 on the following page.

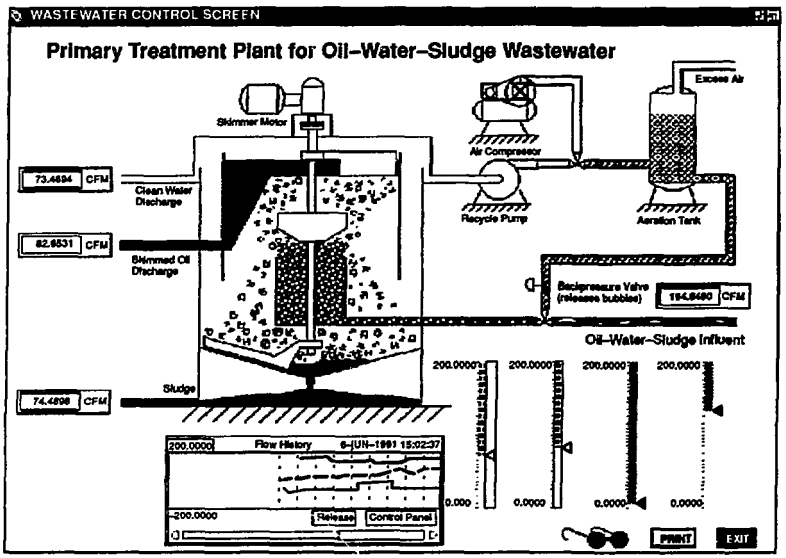

Figure 2. This control screen was created in one day by an experienced Vdraw user. It illustrates the *text*, *bar*, and *strip chart* control tools used to monitor and control the flow of the process.

IV. Connecting to the Hardware

Connecting the Vaccess database to the CAMAC or any other hardware is as simple as including the name of a handler when defining the Vaccess database and supplying the hardware address (branch, crate, station and sub-address in the case of CAMAC). An example handler is provided in source form with Vsystem. From this template, handlers are easy to develop.

Vscan, a generic reader for input channels, is also supplied in source form so that it can be used as a basis for application specific needs. One of the advantages of the change notification feature of Vaccess is that the real-time database can maintain an interest count and then readers like Vscan can change the scan rate of a channel depending on the level of interest in the channel.

V. Logging and Playing Back Data

Vlogger, Vsystem's data logging component, reads data from the Vaccess database and writes the data to standard output devices such as disk files. Vlogger will also play data back into the Vaccess database. This playback function can include all the channels logged in a log file or just a selection of channels. Data can be logged at specified time intervals, on changes in channel value, or by an event defined by a binary channel. Logging can be gated by a binary channel. The list of channels being logged by Vlogger can be increased or reduced dynamically.

Vlogger includes facilities to take a "snapshot" of all or part of the channel settings in a Vsystem database and restoring those settings at a later time.

Vlogger is based on the client–server model and is fully networked so that a user can connect to any logging process in the system without logging on to that system and obtain the status of that logging task or modify the logging in any way.

VI. Monitoring Alarm/Warning States and Errors

Valarm, the interactive alarm module of Vsystem, monitors channel alarms and exceptions. Valarm will monitor all channels in an application or a defined subset. Users acknowledge and reset alarms simply by clicking with the mouse.

Alarm data can also be logged to a printer, to a disk file, or, as the result of a recent industrial requirement, to a DECtalk unit.

VII. Using Scripting Facilities Designed for Non–Programmers

The first industrial application required that there be a simple way to define the sequences of operations required to collect data from an experiment or to control a process. Vscript, as described above, was the result and it is being heavily used by two industrial customers.

Vscript provides a programming environment in which non–programmers can use scripting facilities to define sequences of operations. Users interact with the Vaccess database using a simple English–like syntax that allows anyone to develop scripts. The Vscript syntax is keyword based. For example:

```
Raise TANK_12:PRESSURE to 1100 psi
                or
TANK_12:PRESSURE should be 1100
```

Both these lines are acceptable and would put a value of 1100 into the channel named TANK_12:PRESSURE.

Vscript supports a full range of mathematical functions, file I/O capabilities, control flow constructs and logical operators. Vscript also contains several error handling facilities and supports an interface to the operating system command processor for running other programs or processes.

VIII. Industrial Influences

E. Performance

One would imagine that it would be the research environment that would emphasis performance but in our experience so far it has been in the industrial testing environment where overall performance has been emphasized over function. This has involved the re–writing of many key Vaccess subroutines for performance and adding new routines that further reduced the per–update overhead. We have achieved performance increases of up to six–fold by this effort.

Vdraw display update rates have also been dramatically increased for some of the key tools. Here, similar performance increases have been achieved by careful use of X–windows and in one case offering the user a choice of X–window display techniques.

F. Reliability

The tolerance to software and other errors in industrial applications is much lower than in the research environment. A consequence is that industrial applications are more fully tested and problems are reported for correction. This has resulted in a continuing decrease in the number of errors in the released software and an improvement in the error handling.

G. Report Generation

The reporting from an industrial system is vital to the corporation. This might be the paper report of the shift or test or the reporting to the company database as part of the manufacturing system. To support this we have added mailbox options to some of the tools and we will be extending the capabilities here over the coming months to provide standard interfaces to common database/report systems.

H. Port to a Real–time Kernel, VAX/ELN

While VAX/ELN is almost unknown in experimental physics it has many enthusiastic users in the industrial market. This is because it is extremely well integrated with the development environment on VAX/VMS and well supported by Digital Equipment Corporation. More recently, the availability of a real_time VAX (rt300) and a component to mount on a controller board has resulted in the rt300 VAX with a VAX/ELN license becoming available in VME and CAMAC as well as in other I/O systems. It was these developments that encouraged the porting of Vaccess (and soon the other components of Vsystem) to VAX/ELN.

IX. Summary

With the industrial applications, Vsystem has grown in capability, flexibility, and consistency in ways that benefits all applications, including physics applications. All of the areas of influence described above are of value to the physics community and most are of considerable value.

The strength of Vsystem lies in the X–windows graphics, the networking, the openness and the broad applicability of the system. The influence of the industrial applications has added considerably to the system. For the future we are developing new automation tools and new ports of the package.

X. References

[1] Network Performance for Graphical Control Systems, Clout, P., Geib, M., Westervelt, R. *These Proceedings*