THE ISL-CONTROL-SYSTEM UPGRADE: A MOVE FROM AN IN-HOUSE IMPLEMENTATION TO A COMMERCIAL CONTROL-SYSTEM

W. Busse, C. Rethfeldt
Hahn-Meitner-Institut Berlin, Glienicker-Str. 100, 14109, Germany

Abstract

The control system software of the ISL-accelerator facility at the Hahn-Meitner-Institut Berlin is presently being converted and upgraded from a system which was developed in-house to the commercially available Vsystem by VISTA Inc. The upgrade plan for the continuously running facility has to cope with the maintenance of the installed hardware, with the maintenance of the old control system platform, the port of the knowledge base and tools contained in the old system to the new platform and the maintenance of the new platform. The maintenance includes updates due to modifications of the accelerator, of accelerator components or of operational rules. All this has to be provided with as little interference with every-day operation as possible.

The paper discusses the added value gained by using a commercial system and to which extent the commercial system covers tools and features which were part of the previous system and its operational procedures.

1 THE UPGRADE PATH

The old ISL control system dates from the late 1970's and was originally designed for the VICKSI accelerator complex [1]. It is based on a single server processor (PDP11), a CAMAC field bus, CAMAC interface modules and software which has been developed in-house. While all the installed hardware was or was made commercially available to HMI specifications, the software implementation used up a comparatively considerable amount of manpower, which was no longer available after the implementation and running in phases.

Despite of its still modern internal design and its extensibility on the hardware level, the system has encountered considerable draw-backs because of decreasing performance when, after the extension of the accelerator facility, several operating teams wanted to have independent but simultaneous access, because of poor graphics and because of the fact that any upgrade of the control system software (e.g. increase of the processing power) would have required additional personnel and, above all, the previous know-how.

Therefore it was decided to move to a new system with distributed processors and data bases, with modern graphics and as much commercial software as possible, but to stay with the CAMAC field bus and interfaces, at least as long as maintenance can be guaranteed. The move had to be seamless to allow for continuing accelerator operation and it had to be achieved with the limited man-power available.

The choice taken in the course of the 1990's consists of an OpenVMS cluster with workstations and X-Terminals for the presentation and human interaction levels, with servers for the distributed data bases and transaction libraries, with rt-VAX's as front end servers for concurrent access to the field bus hardware, with PVWave (Visual Numerics Inc.) to provide graphics for data evaluation as an intermediate step and with Vsystem (VISTA Inc.) as the basic control system software [2].

Vsystem comes along with a general set of tools and application programs which control systems have to supply anyway. However, it goes without saying that accelerator specific applications and field hardware specific drivers and conversion routines are at the customer's responsibility.

Both versions, the old and the new, of the ISL control system will be operating in parallel for some interim time. Vsystem based applications will only be allowed read access to accelerator parameters in the Phase 1 Approach. Active accelerator control is foreseen for Phase 2, by moving adequate parameter groups or accelerator sections from the old to the new platform in complete parts.

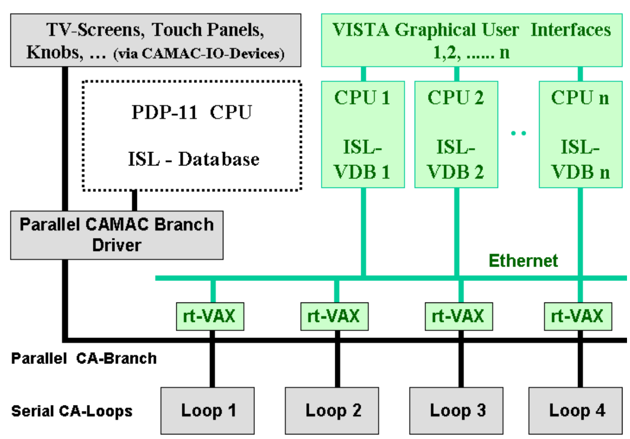

Fig. 1: Schema of the ISL-Control System Upgrade. VISTA Databases (VDB) are distributed among several CPU's, supporting the Graphic User Interfaces on X-Terminals and handling the CAMAC communication by remote procedure calls to Serial Highway front end servers (rt-VAX's). Simultaneous access to the Serial Highways from either the Parallel Branch or the rt-VAX's is co-ordinated by hardware semaphores.

2 THE ADDED VALUE

2.1 The Multi-Platform Solution

Although originally designed and implemented to run under the OpenVMS and VAX-ELN operating systems Vsystem is now supported on a wide range of operating platforms including MS-Windows and UNIX-derivates. Hence the Vsystem user is practically not limited in his choice of operating environment. Together with 2.2 this opens a new area of extensibility for the ISL control system upgrade.

2.2 The Networked Backbone

As in the old ISL control system the backbone of Vsystem is a configuration data base with channel access and processing libraries (cf. Fig.2). The channels, i.e. the data base entries, represent acquisition or set point parameters of the process control facility. The processing libraries provide common API's for channel access by application programs on the human interaction level as well as for calls of conversion routines and drivers on the hardware access level.

On the application level the API is transparent with regard to the location of the data base which holds the channel being accessed. This feature allows for transparent application programming with regard to accelerator parameter access which invisibly may be tied to a local or remote database in a heterogeneous network.

The drivers and conversion routines have to cope with the hardware interfacing the process equipment and the specific behaviour of that equipment. Hence these parts of the transaction libraries have to be implemented by the Vsystem user unless he integrates equipment for which either the manufacturer or VISTA provides the required software, e.g. an OPC based interconnection.

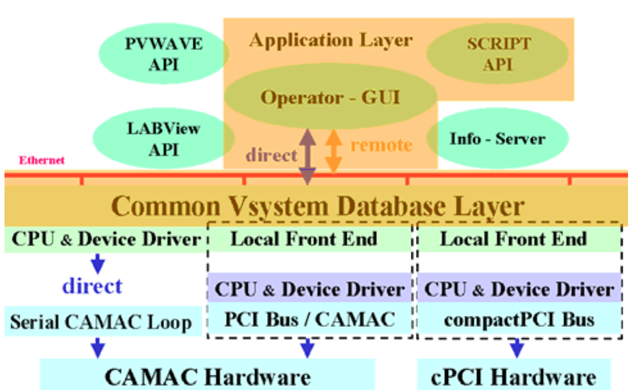

Fig. 2: Tool integration into a uniform control system. Vsystem, indicated in orange, features the unified access of various control applications to a scaleable heterogeneous CPU configuration driving a wide collection of long standing and new hardware.

The above mentioned features will increase the performance and the extensibility of the new ISL control system. The data base can be duly partitioned and distributed after porting, thus increasing the processing power as required. Where local intelligence is needed, additional CPUs with a local data base can be added. They could be equipped with local control panels using the same channel collection, thus assuring data integrity throughout the system (cf. Fig.2).

With Vsystem the upgrade provides a uniform access to centrally driven hardware and front end PC's, gluing together long standing and modern hardware and software into a uniform and consistent control system.

2.3 Graphical User Interfaces

On the application level Vsystem incorporates a drawing tool for graphical applications and a bunch of general application programs such as scripting, logging and alarm handling, all of which can be live linked to respective accelerator parameters. In addition, the ISL controls group has provided adequate API's for the coherent access to these parameters from other commercial tools, such as LabVIEW and PVWave to complete the advantages of GUI's for machine analysis and beam physics.

As part of the Phase 1 Approach numerous graphical information displays, such as the one shown in figure 3, have been implemented in relatively short time. They display a live overview of the status of accelerator sections of interest [3].

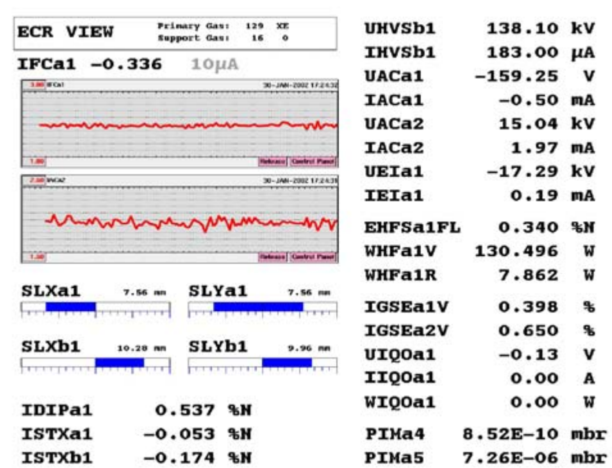

Fig. 3: A typical Vsystem Graphical User Interface. Trend Charts, Live Text and Bars display the actual status of the ECR Ion Source of the RFQ injection beam line.

For remote visualisation and access to archived data a Linux Application Server was installed to collect and save device data and to generate 24h trend charts which can be viewed via the internet [4]. It also supports WAP data exchange with mobile phones, runs predefined checks of the data and sends alarm messages when requested (cf. Fig.4).

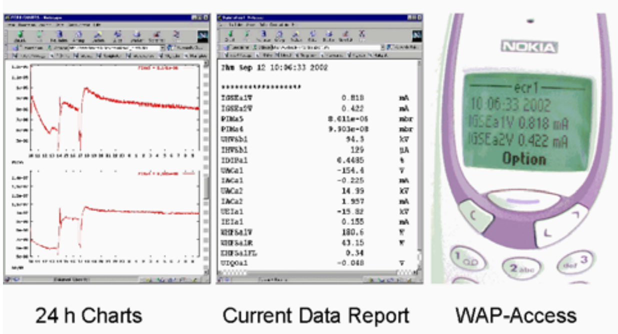

Fig. 4: A Linux Application Server provides remote monitoring of the ISL-Facility. The example illustrates the monitoring of the ISL-Ion Source.

As modern graphical tools often allow for rapid prototyping, machine physicists get more and more interested in developing their own analysis programs for machine study. In addition, physics experiments often ask for access to beam line parameters for logging purposes or even for control access to be able to cycle the experiment through required beam variations. Most of such experiments are run by local PC's equipped with individual experiment specific software products. In the case of LabVIEW, the controls group provides the VI-library for the support of remote channel access. Figure 5 is an example for the LabVIEW integration of a fast local DAQ on a Windows2000 PC and the remote access to the cyclotron radial probe position as reading and setting parameters of the accelerator control system.

Fig. 5: A C-programmed LabVIEW VI-library allows standardised remote access to parameters of the main VISTA control system. This example is part of the ISL control system illustrating the turn-pattern acquisition of a beam in the ISL cyclotron by moving a differential radial probe.

2.4 Small Surprises

In general embedded differences between the old ISL control system and Vsystem become obvious right away when reading the documentation. Different views of what a control system transaction should do or of what the intrinsic features of certain operations should be, are often hidden somewhere in the processing path and not all obvious at first glance.

In the old ISL control system a channel read or write access reports any type of transaction error to the calling application program in an error parameter of the API. Vsystem, on the other hand, has a fixed mechanism to propagate system-errors from the handler to the GUI level which does not allow for user additions of error codes. Therefore device failures, transmission or conversion errors which are detected within drivers or conversion routines cannot be reported to the application level.

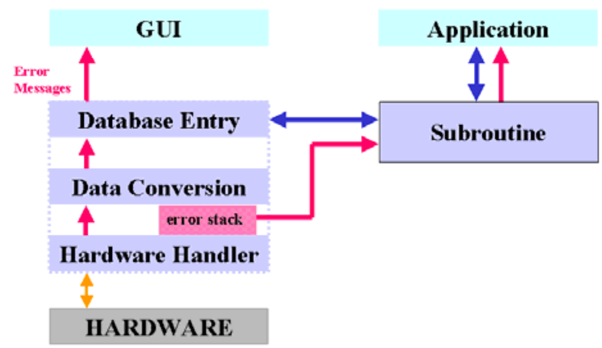

Fig. 6: Example for channel access detours of applications which have to cope with handler or conversion errors

Figure 6 indicates a work-around by using an error stack within the channel's data base entry, which can be checked by application programs if necessary. Data integrity, however, requires several data transfers to assure that transaction and error datum belong together. The application program has to "lock" the channel, do the channel read or write access, acquire the error datum and "release" the channel. Hence, the number of data base accesses, and network transfers ins the case of a remote data base, is increased by a factor of four.

3 SUMMARY AND OUTLOOK

Vsystem is a powerful commercial control system kernel with a collection of integrated tools for implementing and upgrading control systems under the hard conditions of a running facility and of restricted manpower. Our positive experience with transparent API's in heterogeneous networks and on multiple processing platforms leads to continue the distribution of control to front end PC's if it reasonably applies, to have them work as PLC's with a Vsystem kernel and database, and to generally use remote access by operator GUI's at operator locations.

4 REFERENCES

[1] http://www.hmi.de/isl/index.html
[2] http://www.vista-control.com/
[3] C.Rethfeldt, W.Busse, "A Seamless Control System Upgrade", PCaPAC'02, Frascati, October 2002
[4] http://www.hmi.de/isl/ics/datonl/index.html