# Network Performance for Graphical Control Systems

Peter Clout, Mark Geib, and Robert Westervelt
Vista Control Systems, Inc.
134B Eastgate Drive, Los Alamos, NM 87544, USA

*Abstract*

Vsystem is a toolbox for building graphically–based control systems. The real–time database component, Vaccess, includes all the networking support necessary to build multi–computer control systems. Vaccess has two modes of database access, synchronous and asynchronous. Vdraw is another component of Vsystem that allows developers and users to develop control screens and windows by drawing rather than programming. Based on X–windows, Vsystem provides the possibility of running Vdraw either on the workstation with the graphics or on the computer with the database. We have made some measurements on the cpu loading, elapsed time and the network loading to give some guidance in system configuration performance. It will be seen that asynchronous network access gives large performance increases and that the network database change notification protocol can be either more or less efficient than the X–window network protocol, depending on the graphical representation of the data.

## I. Introduction

Performance is one of the considerations when configuring computer control systems. Other considerations are equipment and software costs. In order to help our customers to make intelligent decisions we have made some initial performance measurements on network real–time database access for both synchronous and asynchronous remote access, as well as some measurements to compare network database change notification against network X–protocol for graphical data presentation.

## II. Vsystem's Real–Time, Networked Database

Vaccess is the real–time database component of Vsystem [1]. A library of access routines allows for 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. The library of access routines handles the network transparently to the user. Network access can be either synchronous or asynchronous.

With synchronous access to a remote database, the program making the Vaccess routine call will not continue execution until the request has been sent over the network and a reply received, a process that can take many milliseconds. The network messages contain few useful bytes leading to inefficient cpu and network utilization.

With asynchronous access, a program can make many calls and then call a wait routine, at which point the program will not continue execution until all the calls have been completed. Not requiring an immediate answer means that the Vaccess routines can include many, if not all, the remote database access requests in a single network packet with a consequent dramatic increase in cpu and network utilization.

The arrival of X–windows and X–terminals has recently given the implementers of graphical control systems more configuration options. One can use workstations all running the graphics software and all accessing the data over the network, or one can use a single, powerful, processor running a single copy of the graphics software and serving the users at X–terminals. Both configurations have advantages and disadvantages. Here we attempt to quantify the network issues in this choice.

## III. Remote Database Access Measurements

Vsystem supports VAX/VMS and VAX/ELN. All of these measurements were made between two VAXstation 3100 model 30 workstations rated at 2.7 VUPs, running VAX/VMS V5.4–2, current VAXstations have performances about four to five times that of the VAXstation 3100 model 30. It is important to note that these measurements were intended to compare the different protocols and they can never replace benchmarks on the proposed computers as many factors affect the system performance apart from the cpu rating.

The network protocol used by Vsystem in these measurements was DECnet. The test consisted of a program running in one VAXstation 3100 model 30 which called multiple "RPUT"'s and "RGET"'s to a database channel in a database on another VAXstation 3100 model 30. "RPUT" and "RGET" are Vaccess library calls to put and get a real number to and from a channel. Standard VMS library calls were used to access the elapsed time and cpu time, and VMS NCP was used to report the network bytes and messages.

Table 1 on the following page lists the measurements made for 1000 calls. Measurements for larger and smaller numbers of calls scaled with the measurements in Table 1.

| | Kbytes Sent | Kbytes Received | Messages Sent | Messages Received | CPU Sec | Elapsed Sec | Kbytes/ CPU Sec | Kbytes/ Elapsed Sec |
|---|---|---|---|---|---|---|---|---|
| Synchronous RPUT | 28 | 16 | 1,000 | 1,000 | 3.73 | 8.88 | 7.5 | 3.2 |
| Asynchronous RPUT | 28 | 0 | 21 | 0 | 0.54 | 0.57 | 52 | 49 |
| Synchronous RGET | 20 | 24 | 1000 | 1000 | 4.31 | 9.31 | 10.2 | 4.7 |
| Asynchronous RGET | 20 | 24 | 15 | 17 | 0.87 | 1.14 | 50.6 | 38.6 |

NOTE: All measurements for 1000 calls

*Table 1: Performance Measurements for Network Database Access*

The column showing the number of messages sent and received clearly shows the effect of asynchronous calls, and the effect on cpu overhead can also be seen in the cpu column.

Figure 1 summarizes the results in terms of the throughput against the demand in requests/second. The asynchronous RPUT performance is improved by the lack of need for replies and that accounts for the high throughput. Asynchronous RGET calls return the value and hence the approximately halved performance.

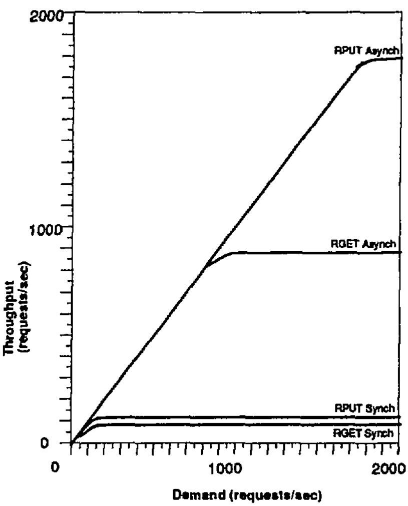


Figure 1: Remote Database Access Performance Between Two VAXstation 3100 Model 30 Computers

The remarkable impact of the requirement of synchronous calls to wait for a reply is shown in Figure 1. The network overhead associated with sending each request as a separate message is about six times the basic call overhead. The elapsed time is further increased over the increase in cpu time by the need to wait for a reply on each call.

## IV. Network Loading

If one considers the message overhead of 48 bytes in addition to the 20 bytes in the "RGET" request message and the 24 bytes in the reply, a total of 140 bytes on the network is associated with each "RGET". Allowing an arbitrary 50% network loading and assuming no collisions, one can achieve a total network throughput of about 4,500 "RGET"'s/second. With asynchronous "RGET"'s this network throughput can be doubled as about 50 "RGET" calls can be included in one network message with the 48-byte overhead. Using the same arguments, asynchronous "RPUT"'s will have the bandwidth further doubled to about 18,000 "RPUT"'s/second because there is no reply message.

All the Vsystem tools use a feature in Vaccess by which any process on the network can request notification of a significant change in any field in a channel. Thus, Vdraw will take a local copy of the value and request notification of a significant change. It is then the responsibility of the remote database to notify Vdraw of the change when it happens. This ensures that the network and computer processing bandwidth is only taken with required information. With change notification, the need for polling is removed and while the network throughput will be about 4,500 changes/second (the network loading of a change notification is about twice that of an asynchronous "RGET") those changes will all be significant, requested changes rather than mostly checks on unchanged data.

## V. Graphics Measurements

For these measurements the same VAXstation 3100 model 30 computers were used with one containing a Vaccess database and a program to generate changes in that database. For the Vaccess communications measurements, Vdraw was run in the other VAXstation 3100 model 30 and the network traffic measured as the data was displayed with different Vdraw control tools. The tools used were a text display, a bar displaying two values, the second value as a small triangle, and a slider with a readback channel. For the X-window network protocol measurements the same test was run but with Vdraw running on the same VAXstation 3100 model 30 as the database and the other VAXstation 3100 model 30 acting as an X-server. In this case, the network traffic was the X-protocol. In this case measurements were made with the X flush rate set both fast, 0.01 seconds, and slow, 10 seconds. Different flush rates did not change the number of bytes transferred but dramatically changed the number of messages. When the flush rate was set slow, the network messages were full. Table 2 shows the results of these measurements and compares the number of bytes transferred.

| Text Display | Bytes/ Update | Updates/ Message Sent | Updates/ Message Received |
|---|---|---|---|
| Vaccess | 85 | 16 | 39 |
| X-Protocol Slow Flush Rate | 48 | 28 | 103 |
| X-Protocol Fast Flush Rate | 48 | 1.7 | 10 |

| Bars | Bytes/ Update | Updates/ Message Sent | Updates/ Message Received |
|---|---|---|---|
| Vaccess | 168 | 8.2 | 28 |
| X-Protocol Slow Flush Rate | 81 | 16 | 26 |
| X-Protocol Fast Flush Rate | 81 | 11 | 13 |

| Sliders | Bytes/ Update | Updates/ Message Sent | Updates/ Message Received |
|---|---|---|---|
| Vaccess | 168 | 8.2 | 28 |
| X-Protocol Slow Flush Rate | 375 | 3 | 3.5 |
| X-Protocol Fast Flush Rate | 375 | 0.2 | 0.5 |

Table 2: Network Performance for Vaccess and X-Protocol Communications

For simple graphic objects, the X-protocol is more efficient in network usage than the Vaccess AST protocol. One reason for this is the amount of data transmitted with the AST to the requesting process in addition to the changed value. This is to minimize the possibility that the process will have to access the remote database further in order to complete the processing of the change. The difference in network efficiency also reflects the optimization in the updating of text and slider data displays [1]. Bar updates can get considerably more complex depending on the options chosen. The large number of bytes transferred to update the slider reflects the complexity of the slider and the fact that it displays the data in graphic and text form. Here the Vaccess mechanism is more efficient in terms of network loading.

Vdraw offers the user the possibility of displaying information with objects of arbitrary complexity and the network loading could become a factor in designing and configuring a system.

## VI. Conclusion

From the measurements it is quite clear that wherever possible, asynchronous database access should be used. When data is being monitored over the network, the performance increase in using change notification will of course depend on the number of changes but in most systems the increase in performance will be dramatic.

The measurements of the network loading of the X-protocol are quite striking in their efficiency. The setting of the X flush rate was observed to affect the perceived performance of the graphics.

Further measurements will be made to understand system performance and identify areas for optimization.

## VII. References


[1] The Influence of Industrial Applications on a Control System Toolbox, Clout, P. *These Proceedings*