# NEXT GENERATION TELEMETRY DATA ACQUISITION WITH WINDOWS® NT

Larry J. Heminger
SBS Technologies, Inc.
2265 Camino Vida Roble
Carlsbad, Ca. 92009 USA

## ABSTRACT

There is a wave of change coming. It started in the industrial automation community and it is slowly and surely working its way into aerospace, satellite and telemetry applications. It's called the PC, and its not just for simple quick-look data anymore. Using state-of-the-art commercial hardware and software technologies, PC-based architectures can now be used to perform self contained, reliable and high performance telemetry data acquisition and processing functions – previously the domain of expensive, dedicated front end systems. This paper will discuss many of the key enabling technologies and will provide examples of their use in a truly next generation system architecture based on the Microsoft® Windows NT Operating System and related features.

## KEYWORDS

PC-based Applications, Telemetry Architectures, Windows NT, Graphical User Interface (GUI), ActiveX®, Network Distributed.

### INTRODUCTION

What exactly is a next generation telemetry system? There are probably as many correct answers to this question as there are readers of this paper. In this writer's opinion, however, the key concepts that differentiate a next generation architecture from older, traditional designs include: (1) Near Unlimited Scalability, (2) Easy to Tailor and customize by the End-User and (3) Open, Easy and Immediate Access to Data.

**Scalable** - From single-user laptop to a network of high-end servers, a next generation system scales gracefully to the application at hand. Scalable designs are typically based on a network distributed architecture and take advantage of symmetric multiprocessing.

**Customizable** - Allows users to easily tailor the system to meet specific and changing mission requirements. The software is designed from the ground up to accommodate the addition of new processing algorithms, I/O hardware, server functions and custom display screens.

**Open Data Access** – Provides well-defined, standard mechanisms for data exchange between a wide variety of software products in both a stand-alone and network-based environment. Data is provided in both off-line and near real-time forms using popular defacto and industry standards enabling test engineers and scientists to focus on data, not programming.

These features, easily achievable in the Windows® environment, have been available for years in the systems that monitor and control many of our world's nuclear power plants, factories and manufacturing shop floors. Research and Development are currently being performed focused on maximizing the use of these commercial technologies for use in telemetry and aerospace data acquisition and processing applications. This paper presents a number of these technologies and describes how coupled with powerful PC platforms, a revolution in price/performance for telemetry computing is underway.

## HIGH PERFORMANCE, LOW COST PLATFORMS

A measure of a computer's performance can be expressed in terms of two important ingredients: CPU processing and I/O bandwidth. As computing technology is changing so rapidly, it is important to examine both current capabilities as well as trends. Trends in Windows NT based computing are symmetric multiprocessing architectures using high performance Pentium Pro, Pentium II and Alpha-based CPUs. These processors provide computational bandwidth on a par with their traditional RISC/Unix-based big brothers as illustrated in the following table:

| Processor | SPECint95[(1)] | SPECfp95[(1)] |
|---|---|---|
| Popular Windows NT Platforms | | |
| 200 MHz Intel Pentium Pro | 8.2 | 6.8 |
| 266 MHz Intel Pentium II | 10.8 | 6.9 |
| 333 MHz Digital Alpha 21164 | 9.8 | 13.4 |
| 500 MHz Digital Alpha 21164a | 15.0 | 20.4 |
| Popular Unix/RISC Platforms | | |
| 180 MHz MIPS R10000 | 10.7 | 19.0 |
| 300 MHz Sun Microsystems UltraSPARC II | 12.1 | 15.5 |

As one can see, NT-based platforms provide similar compute capabilities as high-end Unix workstations. Fully configured Pentium systems, however, are typically a quarter the price (or less) of their similarly configured workstation counterparts.

At 133 MBytes/Second (MB/S) peak transfer rate, PCI currently dominates the personal computing market for a plug-in, high performance peripheral I/O bus. Performance is easily on par with other open standards such as VME (40 MB/S), VME64 (80 MB/S) and Multibus II (80 MB/S), and even compares favorably to high performance proprietary busses when price is considered. 100 Mbit Ethernet, Ultra SCSI and high performance video cards can all be easily procured for the PCI bus at under $200 (US) each.

For telemetry data acquisition applications, the key enabling technology is buffered, PCI bus mastering I/O boards. These boards can be configured, via a Windows NT device driver, to transfer data directly into the CPU's memory at rates approaching full bus bandwidth. These devices require very few CPU cycles to operate, and allow bus bandwidth to be shared nicely among the various I/O cards in the system. Once transferred into memory, telemetry data may be processed, displayed and archived by the single or multiprocessor PC.

## WINDOWS NT FACTS

Windows NT is not a real-time operating system. As such, buffered bus mastering I/O boards, as previously discussed, are required in order to handle high data rates. On-board hardware time stamping is also required in order to provide accurate, microsecond (or better) resolution of telemetry, avionics and analog input data. These features can be found in many of the new PCI board designs.

Windows NT does provide real-time priority levels, multitasking, multithreading, asynchronous I/O and interprocess communication capabilities, sufficient for the implementation of soft real-time applications. Properly designed data acquisition and device driver software can provide deterministic response of a few milliseconds on average, with a worse case of 10 ms (running at real-time priority).

Windows NT was designed by many of the same engineers who pioneered Digital's VMS Operating System. Unlike its Windows-95/3.1 counterparts, NT was designed to be a highly reliable, stable and secure operating environment for the commercial telecommunications, industrial automation and enterprise computing markets. With features such as C2 rated network and file security[3], built-in RAID and UPS support, clustering and protected virtual memory address spaces, NT provides a robust platform for demanding telemetry applications[2].

## COMPONENT TECHNOLOGY

One of the key enabling technologies provided by Microsoft is its concept of software components, called ActiveX Controls. ActiveX Controls (formerly OLE Controls) are shared objects that can be inserted into and used by various Windows-based applications at development and run-time. This technology allows a developer/vendor to encapsulate a particular function, interface and/or graphical control within an ActiveX Control, isolating the client application from many of the associated low level details. An application which uses one or more Controls is called a container, as it 'contains' the Control. Containers can be just about any Windows-based application developed using Microsoft and/or third party tools such as Visual C/C++, Visual Basic, Pascal, etc.

ActiveX Controls are similar in concept to a Dynamic Link Library (DLL) or shared library, except that they provide many additional benefits such as better versioning control, and the ability to operate on them as an abstract object, rather than just a set of subroutines. In addition, ActiveX Controls may also be contained in the Microsoft Internet Explorer Browser, providing access and control of the system via the Web. Figure 1 below illustrates these concepts.

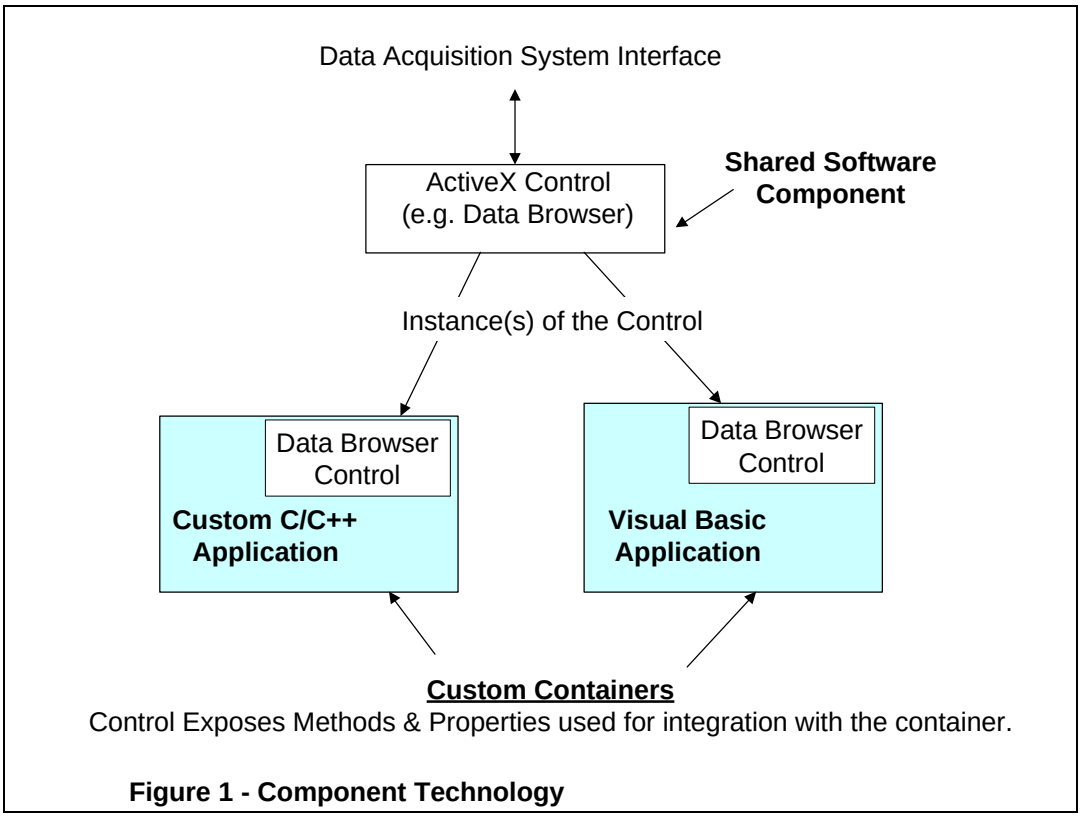


**Figure 1 - Component Technology**

The use of software component technology provides tremendous benefits to the data acquisition system user and developer. These benefits include:

**Software Reuse** – The same exact Control may be used and re-used by many Windows-based applications (containers), simultaneously. Thus, duplication of similar functionality is minimized.

**Rapid Prototyping** – Custom Windows-based applications can quickly be developed and prototyped using a suite of Controls as a toolbox. The developer need not be concerned with low level details, only the high level methods and properties which have been exposed.

**Better Configuration Management** – Controls can be designed as self-contained objects, which encapsulate a particular interface or function. An upgrade and/or bug fix to the Control will be immediately be reflected by all container applications which use it.

Extensive use of component technology has been made in the next generation data acquisition system. The user interface to this system is comprised of a series of ActiveX Controls providing configuration management, system setup and operational capabilities. The configuration database interface is also encapsulated in a Control, providing built-in versioning and independence from the underlying relational database being used. Customers may use these Controls both as an end-user, and as a developer. If desired, a developer can create custom system operation, data analysis and/or utility programs quickly and easily using the available ActiveX Controls as rapid prototyping tools.

## DISTRIBUTED NETWORK ARCHITECTURE

A next generation data acquisition system would not be complete without a network distributed architecture. Demands for greater numbers of I/O interfaces, numbers of simultaneous operators, as well as the need to integrate in near real-time with other computing platforms dictate this fact. We cannot assume anymore that the term PC is synonymous with ‘single user’.

Using ISO/IEEE standard network interfaces and protocols, it is entirely feasible for a Windows NT based system to acquire, process and distribute data over the network in a high performance, near real-time manner. Windows NT includes built-in support for TCP/IP, Novel, Streams, Point-to-Point and AppleTalk protocols, while still others are available from third parties. Popular network interfaces such as standard 10 Mbps Ethernet (10baseT), 100 Mbps Ethernet (100baseT), ATM, FDDI and dial-up links are well supported, and are very inexpensive. All high performance network interfaces are available for the PCI bus.

In order to tackle the larger scale telemetry data processing applications with PC technology, a network of Windows NT based PCs will be required. The various data

acquisition, processing, archiving, display and post processing tasks can be distributed across the network and assigned to individual PC platforms based upon loading and responsiveness requirements. For instance, a multiprocessor server may be used for data acquisition and archival tasks, while multiple single processor PCs may be used for system setup and dynamic data displays. A sample distributed architecture is illustrated in Figure 2.

There are many ways in which a distributed network architecture can be implemented in Windows NT. Sockets and Remote Procedure Call (RPC) are two standard mechanisms which have been used in the past for implementation of traditional client-server architectures. Next generation architectures, however, typically require a higher level of network functionality and abstraction. Today, there are many commercial vendors providing network 'middleware' products. Middleware products provide high level features such as message-oriented, object-oriented and publish-and-subscribe protocols which layer on top of the standard lower-level protocols. Middleware products are ideal for designing robust, highly reliable and fault-tolerant data acquisition systems.

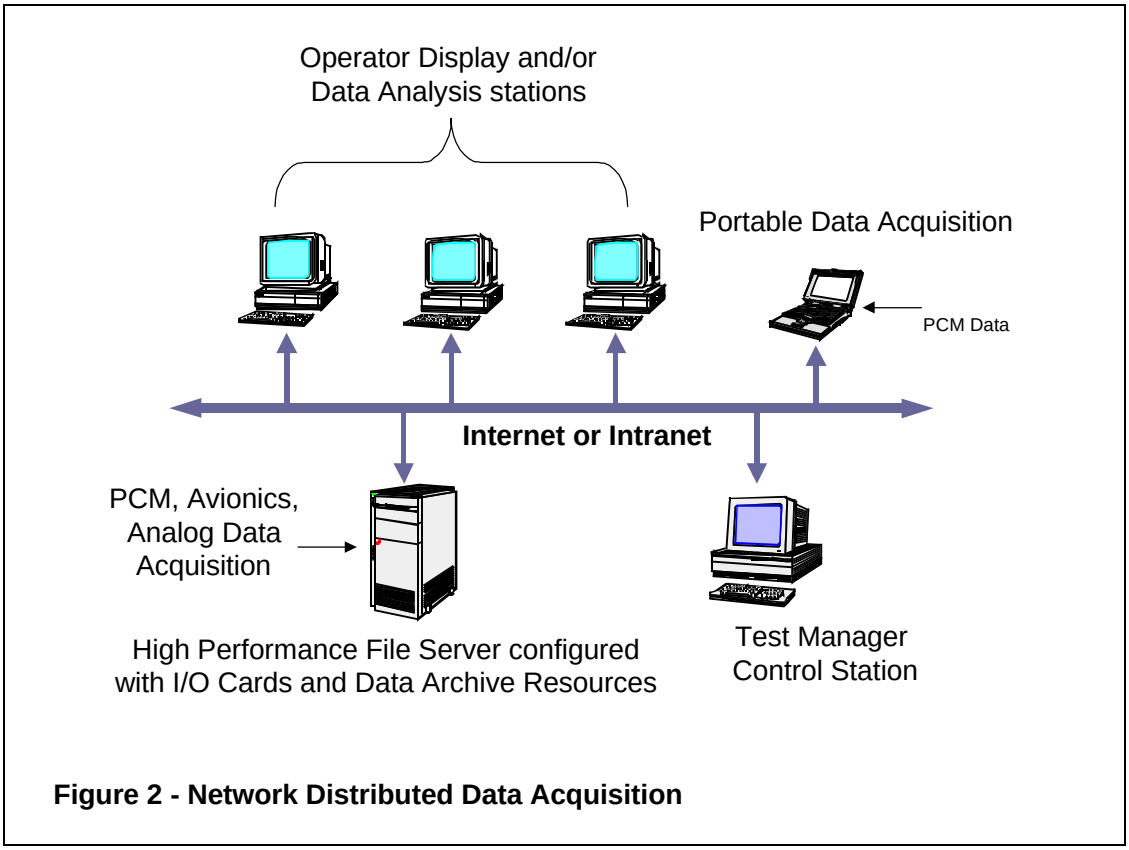


**Figure 2 - Network Distributed Data Acquisition**

A commercial middleware-style product has been selected for use in the next generation architecture. The architecture provides both a Network Applications Programming Interface (API) in conjunction with a memory-resident real-time database, used as a Current Value Table (CVT). The Network API provides a sort of software backplane,

facilitating near real-time communication between clients, servers and other external applications. Servers are responsible for keeping CVT data up to date, while clients provide data displays, processing and analysis functions. Any client and/or server may access data in any CVT on any PC.

The power of the network distributed concept has been combined with the software component technology described earlier, as illustrated in Figure 3 below. This architecture encapsulates the Network API within a suite of ActiveX Controls, which provide high level, network distributed access to the data. Custom applications may now be rapid prototyped using the available ActiveX Controls to completely manage the system, display and/or process data in near real-time, from anywhere on the network with a minimum of programming.

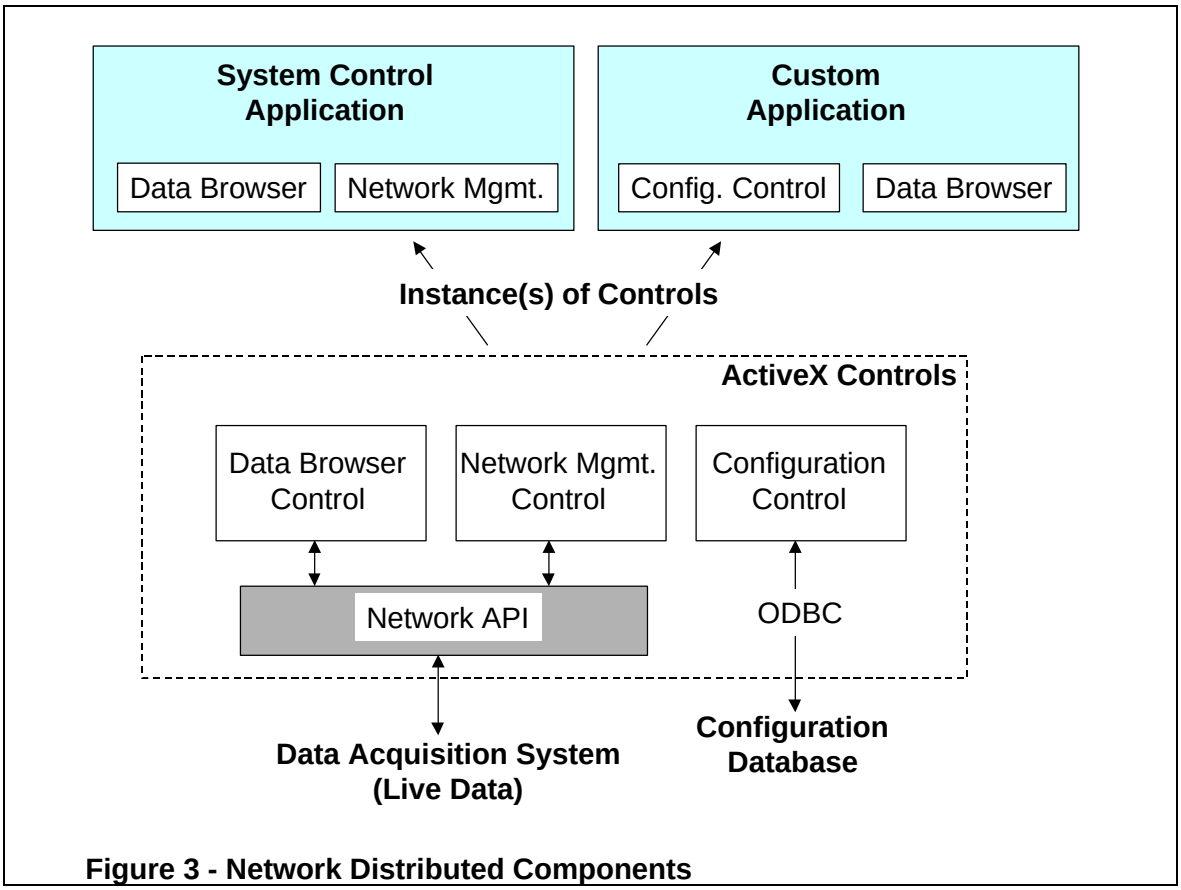


**Figure 3 - Network Distributed Components**

## OTHER TECHNOLOGIES

There are many other technologies and concepts provided by the Windows environment which facilitate easy data exchange, software re-use and application rapid prototyping. By acquiring a working understanding of these techniques, the data acquisition system user will gain unprecedented flexibility and ease of use in the ways data can be presented, reported, displayed, and analyzed. These technologies include:

**Open Database Connectivity (ODBC)** – An open, vendor independent software interface for relational database connectivity. With ODBC, applications may be developed and deployed without a pre-arranged knowledge of the underlying database engine. ODBC uses SQL (Structured Query Language) as a call-level interface, and builds SQL statements at run-time, appropriate to the actual database engine being used. It allows the end-user to select a database engine of choice, rather than it being dictated by a particular system vendor.

**Component Object Model (COM)** – The COM is a foundation for the object-based system that focuses on reuse of interfaces, and it is the model which facilitates OLE and ActiveX programming. It is an operating-system level object model and the interface specification from which any number of interfaces can be built.

**Visual Basic (VB) and Visual Basic for Applications (VBA)** – VB is a powerful Microsoft development environment based on the Basic programming language. VB is one of the fastest, easiest to use, rapid prototyping tools for GUI screen development available. VB programs may make use of component technology as previously discussed, to quickly and easily develop custom applications using one or more ActiveX/OLE Controls. VB is also an important and productive tool for engineers and scientists who are not proficient C/C++ programmers. VBA, a subset of VB, is a convenient mechanism for extending and customizing the Microsoft Office suite of applications and to interface them with a variety of data sources.

**Microsoft Foundation Class (MFC)** – A set of object oriented C++ classes for use in the development of C++ based GUI programs. MFC provides a rapid prototyping environment for C++ applications, encapsulating many of the standard Windows-style GUI constructs in easy to use classes. MFC is a key technology, as compliant applications provide a consistent look-and-feel, with all other MFC based programs.

## CONCLUSION

While Windows NT based PC platforms will not satisfy all real-time data acquisition requirements, they can certainly be used as a next generation solution for a vast number of them. By carefully selecting high performance systems, I/O boards and peripherals, many pitfalls may be avoided. Coupled with the numerous time saving and rapid prototyping software technologies widely available, a PC-based system must be considered as a serious possibility for any new data acquisition requirement. The use of these key software technologies allows the end-user to select the right tool for the job, taking the definition of ‘open system’ to the next level. Best of all, even the highest performance PC platforms are commercially available at a fraction of the cost of specialized workstation and front-end equipment.

## REFERENCES

(1) Benchmarks based on results published by The Standard Performance Evaluation Corporation, 10754 Ambassador Drive, Suite 201, Manassas, VA. 20109. SPECint95 is a set of eight compute-intensive integer benchmarks. SPECfp95 is a set of 10 compute-intensive floating point benchmarks.

(2) Ragen, Matt, “Real-Time Systems with Microsoft Windows NT”, Microsoft Windows NT Technology Brief.

(3) Article ID: Q93362, “C2 Evaluation and Certification for Windows NT”, Microsoft Knowledge Base, 24 March, 1997.