

# **NETWORK DATA ACQUISITION AND PLAYBACK OF MULTIMEDIA DATA**

**Michael Portnoy and Hsueh-Szu Yang**  
**Teletronics Technology Corporation**  
**Newtown, PA USA**

## **ABSTRACT**

Traditional data acquisition systems have relied on physical connections between data sources and data receivers to handle the routing of acquired data streams. However, these systems grow exponentially in complexity as the number of data sources and receivers increases. New techniques are needed to address the ever increasing complexity of data acquisition. Furthermore, more advanced mechanisms are needed that move past the limitations of traditional data models that connect each data source to exactly one data receiver.

This paper presents a software framework for the playback of multiplexed data acquired from a network acquisition system. This framework uses multicast technologies to connect data sources with multiple data receivers. The network acquisition system is briefly introduced before the software framework is discussed. Both the challenges and advantages involved with creating such a system are presented. Finally, this framework is applied to an aviation telemetry example.

## **KEY WORDS**

Networked Data Acquisition, IP Multicast, Real-time Data Playback, Data Routing

## **INTRODUCTION**

Modern Internet browsers conveniently allow users around the world to remotely access information via an easy to remember URL (uniform resource locator or address). Regardless of the complexity of the highly distributed nature of this information, the user is abstracted from the underlying data sources. Data acquisition systems must also retrieve information from remote servers, which in the realm of flight test telemetry, range from simple transducers to even more complicated networks of data acquisition systems. As the demands on flight test instrumentation increase in response to the growing complexity of new aircraft, it becomes ever more difficult to maintain this abstraction.

Traditional data acquisition systems have used physical connectivity methods to link data sources with data receivers. These methods require cabling directly between data sources

and data receivers. Historically, this physical cabling of sources directly to receivers has been sufficient. However, these methods are not scalable and begin to break down as the number of data sources and receivers increases. Specifically, the number of physical connections that are required grows exponentially as sources and receivers are added to the acquisition system. Manual routing of data is time consuming, error prone, and insufficient when multiple receivers need access to a single data source. Optimizing these scenarios is necessary to minimize bandwidth utilization.

One approach to addressing this problem is to shift the burden of data routing and retrieval away from hardware data acquisition systems to routers and software-driven workstations. The great flexibility of software can then be leveraged to help simplify this process. This paper examines a software framework for efficiently transporting multiplexed data to different software components such as video playback, strip chart plotters, and alarm displays.

The framework that is presented uses a new, emerging model of data acquisition. This model provides for the easy retrieval of data from different data sources without the need to reconfigure physical links. Instead, the configuration burden is handled via the software framework. However, the task of efficiently multiplexing and processing data streams from different sources is quite challenging since multiple destinations can share the same data packet. This system expands upon the traditional end-to-end communication of data acquisition systems by allowing communication between a single data source and multiple destinations.

## **EXAMPLE SCENARIO**

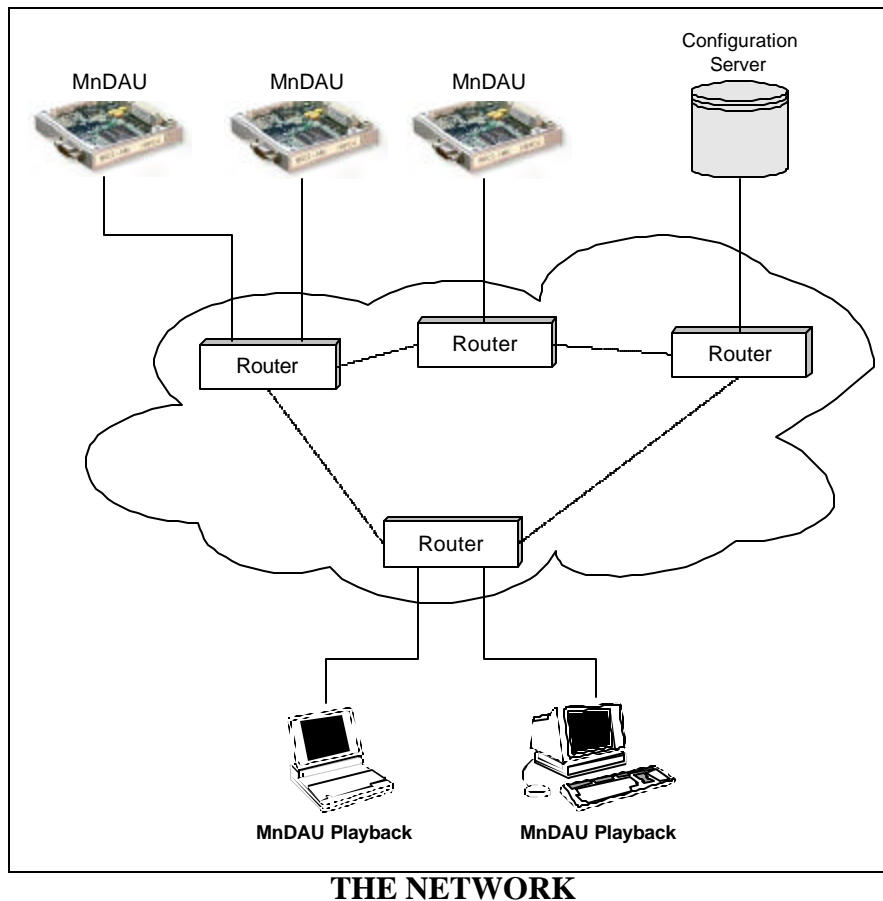
A user logs into an arbitrary workstation to monitor a lab environment where there is a collection of many sensors and cameras, each with varying degrees of data criticality. Each sensor and camera is a remote data source and multiple workstations will need to share data from the same source. Workstations will be used to decode and play back video data from the cameras and to visualize sensor reading using charts, graphs, and indicators. A data acquisition system for this scenario should meet this minimal set of requirements:

1. Dynamically connect data sources and data receivers.
2. Support data transfer between multiple data sources and receivers within the same session.
3. Process extremely large quantities of data.
4. Transfer data in real-time.
5. Simultaneously display multiple data streams synchronously on a workstation.

## SYSTEM OVERVIEW

This data acquisition system is logically divided into a back-end data acquisition network and a front-end software framework, which contains the data analysis and visualization components such as video playback, charts, and alarms. The network physically links the network acquisition system (referred to as MnDAU or miniature networked data acquisition unit), routers, configuration servers, and workstations that run the MnDAU playback software, which is described in this paper. Figure 1 illustrates the connectivity of the network.

**Figure 1. The Data Acquisition Network**



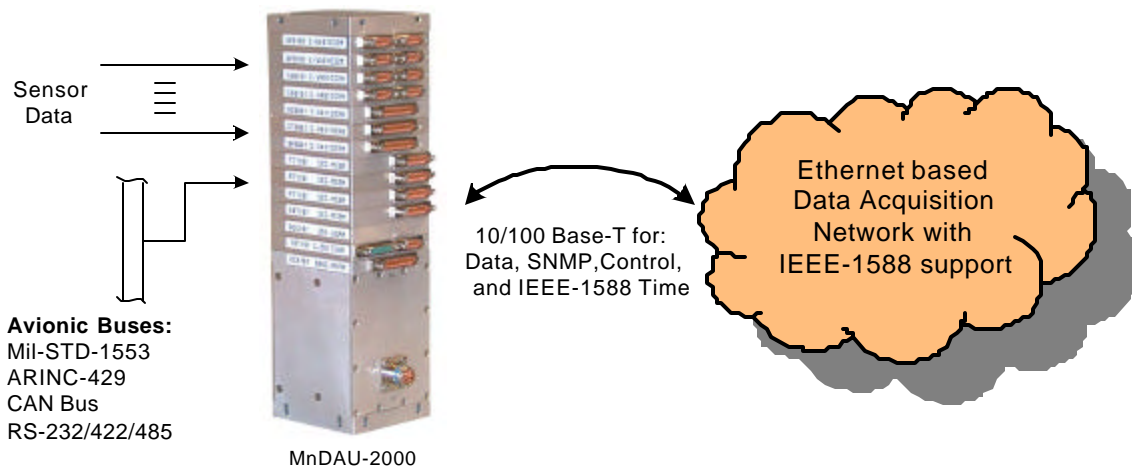
The four major components of the network are:

1. MnDAU(s) (miniature networked data acquisition unit)
2. Router(s)
3. Configuration Server
4. Workstation running the MnDAU playback software.

**MnDAU:** The MnDAU is a network data acquisition unit, which is a system that conditions and acquires multiple input sensor channels with similar or dissimilar data

types, and combines them into data packets for transmission over the network fabric. The network interface also provides the unit with a gateway for setup and configuration, SNMP status and control, and time synchronization using the IEEE-1588 time standard. The unit acquires data from accelerometers, strain gages, various temperature sensors, pressure sensors, synchro/resolver sensors, LVDT, discrete signals, video, and an entire host of avionics buses. Acquired IEEE-1588 time is distributed within the acquisition unit for time tagging sensor data and avionics bus data. Time can also be used to trigger time dependent events, such as simultaneous sampling, within the acquisition unit or across multiple acquisition units on the network. This unit supports multicast technology, which allows data to be received by more than one workstation running the MnDAU playback software.

**Figure 2. Network Data Acquisition Unit**



**Routers:** Routers are the network devices that connect all of network nodes together. They optimize bandwidth usage in addition to performing the basic functions of relaying and delivering data. The router forwards no data unless a receiver has been defined even if the MnDAU is putting out IP packets over the network fabric. Routers also optimize the path between the source and the destination.

**Configuration Server:** The configuration server stores information about the data sources and the data formats. Information from the configuration server is provided on demand and is not transmitted with the data stream. Virtual connections to the data sources are created using the descriptions provided by the configuration server.

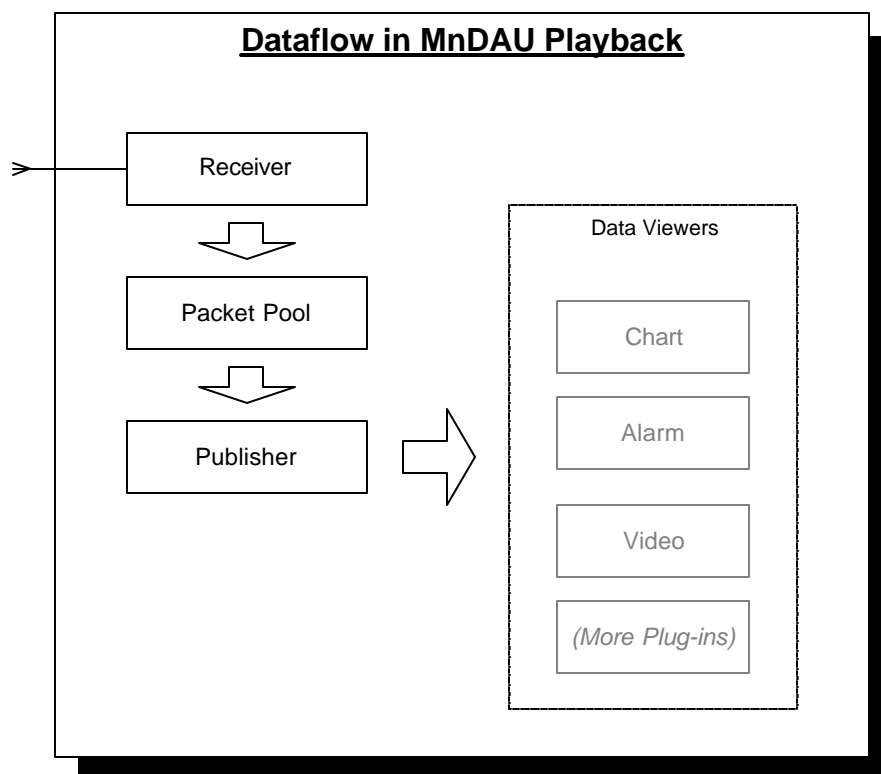
**Workstation:** The MnDAU playback software communicates with the configuration server, enumerates the data sources, and presents them to the user via a graphical user interface. The data model is then dynamically created based on the user's input. Once a source is specified, the MnDAU playback software sends a message indicating that it wants to join the multicast session. Each session contains one source and an unlimited number of receivers. The router then builds a tree structure that links the data source, router, and data receivers. Data packets are efficiently sent along the tree without duplication.

## MNDAU PLAYBACK SOFTWARE DATA FLOW

The MnDAU playback software consists of three fundamental modules as illustrated in Figure 3:

1. Receiver Module
2. Packet Pool Module
3. Publisher Module

**Figure 3. Dataflow in MnDAU Playback**



The receiver module accepts packets from the network and moves them into memory. The packet pool module buffers data from the receiver module and shares the data with the data viewers. The publisher module exposes an API through which the data viewers access data. Data viewers are notified by the publisher module when a new packet arrives in the packet pool module.

This design has three main benefits: high performance data routing, efficient memory management, and a flexible API. The high efficiency of data movement from the network saves CPU time for other data processing. Since the receiver module's thread has the highest priority, it is guaranteed to have CPU time whenever data is available from the network. Furthermore, the short data path from the receiver module to the

packet pool speeds up the buffering process of the highly volatile network data. Finally, the optimized memory management in the packet pool reduces the time for memory allocation.

Memory usage is extremely efficient since all memory is shared between the data viewers; no memory copying operations are invoked. The publisher signals the data viewers when a new packet arrives by passing a reference of the newly arrived packet to the data viewers. If a packet is no longer referenced by a data viewer, then the memory will be returned to the packet pool.

Finally, the generalized API design is flexible and extensible. New data viewers can be easily created as new scenarios arise. Since there is a clear abstraction layer between the core MnDAU data playback mechanism and the data viewers, all complexity about the architecture is hidden from the data viewers. End users can extend this model by creating new data viewers since data can be retrieved as one would normally read a block of memory in a high level programming language.

## **ADVANTAGES**

The MnDAU playback architecture has many advantages. Due to its foundation on IP technology, the architecture is scalable, flexible, and reliable. Many existing technologies and commercial off the shelf (COTS) products can be leveraged and used with this system. Furthermore, continuing, industry wide research in IP technologies will only serve to expand upon its possibilities. For example, several newer router algorithms related to multicast help to reduce bandwidth usage. Other technologies such as deterministic Ethernet are also useful for real-time data transportation.

Physical reconfiguration of connectivity is completely eliminated when a workstation switches from one data source to another one, as long as they are within the same MnDAU network. With the help of the MnDAU playback software, the workstation and the network will find a path that allows data to flow from the source to the workstation. Users only need to specify the name of the source, just as they would enter a URL into an Internet browser.

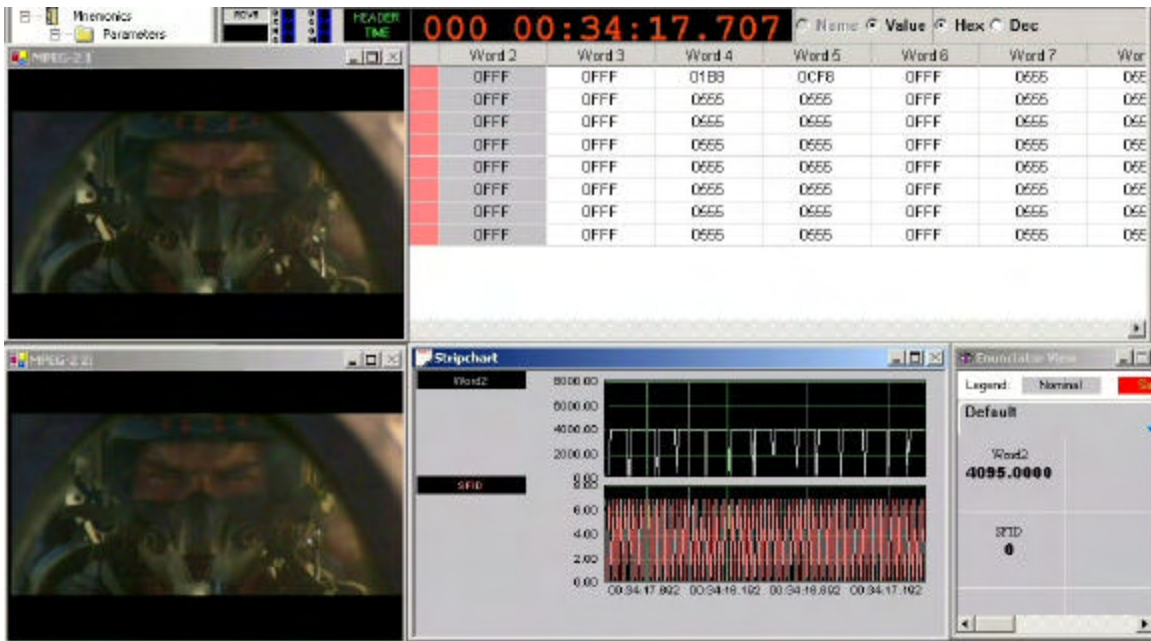
An emerging concept within the data acquisition community is that the data acquisition network itself can act as a database for real-time data. Conventional databases enable clients to query information from a data depository using traditional queries. Applying this technique to the data acquisition network allows a user to similarly retrieve data via queries. This is significant since many data analysis and data mining tools are built upon database queries. These techniques can be further extended to allow real-time querying of the data acquisition network.

## AVIATION TELEMETRY APPLICATIONS

The network acquisition system and MnDAU playback software presented in this paper have many applications to aviation telemetry. In the event of a missile separation, for example, a flight test engineer would be interested in the following measurements: time, MPEG-2 video from a camera, weapons messages on a 1553 bus, and wing strain data from a strain transducer. Each of these types of measurements falls into a different category: time, audiovisual, digital, and analog, respectively. For this event, it is essential that all of the data be time-correlated and multiplexed so that the entire state of the aircraft at the time of the separation can be examined. The MnDAU network acquisition system can be used to acquire this data. Multiple test engineers on the aircraft can connect to the network simultaneously to analyze and visualize the data in different ways using the MnDAU playback software.

Figure 4 shows a screenshot of the MnDAU playback software receiving and playing back time, video, and analog strain data in a situation similar to the one described above:

**Figure 4. MnDAU Playback Software Screenshot**



In this scenario, the MnDAU playback software functions as a software Ethernet decommutator for the data. Incoming, packetized data streams are received by the MnDAU playback software via multicast and are then demultiplexed and routed to the appropriate data viewer. This operation is similar to a traditional hardware decommutator except that this solution is much more flexible and extensible.

## CONCLUSION

The telemetry industry's continuing adoption of network acquisition systems opens up an endless realm of possibilities. By leveraging off the shelf IP technologies such as multicast and automatic routing, intricate networks of data sources and data receivers can be quickly constructed without the need for complex physical connections. The software framework that is presented in this paper allows users to quickly and easily acquire, analyze, and visualize telemetry data using these virtual networks.

Much research is currently devoted to analyzing the process of acquiring real-time data. Streaming databases are one of the most active areas that are being researched. By applying this technology it becomes possible to easily retrieve important information from large quantities of real-time data. The software framework presented in this paper is a first step toward realizing the full potential of streaming databases.

Hopefully, the industry will continue to research and utilize newer network technologies so that data acquisition can continue to be simultaneously simplified and enriched.

## REFERENCES

1. Berdugo, Albert, and John Roach, "Migrating from PCM to Network Data Acquisition System", Proceedings of the 26th European Telemetry Conference (ETC 2006) Germany, May 02-05, 2006.
2. Berdugo, Albert, and John Hildin "A System Approach to Network Centric Data Acquisition System", Proceedings of the 42<sup>nd</sup> Annual International Telemetry Conference (ITC 2006), USA, October 23-26, 2006.
3. Eric Pesciotta and Portnoy, Michael. "An Implementation Of Dynamic Data Acquisition Measurements", Proceedings of the 42<sup>nd</sup> Annual International Telemetry Conference (ITC 2006), USA, October 23-26, 2006.
4. Kupferschmidt, Benjamin and Albert Berdugo. "Designing An Automatic Format Generator For A Network Data Acquisition System", Proceedings of the 42<sup>nd</sup> Annual International Telemetry Conference (ITC 2006), USA, October 23-26, 2006.
5. Tim O'Connell . "PACKET-BASED TELEMETRY NETWORKS OVER LEGACY SYSTEMS", International Telemetry Conference 2005.
6. Larry L. Peterson and Bruce S. Davie., "Computer Networks – A Systems Approach."