ArcoNet: A Proposal for a Standard Network for Communication and Control in Real-Time Performance

Kristi Allik

Department of Music Theory and Composition
Talbot College, University of Western Ontario
London, Ontario, N6A 3K7, Canada

Shane Dunne

Department of Computer Science
Middlesex College, University of Western Ontario
London, Ontario, N6A 5B7, Canada

Robert Mulder

LumiArt
579 Colborne Street #3
London, Ontario, N6V 2B1, Canada

ABSTRACT

This paper proposes that a standard local-area network (LAN) be developed for real-time control and communication among performance systems, such as music synthesizers, tape players, lighting equipment, and slide, laser, or video projectors. The existing MIDI standard, although useful, is too limited for such applications, and most commercial LANs are not designed for the requirements of real-time control. Hence a new network standard is proposed, provisionally called ArcoNet.

After introducing the proposed network and discussing its applications in the studio and in live performance, the paper examines the technical requirements for the network. A brief treatment of issues in local-area network design is given, including discussion of the ISO reference model for Open Systems Interconnection and the CCITT X.400 recommendations for Message Handling Systems. A critical survey of existing LAN technologies is presented, followed by a set of suggestions for the ArcoNet design. Finally the question of how to establish and promote the ArcoNet standard is discussed.

Following the paper is an appendix which contains a set of specific suggestions for the ArcoNet standard. Though tentative and incomplete, it provides a basis for discussion of the ArcoNet proposal in concrete terms.

1. INTRODUCTION

Until recently, the concept of using computer networks in the studio or in real-time performance was only an engineer's dream. That suddenly changed when the Musical Instrument Digital Interface (IMA (1985), Loy (1985)) gave thousands of artists a glimpse of the potential of such integrated systems. However the full potential cannot be realized with MIDI alone. Hence we propose that a new network standard be developed for this purpose.

Connecting all devices involved in a live performance to a common network would give the artist precise control with no need to worry

about operators missing their cues. In the recording or film/video production studio, having several devices on a network would save time by allowing complex sequences of events to be programmed in advance, without the need to change physical device interconnections.

Such networks could include not only musical instruments and signal processors, but lighting and projector controllers, tape transports, gestural and other input devices. Potentially any device which generates or is controlled by digital data streams could be connected. Thus in this paper we will speak of "integrated multi-media networks".

The MIDI standard, while it represents a major advance, is not a suitable foundation on which to build even moderately complex integrated systems. Its basic design lacks any provision for detection and correction of transmission errors, and its bandwidth is too low to permit useful multiplexing of connections.

To realize the concept of an integrated multi-media network, some kind of local-area network (LAN) standard will be needed. We claim that although existing LAN designs might be useful, they cannot in general meet the demands of real-time performance applications. We further claim that because of market pressures, equipment manufacturers cannot be expected to design a suitable communication standard which will be non-proprietary. Therefore it is up to the community of potential users to develop and promote a suitable network standard.

2. WHAT IS BEING PROPOSED

We propose that there be a standard digital network, for real-time data communication between devices used in the performing and recording arts (here referred to as "multi-media equipment" to emphasize that our attention is not restricted to musical equipment alone). To streamline discussion, we have adopted the name *ArcoNet* (Artist's Computer Network) to refer to the proposed system.

The four fundamental purposes of ArcoNet are as follows.

1. *to permit now-incompatible types of multi-media equipment to be interfaced, at least to some degree.*

There are now many devices on the market which produce, or can be controlled by, digital data streams. There are also a plethora of different methods for synchronizing different pieces of equipment, such as SMPTE and MIDI codes, direct digital "strobe" connections, and so forth. The first aim of ArcoNet is to provide a standard network to which these devices may be connected, so that they may exchange digital data and event signals.

A standard network will not by itself permit any two pieces of equipment to be usefully connected. This will usually require specialized hardware (to connect each device to the network), software (to control the interaction, translate codes, etc.), and a defined approach to handling the interaction. A standard network will provide a foundation for such efforts on a technical level, and also make them more reasonable on a practical level. Efforts to interface disparate machines can be viewed as developing a system of general use, rather than an expensive white elephant useful only within a single organization.

2. *to allow general-purpose computers to be interfaced for network control plus data generation, processing, storage and distribution.*

An ArcoNet system would include at least one computer, which might serve as (part of) a master production control console, where an operator could monitor and modify the progress of programmed sequences.

The term *sequence*, as used in this paper, implies any programmed sequence of events. Examples would be musical note sequences, lighting control sequences, and sequences of operations performed by slide, laser or video projectors.

Software for developing and executing such sequences would be the key to ArcoNet's usefulness. When possible, the network should support existing sequencing software such as that developed for MIDI systems. In other cases, the existence of a standard network with publicly-available specifications will foster the development of new software.

3. *to provide for synchronization of multiple devices, by means of a system-wide master timing reference and asynchronous event signalling among sets of devices.*

Some consideration must be given to providing a master timing reference (or "system clock") to synchronize devices across the network. This facility should be independent of existing disparate timing references (such as 24 frame/sec film and 30 frame/sec video).

In this paper, the term *event* means any single event within a performance, such as the sounding of a musical note, the projection of an image, a change in lighting, a particular movement of a human performer, and so on. ArcoNet's usefulness derives from the fact that it allows events to be *signalled* across the network to trigger, or *cue*, other events. In this way, a performance may be

keyed to the actions of human performers (or non-deterministic devices such as kinetic sculptures incorporating some degree of random control).

It should be possible to elaborate a general model of event signalling as part of the ArcoNet specification. This would foster the development of sequencing software such that sequences written for any device could include instructions to wait for cues from other devices, or send cue signals to other devices.

4. *to define a standard set of circuits, methods and protocols (subject to controlled revision) for interfacing existing and future multi-media equipment, which shall be in the public domain.*

For the ArcoNet idea to succeed, it is vital that it be accepted as a *de facto* standard among users and manufacturers. Putting the ArcoNet specifications into the public domain is one way to ensure this. Any proprietary measures such as patents and licensing fees will in general only impede the establishment of a *de facto* standard. With a truly public standard, even the smallest vendors will have full access to the information they need to produce compatible products.

3. CONCEPTION OF THE ARCONET SYSTEM

An ArcoNet network would take the form shown schematically in figure 1. At least one general-purpose computer would be included for controlling the network, and a set of hardware node processors would be used to interface other devices to the network.

Figure 1: schematic diagram of an ArcoNet network

ArcoNet will be useful both in live performance and in the recording or film/video production studio.

In performance, which here connotes music, integrated media, and theatrical performance, all digitally-controllable devices used in a performance will be linked by ArcoNet, along with one or more computers for control. It is also possible that specially built devices, such as a sensor to detect the position of a dancer or motors to control the movement of sets, could be included in the system and interfaced to the ArcoNet.

As a standard network, to which existing equipment could be interfaced, ArcoNet would provide a basis for simpler and more.

flexible device interconnections. Standard protocols defining how events are signalled across the network would provide the basis for general-purpose sequence development software, a subject we discuss further in sections 4 and 6 of this paper.

In the studio, ArcoNet will provide a more stable foundation for the kinds of studio applications now being developed using MIDI. One possibility would be to use a general-purpose computer as a "software front panel" for a studio full of synthesis and signal-processing devices linked by ArcoNet. This would permit not only static device programming, but specification of sequences for automatic execution.

In the studio a standard network, together with an evolving set of protocols governing the interaction of the networked devices, would foster sharing of equipment and software between installations and simplify the introduction of new equipment.

Returning to figure 1, we can add some more specific ideas about a typical ArcoNet system.

Each node processor will contain a microprocessor and memory as well as network interface circuits. Basic network functions would be "wired into" each node using read-only memory (ROM). Further configuration of the network—specifying what each node is to do and when—could then be done by down-loading programs and/or data into the nodes via the network itself. This approach allows many different configurations to be constructed from a few basic node types as "building blocks" (e.g. ArcoNet/MIDI, ArcoNet/SMPTE, and ArcoNet/RS232 nodes).

Not every device connected to the network will require its own node processor. It is entirely reasonable to have one node processor serve two or more devices.

Each ArcoNet setup will contain at least one general-purpose computer equipped with a network interface—probably a more sophisticated one than that possessed by the other nodes. This computer will configure the network with programs and data stored on disk, and will then provide a "front panel" from which the operation of the network can be controlled and monitored. Note that some nodes may also be complete computers with integral mass storage, in which case they could (at least partially) configure themselves.

In live performance applications, it will be necessary to set up and take down the network repeatedly. This suggests that ArcoNet should be based on inexpensive cable, and that it should be self-configuring and self-testing. The latter two capabilities should be built into the standard communication protocols.

Having all stations in a performance set-up linked by high-bandwidth cable suggests other possible uses for the cable. It should not be difficult to include digitized voice signalling for traditional "headset" communication over the network. Digitized voice could also be used to play back pre-recorded cues to human performers or technicians.

4. REQUIREMENTS FOR ARCONET

The design of ArcoNet should reflect the following three fundamental issues.

1. *The importance of basing the network design on sound engineering principles.* It is vital that we study the general principles of network design, and specific networks, before attempting to develop a new system. Cost-reducing compromises should be subjected to rigorous examination before being adopted as part of the design.

2. *The need to anticipate technological changes.* The most promising communication medium at the moment is coaxial cable, but an engineering breakthrough might tip the scales in favour of optical fibres in a few years. The design of ArcoNet must be structured so that new technology can be embraced with minimal redesign.

3. *The need to accommodate as many existing technologies and standards as possible.* For ArcoNet to succeed it must become a *de facto* standard. This cannot happen if an important technology (such as SMPTE synchronization or MIDI) cannot be used with it.

It is also important to keep the cost of ArcoNet within reasonable bounds. An exorbitantly expensive system will never become a *de facto* standard. But as already mentioned, the final design must not compromise sound engineering for cheapness. It should be possible to market standard ArcoNet node processors for under \$500 (U.S.) each, and ArcoNet controllers (plug-compatible with off-the-shelf computers such as the IBM PC) for under \$2000. Even these prices are conservative—it should be possible to bring them down lower.

The design of ArcoNet should follow logically from engineering experience with other LANs. Therefore it should have a layered architecture, permitting some measure of hardware independence. This issue is elaborated in the next section.

Experience has shown that it is often desirable to connect several LANs to form a *supernetwork*. This requires hardware bridging devices called *gateways*, which act as nodes on two or more networks simultaneously and allow information to pass from one to another, and internetworking protocols, which regulate this information exchange. Internetworking capability should be designed into ArcoNet from the outset.

Standard protocols for ArcoNet's basic functions of sequencing, cuing, and timebase synchronization must be defined and published. Protocols for other types of messages, such as packet voice communications and MIDI virtual circuits (see next section) should also be developed. These public standards will form the foundation for third-party software (programs written neither by the software vendor).

Putting ArcoNet's specifications into the public domain will encourage its acceptance as a *de facto* standard. If this does not happen, the market could become dominated by one manufacturer's proprietary system, which is sure to be less powerful, less general, and less accessible than ArcoNet.

Initially, only a few classes of ArcoNet node processors should be

needed, to allow connecting various classes of equipment to the network. An ArcoNet/MIDI box would probably support between two and eight MIDI (IN and OUT) connections. An ArcoNet/RS232 box, with two to eight RS232 serial connections, could be used with devices controlled through that interface, as well as ordinary data terminals. Plug-in boards for connecting computers should be defined for common buses, such as Apple II, IBM PC, and Multibus or VME bus (e.g. Sun workstation). In all, less than a dozen basic designs should be enough to build even very complex networks.

These standard nodes will allow ArcoNet to embrace many existing interface standards, at least at the hardware level. We can go further and ensure that ArcoNet supports the encoding schemes (or presentation-level protocols—see next section) as well. For example, since the MIDI presentation-level protocol defines message types for music playback, there is no immediate need to design a new protocol, especially not to support existing synthesizers. We need only define a standard way for MIDI messages to be transported over ArcoNet, with ArcoNet's advantages of error detection and recovery.

Of course, once ArcoNet became widely used, it would be useful to define new protocols to take better advantage of the network's speed and interactive capability. Advanced synthesizers and signal processors could be built with integral ArcoNet interfaces, allowing much finer control over operation than is now available under MIDI.

5. ISSUES IN LAN DESIGN

This section briefly introduces the most important aspects of LAN design—it is by no means comprehensive. Tanenbaum (1981) gives an excellent treatment of networks in general, while Wood (1985) covers local networks in particular.

5.1 Packet Switching

Most network designs are based on the concept of packet switching. Long communications between pairs of nodes are divided into small blocks of data called packets. Each packet includes some information specifying the intended recipient. This technique, combined with effective rules to ensure that no one node hogs the network for too long, allows many simultaneous communications to be interleaved. Through packet switching, several devices which need to communicate can share the same transmission facilities, reducing costs and resulting in a simpler overall system.

5.2 The ISO Reference Model for Open Systems Interconnection

The complexity of computer network designs requires that they be broken up into smaller entities. The usual approach is to consider a network as a hierarchy of layers or levels, such that higher layers are constructed using the resources provided by lower ones.

Recently the International Organization for Standardization (ISO) has published a standard reference model for networks (Day and Zimmerman (1983)), which has been widely accepted. This model,

which is shown schematically in figure 2, identifies seven distinct layers which may exist in a network design. Each layer provides a set of services to the layer above, by building upon the services provided to it by the layer below. Each layer is implemented in hardware, software, or a combination.

Figure 2: OSI model showing 2 connected stations

These seven layers are not cast in stone. Some network designs may omit an unnecessary layer or implement the functions of two or more adjacent layers in a single subsystem.

Most networks consist of a set of computers called *hosts*, connected by means of a *communications subnet*. In larger networks, the subnet often consists not only of communication links but also of computers which serve to route data around the subnet. Figure 2 shows two hosts, whose communication is being handled by two intervening subnet processors. The latter need only implement the lower layers of the ISO model (those concerned with actual data transmission). Most local area networks are sufficiently simple that no subnet processors are needed to route data between hosts.

The subnet is usually considered to be formed by the physical, data link, and network layers of the OSI model. The physical layer consists of the transmission medium and the mechanical and electrical standards for accessing that medium. It provides the service of physically linking all stations on the network. The data link layer consists of the network interface circuits, and provides the means to send data between stations and to detect transmission errors. The network layer controls the operation of the subnet and ensures that transmission errors are corrected, usually by retransmission of bad data blocks.

Higher layers in the OSI model are concerned with how the basic communication facility provided by the subnet is used. The transport layer provides a straightforward way for programs to communicate over the network, which is independent of the details of the subnet's operation. The session layer provides more application-oriented services such as ways for hosts to identify themselves. The presentation layer is concerned with how data are expressed in binary form for transmission over the network. Finally, programs at the application level implement the operations desired by the network's users.

With a hierarchically layered design, each layer can be implemented without regard for the details of how other layers operate. For example, a subnet based on optical fibres could be substituted for one based on coaxial cable, without having to change the transport, session, presentation and application layers of a system.

Another consequence of hierarchical design is that protocols—sets of rules governing communication—can be designed separately for separate levels. When host *A* communicates with host *B*, one may consider that layer *n* on host *A* is communicating directly with layer *n* on host *B* and vice versa. The fact that the communication must pass through layers 1 through *n-1* on both machines is of no consequence. The two processes at layer *n* are called *peer processes*, and their communication is governed by the *layer-n protocol*.

5.3 The CCITT X.400 Series Recommendations

Another international standard which may have a bearing on ArcoNet's design is the X.400 series recommendations from the Comité Consultatif International Télégraphique et Téléphonique (CCITT (1983)). These define how networks based on the ISO model can be used for the exchange of messages between users, as in electronic mail. ArcoNet's primary purpose is to carry messages such as event signals between devices, hence the X.400 recommendations may provide a useful framework in which to express its design.

Recommendation X.409 will be particularly important. It gives a standard method for defining how data are to be encoded at the presentation level, including a standard syntax for expressing such definitions in print. This will be useful in defining new presentation-level protocols.

5.4 Subnet Design Issues

Wood (1985) identifies seven major design parameters for a LAN. These are *topology*, *transmission medium*, *signalling mode*, *access method*, *data rate*, *interfacing device*, and *type of interfaces*.

A network's *topology* is the structure formed by its communication links. Typical topologies are the *star*, *bus* and *ring*, shown schematically in figure 3. Note that the topology may not be apparent from the physical layout of a network. Figure 4 shows how a ring network might be laid out in the form of a star.

The most popular transmission media are twisted-pair cables and coaxial cables. The former offer better noise rejection, while the latter offer higher bandwidth. Optical fibres, which surpass other media in virtually all respects, still require extremely expensive terminating hardware.

Networks may operate using either *baseband* or *broadband* signalling. In baseband, two voltage or current levels, representing binary 1 and 0, are impressed directly onto the transmission medium. With broadband signalling, the binary signals are used to modulate a higher-frequency *carrier* signal.

The *access method* determines which station may transmit on the subnet at any particular instant, and is part of the ISO network layer. *Time-division* methods allow each station a certain amount of

Figure 3: star, bus and ring topologies

Figure 4: star-connected ring

time on the network. *Polling* methods use a central computer to tell other nodes when it is their turn to transmit. In *token-passing* methods, a special data pattern called a *token* circulates from node to node around the network, and the node in possession of the token may transmit a message before passing it on. In *contention-based* methods, stations may transmit whenever the transmission medium is free, with special action being taken when two messages "collide".

A network's *data rate* places an upper limit on the total amount of information which may be exchanged between nodes in a given time. Most LANs today operate at rates between 1 and 10 million bits per second (Mbps). MIDI connections require 31250 bits per second (0.03125 Mbps), so a 1 Mbps network could potentially support over 30 simultaneous MIDI connections without introducing significant delays.

To be cost-effective, network interfaces must be based on commercial large scale integrated circuits (LSI chips). Such chips, which implement most of the data link layer of the ISO model, are now available for bus and ring topology networks. Some form of microprocessor will also be needed, with DMA (direct memory

access) capability to move information to and from the network at high rates.

The IEEE has proposed a set of draft standards for LANs (Nelson (1983)), which includes standards for both base- and broadband CSMA/CD and token buses, and baseband token rings. These standards will likely be implemented in new LSI chips. The token-passing designs may be useful for ArcoNet, since token-passing protocols offer predictable message delays.

As mentioned earlier, only a few types of interface boxes will be needed, including ArcoNet/MIDI, ArcoNet/RS232, and plug-in cards for popular computers.

5.5 Low-level protocols

Three issues are important in designing data link, network, and transport protocols for ArcoNet. The first is error detection and recovery. It must be possible to detect and correct errors due to noisy transmission media, and ArcoNet's real-time nature further demands that this be done quickly.

The second issue is that different levels of transport service will be required. A big file transfer from a computer to a sequencer cannot be allowed to clog up the network and delay event-signal messages.

Finally the protocols must avoid *deadlock* and *starvation*. Deadlock occurs when two stations are stuck, each waiting for a message from the other before proceeding. Starvation occurs when some station is never allowed to transmit (or is not allowed to do so sufficiently often). These two possibilities must also be avoided at the session and application levels.

5.6 Presentation Level Protocols

ArcoNet will need a common format for signalling events across the network. Each event will typically be represented by a short message from one station to one or more others, indicating only a single number representing the type of event. Although this is a presentation-level concept, it must be supported by lower-level protocols to ensure that event signals can be sent and received with very short and deterministic delays.

A variety of useful presentation-level protocols, or data-encoding schemes, have been defined. These include ASCII for text, NAPLPS (ANSI (1982), Lax and Olsen (1983)) for videotex, and the MIDI messages for musical events. It is important to note that ArcoNet can and should support these existing protocols, as well as new ones which may be defined later.

6. SURVEY OF EXISTING DATA COMMUNICATION TECHNOLOGIES

This section discusses some of the leading LAN and data-interface technologies in current use, and assesses their applicability to ArcoNet.

6.1 MIDI: A Discussion and Critique

The Musical Instrument Digital Interface (IMA (1985), Loy (1985)) is an interface standard primarily designed for point-to-point connections between synthesizers, computers, and signal processors. In terms of the ISO reference model, MIDI defines a physical layer (MIDI cables and connectors), a data link layer (the 31.25 Kbps serial transmission method), and a presentation layer (the various messages and their meanings).

It is a tribute to the success of this standard that most new synthesis and signal processing products since about 1982 have been equipped with MIDI interfaces. This success has led some manufacturers to see MIDI as the basis for integrating all equipment in a studio or real-time setup in a kind of LAN. However MIDI is not suitable as the basis for a usable LAN for several reasons.

First, MIDI is a *simplex* medium. Each MIDI connection carries data in one direction only, from a sender to a receiver, without permitting the receiver to respond in any way. Such an acknowledgement capability is vital for error-correcting protocols. (While error-correcting codes exist for use with simplex channels, they are quite inefficient.)

Second, MIDI's data rate of 31250 bits per second is just too slow for use in a multi-access network. When driving several synthesizers on different MIDI channels from a single MIDI port, the narrow bandwidth limits playback speed.

Third, MIDI provides only 16 logical channels, or station addresses. To be useful, a LAN should provide at least 256 addresses. A related problem is that it is usually necessary to manually switch MIDI-based devices onto the desired channel after each power-up (and manually reset them, if something goes wrong).

Fourth, MIDI is not a multi-access system. It is possible to link several devices to a single MIDI line, but with only one transmitter. Although various "MIDI mixers" have been developed in an attempt to circumvent this, the MIDI standard itself makes absolutely no provision for multiple transmitters sharing the same line.

It is possible to design a star-type network based on dedicated MIDI connections (both send and return) between a central computer and each device. This approach circumvents each of the above problems, and will undoubtedly be tried commercially. However it still has all the problems inherent to star networks, primarily the fact that each new device added to the network will require new MIDI interface hardware on the central computer.

Finally, although a revised "MIDI 2.0" standard is expected soon, which presumably will clear up some of MIDI's present defects, it is unlikely to be sufficiently improved to form the basis for a useful LAN. This is because MIDI is essentially an *interface* standard—for connecting one device to another—rather than a multi-access *network* standard.

6.2 Commercial LAN Products

Virtually all commercial LANs are designed to allow computer resources (i.e. processors, printers, disks) to be shared within an

office environment. Hence they tend to be designed for large data transfers rather than short messages, and for efficient use of costly cable rather than real-time performance over inexpensive cable. These differences make it unlikely that commercial LAN products would be usable, unchanged, for ArcoNet.

Probably the best-known LAN design is Ethernet (DEC (1980)). This is a contention-based, bus topology network using baseband signalling (usually at 10 Mbps) over special coaxial cable. Ethernet pioneered the CSMA/CD access method (Carrier Sense Multiple Access with Collision Detection), which uses the cable bandwidth well but places no upper limit on the time taken by a message to reach its destination. The latter makes Ethernet unsuitable for demanding real-time applications.

Haugdahl (1984) has surveyed the LAN product offerings for the IBM PC and compatible machines, and reports that in 1984 a plethora of totally incompatible systems were available. None of these products are likely to be usable unchanged for ArcoNet, but in some cases they embody excellent design ideas which should be examined carefully.

Mitrenet (Wood (1985)) is a broadband bus network which uses standard community antenna television (CATV) cable. It carries one or more digital data channels using CSMA/CD, and permits several video, FM radio and voice channels to share the same cable. Elements of this product's subnet design might be adapted for ArcoNet.

IBM Corporation has recently announced its token-ring network (IBM (1985), Strole (1986)), which is a baseband coaxial cable ring network using token passing for access control. The fact that this network conforms to the IEEE 802 standard for token ring systems adds extra weight to the 802 standards. The cost of IBM's network components is very high, however, so they will probably not be usable for ArcoNet.

6.3 Computer Buses and the GPIB Bus

Many experimental systems have been built (including some for computer music) in which special-purpose devices are plugged directly into the control, address and data buses of a general purpose computer. This approach has some distinct advantages, but could not be used as the basis for ArcoNet because of the high cost of bus interfaces and cables, and the fact that bus cables cannot be run over distances greater than a few feet.

Another bus technology is the General Purpose Interface Bus (GPIB, also called the IEEE 488 buss) developed by Hewlett-Packard corporation (Stone (1982)). This is a parallel bus (where several bits can be sent at once) designed for connecting electronic test equipment to a computer. The GPIB's distance restrictions are not as serious as those of a general-purpose computer bus. Once again, though, the cost of multi-conductor cables makes this technology unsuitable for ArcoNet.

6.4 Other High-speed Interfaces

Some synthesizer and signal-processor manufacturers have looked to higher-speed standard interfaces, available on some micro-computers, as alternatives to MIDI. An example is the Kurzweil

Instruments' use of the RS-422 port on the Apple Macintosh computer, described by Byrd and Yavelow (1985).

These interfaces, offering speeds up to about 1 Mbps, are still designed for point-to-point connections rather than networking. In context of a true LAN, however, such connections would be useful for moving data between specific devices and the network itself.

7. DESIGN DIRECTIONS FOR ARCONET

The final ArcoNet specification must represent a consensus of opinion of all interested parties, so at this stage it is too early to prescribe the network design in detail. However, given the discussion of LAN design and technology in the previous two sections, we can identify at least some desirable aspects of the proposed network.

The ISO reference model for Open Systems Interconnection has been accepted as an international standard, and its terms of reference are rapidly being assimilated into the vocabulary of design engineers. Hence the ArcoNet architecture should be specified in terms of this model.

The following paragraphs roughly follow the ISO reference model from bottom to top in discussing aspects of ArcoNet's design.

The IEEE 802 draft standards for Local Area Network subnets are also being accepted by engineers, and it seems likely that these standards will be implemented in commercial LSI chips. ArcoNet will benefit by following these standards, by being able to use the chips.

Existing LAN subnet designs should be examined closely to see if they can be used for ArcoNet, because using existing components is nearly always cheaper than designing new ones.

The subnet will likely have a bus or ring topology, using coaxial or twisted-pair cables. The cables should be inexpensive but also durable and reliable, since ArcoNet networks will need to be set up and taken down regularly in live performance applications. CATV cable would be a good choice, because it is inexpensive and very easy to terminate (connectors are cheap and can be assembled without soldering).

Existing physical and data-link standards, such as MIDI, RS-232 and RS-422, should be embraced for connecting equipment to ArcoNet node processors. As already mentioned, only a few station types, each having a different kind of external device interface, should be needed for most applications.

Each node processor should contain a microprocessor, memory, and a network interface based on a standard LSI chip. This chip will implement a standard data link protocol, and firmware for the microprocessor will implement a network level protocol which provides for delivery of messages from any station to any other station or group of stations, with automatic error detection and acknowledgement of receipt. For reliable real-time operation, a deterministic-delay access method such as token passing should be chosen.

At the transport level, at least three classes of service should be available. A raw service, providing for transmission of messages

from one station to one or more others, with acknowledgement but no error correction, could be used for digitized voice transmission. A *datagram* service would provide message transmission with error correction, and would be used for most event signalling. A *virtual circuit* service, using automatic buffering to allow stations to communicate on a byte stream basis, would be useful for MIDI information.

At the session level, *polling* and *acknowledgement* message classes should be defined, together with a way of sending programs to the microprocessors in each station. At the presentation level, a class of *event-signal* messages would also be needed, each event being represented by a number in the message, with network control software (at the application level) defining the correspondence of events to event numbers.

Standard ways of using existing presentation-level protocols within ArcoNet should be defined. For example, MIDI messages would usually be sent over virtual-circuit connections, with packet buffering software knowing enough about MIDI to operate efficiently without delaying messages.

The ArcoNet specification will have to include some statements about the kind of network management software which will exist at the application level. Such software will probably reside in one "master" station, and will take care of network configuration, testing, detection of station failures, assignment of event numbers to potential events, and synchronization of real-time clock signals across the network.

After configuration, software distributed across the network would be concerned with the playback of sequences (see sec. 2). Since sequences would be cued by predefined event signals, it would be possible to specify alternative subsequences, to be executed conditionally depending on which of several cues was received at a given time. This would help the artist escape the "tyranny of tape", which implies that once a sequence playback is started, it generally cannot be stopped or modified to match live action.

Application-level software for ArcoNet will operate primarily through the transfer of brief messages between stations. In keeping with earlier suggestions about conformance to international standards, this software should conform to the CCITT's X.400 recommendations for Message Handling Systems insofar as this is possible.

Further ideas for ArcoNet, some of which are tentative, are given in the appendix.

8. GETTING THERE FROM HERE

Accepting that a standard LAN, designed specifically for the needs of artists working with integrated media, is desirable, we must take steps to develop and promote it.

We have already mentioned the importance of establishing ArcoNet as a *de facto* standard among users and manufacturers. This can only be accomplished by setting out a complete, or nearly complete, description of the system we want, before manufacturers have committed themselves to a different design. As potential users we can publicly declare what we would like to have, but *de facto*

standards are created by the proliferation and use of real, working products.

It is vital that the ArcoNet architecture and protocols be fully public, both to avoid any one manufacturer gaining excessive advantage and to ensure that individual users can design their own interfaces and software when they must. It must be made clear that for the most part, ArcoNet is intended to embrace existing technologies rather than replace them.

The ArcoNet specifications will have to be worked out by agreement among all interested parties. Some person or group of people will be needed, to coordinate discussion and dissemination of ideas. The help of existing groups such as the Computer Music Association and the International MIDI Association should be solicited. Supporting funds must be secured without compromising the public nature of ArcoNet in any way.

When the ArcoNet specifications are complete to the point of being usable, they must be published in some form and distributed. This would be handled either by the coordinating group mentioned above or by a commercial publisher.

Considerable effort will be required to define and promulgate the ArcoNet standard. The alternative, however, is anarchy, which seldom yields an optimal outcome.

ACKNOWLEDGEMENTS

We would like to thank Dr. Helmut Jurgensen of the University of Western Ontario Computer Science department, who granted Shane Dunne leave from another research project in order to write this paper. Thanks are also due to James Hong and David Wiseman of the same department, for their helpful suggestions and assistance in locating reference material. The references provided by J. A. Moorer of The Droid Works are also greatly appreciated.

This paper was typeset using the new device-independent *troff* program and other document-processing utilities under Berkeley 4.2 UNIX, and printed on a QMS ps-800 laser printer. We would like to express our whole-hearted appreciation to Jeff Lee of the U.W.O. Computer Science department, for his expert assistance in wrestling *troff* into submission, and for providing invaluable suggestions and references for the paper itself.

APPENDIX: TOWARDS A FIRST-LEVEL SPECIFICATION FOR ARCONET

The preceding paper was intended to propose that there be an ArcoNet standard, not to define the content of that standard, and hence no detailed technical recommendations were given in the paper. This appendix gives a set of specific suggestions for ArcoNet which, while nowhere near complete, should provide a starting point for practical discussion.

- *Subnet topology*: A ring should be used, because as is explained below, it is possible to automatically configure a ring without assigning "hard-wired" addresses to nodes. However, each node must include circuitry to bypass it if the power fails.

- **Internetworking:** Can be accomplished using a two-level hierarchy of rings. Each local network ring in a supernetwork will also be connected to a "global ring" for internetwork communication. That is, the master nodes on each local ring will be ordinary nodes on the global ring.
- **Access method:** token passing, since this approach yields strictly deterministic message delays.
- **Transmission medium:** CATV coaxial cable, because of its availability, high bandwidth, and simple termination.
- **Signalling mode:** baseband, for simplicity. Although baseband imposes length limitations on coax cable due to its frequency response, these are not serious for lengths up to about 200 feet, and can be circumvented by equalization circuits (Pickens (1981)).
- **Node processors:** single board design, based on the Western Digital WD2840 Ring Network Interface chip, the Hitachi HD64180 microprocessor with standard Zilog Z80 series peripheral chips, at least 64K of dynamic RAM and about 8K of EPROM. Standard nodes would be defined by the choice of I/O circuits, e.g. MIDI (4 or 8 IN/OUT connections per node) and RS-232 (4 or 8 per node), and digital headset (see below).
- **Computer interface boards:** Every ArcoNet setup will include at least one microcomputer for network control. Plug-in boards, based on the node processor design, should exist for IBM PC and other micros such as Amiga, Macintosh, etc., and for VME bus (Sun Workstation). To allow the network to operate even when the number of nodes is less than the token length, each micro interface card should also include an 8-bit shift register.
- **Network control hard/software:** time-critical operations should be supported by firmware on the interface card. Higher-level details will be supported by an application program, which will communicate with the card via DMA.
- **Data link protocol:** provided by WD2840 chip, similar to HDLC. Provides CRC for error detection.
- **Network level protocol:** Both point-to-point and broadcast addressing must be possible (using a special address bit pattern to mean "everyone").
- **Transport service:** raw, datagram, and virtual-circuit services are needed, as mentioned in the paper.
- **Digital headset:** at least one packet voice channel should always exist, supported by raw messaging, and duplicating normal broadcast "headset" communication. Digital headset nodes provide normal voice I/O, and pre-recorded voice messages (such as operator cues) can be put onto the headset channel by computers. Standard headset nodes might also include a numeric display for countdowns, and lights for general signalling.
- **Automatic configuration stage 1 (self test):** on power-up, the master node will send a broadcast "reset" message to all nodes. If this message does not come back around the ring, an error is reported. Then a token is generated, and each node in turn grabs the token, sends one message to the master, and then releases the token. The master counts the responses and reports an error if the count is wrong.
- **Automatic configuration stage 2 (address assignment):** the master sends a token around the ring again, and each node in turn grabs it and returns a number, starting at one and incrementing for each new node, then passes the token on. These numbers become the

node addresses.

- **Automatic configuration stage 3 (software distribution):** the master node distributes operating programs to each node as a series of ordinary datagrams. Each node will return a checksum of the program bytes, to ensure that its RAM memory is operational, and if this is approved, it runs the program.
- **During regular operation:** the master node periodically polls all others to ensure that they are working, and is responsible for detecting and restoring lost tokens.
- **Sequencing software:** will provide not just ordinary sequence playback but programming-language features based on message-passing paradigms. Sequences may include wait-for-cue and signal-cue primitives, conditional and looping operations, references to the current state of timer variables (e.g. how many seconds since cue #12 was received?), and asynchronous cue responses (like the *signal* primitive under UNIX).

REFERENCES

- ANSI (1982). *Videotex/Teletext Presentation-Level Protocol Syntax (North American PLPS)*. American National Standards Institute X3.2.1 Videotex/Teltext Character Sets Standing Task Group and CVCC/CSA Working Group on Videotex T500-X3L2.1. June 18, 1983.
- Byrd, D. and Yavelow, C. (1985). The Kurzweil 250 Digital Synthesizer. *Computer Music Journal* 10,1 (Spring 1986).
- CCITT (1983). *Draft Recommendations X.400-X.430: Message Handling Systems*. Technical Report, Comité Consultatif Internationale Télégraphique et Téléphonique. November, 1983.
- Day, J. and Zimmerman, H. (1983). The OSI Reference Model. *Proceedings of the IEEE* 71,12 (December 1983): 1334-1340.
- DEC (1980). *Ethernet Specification*. Digital Equipment Corp., Intel Corp. and Xerox Corp. September 1980.
- Haugdahl, J. (1984). Local-Area Networks for the IBM PC. *Byte* 9 (December 1984): 147-173.
- IBM (1985). *IBM Token-Ring Network*. Product Announcement, International Business Machines Corp. October 22, 1985.
- IMA (1985). *Musical Instrument Digital Interface detailed specification 1.0*. International MIDI Association.
- Lax, L. and Olsen, M. (1983). NAPLPS Standard Graphics and the Microcomputer. *Byte* 8,7 (July 1983).
- Loy, G. (1985). Musicians Make a Standard: the MIDI Phenomenon. *Computer Music Journal* 9,4 (Winter 1985): 8-26.
- Nelson, J. (1983). 802: A Progress Report. *Datamation* 29 (Spring 1983):136-152.
- Pickens, R. (1981). Wideband Transmission Media III: Guided Transmission: Wireline, Coaxial Cable, and Fiber Optics. In *Computer Communications, Volume I: Principles* Chow, W., ed. Prentice-Hall, Inc. Englewood Cliffs, N.J.
- Stone, H. (1982). *Microcomputer Interfacing*. Addison-Wesley, Reading, Mass. pp 215-227.
- Strole, N. (1986). Overview: How IBM addresses LAN requirements with the token ring. *Data Communications*, February 1986.
- Tanenbaum, A. (1981). *Computer Networks*. Prentice-Hall, Inc. Englewood Cliffs, N.J.
- Wood, D. (1985). Local Networks. In *Computer Communications, Volume II: Systems and Applications*. Chow, W., ed. Prentice-Hall, Inc. Englewood Cliffs, N.J.