

# UBIQUITY by **EAW**

**THE UBIQUITY NETWORK**

*No Rules. No Limits.*

*When your only tool is a hammer, all problems begin to resemble nails.*

- Abraham Maslow

### **Unique Benefits of the Ubiquity Network, from Loud Technologies, Inc.**

- Unlimited channel capacity
- No topology restrictions
- Powerful integration of control & audio
- Zero administration
- Self healing
- Telephone-style addressing
- No servers required
- Inherent congestion avoidance
- Seamless integration with other audio & control networks
- Simplicity for small networks, flexibility and redundancy for large networks
- Implementation cost aligned with product requirements

### **Have We Lost Our Minds? or, "Oh no! Not another audio network!"**

The existence of this document should make it obvious that Loud Technologies has entered the audio-system networking arena. Have we lost our collective minds? Are we not aware that it is already an overcrowded market and that one of the biggest challenges facing integrators is the proliferation of incompatible network protocols?

Of course we're aware of these issues. Trust us - we didn't set out to create "yet another network". We set out in search of innovative new solutions to long-standing challenges. One of the governing principles of EAW's approach has always been to let the challenges define the solutions. You won't find any Maslow's hammers in EAW's audio toolbox.

So what led us to create yet another network? The answer is a combination: Current networks have limitations that render some of our new ideas inconvenient, difficult, or downright impossible to implement; and the new concepts that arose from our analysis were just too cool to let lie.

The first part of this paper examines some of the inherent problems shared by existing networks. As you might guess, the second part lays out the defining details of our new network concept, which we have dubbed "Ubiquity", and shows how it avoids the pitfalls into which existing networks have fallen.

### **Requirements of an Audio/Video distribution and Control Network**

One of the central strategies of EAW's research and development activities over the last ten years has been to pioneer harnessing the power of DSP (digital signal processing). While DSP has been widely available in audio products for more than a decade, it's ability to fundamentally improve the performance of audio systems had been largely untapped. As we started contemplating network support for some of the DSP-based solutions we are planning, we were faced with the question, "Which network protocol should we adopt?" To make sure we covered all the bases, we considered numerous, current applications of networked audio, and brainstormed a large number of potentially beneficial techniques for both large and small systems. Then, we evaluated each of the existing networks, intent on identifying the one that would allow our ideas to be realized.

In addition to the broadly recognized attributes, such as “uses standard cabling” and “must have a high channel count”, we determined that the ideal network should also:

- be absolutely timing sure: on-time audio delivery must have the highest priority
- require very little or no administration
- make device identification and addressing as simple as possible
- work equally well in small, serverless, point-to-point systems and large, complex systems
- use generic, easily obtainable hardware, or not require any
- support daisy chain and star topologies - the less restrictions on topology the better
- carry both audio and control on one cable
- tunnel generic information traffic (like TCP-IP)
- support both broadcast and routed audio plus control
- have low, deterministic audio latency, even when routed
- offer simple-to-implement redundancy
- offer simple ways to overcome channel capacity limitations
- be economically viable in both simple single-channel and elaborate multi-channel devices

You’ve probably guessed that we found existing solutions sorely wanting, when measured against these criteria. The more we studied the implementation rules, design guidelines, and cost structure for each of the networks, the more obvious it became that we were looking at a collection of Maslow’s hammers. Each of them was an elegant attempt to make an information network do something it wasn’t intended to do.

Three of the principles underlying the design of information networks are:

- *Delivery must be guaranteed, but timing may vary*
- *Access must not be denied, but speed may vary*
- *Networks are administrated*

These “first principles” of information network design are fundamentally different than the first principles that should govern constant-bit-rate network design. Consequently, the purpose-designed behaviors of subsystems designed for use in information networks have to be suppressed or circumvented in a constant-bit-rate network. The result is a set of implementation rules that are not aligned with the requirements of audio systems.

When you focus on the challenges of the applications, rather than on the Maslow’s hammer in your hand, you end up with something completely different. You end up with Ubiquity.

## **Ubiquity Networking**

### *What is Ubiquity?*

Ubiquity is first, and foremost, a CBR (constant-bit-rate) network. Rather than allowing random access on a first come first serve basis, it passes constant bit rate streams at scheduled times. Control messages are only passed during the blocks of time allotted to information traffic or in unused CBR bandwidth. This prioritization of traffic is the essential feature missing from conventional information LANs. Ethernet™, for example, is “egalitarian”, which means that all devices and packets of information are normally treated with the same priority. It may be appealing to some in a socioeconomic sense, but it makes it very difficult to guarantee on-time CBR delivery in the presence of generic information traffic. That leaves implementers to ensure that only audio devices are connected to the audio network. This implementation rule is not necessary in a Ubiquity network.

Ubiquity is also efficient. Rather than passing information in Ethernet frames, it passes data in a format based on ATM (asynchronous transfer mode) cells. ATM is a ubiquitous technology used in many large-scale networks such as ISDN, ADSL, and in telecommunications.

*What are these frames and cells all about?*

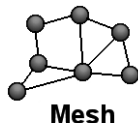
They're really nothing more than data structures. Low-level network services format the information they transfer in a very particular way. An Ethernet frame is a data structure with a 20-byte header containing addressing information, a description of the data format, and the number of bytes in its payload. An Ethernet payload can vary between 46 and 1500 bytes. Audio networks that are built on Ethernet pass 1500 bytes of audio per frame, in order to minimize the amount of bandwidth wasted on headers. 1500 bytes represents 10.4 milliseconds of 24-bit, 48 kHz audio, which is an unacceptable amount of latency. That's why Ethernet-based networks put anywhere from 8 to 64 different audio channels in each frame. As long as you're OK with sending all of those channels to the same place, this is an acceptable solution.

By contrast, ATM, which refers to its data packets as "cells", uses only a 5-byte header and a 48-byte payload. The reason it can get away with such a small header is that it is a connection-based protocol. Rather than individually addressing each packet of data, as Ethernet does, connections are established with both delivery schedules and channel routings determined ahead of time. While Ethernet is analogous to posting a letter, ATM is analogous to placing a phone call.

*"OK - so Ubiquity protects its CBR streams from its information traffic. Why do I need information traffic? I'll just build a second network for information. After all, I'm getting all the parts from Radio Shack."*

True - most large systems are currently implemented with separate audio and control networks. It is not a horrible compromise, but it can be done better. If the low-level network can carry information without risk to the integrity of the CBR streams, the high-level network can be much smarter. The much smarter network I'm referring to makes current networks look downright prehistoric. That brings us to:

## Ubiquity Mesh Networking



*"Mesh networking is a way to route data, voice and instructions between nodes. It allows for continuous connections and reconfiguration around blocked paths by "hopping" from node to node until a connection can be established. Mesh networks are self-healing: the network can still operate even when a node breaks down or a connection goes bad. As a result, a very reliable network is formed."*

-Wikipedia, The Free Encyclopedia, "Mesh Networking", <http://en.wikipedia.org>

The way Ubiquity mesh networking works is fascinating. Each individual node only keeps track of its immediate neighbors. When one node wants to transmit a message to another node, it broadcasts a "route request" which is passed around from node to node until it reaches the requested destination. The destination node then broadcasts a response to all of its neighbors, which may arrive back at the original node by multiple routes. Each node along the "route home" may receive the response from more than one neighbor, but information is appended to the message at each hop (like the hop count). This allows each node to determine and save only the "best next hop" from itself to that particular destination.

Later, when the message is actually sent, it is passed from node to node via each node's "best next hop", arriving at the destination by the best route possible. The interesting part is that no individual node knows the entire route between points. The nodes form a sort of "organism" that passes information in the most efficient way possible, even though no individual node is aware of anything beyond its immediate neighbors. This process, called "route discovery", can be very fast and the network can adapt to changes very quickly.

As connections come and go, the network continually rebuilds its routes. This is how wireless ad-hoc networks keep working, even though individual wireless nodes continually join and leave the network.

*“OK - it’s a cool concept. But what does it mean for an audio network?”*

### **Benefits of The Mesh**

Ethernet must be wired in a star topology. Well, that’s not exactly true, because a series of 1-in/2-out switches can make a star network look like a ring. Nevertheless, the essence of the topology restriction on conventional network wiring is that there can be only one connection between any two points in a subnet. A mesh network does not have this restriction. This is a nicety for those situations where the physical layout of a system doesn’t fit naturally into a star topology. However, its real power lies in the useful things we can do with those extra paths between points.

#### *Unlimited Capacity*

In a star network, the maximum number of channels that can be passed between any two points is limited to the carrying capacity of one link (remember, there can only be one route between any two points in a subnet). Not so in a mesh network. If the capacity of one link is insufficient, just add a second link. The mesh routers at either end will automatically use the added capacity - automatically! There is no programming required, no subnets to define, no routers to program. Just add a cable and the capacity increases. Still not enough capacity? Add another cable. Are the routers at either end completely plugged up? Just add more routers. Connect the routers together any way you like. The mesh will figure out how to make it work. Simple.

#### *Self Healing*

Route discovery is an inherent capability of the mesh. When the system is first started up, the routes are initialized, and audio and control messages start being passed. If part of the network goes off line for any reason, the mesh will automatically seek new routes around the dead part of the network. Whether the dead part is a cut cable, an unplugged router, or a failed component, the mesh will “self heal”. As long as there is sufficient capacity in the rest of the network, there will be no long-term interruption in service. This means designing redundancy into a system is a simple matter of providing multiple paths. Simple.

#### *Telephone-Style Addressing*

Because the mesh chooses the links that carry information between points, routing of audio and control is specified by source and destination. When you make a phone call, you don’t select the particular phone lines you want your call to use - you dial a telephone number. That’s how mesh networking works. Each audio source and destination is identified by a device’s unique ID number and a channel identifier. The infrastructure of devices and wiring can be changed and the mesh figures out which links to use to make the connections. Simple.

#### *Zero Administration*

You may have noticed that nowhere in this description did we say, “The administrator can”, or the “The administrator should.” There is no administrator. There is only an architect. The decisions that are made as to how many links are required and where to provide cross-links to provide sufficiently granular redundancy are design-time decisions. Upon implementation and after changes are made, the mesh handles the drudgery of router programming and channel assignment. In simple systems, there aren’t even any rules. As long as every device in the system is connected in some way to the mesh, everything works. Simple.

## Communicating With the non-Ubiquity World

To preserve the integrity of CBR timing, the Ubiquity network can only be accessed directly by Ubiquity-aware devices. However, other networks can use a Ubiquity infrastructure, indirectly, by way of edge servers. It will be possible to transparently tunnel conventional Ethernet packets from one edge server to another. It will also be easy to interface Ubiquity audio to other audio networks. Unfortunately, the benefits of mesh networking must necessarily stop at the edge of the Ubiquity network. Other types of networks must still be administrated in their usual manner.

## Implementation Costs

One of the key factors that has limited the widespread adoption of networked audio technologies has been poor alignment between cost and functionality. Per-node licensing arrangements and algorithmic complexity (largely to subvert the natural behaviors of Ethernet) have made network audio connections relatively expensive for low-cost devices, such as individual loudspeakers. Ubiquity permits a very simple, inexpensive interface for terminus devices, such as loudspeakers. More elaborate functionality, such as mesh routing, can be provided by products with spare processing power or a cost profile that allows the addition of the required technology.

As for the supposed advantages that can be gained by borrowing parts from general purpose information networking, the most-touted benefit is the ability of Ethernet-based audio networks to use inexpensive, mass-produced switches. By now, the astute reader will have realized that mesh networks don't require switches. Zero is less than inexpensive.

## When Does All of This Become Real?

In March 2006, the first implementation of Ubiquity was demonstrated in the NTL720 and NTC series of products. This level of implementation, referred to as Ubiquity Level 2.5, realizes the physical, datalink, and adaptation layers of the Ubiquity protocol (see Appendix A for an explanation of network layers). Future versions will implement additional adaptations, as well as the fully realized routing protocols. This will be known as Ubiquity Level 3. The timing for the release of future realizations of Ubiquity will be dictated by the release of products that make effective use of the more advanced features.

## Summary

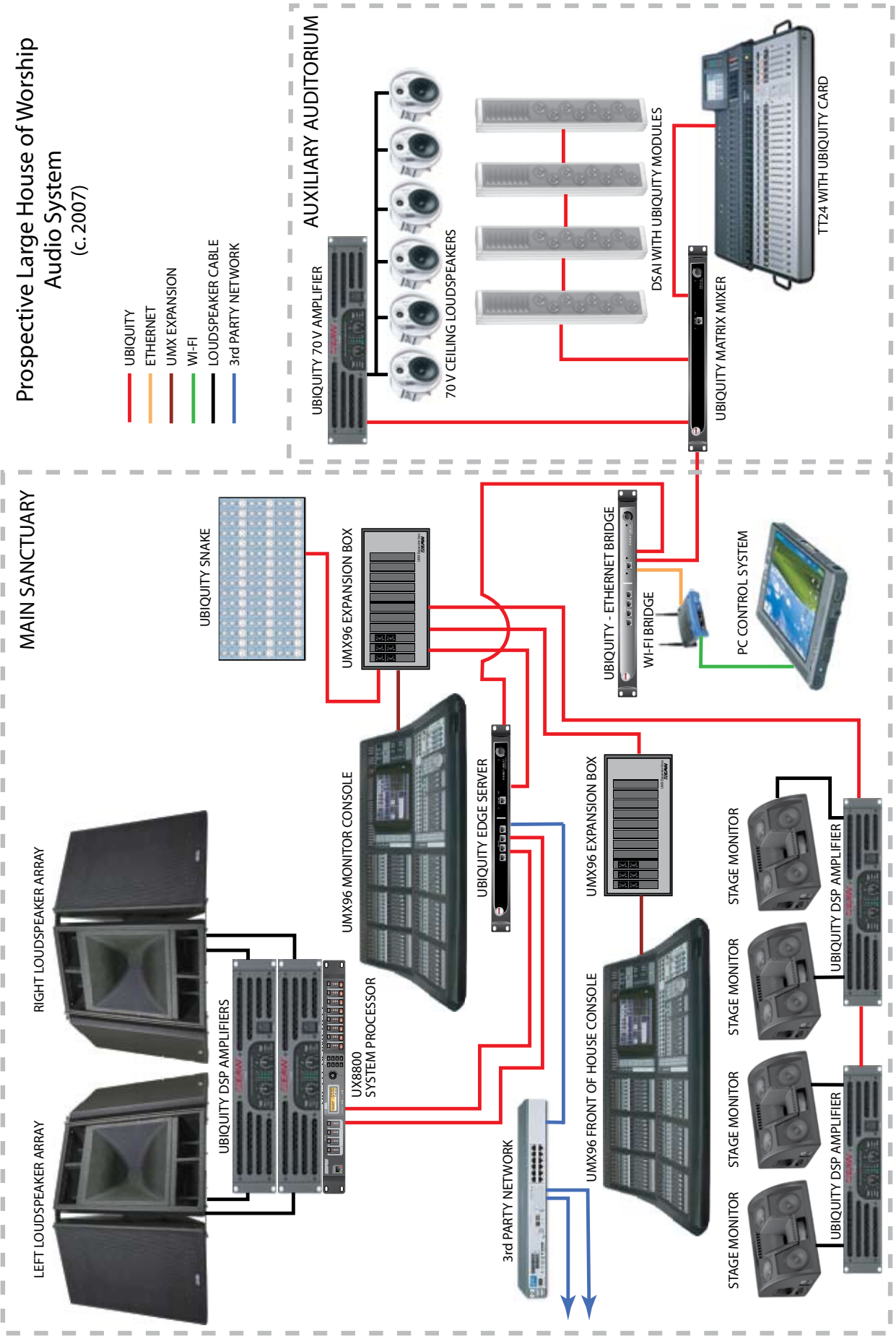
The unique benefits of a Ubiquity network provide the best summary:

- Unlimited channel capacity
- No topology restrictions
- Powerful integration of control & audio
- Zero administration
- Self healing
- Telephone-style addressing
- No servers required
- Inherent congestion avoidance
- Seamless integration with other audio & control networks
- Simplicity for small networks, flexibility and redundancy for large networks
- Implementation cost aligned with product requirements

It's simple.

# Prospective Large House of Worship Audio System (c. 2007)

- UBIQUITY
- ETHERNET
- UMX EXPANSION
- WI-FI
- LOUDSPEAKER CABLE
- 3rd PARTY NETWORK



## Appendix A: The OSI Seven-Layer Model

The OSI Seven-Layer model is a framework for describing responsibility for the various functional requirements of a network. Those who are familiar with this framework may wonder how Ubiquity's components fit into the framework.

The Seven-Layer model looks like this:

LAYER	FUNCTION	DESCRIPTION
7	Application	Do useful things with information
6	Presentaion	Data syntax (html, MIME, etc)
5	Session	Manage connections between different computers
4	Transport	Detect packet loss and retransmit (guarantee delivery)
3	Network	Addressing & routing
2	Data Link	Data packet format
1	Physical	Electrical systems, wires, etc.

For the current discussion, layers 4 through 7 can all be rolled up into one - we'll call it the "Extended Application" layer. In the ATM world, it is common to identify a layer between 2 & 3. This layer gives ATM the required flexibility, but without the efficiency penalty of more complicated data link layers, such as Ethernet frames.

The Ubiquity model looks like this:

LAYER	FUNCTION	DESCRIPTION
4 - 7	Extended Application	Audio/Control senders/recievers, generic networks
3	Network	Ubiquity Mesh Routing
2.5	Adaptation	Ubiquity ATM Adaptations (sample rate, bit depth)
2	Data Link	ATM Cells
1	Physical	Standard, inexpensive, Ethernet electronic parts



**UBIQUITY NETWORK** from Loud Technologies, Inc.

[WWW.EAW.COM](http://WWW.EAW.COM)

One Main Street, Whitinsville, MA 01588 800 992 5013 / 508 234 6158