Synchronous Adaptive INfrastructure (SAIN) Fundamentals

Ray W Sanders, Chairman, CircuitPath Network Systems Corporation

Abstract

This paper is a tutorial describing the foundations of Synchronous Adaptive INfrastructure (SAIN) technology. The goal of SAIN is to enable building both small and large networks using a single paradigm that separates bandwidth-on-demand connectivity from higher-level protocols. This approach greatly simplifies network design and operation and provides protocol-agnostic connectivity to and from any current or future protocol — user or network. Its initial uses are in Broadband Access Networks. However, the approach enables building Metropolitan Area and Wide Area Networks that can interface with legacy IP, ATM and PSTN networks without requiring modification of such networks.

Introduction

Connectivity is the fundamental aspect of communications – in other words, establishing network paths among terminal devices to support information flows.

Connectivity depends on systems that support:

- unicasting (one-to-one terminal connectivity),
- multicasting (one-to-many connectivity) and
- conferencing (many-to-many connectivity).

Affordable networks must include ways of using high-speed links interconnecting network nodes to carry a number of connections. This means that the outputs from information sources must be connected through devices that *multiplex* the use of the links. Of the known methods of doing this, dividing use of a link by time, where specific periods are dedicated successively to a single connection's information flow, provide the greatest utility.

Two methods have been developed to do this. One is Packet Multiplexing (PM) that divides use of a high-speed link into variable length time slots. The other is Time Division Multiplexing (TDM) where fixed period time slots are used. PM (and its companion, Packet Switching (PS)) has been the mainstay of data networking for the past 30 years. TDM (and its companion Time Division Switching (TDS)) has supported voice networking over the same period.

Each technology (PM and TDM) focused on the unique market requirements for which its use was designed, leading to the current dichotomy between voice and data networking. PS is difficult to apply to voice applications because of the variable delay it engenders. Time Division Switching is too restrictive in its ability to channelize high-speed links to match the characteristics of data traffic.

The world wants a "converged" network — a single network that efficiently and reliably handles all voice, video and data traffic in a common fabric. Today's main efforts are to use PS and Cell Switching, which is essentially PS using fixed length packets, called *cells*. Obtaining convergence using packet-based switching has proved a difficult and complex "work in progress".

The focus of this paper is on a simple alternative. It bases network connectivity on TDM, but with a major difference from past practice focused on voice applications. The difference is a new multiplexing paradigm that overcomes the deficiencies of past approaches. It allows connections to be *established* in milliseconds, and allows a connection's data rate to be *changed* in milliseconds, an approach that has heretofore been considered too difficult to be practical.

This new technology, called Synchronous Adaptive INfrastructure (SAIN) performs these tasks in a very simple manner. It allows networks to be built that separate *network connectivity* from *network interfaces*. Flexible connectivity is supported by a simple paradigm. The connectivity supports interfaces to all existing and future user and network communication protocols, whether voice, video or data. SAIN connectivity is agnostic to *any* interface protocol.

All end-to-end SAIN connections emulate simple clocked modem pairs connecting sources to destinations.

This is a new approach to building modern networks. It profoundly simplifies the way networks are built and managed while, at the same time, providing better performance and increased reliability compared to existing approaches.

A number of companion papers discuss the application aspects of SAIN. This paper focuses on the basics of SAIN technology.

Multiplexing

A multiplexing system permits the sharing of communication media (typically serial high speed links) among a number of data flows. Data flows may comprise voice, video or data information, or combinations thereof.

For each source to destination flow, the interconnecting transmission media can be made to appear exclusively dedicated to the specific flow. For example, if there are three flows labeled *A*, *B*, and *C* between a source and destination, a transmission link is devoted exclusively to each data flow for short periods of time. A segment from data flow *A* might be followed by one from *B*, followed by one from *C* and followed again by one from *A*, and so on.

Each interleaved flow is carried over a *connection* embedded in a communication link. A connection is not physically hard-wired. Rather, it is a partition of the overall transmission capacity of a link that *appears* to be a (temporary) physical modem pair connection. The aggregate of all connections (including, possibly, one or more additional empty spare capacity connections) comprises the total capacity of the high-speed link.

In this manner, each link carries data generated by one or more information sources destined to one or more data sinks. A multiplexed stream of information is a sequence of individual units of data, each uniquely associated with its respective data source(s) and sink(s), and sent over transmission media in an interleaved fashion.

For purposes of this paper, a smallest data unit will be taken to be one or more information bits. Each individual data unit carried over the link must be associated with a specific connection. Typically this is accomplished by dividing the information transmission capacity of a communications medium into segments assigned to the individual data units. There must always be a means for uniquely identifying each segment with a specific connection between source(s) of data and to see to it that only data units from such source(s) are embedded within those segments so identified.

There are three ways of defining data segments. One is a *packet*; the second is a *cell*. A packet is a collection of bits that are sent as a group. The number of bits in a packet may or may not be fixed. If not fixed, packet lengths are usually set within a range dictated by a network standard.

A cell is also a collection of bits sent as a group. However, a cell is of fixed length that conforms to some network standard. Each time division segment containing a cell comprises the smallest element of transmission capacity assignable to a connection. Asynchronous Transmission Mode (ATM), for example, uses globally fixed length cells.

A third way of defining segments is to name them *cellets*. A cellet's size is fixed for a specific link. A stream of cellets is divided into frames of one or more cellets. Cellets are the basis of the SAIN architecture that is defined more completely in later in this paper.

Explicit and Implicit Addressing

In all digital switching systems, each information segment produced by one or more sources is embedded into a uniquely configured stream of segments. Associated with each segment is a connection identifier called a *connection address* that identifies to the link's receiving end the specific connection assigned to the segment. Connection addresses can be either *explicit* or *implicit*.

An *explicit address* is one where the connection identifier physically comprises part of the packet or cell with which it is associated. These Protocol Data Unit (PDU) packets have explicit address headers.[1]

An *implicit address* is one where the connection identifier associated with a TDM segment is defined by its unique position embedded within a frame. Each multiplexing termination at the source and receiving ends of a time division multiplexed *link* is constrained to use a predefined convention. This paper focuses on cellet switching (CS).

BENEFITS OF ATDM NETWORKS

Figure 1 a) and b) are illustrations of the two types of addressing noted above. Figure 1 a) shows an explicit addressing example while Figure 1 b) shows an implicit case. In the Figure 1 a), the contents of each cellet is composed of two parts, an Address Portion, such as α_A, and a Payload Portion, such as **A**. In Figure 1 b), each cellet contains only a payload portion. As noted above, the connection to which a particular cellet is assigned is implicitly connoted by the cellet's location in time (i.e. its physical location within a frame).

ASYNCHRONOUS & SYNCHRONOUS TDM

Explicit addressing based systems on are called *Asynchronous TDM* (ATDM) systems. They are asynchronous since segments in ATDM systems have their connection addresses attached as a part of Protocol Data Units that are not position referenced in time for purposes of identifying the connection to which the cellets belong.

Those systems based on implicit addressing are called *Synchronous TDM* (STDM) systems. They are synchronous so that the sending and receiving multiplexers have the same frame of reference for identifying the connection to which each cellet belongs based on its unique position in time[2].

In certain contexts, it is an advantage that explicit addressing ATDM systems can send PDUs at any time and therefore in any order, while in STDM systems, cells must be sent synchronously (i.e., coherently) in a predetermined order. Arbitrary ordering of ATDM mandates queuing of PDUs to smooth out data flows. Queuing and related topics have been the subject of much research and many system implementations over the past 30 years. The history of these efforts has been well chronicled in a paper by A.G. Fraser, "Early Experiments with Asynchronous Time Division Networks", IEEE Network, Vol. 7, no. 1, pp. 12-26, (January 1993).

ATDM systems have been developed primarily for data communications. The asynchronous character of ATDM is well suited to deal with the so-called *bursty* nature of data transmissions. Typically in data communication applications, the need for sending information occurs sporadically, in bursts. When a particular source wants to transmit information, it is often desirable to send it quickly. Thus, there is need for adequate communication connection transmission capacity that can be used on demand.

STATISTICAL MULTIPLEXING

In conventional STDM circuit switching systems such as the PSTN, link capacity is allocated up front for each connection through establishment of a physical circuit; a connection handling bursty data is therefore idle between bursts. This results in inefficient utilization of overall link transmission capacity.

The desire to achieve more efficient use of link capacity in the presence of bursty data has led to the notion that a high capacity communication link can be shared among data communication sources (and thus connections) on a statistical basis — in other words, by using *statistical multiplexing*.

This idea rests on the assumption that the probability is small that more than a limited number of communication sources will have data to send at the same time. The theory is that, even if the sum of the instantaneous transmission rates from the sources exceeds the capacity of the communication link, the probability is high that there will be enough link

[1] In some situations, the word "synchronous" is a misnomer. It is not always necessary to synchronously clock a link. What is necessary is that data be coherently timed. This means that at the destination of a transmission, the order of each cellet must be known with respect to all others within a frame.

[2]

capacity to handle the average load at any given time. This will be true *if* the average to be sent from the sources is *really* less than link capacity.

ATDM techniques have become the *de facto* method of designing and building high speed multiplexing and switching systems. Devices currently in the market include statistical multiplexers and packet switches, designed for data traffic. Asynchronous Transfer Mode (ATM) multiplexers and switches are being implemented as a part of Broadband Integrated Services Digital Networks (B-ISDN) as well as local network access multiplexers and switches.

VIRTUAL CONNECTIONS

A *connection* is an association over network links between one or more signal sources and one or more signal sinks involved in a specific data flow via communications media. A *virtual connection* is a circuit that is established logically within a network, but is not physically connected until a sending terminal presents information to transmit.

An advantage of ATDM technology is its ability to support virtual connections. A virtual connection requires that establishment of a connection be performed only once. Although link transmission capacity has not yet been physically allocated, the logistics of establishing such a connection have already been put in place so that the physical connections necessary to permit transmission over the link can be made quickly when necessary.

This process establishes a physical route over one or more tandem links that does not waste network capacity when not needed and which maintains a high probability that sufficient capacity will exist when information must be transmitted.

DISADVANTAGES OF ATDM NETWORKS

Although there has been great emphasis on ATDM technology, it possesses a number of significant disadvantages:

"HEAD TAX"

As can be seen from Figure 1a, explicit addressing demands that some of the communication link capacity be devoted to addressing, resulting in wasted capacity that cannot be allocated to information transmission.

Because of required addressing overhead, a tradeoff must be made between the portion of available capacity devoted to address information versus payload information. It is desirable to maximize the number of payload bits per PDU compared to header bits, i.e., to send large payloads per PDU.

Because of required addressing overhead, a tradeoff must be made between the portion of available capacity devoted to address information versus payload information. It is desirable to maximize the number of payload bits per PDU compared to header bits, i.e., to send large payloads per PDU.

Unless the PDU size is fixed, additional overhead is required to designate the length of each PDU (as is done in most PS systems such as Frame Relay networks). This is one reason that newer cell-switching approaches, such as ATM make use of fixed cell sizes.

ACCESS BUFFER-FILL TIME DELAY

If an information source must deliver information to a network through a small bandwidth connection, payload sizes must be kept small to keep network access delay low. One of the drawbacks of limiting PDU payloads to high values is the amount of time it takes to fill a PDU buffer with data from a low speed source. Until the entire PDU is available for transmission, it cannot be sent over a high-speed packet network. There are three options available to address this problem:

- Suffer the bandwidth penalty caused by a large headers to keep PDU payloads small.
- Use fixed cell sizes with empty pad bits.
- Require time multiplexing of multiple connections within a PDU.
- Use header compression with or without TDM.

The third alternative may be comparable in some instances to the methods described in this paper. However, using these methods in recommended ways completely avoids PDU overheads and the other problems of ATDM networks.

LIMITATIONS OF STATISTICAL MULTIPLEXING

A simple Markoff queuing model shows that In practical ATDM applications, delays for individual users can become large unless the average traffic load is substantially less than the total link capacity. Classical queuing theory shows that the average delay for PDUs approaches infinity as the average offered load approaches 100% of a link's capacity[3].

The average amount of traffic must stay far below full utilization of a link's capacity to assure low queuing delay. However, recent studies show the real world is even more exacting. The simple model is overly optimistic. Studies of real-world Internet traffic show that traffic arrivals behave in a fractal manner rather than as a pure Markoff process. This behavior defines traffic as being *self similar*, a topic that is now the subject of intense study.

There is now consensus that statistical multiplexing based on PDU multiplexing and switching is less useful in maximizing utilization of link capacity than conventional wisdom believes.

EFFECTS OF COMMUNICATION ERRORS

If a communication error occurs during transmission of the address portion of a PDU, a switching node cannot deliver the payload to the proper destination. To help overcome this problem, a certain amount of the transmission capacity is also devoted to error detection; the result is a further increase in overhead. (For example, ATM has standardized on a cell size of 53 bytes (424 bits) of which 48 bytes (384 bits) are payload data, 4 bytes (32 bits) are address information, and 1 byte (8 bits) is devoted to error detection).

[3] An approximate model for expected delay is based on assuming that arrival rates of packets behave as a Markoff process with a Poisson distribution. In other words, arrival rates follow an exponential distribution.

Link error rate can have a profound effect on ATDM systems. In data applications, an error in transmitting a segment of a PDU from a source to a sink can be detected by the application and a request for retransmission can be sent to the source. This approach is tolerable because data transfers are usually not extremely time dependent. For most non-data applications, however, retransmission schemes can insert delays that can be intolerably large and variable. An error in sending a cell's header is particularly disastrous because switching nodes cannot determine the destination of the cell. The entire cell must be discarded. This characteristic of cell-based systems necessitates very low error rate transmission facilities for delay-sensitive applications such as voice and particularly video.

DELAY CAUSED BY INTERNAL NETWORK BUFFER STORAGE

It is desirable to keep the overall PDU size small because larger PDU sizes increase buffering requirements at switching nodes. Buffer storage results in transport delay through a switch node that can seriously degrade communication service where total communication link capacity is limited. For example, if ATM cells are sent to support a DS0 telephone connection and if no queuing is required at tandem nodes, a minimum (payload) connection speed of 3.072 Mbps would be required to meet the same transit delay as occurs in a PSTN network for a DS0 (64,000 bps) connection. ATM has been designed for communication facilities of the order of 45 megabits per second (Mbps) or more. Large delays do not then occur. Over lower speed link s, the delays can limit interactive communication.

NETWORK COMPLEXITY REQUIRED TO SEND ISOCHRONOUS DATA AND OTHER INFORMATION FLOWS

In an integrated network that is devoted to voice and video communication along with data, data traffic may overtake voice over the next few years. Over the longer term, however, video will likely predominate over both voice and data in many networks.

Voice and video services are isochronous in nature; they send information at fixed (or slowly varying) clocking rates. Information flows endure for long periods of time (tens of milliseconds to many hours).

These are not the much-studied "bursty" information sources on which most ATDM network theory has been based. In fact, it is now evident that most data sources (as well as voice and video) produce information flows that persist over periods that encompass sending a multiplicity of PDUs.

This artifact is becoming more prevalent as time goes by. Upwards of 75% of the data traffic on the Internet is now World Wide Web based, where typical Web pages include tens of kilobytes of information. With the increased use of the Internet for voice and video, it is reasonable to conclude that except for UDP control PDUs (which require only a small amount of bandwidth), essentially all Internet traffic will be flow based.

COMPLEXITY OF PROVIDING QUALITY OF SERVICE ASSURANCE

The requirements for managing flows depart markedly from the past emphasis of merely delivering PDUs. By its very definition, a flow relates to the number of bits being transmitted per second. In other words, for a given Quality of Service, the flow related to a particular application requires that the flow (a given number of bits) be delivered from a source to a destination within a given number of seconds. The single most important parameter for managing Quality of Service of a flow is to manage its connection's bandwidth.

Using ATDM networks means changing significantly the nature of handling PDUs from its simple beginnings for data only networks. Additional protocols must be invented that adapt historical ATDM networks into robust managed bandwidth networks.

This is no easy task. The problem is to overlay complex protocols on top of an Asynchronous Time Division Multiplexed network so that it behaves like a Synchronous Time Division Multiplexed network.

Network flows within ATDM networks are implemented by managing access and internal buffers. PDUs must be prioritized, which increases control complexity. There is no direct paradigm to change traffic prioritization into bandwidth management.

To accommodate isochronous connections, ATDM networks require complex clocking subsystems. ATM has developed a set of ATM Adaptation Layer protocols to deal with these and other issues. However, they are complex when applied in the manner initially conceived (adapting ATM to all network protocols). In fact, ATM is finding its largest market in backbone networks because of its ability to manage bandwidth between network nodes.

This use of ATM highlights a significant need within operational networks. A fundamental issue is managing bandwidth at a fine enough granularity to meet Quality of Service objectives for individual connections.

Within the Internet community, intensive work has been performed for years to provide a standard to meet Quality of Service needs. There is progress being made, but, as yet, there is no set of agreed-to standards that have been released even though individual vendors have implemented their own (preliminary) versions. To get to a universally acceptable standard is a daunting task that could take many years to reach, if ever.

NEED FOR POLICING AND TRAFFIC SHAPING

More complexity must be implemented within ATDM networks in order to deal with their asynchronous nature. One of the requirements for service providers is to be able to specify various service levels that they guarantee to their subscribers. These often include maximum information rates for which the service provider guarantees delivery. Using an ATDM network, a service provider must monitor the rate at which PDUs are sent to a network to be sure that the user is not exceeding the guaranteed maximum. If he does, excess PDUs must be marked for discard. This artifact increases complexity for user as well as the service provider. (In ATM networks, this capability is embedded within User Parameter Control functionality.)

Another guarantee is delay variation of PDUs delivered at the distant end of a connection. Buffering required within

ATDM networks to smooth out traffic bursts can cause PDUs belonging to a given flow to bunch up within buffers. This requires implementation of *traffic shaping* to clock out buffered PDUs in a more uniform manner than occurs within basic ATDM networks.

COMPLEXITY OF PROVIDING MULTICASTING AND CONFERENCING SERVICES

Multicasting and conferencing can be accomplished by replicating PDUs or by defining "group addresses" that are interpreted at each node to define one or more destinations. Both approaches require complex manipulation of connection addresses. In conventional voice and data communication applications, multicasting and conferencing are fringe services. For video and multimedia applications, however, the ability to support multicasting becomes a much more important network requirement.

NETWORK RELIABILITY ISSUES

Connectionless networks such as the Internet provide for data delivery with high probability in the event of catastrophic failures of a single node or link. On the other hand, the main reliability issues of service provider based networks do not require connectionless architectures to overcome failures. The Public Switched Telephone Network (PSTN) is far more reliable than the Internet, and it is connection-oriented.

The main reliability concerns of ATDM networks relate to the requirement that these networks rely on complex software that is procured from multiple vendors for reliable operation. Guaranteeing error-free interoperability among networks is a daunting challenge.

COMPLEXITY OF MANAGING NETWORKS AND DIFFICULTY OF TRAINING OPERATIONAL PERSONNEL

Even if standards to provide Quality of Service capabilities within ATDM networks eventually emerge, the complexity of building them will make management and further extensions of networks difficult. We, as an industry, are trying to adapt an underlying paradigm, Asynchronous Multiplexing, to solve multiplexing problems for which it was never intended. Design challenges will remain intimidating to many.

For a service provider, training personnel to achieve sufficient understanding and skill becomes difficult and expensive. Operational costs are now based mostly on the number of people required to operate a network rather than equipment cost. The more complex the implementations, the more difficult the task becomes, particularly as networks must be scaled to higher data rates and in number of different services offered.

BENEFITS OF TODAY'S STDM NETWORKS

As the discussion of the last section makes clear, the disadvantages of ATDM networks almost all derive from the fundamental architectural principles on which the networks are based — using asynchronous structures that depend on independently clocked buffers for multiplexing and switching.

When looking at the future of networking, it is clear that clocked network structures could overcome many of today's difficulties. But, the obvious question is, “Why aren't we using STDM in networks other than the Public Switched Telephone Network?”

DISADVANTAGES OF TODAY'S STDM NETWORKS

The answer to the question above is simple. Today's STDM networks fail to meet the challenge of providing the diversity of data rates and flexibility of PDU-based networks.

INFLEXIBLE CHOICES OF CONNECTION AND LINK SPEEDS

Connection and link speeds of the only large scale STDM network (the PSTN) are constrained to conform to digital time multiplexing hierarchy standards that established by national and international standards organizations. Even though the standards facilitate interworking between equipment of different manufacturers and between different countries, the number of speeds is limited and they are extremely inflexible during system operation.

LONG CONNECTION TIMES

Current STDM switching systems have complex signaling systems that require large amounts of information to be transmitted to establish connections. Although new connection routing strategies have been implemented, the time to set up a connection is long compared to that required for data-centric connectionless networks.

INFLEXIBLE MULTIPLEXING

Almost all STDM switching systems are constrained to minimum data rates of 64,000 bits per second (bps) and to, at most, integer multiples thereof. There is a single standard frame rate (8,000 per second) throughout the PSTN. While this is a good choice for its intended purpose, multiplexing and switching 64,000 bps voice connections, it is not a good choice for other purposes such as switching data, video and low-speed (compressed) voice connections.

Nothing is fundamentally compelling about using 125 μsec frame periods. It is just the way it's been done. STDM is inflexible, not because it need be, but rather, that is the way standards have emerged.

VIRTUAL CONNECTIONS NOT SUPPORTED

Providing virtual switched connections is not practical with existing architectures. At least, virtual connections with the flexibility that exists within ATDM networks cannot be supported. Again, this is the result of standards bodies focusing on voice switching needs, and not those of a larger context.

LIMITED MULTICASTING AND CONFERENCE SUPPORT

Providing multicasting and conferencing are not inherently a part of STDM switching systems. They are add-on modules and subsystems that must be provisioned separately.

What's needed?

It is clear that both ATDM and STDM architectures have capabilities that neither can provide alone. But, how can we implement the best of both?

A plausible answer would exist if only we could overcome the current deficiencies of STDM that are its limited choice of available standard bandwidths, its inability to set up connections quickly and its inability to change a connection's bandwidth once the connection is set up.

CHANGING THE STDM PARADIGM

Suppose we were able to do the following three things:

1. support any bandwidth,
2. set up connections in milliseconds, and
3. change bandwidth of connections already established in milliseconds.

If this is possible, it represents a new paradigm building networks.

Changing the STDM paradigm is the focus of SAIN. By doing do, designers can focus on the very best attributes of both ATDM and STDM in a combined network architecture.

CHOOSING THE BEST OF ATDM AND STDM

It is necessary to look at specific network architectures and protocols to choose the most important attributes. The two most important ATDM architectures are IP and ATM. There is currently no STDM architecture that can be pointed to as "most important". This paper suggests that SAIN has the potential of gaining that distinction. The following discussion focuses on this potential.

THE BEST OF IP

Arguably, the most important aspect of IP is its universal addressing capability. It is rapidly becoming the *lingua franca* of communications for all voice, video or data applications. Another key benefit is the ability of IP to send and deliver variable length packets, an attribute that fits with the von Neuman computer architecture standard. In addition, IP provides a basis for dividing large networks into domains and subnets that, while logically separate, interoperate to become the global Internet.

THE BEST OF ATM

The most important attribute of ATM is its connection-oriented architecture that defines both channels and paths and the categorization of traffic types (Constant Bit Rate [CBR], Variable Bit Rate [VBR], Available Bit Rate [ABR] and Undefined Bit Rate [UBR]).

THE BEST OF TODAY'S ATDM AND STDM NETWORKS

Arguably, the most important features implemented in modern networks are the ability of implementing *Virtual Private Networks* (VPNs).

How can SAIN exploit these key IP and ATM attributes?

The answer is by focusing on a new network architecture that separates connectivity from higher level user and network protocols. This greatly simplifies network architectures since high level protocols like IP, ATM and a host of others (802.3, 802.11, DOCSIS, IPX, SNA, Frame Relay, etc.) need to be implemented *only* at the edge of a network.

SAIN's role is to provide bandwidth-on-demand connectivity at the physical network layer plus a signaling layer that allows near-instantaneous management of connections and their bandwidth.

SAIN leads to the disappearance of the complexities of asynchronous switching internal to a network.

SAIN enable a new network architecture paradigm by supplying the missing ingredients that have not yet been a part of STDM networks.

By supplying the missing elements of current STDM connectivity networks, a new architecture emerges that greatly simplifies network design and operation.

No longer need we face the prospect of requiring continual upgrades of network routers and switches to keep up with the latest versions of various complex protocols intended to overcome the limitations asynchronous multiplexing and switching impose.

To implement a large SAIN-based network requires substantial effort. However, the basics of the approach apply uniquely to broadband access networks (the on-ramps of the Internet), the current bottlenecks to enjoying fully the Internet potential. Implementation of distribution networks using SAIN technology is quite straight-forward.

Combining the best of both ATDM and STDM in a common network fabric is the mission of developing Synchronous Adaptive INfrastructure (SAIN) technology.

The rest of this paper focuses on its technical underpinnings.

Brief description of SAIN systems

Objectives of SAIN technology

ENABLING COMBINING THE BEST OF ATDM AND STDM

An objective of Synchronous Adaptive INfrastructure (SAIN) technology is to provide the advantages of both ATDM and STDM and concomitantly to overcome their separate disadvantages. SAIN technology implements a new method of assigning cellets to form embedded connections within communications media. The method is based on Time Division Multiplex using implicit addressing.

ENABLING VIRTUAL CONNECTIONS IN AN STDM ENVIRONMENT

A SAIIN has as one of its advantages the ability to support the virtual connections (heretofore an advantage associated almost exclusively with ATDM systems) to better handle the vagaries of bursty data, but it also provides the advantage of implicit addressing (and its low overhead) heretofore associated only with conventional STDM systems.

FAST CONNECTION SETUP AND CAPACITY CHANGES

A further objective of SAIN technology is to provide a methodology that can reassign cellets to embedded connections in an automated fashion in real time. The process responds to system requests both to add or delete connections and to increase or decrease capacity for established connections.

ATDM systems have been favored because they are able to respond in real-time to changes in demand most often associated with bursty data sources for link transmission capacity. More efficient use of link capacity results than obtains from assignment of a full-period connection. However, this advantage is still much less than can be achieved by implementing bandwidth-on-demand scheduling at the physical network level.

STDM systems require very little or no capacity expenditures to implement implicit addressing. As currently implemented, however, these highly standardized approaches are inflexible and ill-suited for responding in real-time to changes in demand for connection capacity.

As with current STDM systems, SAIN systems form embedded connections within a link by assigning a subset of all cellets within a Time Division Multiplex frame. Unlike current STDM systems, however, SAIN implements methods of assigning cellets to connections that permits their reassignment in real time based on increasing or decreasing demand for connection capacity.

TDM as a transform process

In the past, TDM has been implemented as *hierarchical processes* where a high bandwidth link is divided into a number of smaller bandwidth connection s using cellets spaced throughout a fixed-length frame. Further subdivisions of connection s are often made to achieve a large number of small bandwidth connections. Also, these small bandwidth connections usually become the quantum that must be handled within switches.

SAIN overcomes the rigidity of hierarchical derivation of connections by implementing TDM as a *transform process*. The chosen transform for SAIN's first embodiments is a *binary transform* illustrated in Figure 2.

EXAMPLE OF HOW BINARY TRANSFORMS WORK

Figure 2 shows a table that has 32 rows divided into two parts. Each row corresponds to a cellet in an integer power-of-two-cellet length Time Division Multiplex frame. (The 32-row table has been divided in the middle to allow it to be printed in a compact fashion.)

The first two columns denote the *Connection Domain Address* of each cellet when viewed in the *Connection Domain*. This domain allows the assignment of contiguous cellets in a frame to a specific connection.

The last two columns denote the *Virtual Time Domain Address* of each cellet when viewed in the Time Domain. This domain relates to the time at which each cellets is transmitted during a frame.

The way the transform works is illustrated by the waveforms shown below the table. The cellets within the waveforms shown in the Connection Domain portion of the figure show the two 4-cellet ranges (delineated in the table) being assigned to two different connections. Note that they appear as 4 contiguous Connection Domain Addresses.

The waveforms in the Time Domain section of the table show how these Connection Domain Addresses assignments result in time domain Time Domain Address assignments by invoking the transform. In the Connection Domain, the cellets are contiguous. In the Virtual Time Domain, they are spread out.

It should be carefully noted, that only the Time Domain Address order of the cellets is relevant to the physical multiplexing of data. Connection Domain Address order is a logical convention that enables a shorthand description of specific cellets that belong to a specific connection. For example, connection **(a)** is a 4-element connection with a Connection Domain Address range of 4. This is much simpler than designating Time Domain Addresses **4, 12, 20,** and **28**.

Using Connection Domain Address ranges to denote

Connection Domain		Virutal Time Domain		Connection Domain		Virtual Time Domain	
0	00000	00000	0	16	10000	00001	1
1	00001	10000	16	*17*	*10001*	*10001*	*17*
2	00010	01000	8	*18*	*10010*	*01001*	*9*
3	00011	11000	24	*19*	*10011*	*11001*	*25*
4	*00100*	*00100*	*4*	*20*	*10100*	*00101*	*5*
5	*00101*	*10100*	*20*	21	10101	10101	21
6	*00110*	*01100*	*12*	22	10110	01101	13
7	*00111*	*11100*	*28*	23	10111	11101	29
8	*01000*	*00010*	*2*	24	11000	00011	3
9	01001	10010	18	25	11001	10011	19
10	01010	01010	10	26	11010	01011	11
11	01011	11010	26	27	11011	11011	27
12	01100	00110	6	28	11100	00111	7
13	01101	10110	22	29	11101	10111	23
14	01110	01110	14	30	11110	01111	15
15	01111	11110	30	31	11111	11111	31

Connection Domain
Connection Domain Addresses
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
4 5 6 7 (a) 17 18 19 20 (b)
Virtual Time Domain Addresses
0 16 8 24 4 20 12 28 2 18 10 26 6 22 14 30 1 17 9 25 5 21 13 29 3 19 11 27 7 23 15 31

Virtual Time Domain
Virtual Time Domain Addresses
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
(a) 4 6 5 7
(b) 20 18 17 19
Connection Domain Addresses
0 16 8 24 4 20 12 28 2 18 10 26 6 22 14 30 1 17 9 25 5 21 13 29 3 19 11 27 7 23 15 31

FIGURE 2. CONNECTION DOMAIN TO TIME DOMAIN BINARY TRANSFORM (32-CELLET FRAME)

connection assignments is a conceptually simple way to visualize dividing a long frame into individual connections. Much more important, though, are the implications in building a Time Division Multiplex system that overcomes the limitations of PSTN multiplexing.

Time Division Multiplex trunks depend on synchronization of a receiving end scheduler with its sending end counterpart. Denoting a channel by Element Address ranges requires only a few bits to define. This saves bandwidth compared to packet- and cell-based systems. It also enables channel bandwidths to be changed quickly.

UNIFORM DISTRIBUTION OF CELLS SLOTS VS. NEARLY UNIFORM DISTRIBUTION

There are circumstances where cellets are exactly uniformly distributed throughout a frame using the binary transform approach. There are three requirements that must be met:

1. The number of cellets per frame must be an integer power-of-two.
2. The number of cellets to be assigned to a channel must also consist of an integer power-of-two number of cellets.
3. The lower boundary Element Address of an assigned channel must be an integer multiple of the number of cellets per channel.

In Figure 2, the waveforms labeled **(a)** meet these criteria. As a result, the 4 Cellet Addresses assigned to the channel are exactly 8 cellets apart.

The waveform labeled **(b)** meets the first two criteria, but not the third. The lower bound of the Element Address range is 17 which is not an integer multiple of 4, the number of cellets assigned to the channel so that the cellets are not uniformly spaced. The inter-cell-slot spacing for cellets with Element Addresses 17, 18 and 19 are spaced 8 cellets apart. The cellet whose Element Address is 20 is only 4 cellets away from the Element Address 18 cellet. It, in turn, is 12 cellets away from the Element Address 19 cellet (that occurred in the preceding frame).

APPLYING BINARY TRANSFORMS TO FRAMES THAT ARE NOT AN INTEGER POWER-OR-TWO NUMBER OF CELLETS LONG

In the real world, the number of cellets per frame is often dictated by available standard trunking facilities. Requiring frames to contain an integer power-of-two number of cellets would not appreciably improve system performance. In some cases, it might reduce jitter slightly, but jitter is already small, and in most cases, the difference would not be noticeable.

The transform used in this paper is clearly related to integer powers-of-two numbers. Fortunately, even if the number of cellets in a frame is not an integer power-of-two, the whole SAIN concept is still valid. The method illustrated in Figure 2 can still be used.

Glossary

- CS Cellet Switching
- PM: Packet Multiplexing
- PS: Packet Switching
- TDM: Time Division Multiplexing
- TDS Time Division Switching