

## **Introduction to Sigtran**

*by Jim Darroch, Protocol Development Manager,  
Artesyn Communication Products*

Sigtran is a working group of the IETF, formed in 1999, and tasked with defining an architecture for the transport of real-time signaling data over IP networks. Its work culminated in not just the architecture, but also the definition of a suite of protocols to carry SS7 and ISDN messages over IP.

This protocol suite is made up of a new transport layer -- the Stream Control Transmission Protocol (SCTP) -- and a set of User Adaptation (UA) layers which mimic the services of the lower layers of SS7 and ISDN.

This article describes the Sigtran architecture and protocol suite. It starts by outlining the network architecture within which the Sigtran suite applies, effectively defining the problem solved by SCTP and the UAs. It continues to describe the protocol requirements for transporting signaling information over IP -- presenting an argument why existing protocols (such as TCP) are not suitable for this purpose. Finally, the UA layers themselves are discussed -- covering both their functionality and their applicability.

Before advancing into the topic of transporting signaling information over IP, it is best to consider why such a facility may be required. In essence, discuss the problem before describing the solution.

### **VoIP Background**

The application space for Sigtran is Voice over IP (VoIP) or, more accurately, Media over IP (MoIP). "Media" applies to any real-time traffic: Voice, music, video, etc.

The possibilities of using the Internet to carry voice traffic arose in 1995. PC software already existed to make connections between computers, over the public Internet (by way of dialup connections to an ISP at the local call rate). In the early days of VoIP this facility was enhanced to accept digitized voice, split it into packets, and present it to the Internet connection as data. Correspondingly, packets received from the remote PC would be reassembled into a digital voice stream and sent to the PC's speakers.

While it was feasible to hold a conversation in such a fashion, there are some fundamental problems with this architecture:

- The voice quality was terrible. The Internet connections had no pre-allocated bandwidth and were subject to variable packet delay. These contributed to jittery conversations, with chunks of speech missing.
- There were no standards which defined how the voice was packetized and how connections were established and managed. Subscribers had to use the same VoIP package in order to talk.

- Only simple (phone) network topologies were supported -- point-to-point connections between known IP addresses. There were no defined interfaces between the Internet and the PSTN.
- Only simple call services were supported; basically, conversations. None of the services normally associated with the PSTN (such as call forwarding) were supported.

These limitations made it impossible to deploy VoIP on the scale required to make it usable by businesses or a large-scale home-user base.

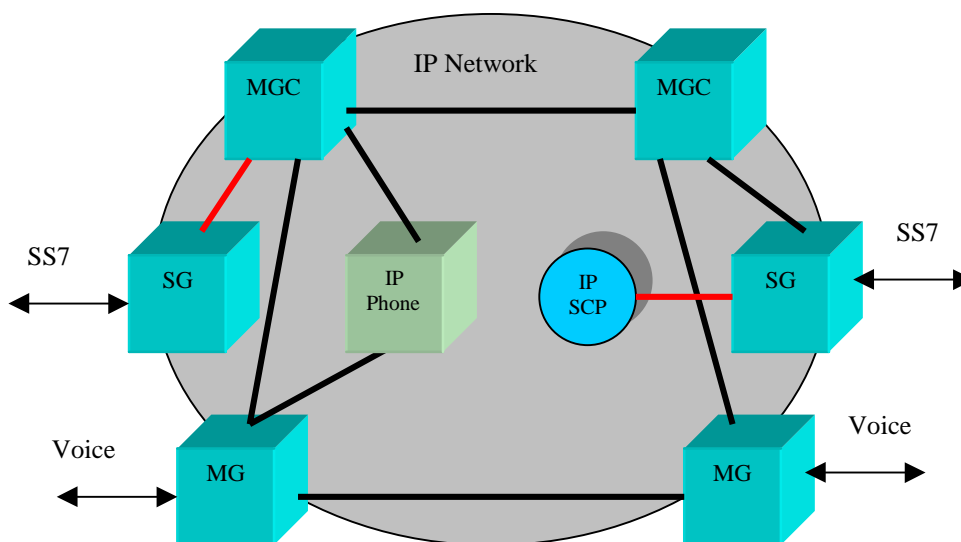
### A Scalable Architecture

The next stage in the evolution of VoIP was to define an architecture which would support integration between the PSTN and IP networks. This would provide a signaling capability for call management as well as defined media paths through the IP network (with reserved bandwidth for real-time media.)

Several groups began work in this area, some cooperatively, others in isolation. These included:

- ETSI -- the TIPHON project.
- ITU -- H.323 working groups.
- The Softswitch Consortium.
- The IETF -- MGCP working group, amongst others.

The work undertaken by these groups established a common architecture which defines the interfaces between the PSTN and IP networks to support voice (or any other streamed media) over IP. Some new network node types have been introduced, and their responsibilities and functions are also defined. The architecture is defined in [2] and illustrated in Fig. 1.



**Fig. 1: Interfaces Between PSTN and IP Networks For Voice**

## **Integrated Network Architecture**

In Fig. 1, the following network elements are defined:

- MGC -- Media Gateway Controller, responsible for mediating call control (between the SG and MG) and controlling access from the IP world to/from the PSTN.
- SG -- Signaling Gateway, responsible for interfacing to the SS7 network and passing signaling messages to the IP nodes.
- MG -- Media Gateway, responsible for packetization of voice traffic and transmitting the traffic towards the destination.
- IP SCP -- an IP-enabled Service Control Point (SCP). This exists wholly within the IP network, but is addressable from the SS7 network.
- The IP Phone (generically referred to as a 'terminal').

These terms are not ubiquitous. In the H.323 world, the MGC is referred to as the "Gatekeeper," while the SG and MG are known singularly as a "Gateway." Often, all three elements are termed "Softswitch."

Note that the MG, SG and MGC are logical entities. In physical implementations the functions may be combined in single network nodes, for example:

- In North American PSTN networks, the signaling and media links are carried on physically separate links. Thus, it makes sense to have physically distinct SG and MG nodes.
- In European PSTN networks, it is more common to have signaling and voice channels sharing physical links, so it makes sense for the SG and MG to be co-located.

This architecture allows network vendors to build Gateways and Controllers which may be deployed in real networks. Network operators may build truly integrated PSTN/IP networks which are scalable and offer viable services to subscribers.

## **Setting The Standards**

In addition to defining the nodes in the network used to interconnect the PSTN and IP networks, the working groups listed before have defined the protocols which run between them. There are two sets of standards which have gained widespread recognition:

- H.323 -- defined by the ITU.
- The IETF model, comprising of SIP, Sigtran, RTP and MGCP or Megaco.
- The IETF define the Sigtran suite specifically for transporting SS7 over IP. This is core to our subject and will be discussed in much more detail than H.323.

## **H.323**

H.323 is not a single protocol specification: it is an architectural description, including references to a suite of protocols required for network integration. Very little of H.323 is

relevant to the matter under discussion here, but the following serves as a brief outline of its components.

The following network elements are defined:

- Gateway -- provides translation of both call control (signaling) and media from the PSTN to/from an IP network; roughly equivalent to a combination of the SG and MG (Fig. 1, again.)
- Gatekeeper -- governs access to the Gateway from both the IP and PSTN side; roughly equivalent to the MGC (in Fig. 1.)
- H.323 terminal -- typically an IP phone (in Fig. 1); this could be any IP device used as such (e.g. a PC.)
- The generic term "H.323 endpoint" is defined as either a Terminal or Gateway -- it refers to the terminus of an H.323 connection.

The following protocols are defined:

- H.225 RAS -- Registration, Authentication and Status protocol. This runs between the Gatekeeper and an H.323 endpoint to provide secure access control.
- H.225 Call Signaling -- establishes connections between H.323 endpoints.
- H.245 Control Signaling -- runs between H.323 endpoints, allowing exchange of control messages (such as terminal capabilities.)
- RTP -- the Real Time Protocol, which carries packetized media between H.323 endpoints.

## **IETF Model**

The IETF, through a number of working groups, have established a similar model to H.323. The network nodes defined in this model are again shown in Fig. 1. The protocols defined are:

- SCTP (and adaptation layers) -- this is the Sigtran protocol suite, which is used to carry SS7 between the SG and MGC.
- Megaco -- Media Gateway Control (also defined by the ITU as H.248.) This is the control protocol between an MGC and MGs.
- SIP -- Session Initiation Protocol. SIP is a call-control protocol running between MGCs or MGCs and IP-based phones.
- RTP -- the IETF specifies the same Real Time Protocol as H.323 for carrying packetized media.

It is in the IETF model that the Sigtran suite of protocols (including SCTP) is of most interest when considering IP as a transport for signaling traffic. In Fig. 1 the red lines represent the connections which carry the Sigtran protocols.

## The Need For SCTP And Adapting The Solution

SCTP is a new transport protocol, designed with the transport of time-sensitive signaling data in mind. It remains flexible enough, however, to be of general use. One question to be addressed is why the Sigtran group went to all the trouble of defining a new transport layer, when they could have chosen to use TCP?

The easiest way, from a protocol implementer's point of view, to run an SS7 layer over IP is to take the chosen layer, define an interface to the IP transport layer (TCP) and plug the two together.

Unfortunately, there are some fundamental flaws with this approach:

- It's inflexible -- once you've got SCCP running over TCP/IP, you are still without a solution to run ISUP.
- It's unlikely that any other equipment vendor will have designed an SS7 to IP interface which is anything like yours. The structure of your IP packets, holding the SS7 information, will appear alien to any other vendor's packets.
- There may be peer-to-peer management issues (such as connection establishment or quality of service negotiation) which have not been addressed.

It's clear from the above that we have to address the issues of standardization and scalability (or reuse). One way of approaching this is to provide an "adapter" over TCP/IP which redefines the transport service in terms of what the upper signaling protocol would expect: In essence, to make TCP/IP look like a lower layer of SS7 (for example, MTP3.) Such an object is referred to as a User Adaptation (UA) layer -- which sits above TCP in the protocol stack.

The primary roles of such a layer would be to:

- Define a standard method by which the upper layer (for example, ISUP or SCCP) would be encapsulated within the TCP message.
- Provide a framework for peer-to-peer management, such as socket interfaces, port numbers and so on.

Having solved the problems of peer-to-peer management and standardization (both ends of the connection use the same UA), scalability and reuse fall into place as the same model is applied to other upper layers -- there would be a UA for use by SCCP, ISUP and so forth.

The downside of such an architecture is that it relies on the services of TCP. We must ask ourselves an important question: Is a *byte-stream* oriented protocol well suited to carrying time-critical signaling *messages*?

## **The Problem with TCP**

There have been questions raised over the suitability of TCP to carry a time-critical protocol like SS7. The questions arise from the very fact that TCP was originally designed for an altogether different purpose.

TCP is byte-streamed, it provides a single stream of data and guarantees that data to be delivered in byte-sequence order. This makes it ideal for delivery of large, unstructured pieces of data, like a file or e-mail message. Unfortunately, that very feature is its downfall.

TCP is particularly sensitive to delays caused by network errors: By loss of bytes, messages, or sequence violation. When this occurs, TCP will hold up delivery of all data (within this monolithic stream) until the correct sequence is restored.

Consider the consequences of this in terms of delivering SS7 messages. If a single TCP stream carries the ISUP signaling for many connections, then the loss of a message relevant to only one resource (such as a telephone call) will result in the delay of all other ISUP messages.

An additional problem is the duration of some TCP timers, which are defined in terms of many seconds. In particular, the length of the Connection, Keep Alive and Retransmission timers may result in excessive delays to connection set-up, determining connection loss and data retransmission.

Recognition of this has resulted in industry experts from many network equipment vendors cooperating to study the problems and propose a solution. This has been done under the auspices of the IETF, in the form of the Sigtran working group.

## **SCTP**

The Sigtran Working Group has defined a Stream Control Transmission Protocol (SCTP) [1] which aims to address the shortcomings of TCP. SCTP itself is a general purpose protocol, a replacement for TCP.

SCTP has the following set of features:

- It is a Unicast protocol -- data exchange is between two known endpoints.
- It defines timers of much shorter duration than TCP.
- It provides reliable transport of user data -- detecting when data is corrupt or out of sequence, and performing repair as necessary.
- It is rate-adaptive, responding to network congestion and throttling back transmission accordingly.
- It supports multi-homing -- each SCTP endpoint may be known by multiple IP addresses. Routing to one address is independent of all others and, if one route becomes unavailable, another will be used.

- It uses an initialization procedure, based on cookies, to prevent denial-of-service attacks.
- It supports bundling, where a single SCTP message may contain multiple "chunks" of data -- each of which may contain a whole signaling message.
- It supports fragmentation, where a single signaling message may be split into multiple SCTP messages in order to be accommodated within the underlying PDU.
- It is message-oriented, defining structured frames of data. TCP, conversely, imposes no structure on the transmitted stream of bytes.
- It has a multi-streaming capability -- data is split into multiple streams, each with independent sequenced delivery. TCP has no such feature.

Aside from enhanced security, the final two points are what makes SCTP so much more suited to carrying signaling messages than TCP.

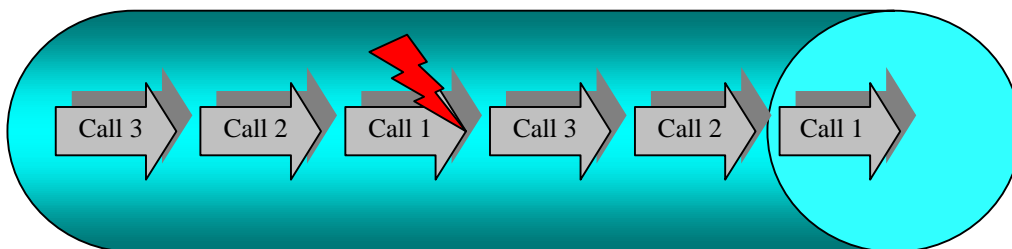
The transmission of bounded, structured frames is a useful feature for the UA layers (as well as any other potential SCTP user.) The transport protocol does all the work to split the stream of data into message segments -- relieving the user's responsibility to interpret a continuous stream of bytes. Such processing would otherwise have to be repeated by each user.

Defining a structure for the messages also makes support of bundling and fragmentation less arduous.

Multi-streaming is the main attraction, however (even the protocol's name implies this: *Stream Control* Transmission Protocol.) This feature was explicitly designed to allow users to partition a single IP connection between two endpoints into separate logical streams of data, and assign each stream to a particular application or resource. The principle is that errors or delays on one stream will not interfere with normal delivery on another.

Consider the example of the ISUP protocol. ISUP carries signaling messages for many PSTN resources (essentially trunk circuits, think of each one as a telephone call.) If three calls are in progress, then one of the calls experiencing a loss of data should not cause a delay in transmission of messages relating to another call.

If ISUP is carried over TCP (see Fig. 2) a single "pipe" of data carries all ISUP messages for all three calls.

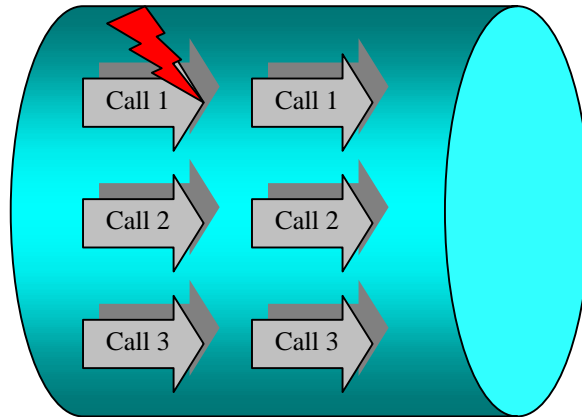


**Fig. 2: SingleData Pipe Carries All ISUP Messages**

## ISUP Over TCP And SCTP

If the establishment messages for all three calls are received correctly, then the calls are in progress. If the calls are then released in the same order, but the release message for Call 1 is lost, then the release messages for the other two calls will be delayed while TCP recognizes the loss and recovers the data. This has the potential to delay the release of Calls 1 and 2 for a large number of seconds, an unacceptable delay to the PSTN.

Consider now ISUP carried over SCTP (see Fig. 3.)

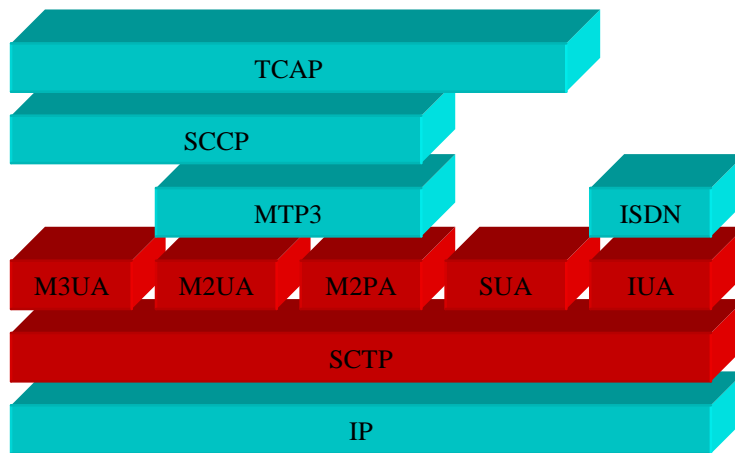


**Fig. 3: ISUP Carried Over SCTP**

In this case, loss of a message relating to Call 1 affects only that stream of data. Calls 2 and 3 are serviced as normal. The resources relating to Call 1 will still be tied up until the transport protocol recovers from the loss of data, but this situation is no different from that in traditional PSTN transport protocols.

## Sigtran Architecture

The Sigtran suite (including SCTP and all the UAs) is shown in 0.



**Fig. 4: Sigtran Protocol Suite**

Some notes on Fig. 4:

- The red areas show the new protocol layers as defined by Sigtran, while the blue areas illustrate existing protocols. For simplicity, ISUP has not been shown -- this would normally run over MTP3 or M3UA.
- The UA layers are named according to the service they replace, rather than the user of that service. For example, M3UA adapts SCTP to provide the services *of* MTP3 -- rather than providing a service *to* MTP3.

The Sigtran adaptation layers all serve a number of common purposes:

- To carry upper layer Signaling Protocols over a reliable IP-based transport.
- To provide the same class of service offered at the interface of the PSTN equivalent. For example, M3UA must provide the same look and feel to its users as MTP3 -- in terms of services, at least (M3UA does not actually replace the features and operations of MTP3).
- To be transparent: The user of the service should be unaware that the adaptation layer has replaced the original protocol (although this is largely dependant on the implementation).
- To remove as much need for the lower SS7 layers as possible.

Sigtran currently defines six adaptation layers, as follows:

- M2UA [4] provides the services of MTP2 in a client-server situation, such as SG to MGC. Its user would be MTP3.
- M2PA [5] provides the services of MTP2 in a peer-to-peer situation, such as SG-to-SG connections. Its user would be MTP3.
- M3UA [3] provides the services of MTP3 in both a client-server (SG to MGC) and peer-to-peer architecture. Its users would be SCCP and/or ISUP.
- SUA [4] provides the services of SCCP in a peer-to-peer architecture, such as SG to IP SCP. Its user would be TCAP, or another transaction-based application part.
- IUA [7] provides the services of the ISDN Data Link Layer (LAPD.) Its user would be an ISDN Layer 3 (Q.931) entity.
- V5UA [9] provides the services of the V.5.2 protocol.

It should be noted that the framework is flexible enough to allow for the addition of new layers as required.

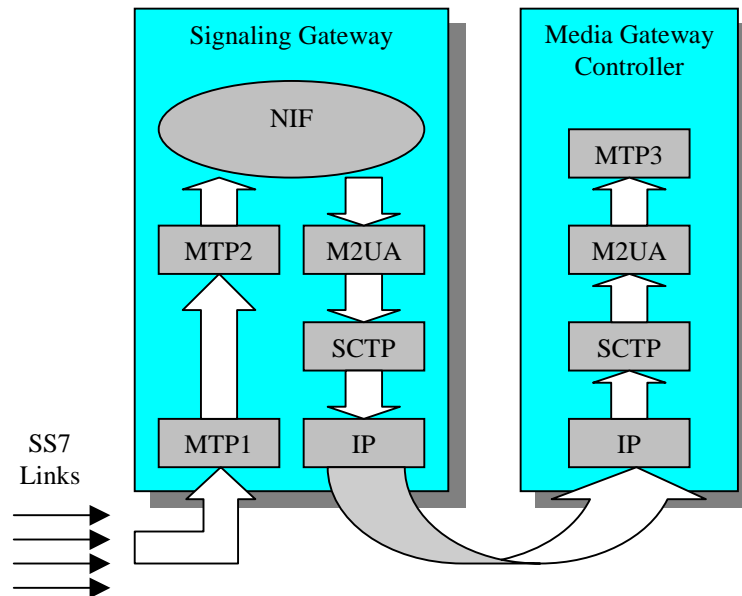
Each UA has a particular applicability, which is discussed in the following sections.

## **M2UA**

M2UA is used to transfer MTP2-user data between the MTP2 instance on a SG and the MTP3 instance on an MGC. As such it operates a client-server model, where the MGC is the client and the SG is the server.

M2UA provides a means by which an MTP2 service may be provided on an MGC. In essence, extending SS7 into the IP network.

The architecture of M2UA usage is shown in Fig. 5.



**Fig. 5: M2UA Architecture**

Effectively, the MTP3 instance on the MGC is the user of the MTP2 instance on the SG! Neither MTP2 nor MTP3 are aware that they are remote from one another. This process, by which signaling messages are passed over IP from the top of one SS7 layer to the bottom of another, is described as *backhauling*.

The MTP3 user at the MGC would usually be ISUP.

This architecture is most applicable in the following circumstances:

- There is a low density of SS7 links at a particular physical point in the network (perhaps as low as one)
- There are a large number of physically separate SG functions (due to the SS7 links being physically located remotely from each other)
- The SG function is co-located with an MG (usually due to one or more of the previous conditions)

In this case, it makes sense to host MTP3 in the MGC. The SS7 address (the pointcode) of the system resides with MTP3: If each SG had its own MTP3 layer, a large number of pointcodes would be required to implement a (logically) single gateway.

This particular Sigtran configuration is likely to be common in European networks, where the SS7 signaling links share the physical medium with voice circuits.

At the SG side, the depiction of M2UA at the peer level to MTP2 is slightly misleading. M2UA is, in many ways, a user of MTP2: The specification defines that M2UA is

responsible for initiating protocol actions which would normally be issued by MTP3, such as:

- Link activation and deactivation
- Sequence number requests
- MTP2 transmit/retransmit buffer updating procedures
- Buffer flushing

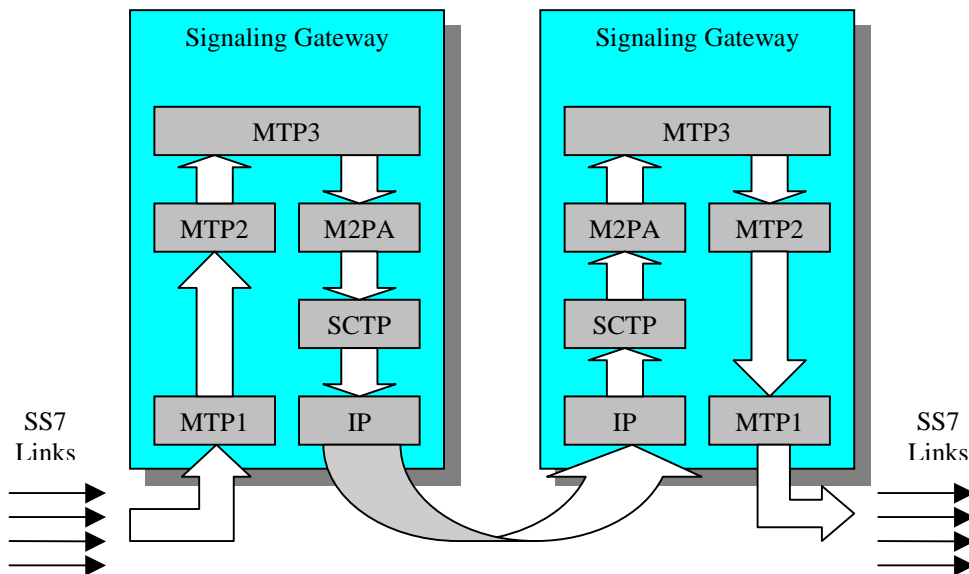
These are implementation issues, however. Such functionality could well reside within the Nodal Interworking Function (NIF).

## M2PA

M2PA is the peer-to-peer equivalent of M2UA. Rather than provide a link between remotely-located MTP2 and MTP3 instances, it replaces an MTP2 link beneath MTP3. The user of M2PA is MTP3 at both ends of the connection (with M2UA, one user is MTP3 and the other is an SG IWF).

M2PA provides a means for peer MTP3 layers in SGs to communicate directly. In essence, it extends the reach of SS7 over the IP network.

The architecture of M2PA usage is shown in Fig. 6.



**Fig. 6: M2PA Architecture**

This architecture is most applicable for an SG to SG connection, used to bridge two SS7 network "islands." In this case, each SG may connect to multiple other SGs, none of which need to know about the upper layer that they are supporting.

MTP3 is present on each SG to provide routing and management of the MTP2/M2PA links. Because of the presence of MTP3, each SG would require its own SS7 pointcode.

Replacing MTP2 links with M2UA is a distinct case from accessing an IP SCP from an SG (the service provided by SUA.) In the SUA case it is known that the higher layer is TCAP (or another application part): The SCCP services can be explicitly provided. M2PA, on the other hand, has no knowledge of the upper SS7 layer.

The significant difference in function from M2UA is that M2PA actually provides an MTP2-like service itself. M2UA merely provides an interface to a remote MTP2 service. This means that M2PA is responsible for:

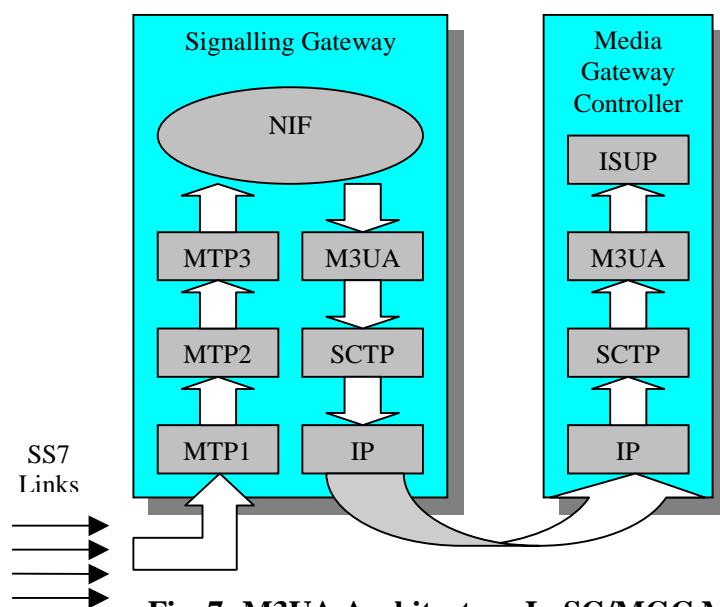
- Link activation/deactivation (in response to requests from MTP3)
- Maintaining link status information
- Maintaining sequence numbers and retransmit buffers, for retrieval by MTP3
- Maintaining local and remote processor outage status

### M3UA

M3UA is similar to M2UA, in that it operates in a client-server way to provide an upper layer SS7 with protocol remote access to the lower layers. Like M2UA, M3UA operates between an SG and an MGC.

M3UA provides a means by which an MTP3 service may be provided on an MGC (thus, terminating the ISUP connection on the MGC) -- essentially extending the reach of SS7 into the IP network.

The architecture of how M3UA fits into the SG/MGC model is shown in Fig. 7.



**Fig. 7: M3UA Architecture In SG/MGC Model**

The MTP3 in the SG is unaware that the ISUP user is located remotely (the NIF will register itself as ISUP with MTP3.) Similarly, the ISUP layer at the MGC will be unaware that it is not served by a local MTP3. This is another example of the MGC backhauling signaling messages from the SG.

This architecture is most appropriate in the following circumstances:

- There is a high enough density of SS7 links to make a standalone SG viable
- The SS7 links are physically accessible at a single point
- These conditions are common in North American networks, where the SS7 links are physically separate from the voice circuits. In this case, a number of links are gathered together into a single physical medium (for example, a T1 line.)

Here each SG has a local MTP3 instance, and so must have its own SS7 pointcode.

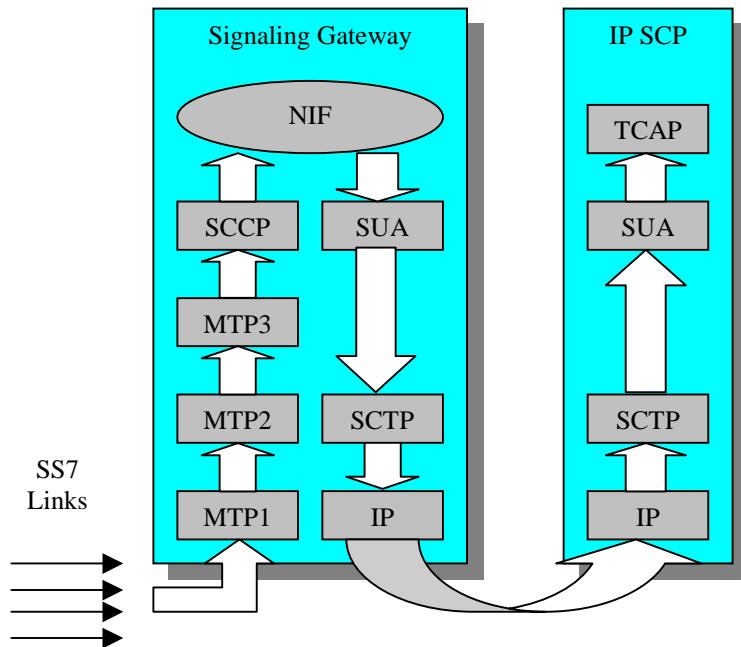
The M3UA layer is responsible for maintaining the MTP3-ISUP interface across an SCTP connection. Commands and requests for data transfer passed down by ISUP on the MGC are carried (over SCTP) by M3UA and presented to the upper interface of MTP3 on the SG. Indications and incoming data messages are passed up from MTP3 on the SG and carried (over SCTP) by M3UA to the lower interface of ISUP on the MGC.

As with M2UA, the depiction of M3UA as a peer of MTP3 may be slightly misleading. Logically, M3UA is a user of MTP3 at the SG. How the two layers interwork is an implementation issue.

## **SUA**

SUA provides a means by which an Application part (such as TCAP) on an IP SCP may be reached via an SG. The network architecture associated with SUA allows for multiple IP SCPs to be reached via a single SG. The IP SCP(s) do not have local MTP3 instances, and so do not require their own S7 pointcodes (MTP3, and the pointcode, reside on the SG.)

The architecture of SUA use between an SG and IP SCP is shown in Fig. 8.



**Fig. 8: SUA Architecture Usage**

The functionality of SUA *could* be provided by the MTP2 or MTP2 UAs. However, SUA provides the mapping between SCCP addresses and IP addresses (at the SG). Without such a function, SCCP would have to be present at each IP SCP and the external SS7 network would require knowledge of each such SCCP instance. SUA can abstract the presence of each IP SCP, providing one SCCP address to cover all nodes.

The services of individual databases are addressed via Subsystem Number (SSN). SUA provides a service not unlike SCCP Global Title Translation (GTT) to map these SSNs into SCCP connection IDs (which are used to route the Application Part messages to the appropriate IP SCP).

SUA is also flexible enough to support Application Parts running between two network nodes wholly within the IP network. This is particularly relevant to emerging networks, where there may be no need for an underlying "traditional" SS7 network. In this case the IP SCP stack would be the same on both (IP-based) nodes.

SUA will further allow Service Databases in the SS7 network to be accessed from the IP network. In this case the architecture would be the same as in Fig 8.

## **IUA and V5UA**

We are concerned here solely with the adaptation of SS7 layers over SCTP, and so IUA and V5UA are not discussed. Refer to 6 and 8 for more information on these UAs.

## References And Glossary

1. RFC 2960 -- Stream Control Transmission Protocol
2. RFC 2719 -- Framework Architecture for Signaling Transport
3. SS7 MTP3 -- User Adaptation Layer <draft-ietf-sigtran-m3ua-06.txt>
4. SS7 MTP2 -- User Adaptation Layer <draft-ietf-sigtran-m2ua-06.txt>
5. SS7 MTP2 -- User Peer-to-Peer Adaptation Layer <draft-ietf-sigtran-m2pa-02.txt>
6. SS7 SCCP -- User Adaptation Layer <draft-ietf-sigtran-sua-04.txt>
7. RFC 3057 -- ISDN Q.921-User Adaptation Layer
8. IETF Web Site: <http://www.ietf.org>
9. V.5.2 -- User Adaptation Layer <draft-ietf-sigtran-v5ua-00.txt>

ETSI	European Telecommunications Standardization Institute
IETF	Internet Engineering Task Force
GTT	Global Title Translation
HDLC	High Level Data Link Control
IP	Internet Protocol
ISUP	ISDN User Part
ITU	International Telecommunications Union
IUA	ISDN User Adaptation layer
M2PA	MTP2 Peer-to-peer user Adaptation layer
M2UA	MTP2 User Adaptation layer
M3UA	MTP3 User Adaptation layer
Megaco	Media Gateway Control (IETF Working Group)
MG	Media Gateway
MGC	Media Gateway Controller
MGCP	Media Gateway Control Protocol
MTP	Message Transfer Part
PSTN	Public Switched Telephone Network
RTP	Real Time Protocol
SCCP	Signaling Connection Control Part
SCP	Service Control Point
SCTP	Stream Control Transport Protocol
SG	Signaling Gateway
Sigtran	Signaling transport (IETF Working Group)
SIP	Session Initiation Protocol (IETF Working Group)
SP	SS7 Signaling Point
SS7	Signaling System No. 7
SSN	SubSystem Number
SSP	Service Switching Point
STP	Signaling Transfer Point
TALI	Transport Adaptation Layer Interface
TCAP	Transaction Capabilities Application Part
TIPHON	Telecommunications and IP Harmonization On Networks
UA	User Adaptation layer
V5UA	V5.2-User Adaptation layer

## About The Author

Jim Darroch has been with Artesyn Communication Products since 1996 and is currently the Protocol Development Manager. Previous positions included Software Architect and Project Manager, working with X.25, ISDN, Frame Relay, SS7 and Sigtran. Before joining ACP, Jim spent four years at HP as a senior software engineer and technical architect working on a distributed surveillance system for SS7 networks. Jim earned a BSc (Hons) in Computer Science from Strathclyde University, in Scotland, and lives in Edinburgh.

He can be contacted at; [jimd@artesyncp.com](mailto:jimd@artesyncp.com), telephone: +44 (0) 131 475 7014

For more information about Artesyn visit its web site at: <http://www.artesyncp.com>

