The Session Initiation Protocol

Report August 2003

About Meridea Financial Software

Meridea Financial Software Ltd is a leading provider of innovative retail financial services solutions. It enables financial institutions to offer advanced, integrated, personalized online and mobile services for global markets. Meridea was founded by Accenture, Nokia, Sampo and 3i in 2001.The Meridea Product Suite provides financial institutions with a complete front-end software package for multi-channel financial services – banking, investment and insurance. It uniquely combines ready-to-run best practice business processes with a high degree of configurability. Meridea's flagship product, the Meridea Product Suite, enables financial institutions to adapt their services to the customers' lifestyles, offering the freedom of choice to use financial services when and where they need it. Working with Meridea means supporting mobility of the consumer, the device and the application.

With the Meridea Product Suite, financial institutions can implement online and mobile services in a cost effective and flexible way, taking advantage of the widest functionality in the market today. Meridea's unique J2EE-based architecture allows institutions to easily customize business processes and to implement innovation services. All channels use the same processes, while channel optimization is handled by separate tiers.

The majority of mobile and online devices are currently supported and supporting new devices is quick and inexpensive. Compared to in-house solutions, Meridea provides more functionality with a solid roadmap into the future, at lower cost and reduced technology risk.

Abstract

"IP over everything and everything over IP" is the *IETF's mantra. The mantra can be split into two parts. Firstly, "IP over everything" describes the convergence of different types of physical networks that the IP protocol runs over. The second part "everything over IP" is about the convergence of different types of media (voice, data, video, etc.), interactions modes (messaging, browsing, rich call), transacting and services on top of one platform, the IP platform.

The Session Initiation Protocol (SIP) is quickly emerging as the industry standard to deliver converged services and applications. SIP can be thought of as the "glue" when creating unified systems of telephony, video, data and real-time web services. The ability to mask many different types of addressing schemes under one address, locate users, modify sessions, invite users to sessions and terminate sessions are key capabilities that used together with other protocols has unlimited potential for new innovations.

At the moment SIP is mostly used to integrate a corporation's phone system and computer system under one common network that can be accessed by a single application. Another important application that SIP is currently being used for is Instant Messaging (IM). Most IM applications today are implemented with proprietary protocols as e-mail was in 1985 before the Simple Mail Transfer Protocol (SMTP) was introduced. It has been said that SIP for person-to-person communication is as important as the Hypertext Transfer Protocol (HTTP) was for the web and SMTP for e-mail.

SIP is an expandable protocol that is suited for many different services and applications. It is impossible to predict everything that SIP will be used for in the future.

* The Internet Engineering Task Force (IETF) is a large open international community of network designers, operators, vendors, and researchers concerned with the evolution of the Internet architecture and the smooth operation of the Internet. It is open to any interested individual.

Table of Contents

Table of Figures

1 Introduction

The Session Initiation Protocol (SIP) is part of the Internet Multimedia Architecture (Figure 1), which consists of many precise protocols that have been designed to perform one function. New services and applications can be created by combining multiple protocols together in a manner best suited for the implementations purpose. This type of design is very flexible and is a driver for constant innovation and development. It is not likely that a service could be created using SIP alone, although SIP does not depend on any other protocol. The current excitement around SIP has made some developers forget that SIP has been designed to be a signaling protocol and should not be used for tasks that are better suited for other protocols, such as data transport.

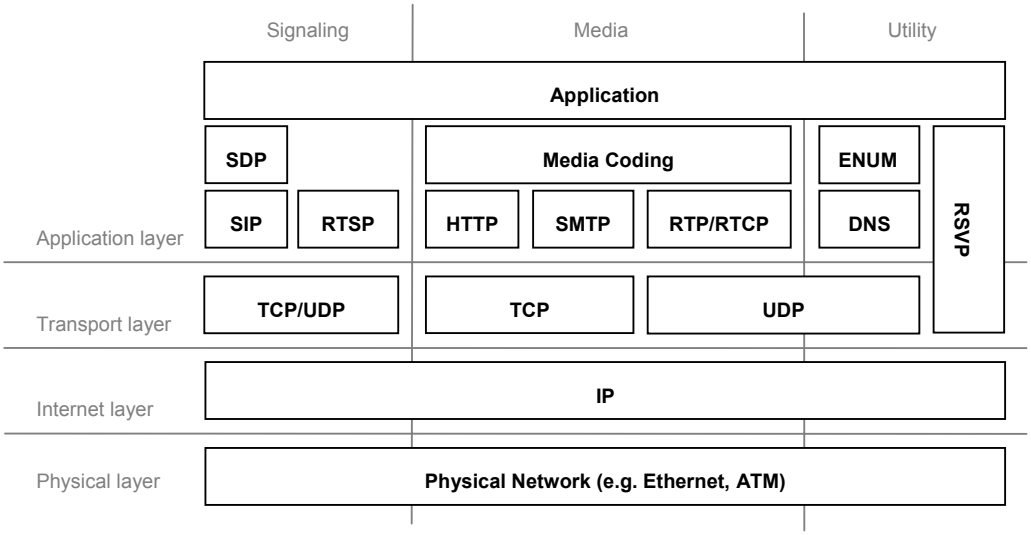


Figure 1 Internet Multimedia Architecture

SIP will replace, as well as, co-exist with other signaling protocols like SS7 and H.323. A unique feature of SIP, compared to other signaling protocols, is that it gives enterprises the possibility to create or integrate their own *IN applications without operator involvement.

The SIP standardization by the IETF is nearing completion with the publication of RFC 3261 (June 2002) and is now mostly in maintenance mode. The core functionality of SIP is sufficient to build the 3G mobile system and corporate PBX's; however more operational experience is still need. The IETF has produced 25 SIP RFC's (823 pages) and the core SIP RFC 3261 is the longest RFC ever. The amount of work done around SIP is already a clear indication of its importance to the Internet and next generation networks.

* Intelligent Network (IN) is a telephone network architecture originated by Bell Communications Research (Bellcore) in which the service logic for a call is located separately from the switching facilities, allowing services to be added or changed without having to redesign switching equipment. According to Bell Atlantic, IN is a "service-specific" architecture.

1.1 Intended Reader

The reader of this report should be familiar with some type of signaling and the Internet architecture in general.

1.2 Purpose of Report

The purpose of this report is to familiarize the reader with SIP. The report describes what SIP is capable of and identifies what SIP should not be used for. After reading this report the reader should be capable of innovating services based on SIP.

The Report is heavily copied from RFC 3261 and other material and is intended for internal company use only. It is by no means meant to be published in any way. All text that is in *italics* has been copied directly from RFC 3261.

2 Overview of SIP Functionality

SIP is an application-layer control protocol that can establish, modify, and terminate multimedia sessions such as Internet telephony calls. SIP can also invite participants to already existing sessions, such as multicast conferences. Media can be added to (and removed from) an existing session. SIP transparently supports name mapping and redirection services, which supports personal mobility - users can maintain a single externally visible identifier regardless of their network location. SIP works independently of underlying transport protocols and without dependency on the type of session that is being established.

SIP supports five facets of establishing and terminating multimedia communications:

1. *[*]User location: determination of the end system to be used for communication*
2. *User availability: determination of the willingness of the called party to engage in communications*
3. *User capabilities: determination of the media and media parameters to be used*
4. *Session setup: "ringing", establishment of session parameters at both called and calling party*
5. *Session management: including transfer and termination of sessions, modifying session parameters, and invoking services*

SIP is not a vertically integrated communications system. ***SIP is rather a component that can be used with other IETF protocols to build a complete multimedia architecture.*** *Typically, these architectures will include protocols such as the Real-time Transport Protocol (RTP) (RFC 1889) for transporting real-time data and providing QoS feedback, the Real-Time streaming protocol (RTSP) (RFC 2326) for controlling delivery of streaming media, the Media Gateway Control Protocol (MEGACO) (RFC 3015) for controlling gateways to the Public Switched Telephone Network (PSTN),*

[*] User location refers to routing and not presence information, i.e. finding the network address of the device that the user has registered in to the network with if any.

and the Session Description Protocol (SDP) (RFC 2327) for describing multimedia sessions. Therefore, SIP should be used in conjunction with other protocols in order to provide complete services to the users. However, the basic functionality and operation of SIP does not depend on any of these protocols.

SIP does not provide services. Rather, SIP provides primitives that can be used to implement different services. For example, SIP can locate a user and deliver an [*]opaque object to his current location. If this primitive is used to deliver a session description written in SDP, for instance, the endpoints can agree on the parameters of a session. If the same primitive is used to deliver a photo of the caller as well as the session description, a "caller ID" service can be easily implemented. As this example shows, a single primitive is typically used to provide several different services.

SIP does not offer conference control services such as floor control or voting and does not prescribe how a conference is to be managed. SIP can be used to initiate a session that uses some other conference control protocol. Since SIP messages and the sessions they establish can pass through entirely different networks, SIP cannot, and does not, provide any kind of network resource reservation capabilities.

The nature of the services provided make security particularly important. To that end, SIP provides a suite of security services, which include denial-of-service prevention, authentication (both user to user and proxy to user), integrity protection, and encryption and privacy services.

SIP works with both IPv4 and IPv6.

3 SIP Architecture

The SIP protocol defines four logical entities: user agents, proxy servers, redirect servers and registrars. Most implementations combine the proxy server, redirect and/or registrar into one server that is commonly called the SIP server. The SIP server also often includes a non-SIP entity called the location server that provides location services. Together these components make up the basic SIP architecture.

The basic SIP architecture is expanded with SIP application servers, SIP gateways and SIP Firewalls and NATs to deliver enhanced and more complex services to users.

[*] *Opaque ~transparent*

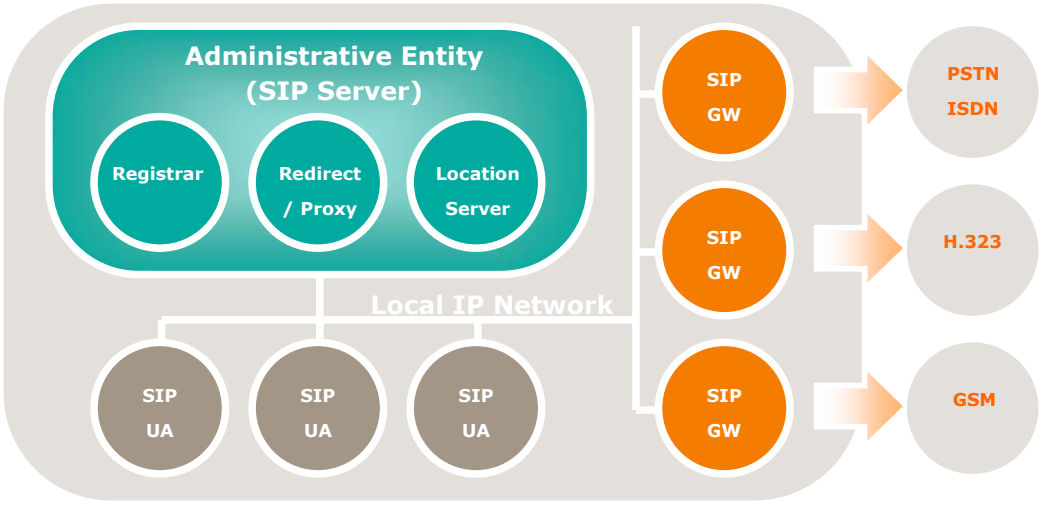


Figure 2 Basic SIP Architecture

It is important to remember that SIP is a protocol and not a complete system; therefore there is no reference SIP architecture. There are many possible architectures depending on the needs of the local installation. The installation can range from a simple PBX with a single proxy for inbound calls to the 'typical' trapezoid architecture with inbound/outbound proxies to more complex systems with multiple proxies in various places.

The 3G mobile telephone system is a good example of a complex architecture that uses SIP. 3G is a powerful driver for SIP, since both UMTS (3GPP) and cdma2000 (3GPP2) have chosen it as their primary signaling protocol. SIP will be used to locate users and by the all IP multimedia subsystem, that will integrate mobile voice communications and Internet technologies together and serve as an application platform for 3G. The 3G SIP architecture is more complex than Internet SIP architecture due to the nature of 3G with less bandwidth (header compression needs) and the walled garden mentality that causes more consideration on billing. The 3G SIP server is divided into three logical entities: the Interrogating CSCF (Call Session Control Function), the Proxy CSCF and the Serving CSFS. The 3G specifications are out of scope of this document and will not be mentioned further. For more information about SIP in relation to 3GPP: http://www.3gpp.org specs 23.218, 23.228 and 24.228 and reports 23.815 and 23.821.

3.1 SIP User Agent

The SIP user agent is a logical entity that can act both as a user agent client (UAC) and a user agent server (UAS). The UAC creates requests and the UAS generates responses. The role of the UA last only for the duration of a transaction. SIP user agents can be software applications or hardware implementations. The SIP User Agent is responsible for interacting with the user. A SIP proxy can act as a SIP user agent. Example SIP user agents are: SIP phones (hardware), SIP based soft phones (PC) and SIP based IM clients (MS Windows Messenger).

3.2 Registrar

A registrar is a server that accepts REGISTER requests and places the information it receives in those requests into the location service for the domain it handles. A registrar is usually co-located with a redirect server and/or a proxy server.

The Registrar enables user mobility which is explained in more detail later in this document.

3.3 Redirect Server

Redirect servers help locate SIP user agents by providing alternative locations where the user can be reachable. Unlike the proxy server, the redirect server does not forward request to new locations, but directs the client to contact an alternate set of URI's with a new request.

3.4 Proxy Server

Proxy Servers are intermediary entities that act as both a server and a client for the purpose of making requests on behalf of other clients. A proxy server primarily plays the role of routing, which means its job is to ensure that a request is sent to another entity "closer" to the targeted user. Proxies are also useful for enforcing policy (for example, making sure a user is allowed to make a call). A proxy interprets, and, if necessary, rewrites specific parts of a request message before forwarding it.

3.5 Non-SIP Entities

The non-SIP entities introduced in this section is not a complete list.

3.5.1 Location Server

A location service is used by a SIP redirect or proxy server to obtain information about a callee's possible location(s). It contains a list of bindings of address-of-record keys to zero or more contact addresses. The bindings can be created and removed in many ways; the SIP protocol specification defines a REGISTER method that updates the bindings.

Location servers can use information from registrars or from other databases. Most registrars upload location updates to a location server upon receipt.

The redirect and proxy servers do not use SIP to obtain information from the location server. Some location servers use Lightweight Directory Access Protocol (LDAP) [RFC1777] to communicate with SIP servers. [SIP Demystified page 106]

3.5.2 Other IETF Protocols

To provide complete services to users SIP should be used in conjunction with other IETF protocols. Below are listed typical IETF protocols that SIP can be used with. The list is not complete and each service implementation will require a different set of protocols.

IETF protocols typically used with SIP to build a complete multimedia architecture:

- RTP: The Real-time Transport Protocol (RFC1889) for transporting real-time data and providing QoS feedback
- RTSP: The Real-Time streaming protocol (RFC2326) for controlling delivery of streaming media
- MEGACO: The Media Gateway Control Protocol (RFC 3015) for controlling gateways to the Public Switched Telephone Network (PSTN)
- SDP: The Session Description Protocol (RFC 2327) for describing multimedia sessions.

IETF protocols for Number resolution:

- ENUM: ENUM is a DNS-based architecture and protocol (RFC2916) by which an E.164 number (a telephone number), as defined in ITU Recommendation E.164, can be expressed as a Fully Qualified Domain Name in a specific Internet Infrastructure domain defined for this purpose (e164.arpa). The result of the ENUM query is a series of DNS NAPTR resource records (RFC2915) which can be used to contact a resource (e.g. URI) associated with that number.
- TRIP: Telephony Routing over IP (RFC2871) is a policy driven inter-administrative domain protocol for advertising the reachability of telephony destinations between location servers, and for advertising attributes of the routes to those destinations. TRIP's operation is independent of any signaling protocol, hence TRIP can serve as the telephony routing protocol for any signaling protocol.

3.5.3 Reused Internet Architecture

The SIP architecture takes advantage of many already existing Internet features such as URIs, MIME and DNS. SIP heavily re-uses SMTP and HTTP structures. From SMTP SIP uses the addressing scheme and from HTTP the SIP message borrows much of the syntax in the SIP message headers. Also many HTTP codes are re-used by SIP e.g. SIP “address not found” is “404” which is familiar from HTTP.

3.6 SIP Application Servers

New services are being created based on SIP. The New services run on application servers (e.g. J2EE, .NET) that support SIP. Instant Messaging (IM) & Presence and IP PBX are good examples of SIP based services that run on SIP enabled application servers. Some application servers also interact with other media and content servers and can be responsible for load balancing across a distributed architecture. Most application servers support SIP, but can offer also multi-protocol support e.g. for H.323. Some application servers support IP and SIP but run proprietary protocols over them.

IM and presence servers typically manage presence, availability, location, and user profile information. The server can collect presence context from IM clients, positioning systems and networks. The

SIP/*SIMPLE based back-end system has a network interoperability layer that supports the exchange of presence information across SIP. IM and Presence services can be implemented using various open (OMA IMPS (Wireless Village™,XMPP) or proprietary (e.g. ICQ, TOC) protocols, however SIP is likely to become the industry standard.

"SIMPLE is a set of extensions to the established SIP protocol which define SIP signaling methods to handle the transport of data and presence. According to observers, one potential problem with SIMPLE is that it is a paging protocol meant to perform signaling but not to carry anything else. "SIMPLE can carry a brief conversation, which is great for single-session IM traffic and SMS traffic, but it is not very good for doing the heavy load to carry things like data signals or video signals on top," IDC's Mahowald says. "There is where you have to deviate from the standard to create your own extensions." Because of the inherent limitations of SIP and because many SIMPLE extensions are still under construction, the existing implementations of the protocol from Microsoft and IBM have included proprietary extensions. Furthermore, SIMPLE is missing core IM-related functionality such as contact lists and group chat capabilities, according to observers. Another potential pitfall with SIMPLE is that SIP uses both TCP and UDP (User Datagram Protocol) as transport layers. TCP includes congestion control, whereas UDP does not, thereby opening the door for packet loss during times of network congestion. According to dynamicsoft's Rosenberg, the IETF will address these issues as the standard evolves."

[XMPP vs. SIMPLE: The race for messaging standards http://www.infoworld.com/article/03/05/23/21FExmpp_1.html]

Originally a PBX (private branch exchange) was mainly a telephone system within an enterprise that switches calls between enterprise users on local lines while allowing all users to share a certain number of external phone lines. The main purpose of a PBX was to save the cost of requiring a line for each user to the telephone company's central office. Most PBX's were made using proprietary protocols that disallowed using other vendors end devices (phones) or creating new services. Today's SIP based IP PBX's do a lot more and are part of converging a company's communication system with the data network. SIP based IP PBX's will be used on one hand to isolate and on the other to integrate a company's VoIP telephone service with the public internet. The need to isolate VoIP services arises from moving voice services to the data network which already today is isolated by most companies from the public Internet by an Intranet.

3.7 SIP Gateways

In order to enable calls and services from other networks to SIP networks (e.g. from PSTN-to-IP) gateways are necessary. In some cases also media transformation gateways are used.

IETF work for telephony services to interwork between the PSTN and the Internet:

- PINT: The PSTN/Internet Interfaces (PINT) (RFC2848) addresses connection arrangements through which Internet applications can request and enrich PSTN telephony services. An

* SIP for Instant Messaging and Presence Leveraging Extensions (SIMPLE) working group focuses on the application of SIP to the suite of services collectively known as Instant Messaging and Presence (IMP).

example of such services is a Web-based Yellow Pages service with the ability to initiate PSTN calls between customers and suppliers. (From IP to PSTN)

- SPIRITS: The Services in the PSTN/IN Requesting Internet Services (SPIRITS) addresses how services supported by IP network entities can be started from IN requests, as well as the protocol arrangements through which PSTN can request actions to be carried out in the IP network in response to events (IN Triggers) occurring within the PSTN/IN. (From PSTN to IP)

IETF work to address the transport of packet-based PSTN signaling over IP Networks:

- SIGTRAN: The primary purpose of this working group will be to address the transport of packet-based PSTN signaling over IP Networks, taking into account functional and performance requirements of the PSTN signaling.
- SIP-T: SIP for telephones (SIP-T) working group has written a best current practise document (RFC3372) for a mechanism that uses SIP to facilitate the interconnection of the PSTN with IP. This is intended to allow traditional IN-type services to be seamlessly handled in the Internet environment. It is essential that SS7 information be available at the points of PSTN interconnection to ensure transparency of features not otherwise supported in SIP. SS7 information should be available in its entirety and without any loss to the SIP network across the PSTN-IP interface. SIP-T defines SIP functions that map to ISUP interconnection requirements.

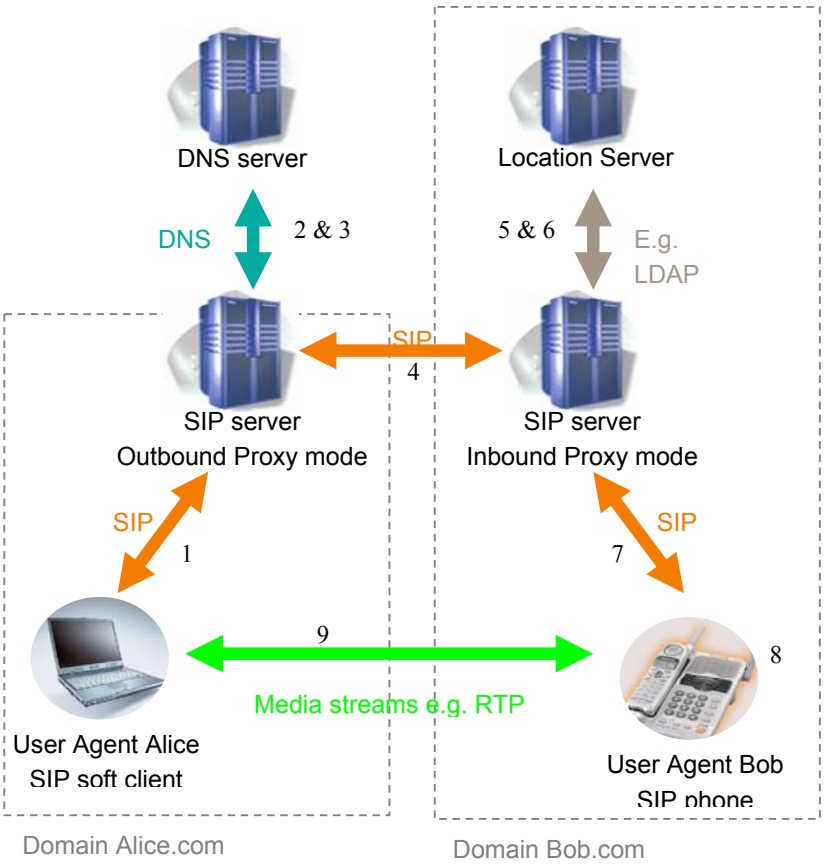
3.8 SIP Firewalls and NATs

As a core part of its functionality, SIP must carry around the ports, IP addresses and domain names needed to describe the sessions it controls. It also causes session traffic to be established (for example, RTP streams with audio and video), often on dynamic UDP ports. As such, there are two issues in getting SIP to traverse NATs and firewalls. The first is getting SIP itself through, and the second is getting the media sessions it initiates through. The latter is by far the harder problem. [draft-rosenberg-sip-firewalls-00.txt]

Newer SIP aware firewalls and NATs have been designed that support SIP and person-to-person communication.

4 SIP Session Setup Example

Figure 2 shows a typical example of a SIP message exchange between two users, Alice and Bob. In this example, Alice uses a SIP application on her PC (referred to as a softphone) to call Bob on his SIP phone over the Internet. Also shown are two SIP proxy servers that act on behalf of Alice and Bob to facilitate the session establishment. This typical arrangement is often referred to as the "SIP trapezoid".



1. Alice "Dials" Bob's SIP URI *sip:Bob@bob.com.* (UA Alice sends INVITE to outbound proxy server)
2. Outbound proxy makes a DNS query to find proxy server for bob.com domain
3. DNS responds with IP address of bob.com proxy server
4. Outbound proxy of alice.com domain sends INVITE to inbound proxy server of bob.com domain
5. Inbound Proxy consults Location Server to check what device Bob has registered with
6. Location Server responds with the SIP URI of Bob's SIP Phone
7. Inbound Proxy server forwards INVITE to Bob's SIP phone
8. Bob's SIP phone is ringing. Bob answers and a media session is established
9. Media Streams are transferred between Alice and Bob using e.g. RTP. In general the end-to-end media packets take a different path from the SIP signaling messages.

Figure 3 SIP Trapezoid

Alice "calls" Bob using his SIP identity, a type of Uniform Resource Identifier (URI) called a SIP URI. It has a similar form to an email address, typically containing a username and a host name. In this case, it is sip:bob@bob.com, where bob.com is the domain of Bob's SIP service provider. Alice has a SIP URI of sip:alice@alice.com.

SIP is based on an HTTP-like request/response transaction model. Each transaction consists of a request that invokes a particular method, or function, on the server and at least one response. In this example, the transaction begins with Alice's softphone sending an INVITE request addressed to

Bob's SIP URI. INVITE is an example of a SIP method that specifies the action that the requestor (Alice) wants the server (Bob) to take. The INVITE request contains a number of header fields. Header fields are named attributes that provide additional information about a message. The ones present in an INVITE include a unique identifier for the call, the destination address, Alice's address, and information about the type of session that Alice wishes to establish with Bob. The INVITE might look like this:

```
INVITE sip:bob@bob.com SIP/2.0
    Via: SIP/2.0/UDP pc33.alice.com;branch=z9hG4bK776asdhds
    Max-Forwards: 70
    To: Bob <sip:bob@bob.com>
    From: Alice <sip:alice@alice.com>;tag=1928301774
    Call-ID: a84b4c76e66710@pc33.alice.com
    CSeq: 314159 INVITE
    Contact: <sip:alice@pc33.alice.com>
    Content-Type: application/sdp
    Content-Length: 142

    (Alice's SDP not shown)
```

The first line of the text-encoded message contains the method name INVITE. The lines that follow are a list of header fields. This example contains a minimum required set. The complete set of SIP header fields is defined in RFC3261.

The details of the session, such as the type of media, codec, or sampling rate, are not described using SIP[*]. Rather, the body of a SIP message contains a description of the session, encoded in some other protocol format. One such format is the Session Description Protocol (SDP) [RFC 2327]. This SDP message is carried by the SIP message in a way that is analogous to a document attachment being carried by an email message, or a web page being carried in an HTTP message.

Since the softphone does not know the location of Bob or the SIP server in the bob.com domain, the softphone sends the INVITE to the SIP server that serves Alice's domain, alice.com. The address of the alice.com SIP server could have been configured in Alice's softphone, or it could have been discovered by DHCP, for example.

The alice.com SIP server is a type of SIP server known as a proxy server. A proxy server receives SIP requests and forwards them on behalf of the requestor. In this example, the proxy server receives the INVITE request and sends a 100 (Trying) response back to Alice's softphone. The 100 (Trying)

[*] SIP separates session establishment from session description. This separation enables using existing SIP infrastructure for providing new services. For example if a VoIP service provider wants to switch to being a gaming service provider it only needs to supply updated SIP UA's that use a e.g. proprietary gaming session description protocol instead of SDP. The service provider does not have to change existing SIP infrastructure. [SIP demystified page 113]

response indicates that the INVITE has been received and that the proxy is working on her behalf to route the INVITE to the destination. Responses in SIP use a three-digit code followed by a descriptive phrase. This response contains the same To, From, Call-ID, CSeq and branch parameter in the Via as the INVITE, which allows Alice's softphone to correlate this response to the sent INVITE. The alice.com proxy server locates the proxy server at bob.com, possibly by performing a particular type of DNS (Domain Name Service) lookup to find the SIP server that serves the bob.com domain. As a result, it obtains the IP address of the bob.com proxy server and forwards, or proxies, the INVITE request there. Before forwarding the request, the alice.com proxy server adds an additional Via header field value that contains its own address (the INVITE already contains Alice's address in the first Via). The bob.com proxy server receives the INVITE and responds with a 100 (Trying) response back to the alice.com proxy server to indicate that it has received the INVITE and is processing the request. The proxy server consults a database, generically called a location service that contains the current IP address of Bob. The bob.com proxy server adds another Via header field value with its own address to the INVITE and proxies it to Bob's SIP phone.

Bob's SIP phone receives the INVITE and alerts Bob to the incoming call from Alice so that Bob can decide whether to answer the call, that is, Bob's phone rings. Bob's SIP phone indicates this in a 180 (Ringing) response, which is routed back through the two proxies in the reverse direction. Each proxy uses the Via header field to determine where to send the response and removes its own address from the top. As a result, although DNS and location service lookups were required to route the initial INVITE, the 180 (Ringing) response can be returned to the caller without lookups or without state being maintained in the proxies. This also has the desirable property that each proxy that sees the INVITE will also see all responses to the INVITE.

When Alice's softphone receives the 180 (Ringing) response, it passes this information to Alice, perhaps using an audio ringback tone or by displaying a message on Alice's screen.

In this example, Bob decides to answer the call. When he picks up the handset, his SIP phone sends a 200 (OK) response to indicate that the call has been answered. The 200 (OK) contains a message body with the SDP media description of the type of session that Bob is willing to establish with Alice. As a result, there is a two-phase exchange of SDP messages: Alice sent one to Bob, and Bob sent one back to Alice. This two-phase exchange provides basic negotiation capabilities and is based on a simple offer/answer model of SDP exchange. If Bob did not wish to answer the call or was busy on another call, an error response would have been sent instead of the 200 (OK), which would have resulted in no media session being established.

In this case, the 200 (OK) is routed back through the two proxies and is received by Alice's softphone, which then stops the ringback tone and indicates that the call has been answered. Finally, Alice's softphone sends an acknowledgement message, ACK, to Bob's SIP phone to confirm the reception of the final response (200 (OK)). In this example, the ACK is sent directly from Alice's softphone to Bob's SIP phone, bypassing the two proxies. This occurs because the endpoints have learned each other's address from the Contact header fields through the INVITE/200 (OK) exchange, which was not known when the initial INVITE was sent. The lookups performed by the two proxies are no longer needed, so

the proxies drop out of the call flow. This completes the INVITE/200/ACK three-way handshake used to establish SIP sessions.

Alice and Bob's media session has now begun, and they send media packets using the format to which they agreed in the exchange of SDP. In general, the end-to-end media packets take a different path from the SIP signaling messages.

5 SIP Methods

Every SIP request contains a field called a *method*, which denotes its purpose. The core SIP specification (RFC 3261) defines six types of SIP methods: INVITE, ACK, CANCEL, REGISTER, BYE and OPTIONS. All SIP implementations support the six core methods. To add functionality beyond the core protocol, a set of extensions have been defined by the IETF. The extensions are ignored by SIP entities that do not support them; however the SIP user agents must support the used extensions or a session cannot be established.

This type of design allows also for industry specific application development on top of SIP. For example a payment application could use a method PAY (a proprietary method for SIP). The sending SIP UA (payee) creates a request that includes the method PAY and sends it to another SIP UA (payer) that supports the same proprietary PAY method. The SIP message then passes through the network with all network elements ignoring the proprietary method. Since the receiving SIP UA supports the method PAY, it accepts the SIP message and executes the PAY method. Depending on the implementation the receiving SIP UA then sends a response to the sending SIP UA.

The number of SIP method extensions is likely to increase as new application ideas constantly arise.

Extensions can also be header, body or parameter extensions.

For more information: Guidelines for Authors of Extensions to the Session Initiation Protocol (SIP)
http://www.ietf.org/internet-drafts/draft-ietf-sip-guidelines-06.txt

5.1 SIP Core Methods

The six core methods defined by RFC 3261:

For registering contact information (binding management):

1. REGISTER: *SIP offers a discovery capability. If a user wants to initiate a session with another user, SIP must discover the current host(s) at which the destination user is reachable. This discovery process is frequently accomplished by SIP network elements such as proxy servers and redirect servers which are responsible for receiving a request, determining where to send it based on knowledge of the location of the user, and then sending it there. To do this, SIP network elements consult an abstract service known as a location*

service, which provides address bindings for a particular domain. REGISTER requests add, remove, and query bindings. A REGISTER request can add a new binding between an address-of-record and one or more contact addresses. Registration on behalf of a particular address-of-record can be performed by a suitably authorized third party. A client can also remove previous bindings or query to determine which bindings are currently in place for an address-of-record.

For setting up sessions:

2. [*]INVITE: *When a user agent client desires to initiate a session (for example, audio, video, or a game), it formulates an INVITE request. The INVITE request asks a server to establish a session. This request may be forwarded by proxies, eventually arriving at one or more UAS that can potentially accept the invitation. These UASs will frequently need to query the user about whether to accept the invitation.*

3. ACK: ACK requests are used to acknowledge the reception of a final response to an INVITE. (SIP has three-way handshake: INVITE-final response-ACK)

4. CANCEL: *The CANCEL request, as the name implies, is used to cancel a previous request sent by a client. Specifically, it asks the UAS to cease processing the request and to generate an error response to that request. CANCEL has no effect on a request to which a UAS has already given a final response. Because of this, it is most useful to CANCEL requests to which it can take a server long time to respond. For this reason, CANCEL is best for INVITE requests, which can take a long time to generate a response. In that usage, a UAS that receives a CANCEL request for an INVITE, but has not yet sent a final response, would "stop ringing", and then respond to the INVITE with a specific error response.*

For terminating sessions:

5. BYE: *The BYE request is used to terminate a specific session or attempted session.*

For querying servers about their capabilities:

6. OPTIONS: *The SIP method OPTIONS allows a UA to query another UA or a proxy server as to its capabilities. This allows a client to discover information about the supported methods, content types, extensions, codecs, etc. without "ringing" the other party. For example, before a client inserts a*

[*] For modifying an existing session a re-INVITE can be used. *This modification can involve changing addresses or ports, adding a media stream, deleting a media stream, and so on. This is accomplished by sending a new INVITE request within the same dialog that established the session. An INVITE request sent within an existing dialog is known as a re-INVITE. Note that a single re-INVITE can modify the dialog and the parameters of the session at the same time. Either the caller or callee can modify an existing session.*

Require header field into an INVITE listing an option that it is not certain the destination UAS supports, the client can query the destination UAS with an OPTIONS to see if this option is returned in a Supported header field. All UAs MUST support the OPTIONS method.

5.2 SIP Method Extensions

Below are SIP method extensions, which have been documented in standards track RFC's:

Session transfer:

1. REFER (RFC3515): The REFER method is a mechanism to transfer SIP sessions to another entity e.g. a secretary transferring a call

Session setup and negotiation:

2. INFO (RFC2976): The INFO method is used for communicating mid-session signaling information along the signaling path for the call e.g. billing information

3. PRACK (RFC3262): PRACK provides reliable provisional response messages

4. UPDATE (RFC3311): UPDATE allows a client to update parameters of a session (such as the set of media streams and their codecs) but has no impact on the state of a dialog. In that sense, it is like a re-INVITE, but unlike re-INVITE, it can be sent before the initial INVITE has been completed. This makes it very useful for updating session parameters within early dialogs.

The ability to request asynchronous notification of events proves useful in many types of SIP services for which cooperation between end-nodes is required. Examples of such services include automatic callback services (based on terminal state events), buddy lists (based on user presence events), message waiting indications (based on mailbox state change events), and PSTN and Internet Internetworking (PINT) status (based on call state events).

The general concept is that entities in the network can subscribe to resource or call state for various resources or calls in the network, and those entities (or entities acting on their behalf) can send notifications when those states change.

Event notification:

5. SUBSCRIBE (RFC3265): The SUBSCRIBE method is used to request current state and state updates from a remote node.
6. NOTIFY (RFC3265): NOTIFY messages are sent to inform subscribers of changes in state to which the subscriber has a subscription. Subscriptions are typically put in place using the SUBSCRIBE method; however, it is possible that other means have been used.

Page-mode message delivery:

7. MESSAGE (RFC3428): The MESSAGE method, an extension to SIP that allows the transfer of Instant Messages. Since the MESSAGE request is an extension to SIP, it inherits all the request routing and security features of that protocol. MESSAGE requests carry the content in the form of MIME body parts. MESSAGE requests do not themselves initiate a SIP dialog; under normal usage each Instant Message stands alone, much like pager messages. MESSAGE requests may be sent in the context of a dialog initiated by some other SIP request.

6 SIP Mobility Modes

Based on Henning Schulzrinne's slides: "Service Mobility", Henning Schulzrinne (with Stefan Berger, Jonathan Lennox, Xiaotao Wu), Columbia University, SIP 2003 – January 2003, Paris, France.

*Mobility is more than cell phones.

6.1 Terminal Mobility

Terminal mobility refers to the traditional mobility that one thinks of when speaking about a mobile phone. The terminal itself is mobile and can move around a geographical space and connect to multiple network attachment points using the same identifier, i.e. a cell phone can move from one cell to another without disconnecting a call.

Terminal mobility can be implemented in different protocol layers. Mobile IP provides a network layer mobility solution that hides address changes from protocols running on top of the network layer (IP). MobileIP does this by providing two IP addresses: one constant address as the end point identifier, and one temporary care-of address for the location of the terminal. SIP on the other hand provides application layer mobility, where the application itself detects the movement of the device and adapts to the new location and can maintain connectivity even if the IP address of the device is changed. SIP handles terminal mobility by modifying (i.e. changing the IP address of) an existing session with the SIP registration mechanism and the re-INVITE method. The SIP registration mechanism binds a user-level identifier to a temporary IP address or host name instead of a permanent IP address.

6.2 User Mobility

User mobility means that the user is reachable under the same identifier on several networks, at several terminals, possibly at the same time. In other words a user can be reached on any device that has been registered into a location service by the use via a single SIP address, making the user's

* Terminal mobility is sometimes referred to as Device mobility. User mobility is sometimes referred to as Personal mobility. Service mobility is sometimes referred to as Application mobility. User mobility is sometimes split into Terminal and Personal mobility.

device choice transparent to third parties that are trying to reach them. This means that the user is able to move across different terminal devices and be reached with the same logical address, the SIP address.

For example a user may want to be reachable via a traditional PSTN phone, a PC and a wireless device. The user may want to use these devices either at the same time or alternate between them. The user may want to use different devices for different purposes, e.g., for private and professional communication. The SIP address incorporates all addresses under it and thus enables user mobility. The user is contacted on the device, in the location and with the interaction mode, that the user has registered himself with for that moment as a preference. SIP proxy servers, redirect severs and registrars handle user mobility.

6.3 Service Mobility

Service mobility is when the user is able to access the same services, independent of device and network used. Example Services are: user address book, speed dial entries and buddy lists, to name a few.

Service mobility is a desirable feature for networks that enable user mobility. Since user mobility assumes that the end user uses multiple devices (personal and non-personal), it follows that the user should have the same services available for all used devices.

It still remains unclear how to implement service mobility in SIP networks. Mostly only requirements of service mobility have been made. Service mobility for the user in GSM networks is handled by the SIM card. However, the SIM card is not suited for multiple networks and is difficult to move from one device to another. Service mobility needs to be device independent and will most likely be implemented so that the users services remain on a server. It should also be possible to update these service definitions from any terminal, without having to then explicitly synchronize them.

The use of non-personal devices (e.g. public phone booths), also enabled by user mobility, to access personal services raises need for secure access devices to services database. Many different options have been discussed ranging from small portable identification tokens to one-time passwords to ensure secure access to the service database. Also it is still unclear what protocol should be used, whether it should be SIP, LDAP or something completely new.

6.4 Session Mobility

Session mobility means that it is possible to move an on-going session to new terminal(s) without terminating the session. Session mobility also incorporates the ability to split a session across end devices into one collaborative application e.g. wall display, cell phone and pc (softphone).

Session mobility is handled in SIP with the REFER method similarly to call transfer. SIP only needs to know that the entity of the new device is the same entity as it was for the previous address (not who the person really is). To initiate the session transfer a REFER request is sent that indicates the new

address were the session is moved to. The receiver of the REFER request then negotiates a new session to the new address using the normal INVITE exchange. If the session is to be split across multiple participants, each participant must be invited separately.

Because different terminals have different capabilities, it is often desirable to be able to also modify sessions after/during session transfer. For example additional media (e.g. video streaming) could be added to a call after moving the call from a cell phone to a PC softphone. The Cell phone might have not supported video streaming or the user might have forgotten his earphones at home (which would make speaking and viewing difficult) or the user might simply not have been willing to pay for video over the mobile network. The session can be modified with the SIP MODIFY method.

Another example of session mobility would be that a user expands an existing session by adding new media and a new device(s). For example an on-going call on a cell phone is split to also include a PC video application. This is done by sending an INVITE to the PC video application address. The session is then modified to include a video stream to the PC video application address. The user can then speak on the cell phone and view the called party from the PC video application.

NOTE! To move an authenticated web bank session to another device is an HTTP issue not SIP, because web session are not established by SIP. For moving HTTP sessions check PPK (private-public key) that was mostly designed for Mobile IPv6.

7 Something to Think About

Below are some example use cases that SIP might be used for. The examples have not been tested and might not work with SIP and might even have no sense or beef in them. The examples are listed here to give something to think about!

7.1 SIP, SMS and MMS

SMS messages use the SS7 signaling path in GSM networks. SIP will replace SS7 in 3G networks as the primary signaling protocol. Therefore all services and applications that have been implemented with SMS can be transferred almost directly to SIP. SIP will enable additional capabilities and remove the 160 character limitation. The price of future SMS type services based on SIP will be lower, because ISP's will be able to offer the service also, since no SMS center will be needed. This also applies for MMS.

MMS is currently implemented using WAP. SIP infrastructure will replace the need for WAP to deliver MMS messages.

7.2 SIP Payload

SIP distinguishes between session establishment and session description. Session establishment is to locate the user, but not what the session is for. SIP does not define how the session should be described or session types. For example SIP together with SDP (Session Description Protocol RFC

2327) can be used to establish VoIP sessions. SIP message content is ignored by SIP serves in the network and it is only the SIP UA's that uses the content of the SIP payload. Therefore it is possible to transport any data in the SIP signaling plane, only the SIP UA's must be able to receive the SIP payload.

SIP Messages are routed in much the same way as e-mail messages. They can also carry multipart message bodies using MIME (RFC 2045). It is important to remember that SIP is not good at transporting large amounts of data since it is not a designed to be a transport protocol. However it is good for delivering instant messages, which are small by definition and probably urgent, and intended to reach the users at their present location. [SIP demystified, page 113]

In other words it might be possible to deliver anything in the SIP payload. One SIP concern is that it will be used to also deliver content for free and certain billing concerns have been raised since signaling is usually free to the user especially if the "call" is not answered (paging mode).

Examples to think about:

1. Deliver small applications e.g. Midlets using SIP:
   The small application could be inserted into the SIP payload.
2. SIP payment: proprietary encryption of payload:
   SIP could maybe be used as a payment (digital cash). The payload could contain information that adds value for the user. The SIP payment could be encrypted in the payload with a proprietary method. Also a proprietary SIP extension could be created for the header field. Encryption could also be done similarly as S/MIME (which is suggested as SIP security method) so that the whole SIP message is proprietarily encrypted and then encapsulated into another SIP message.

7.3 SIP API for J2ME

The SIP API for J2ME can be used always when a Midlet needs a connection somewhere. It can be necessary to establish a session to other applications, to a server or to other devices that support the SIP API for J2ME.

The connection established with the SIP API could be used for example to update Midlets.

7.4 SIP for Device Capability Discovery

SIP is by default able to discover device capabilities during session setup, since SIP is a signaling protocol that uses other protocols to negotiate a session. During session negotiation the sender and receiver agree upon parameters to be used during the session e.g.:

- supported technologies
- supported media
- if and what want to receive if anything
- size limits, etc.

Example by Jonathan Rosenberg (Dynamicsoft):"SIP and MMS":

- SIP for Presence solves the user-level problems with MMS:
    - whether recipient supports MMS
    - what media types they support
    - whether they want to receive an MMS or not
    - size limits

7.5 Virtual Safety Deposit Boxes

User and Service mobility might create a need for virtual safety deposit boxes that store personal services (phone book, IM buddy list and personal small productivity applications) that can be accessed from any device using some pre-established or federated identity.

There are two main requirements of service mobility: 1) Access to personal services from any device with same up-to-date services for all devices. (It might also be practical to be able to run certain applications from the safe storage so that the user is not forced to install the application each time using a new device.) and 2) The services are personal and are only available for the owner of the services.

Financial Institutions could prove to be a natural provider of this kind of service, since they are considered in general as trusted parties and already most financial institutions provide multi-channel access to existing services with high concern on security. The Financial Institution could use already existing device independent authentication infrastructure to recognize the user from any device and allow access to services based on this authentication.

7.6 Distributed Payment

Distributed payment refers to a payment scheme were the user of a service does not necessarily pay for the service or pays for only parts of the service. SIP could enable distributed payment. For example:

- "You pay for audio, I'll pay for video"
- "I'll pay for service you pay for bandwidth"

From the Financial Institutions point of view this could be thought of as a "third party payment", were the financial institution is the third party and handles payment between the other two parties. Maybe one could have a personal SIP entity that accepts all service charges e.g. the entity could be analogous to a credit card. For example:

- "I'll buy this and "my credit card" will pay for it"

"My Credit card" being a SIP entity invited to the session by the purchasing party that represents the Financial Institution on behalf of the purchasing party to the merchant.

At the moment there is no standardized SIP extension for distributed payment. The extension could be easily made and has been discussed, but nobody has needed it. The extensions could be one line e.g. re-direct payment

Distributed payments have also marketing possibilities: “Coca-cola will pay for your calls if you do this”.

Distributed payment could maybe be used to separate service billing from device/address being used e.g. to separate voice charges from service charges or separate user from device being used.

7.7 Conferencing

Inviting users to Mbone session was the original purpose of SIP when it was first commissioned. The protocol has evolved steadily and SIP is currently used to invite users to all types of sessions, including multicast and point-to-point sessions.

Maybe banking and investment could be thought of in terms of conferencing, i.e. the user, financial institution and other related parties together in a conference?

7.8 Unified Messaging

SIP is a good platform for service creation. SIP is perfect protocol to use to combine different services for the user. The similarities with between SIP and HTTP and SMTP make it easy to combine web and e-mail Internet services together with multimedia. SIP not only integrates services, but it also delivers them to the user’s real location. SIP applications integrate Web browsing, e-mail, voice calls, videoconferencing, presence information, and instant messages into unified messaging. [SIP demystified, page 110]

The ability to combine several services and applications together makes SIP ideal for Branch and Call Center service creation.

8 SIP Bits from the Press

SIP Center New Letter August 2003:
A few juicy SIP bits form the press over the last month:

"Today few people know about SIP, but it will fundamentally change the telephone business, just as MP3 has done with the music industry." SIPphone CEO, Michael Robertson.
"The Session Initiation Protocol (SIP) has become a standard supported by almost all of the hardware and software makers in some fashion." Glenn Fleishman, The Seattle Times.

"For Release 5 networks, handsets will be based on the Session Initiation and Session Description Protocols (SIP/SDP)." David Myers, Co-founder and Vice President of Engineering, Dilithium Networks.

"The cable companies are starting to offer phone service and it is clear they will use SIP-based IP telephony." Dave Passmore, Analyst, Burton Group.

"IDC indicated that revenues from packet-based enhanced services worldwide will triple between 2002 and 2004 to $9.8 billion." Ofer Shem Tove, Internet Telephony.

"Vonage, the leading provider of broadband telephony, today announced the completion of more than 40 million calls over its Session Initiation Protocol (SIP) network." - Vonage.

"Hotel Commonwealth in Boston opened last month with an IP network infrastructure that supports voice and text messaging to in-hotel wireless phones and other interactive applications for guests, all relying on the Session Initiation Protocol (SIP)." - Alcatel

9 For More Information

9.1 Books

Camarillo, G., “SIP Demystified”, 2002, ISBN 0-07-137340-3
Sinnreich, H., Johnston, A., “Internet Communications Using SIP”, 2001, ISBN 0-471-41399-2
Johnston, A., “Understanding the Session Initiation Protocol”, 2001, ISBN 1-58053-168-7

9.2 Web sites

SIP Forum http://www.sipforum.org
SIP Center http://www.sipcenter.com
Pulver.com http://www.pulver.com/

9.3 IETF

9.3.1 SIP WG (http://www.ietf.org/html.charters/sip-charter.html)

Internet-Drafts:

- Session Timers in the Session Initiation Protocol (SIP)
- Caller Preferences for the Session Initiation Protocol (SIP)
- Management Information Base for Session Initiation Protocol (SIP)
- Guidelines for Authors of Extensions to the Session Initiation Protocol (SIP)
- The Stream Control Transmission Protocol as a Transport for for the Session Initiation Protocol
- Internet Media Types message/sipfrag
- The Session Inititation Protocol (SIP) 'Replaces' Header
- The SIP Referred-By Mechanism
- Compressing the Session Initiation Protocol

- Session Initiation Protocol Extension Header Field for Service Route Discovery During Registration
- Session Initiation Protocol Extension to Assure Congestion Safety
- A Mechanism for Content Indirection in Session Initiation Protocol (SIP) Messages
- An Extension to the Session Initiation Protocol (SIP) for Symmetric Response Routing
- The Session Inititation Protocol (SIP) 'Join' Header (38455 bytes)
- Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)
- SIP Authenticated Identity Body (AIB) Format
- S/MIME AES Requirement for SIP
- An Extension to the Session Initiation Protocol for Request History Information
- Communications Resource Priority for the Session Initiation Protocol (SIP)
- Indicating User Agent Capabilities in the Session Initiation Protocol (SIP)

Request For Comments:

- RFC 2976: The SIP INFO Method
- RFC 3204: MIME media types for ISUP and QSIG Objects
- RFC 3261: SIP: Session Initiation Protocol (RFC 3261)
- RFC 3262: Reliability of Provisional Responses in SIP
- RFC 3263: SIP: Locating SIP Servers
- RFC 3265: SIP-Specific Event Notification
- RFC 3361: DHCP Option for SIP Servers
- RFC 3310: Hypertext Transfer Protocol (HTTP) Digest Authentication Using Authentication and Key Agreement (AKA)
- RFC 3311: The Session Initiation Protocol UPDATE Method
- RFC 3312: Integration of Resource Management and SIP
- RFC 3420: Internet Media Type message/sipfrag
- RFC 3323: A Privacy Mechanism for the Session Initiation Protocol (SIP)
- RFC 3325: Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks
- RFC 3428: Session Initiation Protocol Extension for Instant Messaging
- RFC 3326: The Reason Header Field for the Session Initiation Protocol (SIP)
- RFC 3327: Session Initiation Protocol Extension for Registering Non-Adjacent Contacts
- RFC 3329: Security Mechanism Agreement for the Session Initiation Protocol (SIP)
- RFC 3313: Private Session Initiation Protocol (SIP)Extensions for Media Authorization
- RFC 3515: The Session Initiation Protocol (SIP) Refer Method
- RFC 3319: Dynamic Host Configuration Protocol (DHCPv6)Options for Session Initiation Protocol (SIP) Servers

Additional SIP Pages references by SIP WG (http://www.softarmor.com/sipwg/)

- Henning Schulzrinne's Marvelous SIP Page
  http://www.cs.columbia.edu/~hgs/sip/
- SIPIT-10 Slides from Robert Sparks
  http://www.softarmor.com/sipwg/references/SIPIT10_NEWRFC.ppt

- Changes in SIP from bis09 to RFC3261 -- Jonathan Rosenberg
  http://www.softarmor.com/sipwg/references/changes-bis09-to-3261.html
- Mailing List: SIP List - Official SIP WG Mail List
  http://www.ietf.org/mailman/listinfo/sip

9.3.2 SIPPING WG (http://www.ietf.org/html.charters/sipping-charter.html)

Internet-Drafts:

- Using ENUM for SIP Applications
- Mapping of of Integrated Services Digital Network (ISUP) Overlap Signalling to the Session Initiation Protocol
- Session Initiation Protocol Service Examples
- SIP Support for Real-time Fax: Call Flow Examples And Best Current Practices
- A Call Control and Multi-party usage framework for the Session Initiation Protocol (SIP)
- Best Current Practices for Third Party Call Control in the Session Initiation Protocol
- A Message Summary and Message Waiting Indication Event Package for the Session Initiation Protocol (SIP)
- Requirements for Content Indirection in Session Initiation Protocol (SIP) Messages
- An INVITE Inititiated Dialog Event Package for the Session Initiation Protocol (SIP)
- A Session Initiation Protocol (SIP) Event Package for Conference State
- Session Initiation Protocol PSTN Call Flows
- Session Initiation Protocol Basic Call Flow Examples
- Session Initiation Protocol Torture Test Messages
- SIP Generic Request History Capability – Requirements
- Authentication, Authorization and Accounting Requirements for the Session Initiation Protocol
- 3rd-Generation Partnership Project (3GPP) Release 5 requirements on the Session Initiation Protocol (SIP)
- Session Initiation Protocol Call Control - Transfer
- Requirements for Connection Reuse in the Session Initiation Protocol (SIP)
- A Session Initiation Protocol (SIP) Event Package for Registrations
- Interworking between SIP and QSIG
- Requirements for SIP User Agent Profile Delivery Framework
- A Framework for SIP User Agent Configuration
- Session Initiation Protocol Call Control - Conferencing for User Agents
- High Level Requirements for Tightly Coupled SIP Conferencing
- A Framework for Conferencing with the Session Initiation Protocol
- Requirements for Session Policy for the Session Initiation Protocol (SIP)
- Guidelines for Usage of the Session Initiation Protocol (SIP) Caller Preferences Extension

Request For Comments:

- RFC 3351: User Requirements for the Session Initiation Protocol (SIP) in Support of Deaf, Hard of Hearing and Speech-impaired individuals

- RFC 3372: Session Initiation Protocol (SIP) for Telephones (SIP-T): Context and Architectures RFC 3324: Short Term Requirements for Network Asserted Identity)
- RFC 3398: Integrated Services Digital Network (ISDN) User Part (ISUP) to Session Initiation Protocol (SIP) Mapping
- RFC 3485: The Session Initiation Protocol (SIP) and Session Description Protocol (SDP) Static Dictionary for Signaling Compression (SigComp)

Additional SIP Pages references by SIPPING WG (http://softarmor.com/sipping)
ITU Communications:

- Liason statement from ITU SG11 on interworking with BICC and ISUP html http://softarmor.com/sipping/references/itu-sg11-bicc.html
- NWB-059 Draft Recomendationn Q.SIPPROF doc http://softarmor.com/sipping/references/nwb059.doc
- NWB-087 baseline text for proposed new Recommendation Q.1912.SIP doc http://softarmor.com/sipping/references/nwb087.doc
- Mailing List SIPPING List - Official SIPPING WG Mail List http://www.ietf.org/mailman/listinfo/sipping

9.3.3 SIMPLE WG (http://www.ietf.org/html.charters/simple-charter.html)

Internet-Drafts:

- A Presence Event Package for the Session Initiation Protocol (SIP)
- An Extensible Markup Language (XML) Based Format for Watcher Information
- A Watcher Information Event Template-Package for the Session Initiation Protocol (SIP)
- Requirements for Manipulation of Data Elements in Session Initiation Protocol (SIP) for Instant Messaging and Presence Leveraging Extensions (SIMPLE) Systems
- SIMPLE Presence Publication Requirements
- A Session Initiation Protocol (SIP) Event Notification Extension for Resource Lists
- Session Initiation Protocol (SIP) Extension for Presence Publication
- Requirements for Efficient Delivery of Presence Information
- Requirements for Filtering of Watcher Information
- Instant Message Sessions in SIMPLE
- Requirements for Presence Specific Event Notification Filtering
- Extensible Markup Language (XML) Configuration Access Protocol (XCAP)Usages for Setting Presence Authorization
- The Extensible Markup Language (XML) Configuration Access Protocol (XCAP)
- A Session Initiation Protocol (SIP) Event Package for Modification Events for the Extensible Markup Language (XML) Configuration Access Protocol (XCAP) Managed Documents
- An Extensible Markup Language (XML) Configuration Access Protocol (XCAP) Usage for Presence Lists
- RPID -- Rich Presence Information Data Format

Request For Comments:

- No Request For Comments