Internet Draft
Document: draft-watson-sipping-req-history-02.txt

Mark Watson
Mary Barnes
Nortel Networks
Cullen Jennings
Cisco
Jon Peterson
NeuStar

Category: Informational
Expires December 2002

June 2002

Generic Request History Capability û Requirements

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at
http://www.ietf.org/ietf/1id-abstracts.txt
The list of Internet-Draft Shadow Directories can be accessed at
http://www.ietf.org/shadow.html.

Abstract

Many services that SIP is anticipated to support require the ability to determine why and how the call arrived at a specific application. Examples of such services include (but are not limited to) sessions initiated to call centers via "click to talk" SIP URLs on a web page, "call history/logging" style services within intelligent "call management" software for SIP UAs and calls to voicemail servers and call centers. While SIP implicitly provides the redirect/retarget capabilities that enable calls to be routed to chosen applications, there is currently no standard mechanism within SIP for communicating the history of such a request. This "request history" information allows the receiving application to determine hints about how and why the call arrived at the application/user.

This draft discusses the motivations in support of a mechanism which records the "request history" and proposes detailed requirements for such a generic "request history" capability.

Table of Contents

1. **Introduction:** Why define a Generic "Request History" capability?

SIP implicitly provides redirect/retarget capabilities that enable calls to be routed to specific applications as defined in [1]. The term retarget will be used henceforth in this draft to refer to the process of a Proxy Server/UAC changing a URI in a request and thus changing the target of the request. This term is chosen to avoid associating this request history only with the specific SIP Redirect Server capability that provides for a response to be sent back to a UAC requesting that the UAC should retarget the original request to an alternate URI. The rules for determining request targets as described in section 16.5 of [1] are believed to be consistent with the use of the retarget term in this draft.

The motivation for the request history is that in the process of retargeting old routing information can be forever lost. This lost information may be important history that allows elements to which the call is retargeted to process the call in a locally defined, application specific manner. The proposal in this draft is to provide a mechanism for transporting the request history. It is not proposing any behavior for a Proxy or UA upon receipt of the information. Indeed, such behavior should be a local decision for the recipient application.

Current network applications provide the ability for elements involved with the call to exchange additional information relating to how and why the call was routed to a particular destination. The following are examples of such applications:

1) Web "referral" applications, whereby an application residing within a web server determines that a visitor to a website has arrived at the site via an "associate" site which will receive some "referral" commission for generating this traffic,

2) Email forwarding whereby the forwarded-to user obtains a "history" of who sent the email to whom and at what time

3) Traditional telephony based call redirection services such as Voicemail, call-center "automatic call distribution", and "follow-me" style services.

Several of the aforementioned applications, and specifically those applications based on email or WWW, define application specific mechanisms through which it is possible to obtain the necessary history information.

In order to prevent differing proprietary mechanisms emerging to obtain the required "request history" information, it is proposed that the SIPPING WG evaluate the requirements and determine a generic mechanism for the transport of such "request history" information.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119.

3. "Request History" Requirements

The following list constitutes a set of requirements for a "Request History" capability. Note that some of these requirements may be met using existing elements within SIP û whether and what SIP extensions would be needed to meet these requirements is out of scope of this draft.

The requirements have been enumerated and tagged to facilitate reference to each requirement:

1) CAPABILITY-req: The "Request History" capability will provide a capability to inform proxies and UAs involved in processing a request about the history/progress of that request. While this is inherently provided when the retarget is in response to a SIP redirect, it is deemed useful for non-redirect retargeting scenarios, as well.

2) GENERATION-req: "Request History" information is generated when the request is retargetted [see section 4.1 for further discussion of this requirement].

3) ISSUER-req: "Request History" information can be generated by a UA, proxy or redirect server. It can be passed in both requests and responses.

4) CONTENT-req: The "Request History" information for each occurrence of retargeting, shall include the following:

4.1) The new URI or address to which the request is in the process of being retargeted

4.2) The URI or address from which the request was retargeted.

4.3) The reason for the Request-URI modification [See section 4.2 for further description of this requirement].

4.4) Chronological ordering of the Request History information.

5) REQUEST-VALIDITY-req: Request-History is applicable to requests not sent within an established dialog. (i.e. INVITE, REGISTER, MESSAGE, and OPTIONS).

6) BACKWARDS-req: Request-History information may be passed from the generating entity backwards towards the UAC. This is needed to enable services which inform the calling party about the dialog establishment attempts.

7) FORWARDS-req: Request-History information may also be included by the generating entity in the request, if it is forwarded onwards.

8) REDIRECT-RESP-req: An entity (UA or proxy) retargeting in response to a redirect or REFER shall include any Request History information from the redirect/REFER in the new request.

4. Further Requirements Related Considerations

This section of the document further addresses some concerns that arise out of the Requirements specification in section 3.

4.1 Further considerations for capturing retargeting

The original request URI of a retargeted request SHOULD identify the user, service or resource, which performed the retargeting, as captured in requirement 4.2 in section 3. In some scenarios, it might be possible for more than one instance of retargeting to occur within the same Proxy. It is recommended that a proxy SHOULD NOT 'internally retarget' a request to a different user, service or resource on the same proxy, without generating Request History information for the 'internal retargeting' as well. It should be highlighted that an underlying requirement is to ensure that any retargeting maintains the privacy associated with the original Request URI. This requirement is addressed, along with additional security specific requirements in Section 5.

4.2 Reason for retargeting

The reason for the retargeting is only known to the application performing the retargeting. However, it does make sense to define a set of reasons, which will be commonly required. It is proposed that [6] provides a reasonable starting point for the definition for the set of reasons.

4.3 Optionality of the "Request History" capability

Requirement 2 in section 3 specifies that "Request History" information is generated when the request is retargeted. In many cases, it is anticipated that whether the history is added to the Request would be a local policy decision enforced by the specific application, thus no specific protocol element is needed. However, due to the capability being "optional" from the SIP protocol perspective, the impact to an application of not having the "Request History" must be described. For example, in a scenario where there is sequential forking and retargeting, some of the destinations previously tried could be retried. The impact of not having the "Request History" information for this sample application is that routing is inefficient. However, another scenario involving a voicemail application, the impact of not having the "Request History" information would be the service could not operate without having the information as to why the call was retargeted and the initial target for the call. Thus, the expectation would be that the policy in a system that intended to support this voicemail application would have to require the entities within its domain which are capable of retargeting to capture "Request History" information. Appendix A of this document in section 8 provides further details of these examples.

5. Security Considerations

The Request History information is being inserted by a network element retargeting a Request, resulting in a slightly different problem than the basic SIP header problem, thus requiring specific consideration. In addition, there may be privacy implications associated with some of the Request History information.

The potential security problems introduced include the following:
1) A rogue application could insert a bogus Request History entry either by adding an additional entry as a result of retargeting or entering invalid information.

2) A rogue application could delete an entry added by a previous retargeting. While this may be a valid scenario for some

applications, this may indicate a loss of integrity of the Request History content, which could significantly impact other applications.

3) Loss of privacy associated with forwarding a specific Request URI in the Request History.

4) A rogue application could re-arrange the Request History information to change the nature of the end application or to mislead the receiver of the information.

Thus, any solution to "Request History" capability must meet the following requirements:

1) SEC-req-1: The entity receiving the Request History must be able to determine whether any of the previously added Request History content has been altered.

2) SEC-req-2: The ordering of the Request History information must be preserved at each instance of retargeting.

3) SEC-req-3: The entity receiving the Request History must be able to determine whether a previously added Request History content has been removed.

4) SEC-req-4: The entity receiving the information conveyed by the Request History must be able to authenticate the source of the information.

It is likely that the solutions to several of the requirements are inter-related. For example, with the requirement for Chronological ordering [Requirement 4.4 in section 3], it is likely that the solution to SEC-req-1 would also meet SEC-req-2. Following on this, if SEC-req-2 is met, then SEC-req-3 could make use of the Chronological ordering to detect if information had been removed.

It should also be noted that these requirements apply to any entity making use of the Request History information, either by retargeting and capturing the information, or as an application making use of the information in a Request or Response. However, to ensure the overall integrity of this information as it traverses the network, an additional requirement with regards to the security of the transport is introduced:

5) SEC-req-5: To ensure the overall integrity of the chain of Request History information, the transport must be secure.

In addition, there are general privacy requirements that MUST be met:

6) PRIV-req-1: The entity retargeting the Request must ensure that it maintains the privacy (as described in [7]) associated with the original Request URI which is retargeted.

7) PRIV-req-2: The entity receiving the Request History must maintain the privacy associated with the information.

It is recognized that meeting the privacy requirements may impact the functionality of this solution. The applicability guidelines for a solution must clearly address this impact.

6. Going forward

The authors request that the SIPPING WG study this contribution and come to consensus regarding the set of requirements necessary for a Generic Request History mechanism. A next step is proposed to document the analysis of the various mechanisms proposed for this problem domain [2][3][4] and [5] and determine the extent to which these meet the agreed requirements. Such an analysis would thus provide suitable grounds for determining what extensions are necessary to SIP in order to support the agreed requirements.

In addition, it is proposed that further analysis of the requirements resulting in a solution would include the following:

1) Further analysis of the security requirements and potential solutions. The solution to some of the security requirements appears to be in the same problem domain as the security requirements for the Referredby header [10] and further analysis is required to determine if this is case and whether there is potential for synergy in the security solutions.

2) Further scenarios, highlighting in more detail some of the issues that will be encountered due to the optionality of the "Request History" capability. This will enable the solution documentation to provide more explicit guidelines on the applicability of the solution.

7. IANA Considerations

This document does not have any implications for IANA.

References

[1] J. Rosenberg et al, "SIP: Session initiation protocol," draft-ietf-sip-rfc2543bis-09.txt, February 27th, 2002.

[2] B. Campbell, R. Sparks, "Control of Service Context using SIP Request-URI", RFC 3087, April 2001.

[3] S. Levy, B. Byerly, J. Yang, "Diversion Indication in SIP", draft-levy-sip-diversion-03.txt, November, 2001.

[4] W. Marshall et al, "SIP Extensions for Caller Identity and th Privacy", draft-ietf-sip-privacy-04.txt, February 27 , 2002.

[5] D. Oran, H. Schulzrinne, "SIP extension for tracking locations attempted", oran-sip-visited-00.txt, August 6, 2000.

[6] H. Schulzrinne, D. Oran, G. Camarillo, "The Reason Header Field for the Session Initiation Protocol", draft-schulzrinne-sip-reason-th 01.txt, February, 28 , 2002.

[7] J. Peterson, "SIP Privacy", draft-ietf-sip-privacy-general-01.txt, June, 2002.

[8] R. Sparks, "The SIP Referredby Header Field", draft-ietf-sip-referredby-00.txt, May, 2002.

Contributors

Robert Sparks contributed excellent feedback and direction for the Security considerations section of this document. In addition, he highlighted the importance of addressing the optionality aspects of the "Request History" capability.

Acknowledgments

The authors would like to thank Chris Hogg for serving as the editor for the initial (-00) version of this draft. In addition, Sanjoy Sen provided useful comments and suggestions related to this draft.

AuthorsÆ Addresses

Mark Watson
Nortel Networks (UK)
Maidenhead Office Park (Bray House)
Westacott Way
Maidenhead,
Berkshire Tel: +44 (0)1628-434456
England Email: mwatson@nortelnetworks.com

Mary Barnes
Nortel Networks Tel: +1 972-684-5432
Richardson, Texas Email: mbarnes@nortelnetworks.com

Jon Peterson
NeuStar, Inc.
1800 Sutter Street, Suite 570
Concord, CA 94520 Email: Jon.Peterson@NeuStar.com

Cullen Jennings
Cisco Systems
170 West Tasman Dr Tel: +1 408 527 9132
MS: SJC-21/3 Email: fluffy@cisco.com

8.Appendix A - Scenarios

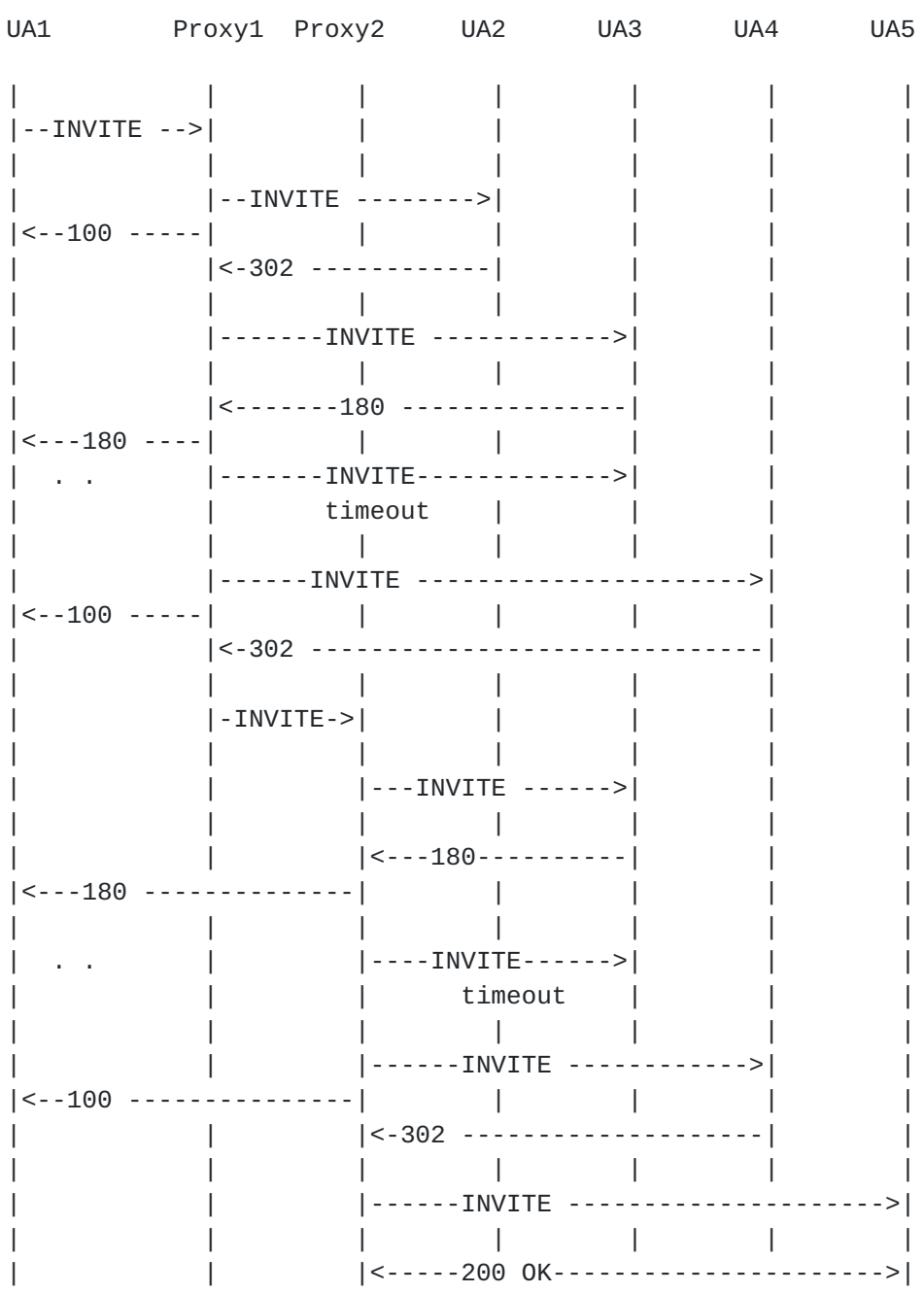
This section highlights some scenarios under which the Request History Capability could be applicable.

Certainly, various other solutions can be applied in some fashion to each of these scenarios, however, the objective of this draft has been to abstract the requirements from these scenarios towards providing a more robust solution for each and at the same time providing fundamental building block(s) applicable to future applications.

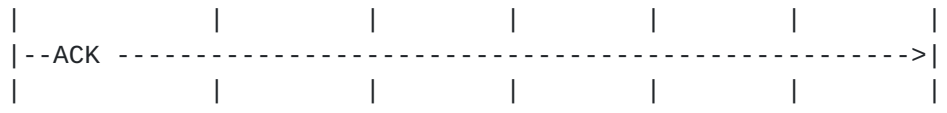
8.1 Sequentially forking with Retargetting

This scenario is as follows:

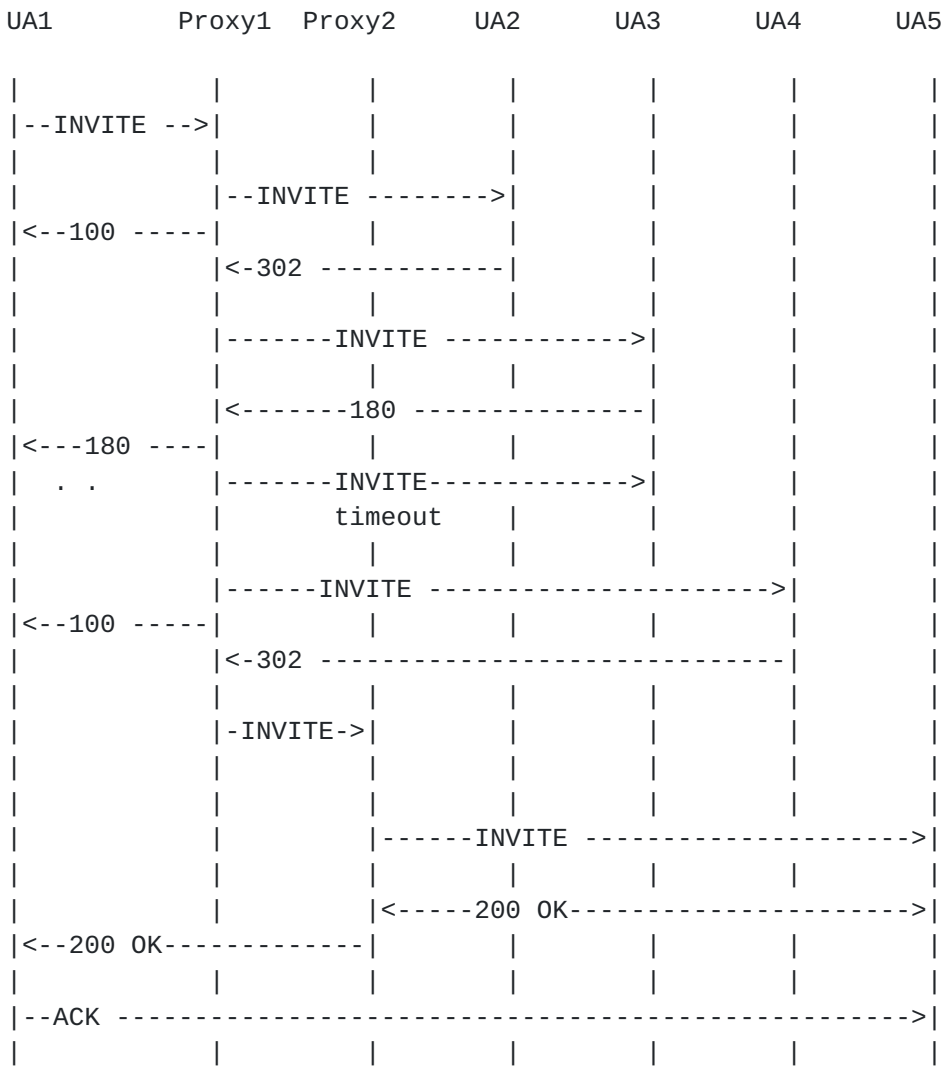
- o UA 1 sends a call to proxy 1. Proxy 1 sequentially tries several places (UA2, UA3 and UA4) before retargetting the call to Proxy 2. Proxy 2 unfortunately tries several of the same places (UA3 and UA4), before completing at UA5.




|<--200 OK-------------| | | | |



This scenario is provided to show the duplication of messaging when there isnÆt sufficient knowledge to optimize a sequential attempt at reaching an end user. With the "Request History" capability, this flow could be optimized as follows:

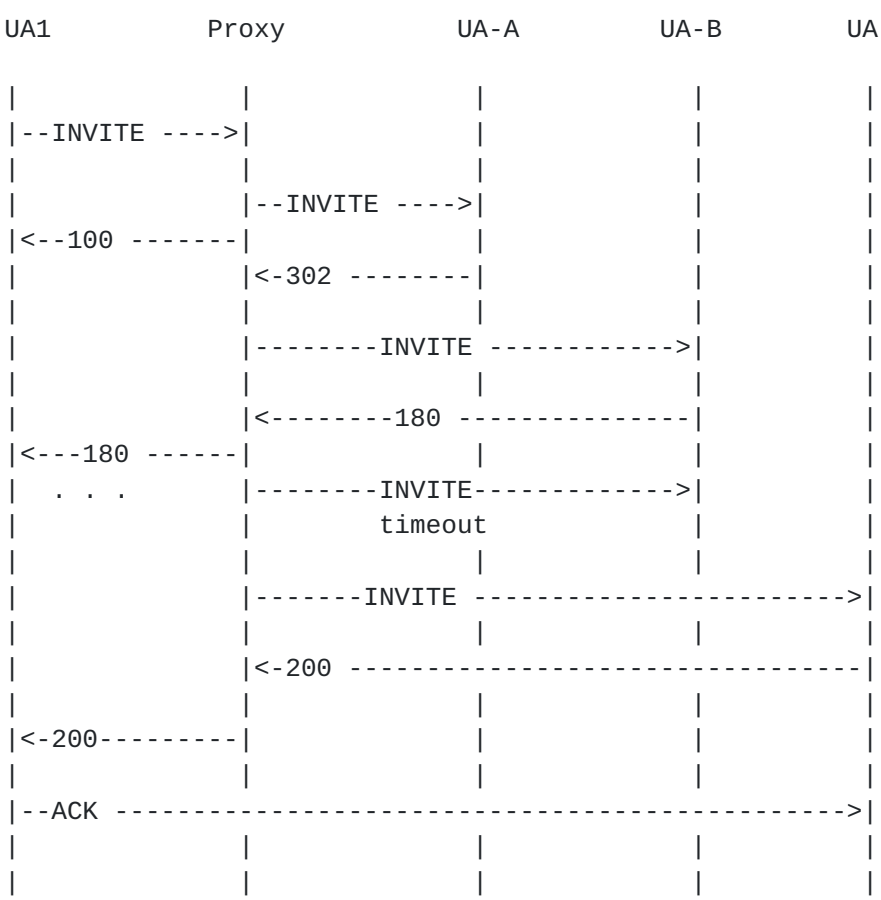


8.2 Voicemail

This scenario is as follows:

- UA 1 called UA A which had been forwarded to UA B which forwarded to a UA VM (voicemail server) which needs information (e.g. reason the call was retargeted, original

Request URI) to make a policy decision about what mailbox to use, which greeting to play etc. This scenario shows that something like the "Request History" capability must be used for this service to function.



Certainly, another valid scenario for the support of voicemail would be that this 'policy decision' on which mailbox to use (etc.) is made by the UA which forwarded to voicemail (UA B), or by the Proxy which performed the forwarding on behalf of B. In this case, the UA or Proxy can put all the information that the Voicemail server needs to identity the correct mailbox, etc., into the Request-URI. This fits with the SIP service paradigm where the Request-URI identifies the resource (namely, the particular mailbox/greeting etc.) that is required.

However, whilst this model is certainly applicable and required in SIP, it places service intelligence away from the system providing the key aspect of the service (the VM server).

The proposal in this draft is to rely on generic information-providing capabilities in the UA/Proxy, allowing the Voicemail system

to provide more and better voicemail-related services without relying

on specific capabilities in the UA/Proxy. This would allow voicemail service providers to innovate independently of the particular UA/Proxy that their customers are using, and its capabilities. Presently, with the information loss problem, VM service providers, and any other similar service providers, are limited in the services they can provide because they do not have complete information about how the call reached them. They rely on the UA/proxy of their customers having the necessary capabilities to formulate a Request-URI identifying exactly what should happen next. Finally, there is obviously a desire to use existing voicemail platforms based on PSTN/ISDN technology which operate according to the paradigm in this example.