SALUD
Internet-Draft
Updates: 3261 (if approved)
Intended status: Standards Track
Expires: April 5, 2013

L. Liess, Ed.
R. Jesske
Deutsche Telekom AG
A. Johnston
Avaya
D. Worley
Ariadne
P. Kyzivat
Huawei

October 2, 2012

Alert-Info URNs for the Session Initiation Protocol (SIP) draft-ietf-salud-alert-info-urns-07

Abstract

The Session Initiation Protocol (SIP) supports the capability to provide a reference to a specific rendering to be used by the UA when the user is alerted. This is done using the Alert-Info header field. However, the reference addresses only network resources with specific rendering properties. There is currently no support for predefined standard identifiers for describing the semantics of the alerting situation or the characteristics of the alerting signal, without being tied to a particular rendering. To overcome this limitation and support new applications, a new family of URNs for use in SIP Alert-Info header fields is defined in this specification.

This document normatively updates [RFC3261], the Session Initiation Protocol (SIP). It changes the usage of the SIP Alert-Info header field defined in the [RFC3261] by additionally allowing its use in all provisional responses to INVITE (except the 100 response).

Requirements Language

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 [RFC2119].

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

This Internet-Draft will expire on April 5, 2013.

Copyright Notice

Table of Contents

1. Introduction

1.1. Motivation

The Session Initiation Protocol (SIP) [RFC3261] includes a means to suggest to a user agent (UA) a particular ringback tone or ring tone to be used during session establishment. In [RFC3261] this is done by including a URI in the Alert-Info header field, that specifies the tone. The URI is most commonly the HTTP URL to the audio file. On the receipt of the Alert-Info header field the user agent may fetch the referenced ringback tone or ring tone and play it to the user.

This mechanism hinders interoperability when there is no common understanding of the meaning of the referenced tone, which might be country- or vendor-specific. It can lead to problems for the user trying to interpret the tone and for the UA wanting to substitute its own tone (e.g., in accordance with user preferences) or provide an alternative alerting mode (e.g., for hearing-impaired users). If caller and callee are from different countries, the understanding of the tones may vary significantly. Hearing impaired users may not sense the specific tone if it is provided as an audio file. The tone per se is also not useful for automata.

There are currently interoperability issues around the use of the Alert-Info header field when not using an external ring file. For example, consider the PBX special ring tone for an external (to the PBX) caller. Different vendors use different approaches such as: Alert-Info: <file://ring.pcm>;alert=normal where ring.pcm is a dummy file or: Alert-Info: <file://normal.ring.pcm> or: Alert-Info: <sip:normal-ringtone@example.com>. As a result, Alert-Info currently only works when the same vendor provides PBX and UA, as only then is the same "fake" proprietary URI convention used.

Another limitation of the current solution is that the referenced tones are tied to particular rendering. It is not possible to provide semantic indications or names for rendering characteristics that signals the intent and allows the recipient to decide how to render the received information in an appropriate way.

To solve the described issues, this specification defines the new URN namespace 'alert' for the Alert-Info header field that allows for programmatic user interface adaptation and for conversion of equivalent alerting tones in the Public Switched Telephone Network (PSTN) when the client is a gateway. The work to standardize an Alert-Info URN will increase SIP interoperability for this header field by replacing proprietary conventions used today.

Using the 'alert' namespace provides syntax for several different

application spaces, e. g.:

- Names for service indications, such as call waiting or automatic callback, not tied to any particular rendering.
- Names for common ring tones generated by PBX phones for cases such as an internal enterprise caller, external caller, ringback tone after a transfer failure or expiration of a hold timer, etc.
- Names for country-specific ringback tones.
- Names for things with specific renderings that aren't purely audio. They might be static icons, video sequences, text, etc.

Some advantages of a URN rather than a URL of a downloadable resource:

- Do not need to download it or deal with security issues associated with dereferencing.
- No formatting or compatibility issues.
- No security risk of rendering something unexpected and undesirable.
- The tone can be stored locally in whatever format and at whatever quality level is appropriate, because it is specified "by name" rather than "by value".
- It is easier to make policy decisions about whether to use it or not.
- It facilitates translation for the hearing impaired.

The downside is that if the recipient does not understand the URN then it will only be able to render a default ringback tone or ring tone.

This document creates a new URN namespace and registry for alert indications and registers some initial values.

1.2. Alert-Info Header Field Usage Change

This specification changes the usage of the SIP Alert-Info header field defined in the [RFC3261] by additionally allowing its use in all provisional responses to INVITE (except the 100 response).

In practice, this specification extends Alert-Info in that it will

cause the use of a new class of URIs and the use of multiple URIs. Backward compatibility issues are not expected, as devices that do not understand an Alert-Info URN should ignore it, and devices should not malfunction upon receiving multiple Alert-Info alert-params (which was syntactically permitted before, but rarely used).

1.3. Terminology

This specification uses a number of terms to refer to the roles involved in the use of alerting indications in SIP. A "specifier" sends an "alerting indication" (one or more URNs in an Alert-Info header) to a "renderer" which then "renders" a "signal" or "rendering" based on the indication to a human user. A "category" is a characteristic whose "values" can be used to classify indications.

This specification uses the terms "ring tone" and "ringback tone". A "ring tone" or "calling signal" (terminology used in [E182]) is a signal generated by the callee's end device, advising the callee about an incoming call. A "ringback tone" or "ringing tone" (terminology used in [E182]) is a signal advising the caller that a connection has been made and that a ring tone is being rendered to the callee.

2. Requirements

This section discusses the requirements for an alerting indication to transport the semantics of the alerting situation or the characteristics of the rendering.

REQ-1: The mechanism will allow user agents (UAs) and proxies to provide in the Alert-Info header field an alerting indication which describes the semantics of the signaling situation or the characteristics of the rendering and allows the recipient to decide how to render the received information to the user.

REQ-2: The mechanism will allow the alerting indication to be specified "by name" rather than "by value", to enable local policy decisions whether to use it or not.

REQ-3: The mechanism will enable alerting indications to represent a wide variety of signals, which have many largely-orthogonal characteristics.

REQ-4: has been deleted. To avoid confusion, the number will not be reused.

REQ-5: The mechanism will enable the set of alerting indications to

be able to support extensibility by a wide variety of organizations that are not coordinated with each other. Extensions will be able to:

- add further values to any existing category

- add further categories that are orthogonal to existing categories

- semantically subdivide the meaning provided by any existing indication

REQ-6: The mechanism will be flexible, so new alerting indications can be defined in the future, when SIP-applications evolve. E. g. Alert-Info URNs could identify specific media by name, such as "Beethoven's Fifth", and the end device could render some small part of it as a ring tone.

REQ-7: The mechanism will provide only an indication capability, not a negotiation capability.

REQ-8: The mechanism will not require an alerting indication to depend on context provided by a previous alerting indication in either direction.

REQ-9: The mechanism will allow transmission in the Alert-Info header field of SIP INVITE requests and provisional 1xx responses excepting the 100 responses.

REQ-10: The mechanism will be able to accommodate renderers that are customized with a limited or uncommon set of signals they can render and renderers that are provided with a set of signals that have uncommon semantics. (The canonical example is a UA for the hearing-impaired, customized with an uncomon set of signals, video or text instead of audio. By REQ-7, the renderer has no way of transmitting this fact to the specifier.)

REQ-11: The mechanism will allow an alerting indication to reliably carry all extensions if the specifier and the renderer have designs that are properly coordinated.

REQ-12: The mechanism will allow a renderer to select a tone that approximates to that intended by the specifier if the renderer is unable to provide the precise tone indicated.

REQ-13: The mechanism will support alerting indications relating to services such as call waiting, forward, transfer-recall, auto-callback and hold-recall.

REQ-14: The mechanism will allow rendering common PBX ring tone types.

REQ-15: The mechanism will allow rendering specific country ringback tones.

REQ-16: The mechanism will allow rendering tones for emergency alerts. (Use cases and values definition are not subject of this specification.)

REQ-17: The mechanism will allow rendering using other means than tones, e.g. text or images.

REQ-18: The mechanism will allow TDM gateways to map ring/ringback tones from legacy protocols to SIP at the edge of a network, e.g. national ring tones as defined in TIA/EIA-41-D and 3GPP2 A.S0014. (Use cases and values definition are not subject of this specification.)

REQ-19: The mechanism will ensure that if an UA receives Alert-Info URNs or portions of an Alert-Info URN it does not understand, it can ignore them.

REQ-20 The mechanism will allow storage of the actual encoding of the rendering locally rather than fetching it.

REQ-21: The mechanism must provide a simple way to combine two alerting indications to produce an alerting indication that requests a combination of the intentions of the two alerting indications, where any contradictions or conflicts between the two alerting indications are resolved in favor of the intention of the first alerting indication.

3. Use Cases

This section describes some use cases for which the Alert-Info URN mechanism is needed today.

3.1. PBX Ring Tones

This section defines some commonly encountered ring tones on PBX or business phones. They are as follows:

3.1.1. normal

This tone indicates that the default or normal ring tone should be rendered. This is essentially a no-operation Alert-Info URN and

should be treated by the UA as if no Alert-Info URN is present. This is most useful when Alert-Info header field parameters are being used. For example, in [I-D.ietf-bliss-shared-appearances], an Alert-Info header field needs to be present containing the "appearance" parameter, but no special ring tone needs to be specified.

3.1.2. external

This tone is used to indicate that the caller is external to the enterprise or PBX system. This could be a call from the PSTN or from a SIP trunk.

3.1.3. internal

This tone is used to indicate that the caller is internal to the enterprise or PBX system. The call could have been originated from another user on this PBX or on another PBX within the enterprise.

3.1.4. priority

A PBX tone needs to indicate that a priority level alert should be applied for the type of alerting specified (e.g. internal alerting).

3.1.5. short

In this case the alerting type specified (e.g. internal alerting) should be rendered shorter than normal. In contact centers, this is sometimes referred to as "abbreviated ringing" or a "zip tone".

3.1.6. delayed

In this case the alerting type specified should be rendered after a short delay. In some bridged line/shared line appearance implementations, this is used so that the bridged line does not ring at exactly the same time as the main line, but is delayed a few seconds.

3.2. Service Tones

These tones are used to indicate specific PBX and public network telephony services.

3.2.1. call-waiting

The Call Waiting Service [TS24.615] permits a callee to be notified of an incoming call while the callee is engaged in an active or held call. Subsequently, the callee can either accept, reject, or ignore the incoming call. There is an interest on the caller side to be

informed about the call waiting situation on the callee side. Having this information the caller can decide whether to continue waiting for callee to pickup or better to call some time later when it is estimated that the callee could have finished the ongoing conversation. To provide this information, the callee's UAS (or proxy) aware of the call waiting condition can add the call-waiting indication to the Alert-Info header field in the 180 Ringing response. As call-waiting information may be subject to the callee's privacy concerns, the exposure of this information shall be done only if explicitly required by the callee.

3.2.2. forward

This feature is used in a 180 Ringing response when a call forwarding feature has been initiated on an INVITE. Many PBX system implement a forwarding "beep" followed by normal ringing to indicate this. Note that a 181 response can be used in place of this URN.

3.2.3. transfer-recall

This feature is used when a blind transfer [RFC5589] has been performed by a server on behalf of the transferor and fails. Instead of failing the call, the server calls back the transferor, giving them another chance to transfer or otherwise deal with the call. This service tone is used to distinguish this INVITE from any other normal incoming call.

3.2.4. auto-callback

This feature is used when a user has utilized a server to implement an automatic callback service [I-D.ietf-bliss-call-completion]. When the user is available, the server calls back the user and utilizes this service tone to distinguish this from any other normal incoming call.

3.2.5. hold-recall

This feature is used when a server implements a call hold timer on behalf of an endpoint. After a certain period of time of being on hold, the user who placed the call on hold is alerted to either retrieve the call or otherwise dispose of the call. This service tone is used to distinguish this case from any other normal incoming call.

3.3. Country-specific ringback tone indications for the public telephone network

In the PSTN, different tones are used in different countries. End

users are accustomed to hear the callee's country ringback tone and would like to have this feature for SIP.

4. Namespace Registration Template

This section describes the registration template for the 'alert' URN namespace identifier (NID) according to the [RFC2141] and [RFC3406]

Namespace ID: alert

Registration Information:

Registration version: 1

Registration date: TBD

Declared registrant of the namespace:

Registering organization: IETF

Designated contact: Laura Liess

Designated contact email: l.liess@telekom.de

Declaration of syntactic structure:

The Namespace Specific String (NSS) for the "alert" URNs is called alert-identifier and has a hierarchical structure. The left-most <label> is called <alert-category> and is separated from the right-side of the <alert-identifier>, the <alert-indication> by a colon. The general form is urn:alert:{alert-category}:{alert-indication}.

In this specification, following alert-categories identifiers are described: "service" , "priority" , "source" , "duration", "delay" and "locale". The <alert-category> set can be extended in the future.

The categories are orthogonal. Any Alert-Info URN defined in this specification is syntactically valid for ring and ringback tones and can be used in INVITE requests or in provisional 1xx responses excepting the 100 response.

The<alert-indication>s are hierarchical identifiers. The set of allowable characters is the same as that for domain names [RFC1123]. Labels used in <standard-name> MUST comply with the syntax for Non Reserved LDH-labels [RFC5890]. Labels used in <private-name> MUST comply with the syntax for Non Reserved LDH-labels or for A-labels [RFC5890]. Comparisons MUST follow the comparison rules for the corresponding type of label. Registered URNs MUST be transmitted as registered. A new name MUST NOT be registered if it is equal by the comparison rules above to an already registered name.

The ABNF [RFC5234] for the Alert -Info URNs is shown below:

```
alert-URN        = "urn:alert:" alert-identifier
alert-identifier = alert-category ":" alert-indication
alert-category   = name
alert-indication = name *(":" name)
name             = standard-name / private-name
standard-name    = label
private-name     = label "." label *[ "." label ]
label            = let-dig [ *let-dig-hyp let-dig ]
let-dig-hyp      = let-dig / "-"
let-dig          = ALPHA / DIGIT
ALPHA            = %x41-5A / %x61-7A   ; A-Z / a-z
DIGIT            = %x30-39 ; 0-9
```

Note: <private-name> is

Relevant ancillary documentation: None

Community considerations: The alert URN is believed to be relevant to a large cross-section of Internet users, including both technical and non-technical users, on a variety of devices and with a variety of perception capabilities. The 'alert' URN will allow Internet users to receive more information and enable them to better make decisions about accepting an offered call, or get better feedback on the progress of a call they have made. User interfaces for the perception impaired users can better render the ringback tone indication based on the Alert-Info URN. The

assignment of identifiers is described in Section 6. The Alert-Info URN does not prescribe a particular resolution mechanism, but any resolution MUST comply with the rules in Section 8.

Namespace considerations: This specification proposes a new URN namespace "alert" for global names representing the semantic of signals or renderings. The particular rendering of an "alert" URN identifier depends on the context in which the rendering is done.

Also the initial scope is the usage in the SIP Alert-Info header, the new "alert" URN namespace is designed for more general usage. The URN may appear in any protocols that allow general URIs (e.g. also in ITU-T protocols) , in web pages or in any end devices software.

For the reasons described above, the usage of the already existing namespace "urn:ietf:params" was not considered appropriate. The URN namespace for Emergency and Other Well-Known Services [RFC5031], was used as a model.

Note, to be deleted for the final version of this draft: Because the work on this draft lasts for about fouryears, the new "alert" URN namespace is already used in already finalized specifications of other SDOs (3GPP) and there are already existing implementations in products and large carrier networks.

Also there do not appear to be other URN namespaces that serve the same need of uniquely identifying the semantic of a signal or rendering feature. Unlike most other currently registered URN namespaces, the "alert" URN does not identify documents and protocol objects (e.g., [RFC3044], [RFC3120], [RFC3187], [RFC3188], [RFC4179], [RFC4195], [RFC4198]), types of telecommunications equipment [RFC4152], people or organizations [RFC3043].

Identifier uniqueness considerations: An "alert" URN identifies the semantic or sensory feature of the rendering at the caller's or callee's end device. The feature identified by a particular "alert" URN is distinct from the feature identified any other "alert" URN. For "alert" URN identifiers based on <standard-name>s this is ensured through the IANA registration of each "alert" URN (that is, each defined combination of <alert-category> and <alert-identifier>) as described in Section 6.

"Alert" URN identifiers with a <private-name> part, IANA delegates the task to act as a registry to the owner of the "reverse FQDN" in the <private-name>.

Identifier persistence considerations: The "alert" URN identifiers based on <standard-name>s for the same indication is expected to be persistent, as long as it is registered with IANA.

The "alert" For URN identifiers with a <private-name>part, the owner of the "reverse FQDN" in the <private-name> is responsible for the correct proper functionality within the private domain.

Process of identifier assignment: The process of identifier assignment is described in Section 6, which includes the extension rules for independent organizations described in Section 7.2.

Process for identifier resolution: Alert-Info URNs are statically resolved according to the IANA registry.

Rules for lexical equivalence: Alert-Info URNs are compared according to case-insensitive string equality.

Conformance with URN syntax: The BNF in the 'Declaration of syntactic structure' above constrains the syntax for this URN scheme.

Validation mechanism: Validation determines whether a given string is currently a validly-assigned URN [RFC3406]. Static validation is performed based on the currently registered Alert-Info URNs at IANA.

Scope: The scope for this URN is public and global.

5. Alert-Info URN Values Definitions

5.1. Alert-category Values Definitions

Following <alert-category> values are defined in this document:

- service

- source

- priority

- duration

- delay

- locale

5.2. Alert-indication Values Definitions

This section describes the Alert-Info URN indication values for the alert-categories defined in this document.

For each <alert-category> , a default indication is defined, which is essentially a no-operation Alert-Info URN and should be treated by the UA as if no Alert-Info URN for the respective category is present. Alert-Info URN default indications are most useful when Alert-Info header field parameters are being used. For example, in [I-D.ietf-bliss-shared-appearances], an Alert-Info header field needs to be present containing the "appearance" parameter, but no special ringtone need be specified.

The "<private-name>" syntax is used for extensions specific to independent organizations, as described in Section 7.2.

5.2.1. Alert-Info URN Indication Values for the alert-category 'service'

- normal (default)

- call-waiting

- forward

- recall:callback

- recall:hold

- recall:transfer

- <private-name>

Examples: urn:alert:service:call-waiting or urn:alert:service:recall:transfer.

5.2.2. Alert-Info URN Indication Values for the alert-category 'source'

- unclassified (default)

- internal

- external

- friend

- family

- <private-name>

Examples: urn:alert:source:external.

5.2.3. Alert-Info URN Indication Values for the alert-category 'priority'

- normal (default)

- low

- high

- <private-name>

Examples: urn:alert:priority:high.

5.2.4. Alert-Info URN Indication Values for the alert-category 'duration'

- normal (default)

- short

- long

- <private-name>

Examples: urn:alert:duration:short.

5.2.5. Alert-Info URN Indication Values for the alert-category 'delay'

- none (default)

- yes

- <private-name>

Examples: urn:alert:delay:yes .

5.2.6. Alert-Info URN Indication Values for the alert-category 'locale'

- default (default)

- country:<ISO 3166-1 country code>

- <private-name>

The ISO 3166-1 country code [ISO3166-1] is used to inform the UA on the other side of the call that a country-specific rendering should be used. For example, to indicate ringback tones from South Africa, the following URN would be used: <urn:alert:locale:country:za>.

6. IANA Considerations

This section registers a new URN namespace identifier (NID) in accordance with RFC 3406 with the registration template provided in Section 4.

6.1. New alert identifiers

Alert URN identifiers are identified by <label>s managed by IANA, according to the processes outlined in [RFC5226] in a new registry called "Alert URN Labels". Thus, creating a new Alert-Info URN identifier requires IANA action. The policy for adding a new alert category is 'Standards Action'. (This document defines the alert categories 'service', 'source', 'priority', 'duration', 'delay' and 'locale'.) The policy for assigning <label>s to <alert-indication>s and the rules to combine them may differ for each <alert-category> and MUST be defined by the document describing the corresponding alert category. The entries in the registration table have the following format:

<alert-category>/ <alert-identifier>;	Reference	Description
foo	RFCxyz	Description of the 'foo' <alert-category>;
foo:bar	RFCabc	Description of the 'foo:bar' <alert-identifier>

Each <alert-category> or <alert-indication> label MUST NOT exceed 27 characters.

6.2. Initial IANA Registration

6.2.1. The "service" alert-category and alert-identifiers

The following table contains the initial IANA registration for the "service" <alert-category> and <alert-identifier>s. The value of this indicator is set to a value different from "normal" if the caller or callee is informed that a specific telephony service has been initiated.

<alert-category>/ <alert-identifier>	Reference	Description
service	RFC XXXX	<alert-category> for "service" <alert-identifier>s
service:normal	RFC XXXX	Normal ring /rinback rendering (default value)
service:call-waiting	RFC XXXX	Call waiting was initiated at the other side of the call
service:forward	RFC XXXX	Call has been forwarded
service:recall:calback	RFC XXXX	Recall due to callback
service:recall:hold	RFC XXXX	Recall due to call hold
service:recall:transfer	RFC XXXX	Recall due to callback
service:<private-name>	RFC XXXX	Reserved for private extensions

6.2.2. The "source" alert-category and alert-identifiers

The following table contains the initial IANA registration for the "source" <alert-category> and <alert-identifier>. The value of this indicator provides information about the user at the other side of the call.

<alert-category>/ <alert-identifier>	Reference	Description
source	RFC XXXX	<alert-category> for "source" <alert-identifier>s
source:unclassified	RFC XXXX	Unclassified ring /rinback rendering (default value)
source:internal	RFC XXXX	User at the other side of the call is internal to the enterprise or PBX system
source:external	RFC XXXX	User at the other side of the call is internal to the enterprise or PBX system
source:friend	RFC XXXX	User at the other side of the call is a friend
source:family	RFC XXXX	User at the other side of the call is a family member
source:<private-name>	RFC XXXX	Reserved for private extensions

6.2.3. The "priority" alert-category and alert-identifiers

The following table contains the initial IANA registration for the "priority" <alert-category> and <alert-identifier>s. The value of this indicator provides information about the priority the alerted user should give to the call.

<alert-category>/ <alert-identifier>	Reference	Description
priority	RFC XXXX	<alert-category> for "priority" <alert-identifier>s
priority:normal	RFC XXXX	Normal ring /rinback rendering (default value)
priority:low	RFC XXXX	Low priority call.
priority:high	RFC XXXX	High priority call
priority:<private-name>	RFC XXXX	Reserved for private extensions

6.2.4. The "duration" alert-category and alert-identifiers

The following table contains the initial IANA registration for the "duration" <alert-category> and <alert-identifier>s. The value of this indicator provides information about the duration of the alerting signals compared to the default alerting signals.

<alert-category>/ <alert-identifier>	Reference	Description
duration	RFC XXXX	<alert-category> for "duration" <alert-identifier>s
duration:normal	RFC XXXX	Normal ring /rinback rendering (default value)
duration:short	RFC XXXX	Shorter than normal
duration:long	RFC XXXX	Longer than normal
duration:<private-name>	RFC XXXX	Reserved for private extensions.

6.2.5. **The "delay"** alert-category and alert-identifiers

The following table contains the initial IANA registration for the "delay" <alert-category> and <alert-identifier>s. The value of this indicator provides information about the delay of the alerting signals.

<alert-category>/ <alert-identifier>	Reference	Description
delay	RFC XXXX	<alert-category> for "delay" <alert-identifier>
delay:none	RFC XXXX	Immediate alerting (default value)
delay:yes	RFC XXXX	Delayed alerting
delay:<private-name>	RFC XXXX	Reserved for private extensions

6.2.6. **The "locale"** alert-category and alert-identifiers

The following table contains the initial IANA registration for the "locale" <alert-category> and <alert-identifier>s. The value of this indicator provides information about the location of the user at the other side of the call.

<alert-category>/ <alert-identifier>	Reference	Description
locale	RFC XXXX	<alert-category> for "locale" <alert-identifier>
locale:default	RFC XXXX	Alerting not location specific (default value)
locale:country:<ISO 3166-1 country code>	RFC XXXX	Country-specific alerting
locale:<private-name>	RFC XXXX	Reserved for private extensions

7. Extensibility Rules

7.1. General Extensibility Rules

The set of Alert-Info URNs is intended to be extensible. An extension "at the top level" creates an entirely new category (or characteristic), an extension "at the second level" creates a new indication value for a category, an extension "at the third level" creates a subdivision of a indication value, etc. Extensions at lower levels are preferred over those at upper levels.

URNs allow in principle infinite subdivision of existing indication values, although most of the standard Alert-Info URNs give only one level of subdivision and a few give two levels of subdivision.

The process for defining new Alert-Info URNs is described in Section 6.1. Adding new categories and adding <alert-indication> values other than via the "private" mechanism described in Section 7.2 is standards action.

7.2. Extensions Rules for Independent Organizations

The "<private-name>" syntax is for proprietary extensions specific to independent organizations. The "<private-name>" is used in the form of a "reverse FQDN" of the entity that defines the extension, possibly followed by further components. Standard URNs will never contain a ".", so proprietary extensions need no further marker. This gives a way of assigning unique names without the need for a new registry. The namespace for each alert category is independent. Those assigning new names must ensure they are in a position to assign names uniquely for the FQDN they choose.

For example, some company SomeCompany.example.org could use urn:alert:service:call-waiting:org.example.somecompany, which is the

SomeCompany's private version for call-waiting or it may have several distinct private versions of call-waiting, e.g. urn:alert:service:call-waiting:abc.org.example.somecompany and urn:alert:service:call-waiting:def.org.example.somecompany (which are siblings in the tree under urn:alert:service:call-waiting). Also it can subdivide its private version of call-waiting in urn:alert:service:call-waiting:org.example.somecompany:abc and urn:alert:service:call-waiting:org.example.somecompany:def (which are siblings in the tree under service:call- waiting: org.example.somecompany).

Adding new categories and adding <alert-indication> values via the "private" mechanism is not a standards action.

8. Combinations of Alert-Info URNs

8.1. Priority Rules

This section describes combination rules for the case when all the Alert-Info header fields only contain Alert-Info URNs. Combinations of URNs and URIs in the Alert-Info header fields of the same SIP-message are not defined in this specification.

In many cases, more than one URNs will be needed to fully define a particular tone. This is done by including multiple Alert-Info URNs, in one or more Alert-Info header fields in a request or a response. For example, an internal, priority call could be indicated by Alert-Info: <urn:alert:source:internal>, <urn:alert:priority:high>. A priority call waiting tone could be indicated by Alert-Info: <urn:alert:service:call-waiting>, <urn:alert:priority:high>.

The sender of the Alert-Info header may include an arbitrary list of Alert-Info URNs, even if they are redundant or contradictory. An earlier URN has priority over any later contradictory URN. This allows any element to modify a list of URNs to require a feature value (by adding a URN at the beginning of the list) or to suggest a feature value (by adding a URN at the end of the list).

The receiving UA attempts to match the received Alert-Info URNs combination with the signal(s) it is able to render.

The implementation is free to ignore any or all parts of the received Alert-Info URNs. The exact way in which a UA renders a received combination of Alert-Info URNs is left as an implementation issue. However, the implementation MUST comply to following rules:

a. Each alert-info URN has precedence over all URNs that follow it, and its interpretation is subordinate to all URNs that precede it.

b. If the UA cannot implement the effect of a URN (because it does not recognize the URN or the URN's effect is precluded by preceding URNs), the UA repeatedly removes either

(1) the final <name> of the URN, or

(2) if the final <name> is a <private-name> with three or more <label>s, the final <label>

until either

(i) the resulting URN is recognized and can be given effect by some signal (without reducing the degree of expression of any preceding URN), or

(ii) the resulting URN is reduced to having no <alert-indication>.

In case (ii), that URN in the series cannot be given effect, so it is ignored.

c. In case that after processing all the received URNs, the UA can generate more than one signal that are equally effective at expressing the URNs (under the preceding rules), one of those signals is selected. When selecting from the set of equally effective signals, no signal should be chosen if a less-specific signal is also in the set. (Specificity is to be judged based on the defined meanings of the signals to the user.) (E.g., if each signal is considered to express certain <alert-indication>s of certain <alert-categories>, one signal is less-specific than a second signal if the first signal's <alert-indication>s are a subset or are prefixes of the second signal's <alert-indication>s.) However, a more-specific signal may be chosen if the choice is based on information derived from the containing SIP message. E.g., a signal implying urn:alert-info:priority:high may be chosen if the SIP message contains the header "Priority: urgent".

In all situations, the set of signals that can be rendered and their significances may change based on user preferences and local policy.

In adidition, they may change based on the status of the UA. E.g., if a call is active on the UA, all audible signals may become unavailable, or audible signals may be available only if urn:alert-info:priority:high is specified.

8.2. Multi-mode signals

There are cases when the device can render two signal modes (e.g., audio and visual, or video or text) at the same time.

Formally, the device must be considered as making its choice from the set of all combined signals that it can render, and that choice must conform to the above rules. However, it can be proven that if the device makes its rendering choice for each of the two modes independently, with each choice separately conforming to the above rules, its combined choice conforms to the above rules, when it is regarded as a choice from among all possible combinations.

In such a situation, it may simplify implementation to make each choice separately. It is an implementation decision whether to chose from among combined signals, or to combine choices made from each signal mode.

9. Non-normative Algorithm for Handling Combinations of URNs

The following text is a non-normative example of an algorithm for handling combinations of URNs that complies with the requirements in Section 8. Thus, it demonstrates that requirements in section 7 are consistent and implementable. (Of course, a device may use any other algorithm which complies with Section 8.)

9.1. Algorithm Description

For each category (feature), there is a tree of possible values. For this description, we will name each tree by the category name, and name each node by the trailing portion of the URN. Each URN thus corresponds to a node in a category tree. Thus, there is a tree named "source", whose root node is also named "source", and which has the children source:internal, source:external, source:friend, and source:family. For example, urn:alert:source:external is placed at the node "source:external" in the "source" tree. (Of course, there are an infinite number of potential additional nodes in the tree for private values, but we don't have to represent those nodes explicitly unless the device has a signal representing the private value.)

We assign similar locations to signals, but each signal has a place in *every* tree. If a signal has a simple meaning, such as "external

source", its place in the "source" tree is source:external, but its place in every other feature tree is at the root node, meaning that it has no particular meaning for that feature.

A signal that has a complex meaning may have non-root positions in more than one feature tree. For example, an "external, high priority" signal would be placed at source:external and priority:high in those trees, but be at the root in all other feature trees.

In order to assure that the algorithm always selects at least one signal, we assume that there is a "default" signal, whose position in every feature tree is at the root. The default signal is set up so that it will never be excluded from the set of acceptable signals for an indication, but will usually be the least-desirable signal for any indication.

The algorithm proceeds by considering each URN in the received Alert-Info header from left to right, while revising a set of signals. The set of signals starts as the entire set of signals available to the device. Each URN excludes some signals from the set, and *sorts* the signals that remain in the set according to how well they represent the URN. (The details of these operations are described below.) The first URN is the "major sort", and has the most influence on the position of a signal in the set. The second URN is a "minor sort", in that it arranges the orders of the signals that are tied within the first sort, the third URN arranges the orders of the signals that are tied within the first two sorts, etc.

At the end of the algorithm, a final, "most minor" sort is done, which orders the signals which have been tied under all the sorts driven by the URNs. This final sort places the least specific signals (within their groups) *first*. (If one signal's position in each category tree is ancestral or the same as a second signal's position in that tree, the first signal is "less specific" than the second signal. Other cases are left to the implementation to decide.)

Once all the URNs are processed and the sorting is done, the device selects the first signal in the set.

Here is how a single sort step proceeds, examining a single URN to modify the set of signals (by excluding some signals and further sorting the signals that remain):

- The URN specifies a specific node in a specific category tree.

- All signals in the set that are positioned at that node, or at an ancestor node of the URN, are kept. All other signals are removed

from the set.

- o Within any group of signals that are tied under the previous sorts, place first those which are at the node of the URN, place second those which are at the parent node of the URN, etc., and place last those which are at the root node of the feature tree.

9.2. Examples of how the algorithm works

The following examples show how the algorithm described in the previous section works:

9.2.1. Example 1

The device has a set of 4 alerting signals. We list their primary meanings, and the locations that they are placed in the feature trees:

Signal 1

Meaning: external

Locations:

- source:external

- priority (that is, the root node of the priority tree)

Signal 2

Meaning: internal

Locations:

- source:internal

- priority

Signal 3

Meaning: low

Locations:

- source

- priority:low

Signal 4

Meaning: high

Locations:

- source

- priority:high

To which we add:

Signal 5

Meaning: default

Locations:

- source

- priority

If the device receives <urn:alert:source:internal>, then the sort is:

Signals at source:internal:

<urn:alert:source:internal>

Signals at source:

<urn:alert:priority:low>

<urn:alert:priority:high>

default

And these signals are excluded from the set:

<urn:alert:source:external>

So in this example, the sorting algorithm properly gives first place to <urn:alert:source:internal>.

9.2.2. Example 2

Let us add to the set of signals in Example 1 ones that express combinations like "internal, high priority", but let us specifically exclude the combination "internal, low priority" so as to set up some

tricky examples. This enlarges our set of signals:

Signal 1

Meaning: default

Locations:

- source

- priority

Signal 2

Meaning: external

Locations:

- source:external

- priority

Signal 3

Meaning: internal

Locations:

- source:internal

- priority

Signal 4

Meaning: low

Locations:

- source

- priority:low

Signal 5

Meaning: high

Locations:

- source

- priority:high

Signal 6

Meaning: external high

Locations:

- source:external

- priority:high

Signal 7

Meaning: external low

Locations:

- source:external

- priority:low

Signal 8

Meaning: internal high

Locations:

- source:internal

- priority:high

If the device receives <urn:alert:source:internal>, then the sort is:

Signals at source:internal: (that is, tied for first place)

- internal

-internal high

Signals at source: (tied for second place)

- low

- high

- default

Signals excluded from the set:

- external

- external low

- external high

Two signals are tied for the first place, but the final sort orders them:

- internal

- internal high

because it puts the least-specific signal first. So the signal "internal" is chosen.

9.2.3. Example 3

The same device receives <urn:alert:source:external>, <urn:alert:priority:low>. The first sort (due to <urn:alert:source:external>) is:

Signals at source:external:

- external

- external low

- external high

Signals at source:

- low

- high

- default

Signals excluded:

- internal

- internal high

The second sort (due to <urn:alert:priority:low>) puts signals at priority:low before signals at priority, and excludes signal at priority:high:

- external low
- external
- low
- default

Excluded:

- external high
- high
- internal
- internal high

So, we choose "external low".

9.2.4. Example 4

Suppose the same device receives <urn:alert:source:internal>, <urn:alert:priority:low>. Note that there is no signal that corresponds to this combination.

The first sort is based on source:internal, and results in this order:

- internal
- internal high
- low
- high
- default

Excluded:

- external

- external low

- external high

The second sort is based on priority:low, and results in this order:

- internal

- low

- default

Excluded:

- internal high

- high

- external low

- external

- external high

So we choose the signal "internal".

9.2.5. Example 5

Let us set up a simple set of signals, with three signals giving priority:

Signal 1

Meaning: default

Locations:

- priority

Signal 2

Meaning: low

Locations:

- priority:low

Signal 3

Meaning: high

Locations:

- priority:high

Notice that we've used the "default" signal to cover "normal priority". That is so the signal will cover situations where no priority URN is present, as well as the ones with <urn:alert:priority:normal>. So we're deliberately failing to distinguish "priority:normal" from the default priority.

If the device receives <urn:alert:priority:low>, the sort is:

- low

- default

Excluded:

- high

and signal "low" is chosen.

Similarly, if the device receives <urn:alert:priority:high>, signal "high" is chosen.

If the device receives <urn:alert:priority:normal>, the sort is:

- default

Excluded:

- low

- high

and signal "default" is chosen.

If no "priority" URN is received, "default" will be put before "low" and "high" by the final sort, and so it will be chosen.

10. User Agent Behaviour

A SIP UA MAY add a URN or multiple URNs to the Alert-Info header field in a SIP request or a provisional 1xx response (excepting a 100 response) when it needs to provide additional information about the

call or about the provided service.

Upon receiving a SIP INVITE request or a SIP provisional response with an Alert-Info header field that contains a combination of Alert-Info URNs, the User Agent (UA) attempts to match the received Alert-Info URNs combination with a signal it can render. The process the UA uses MUST conform to the rules described in Section 8. (A non-normative algorithm example for the process is described in Section 9.)

The User Agent (UA) is responsible for producing a reasonable rendering regardless of the combination of URIs (of any schemes) in the Alert-Info header field.

11. Proxy Behaviour

A SIP proxy MAY add a URN or multiple URNs to the Alert-Info header field in a SIP request or a provisional 1xx response (excepting a 100 response) when it needs to provide additional information about the call or about the provided service.

Following example shows both the network audio resource referenced by the HTTP URI and the URN indication for the call-waiting service transported by the Alert-Info header field in a 180 Ringing provisional response.

```
SIP/2.0 180 Ringing
Alert-Info: <http://www.example.com/sound/moo.wav>,
             <urn:alert:service:call-waiting>
To: Bob <sip:bob@biloxi.example.com>;tag=a6c85cf
From: Alice <sip:alice@atlanta.example.com>;tag=1928301774
Call-ID: a84b4c76e66710
Contact: <sip:bob@192.0.2.4>
CSeq: 314159 INVITE
Via: SIP/2.0/UDP server10.biloxi.example.com;
            branch=z9hG4bK4b43c2ff8.1
Content-Length: 0
```

12. Internationalization Considerations

The <alert-identifier> labels are protocol elements [RFC6365] and are not normally seen by users. Thus, the character set for these elements is restricted, as described in Section 6.

The URNs urn:alert:locale:country:<ISO 3166-1 country code> select

renderings that are conventional in the specified country.

13. Security Considerations

As an identifier, the alert URN does not appear to raise any particular security issues. The indications described by the 'alert' URN are meant to be well-known.

However, the provision of specific indications may raise privacy issues, e.g. indications about the source of the message or about services initiated at the other side. Such provision SHALL always be explicitly authorised by the party (caller or callee) the information in the Alert-Info URN refers to.

Proxies may choose to suppress undesired indications, e.g. from untrusted sources, while allowing them from trusted sources.

14. Acknowledgements

The authors wish to thank Denis Alexeitsev, the editor of the initial draft in BLISS, Anwar Siddiqui for his contributions to the draft, and Adam Roach, Dean Willis, Martin Huelsemann, Shida Schubert, John Elwell and Tom Taylor for their comments and suggestions.

15. References

15.1. Normative References

[RFC1123] Braden, R., "Requirements for Internet Hosts - Application and Support", STD 3, RFC 1123, October 1989.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2141] Moats, R., "URN Syntax", RFC 2141, May 1997.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

[RFC3406] Daigle, L., van Gulik, D., Iannella, R., and P. Faltstrom, "Uniform Resource Names (URN) Namespace Definition Mechanisms", BCP 66, RFC 3406, October 2002.

[RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.

15.2. Informative References

[E182] "Application of tones and recorded announcements in telephone services", http://www.itu.int/rec/T-REC-E.182-199803-I/en .

[I-D.ietf-bliss-call-completion] Worley, D., Huelsemann, M., Jesske, R., and D. Alexeitsev, "Call Completion for Session Initiation Protocol (SIP)", draft-ietf-bliss-call-completion-16 (work in progress), September 2012.

[I-D.ietf-bliss-shared-appearances] Johnston, A., Soroushnejad, M., and V. Venkataramanan, "Shared Appearances of a Session Initiation Protocol (SIP) Address of Record (AOR)", draft-ietf-bliss-shared-appearances-13 (work in progress), July 2012.

[ISO3166-1] "ISO 3166-1 English country names and code elements", http ://www.iso.org/iso/ english_country_names_and_code_elements .

[RFC3043] Mealling, M., "The Network Solutions Personal Internet Name (PIN): A URN Namespace for People and Organizations", RFC 3043, January 2001.

[RFC3044] Rozenfeld, S., "Using The ISSN (International Serial Standard Number) as URN (Uniform Resource Names) within an ISSN-URN Namespace", RFC 3044, January 2001.

[RFC3120] Best, K. and N. Walsh, "A URN Namespace for XML.org", RFC 3120, June 2001.

[RFC3187] Hakala, J. and H. Walravens, "Using International Standard Book Numbers as Uniform Resource Names", RFC 3187, October 2001.

[RFC3188] Hakala, J., "Using National Bibliography Numbers as Uniform Resource Names", RFC 3188, October 2001.

[RFC4152] Tesink, K. and R. Fox, "A Uniform Resource Name (URN) Namespace for the Common Language Equipment Identifier (CLEI) Code", RFC 4152, August 2005.

[RFC4179] Kang, S., "Using Universal Content Identifier (UCI) as Uniform Resource Names (URN)", RFC 4179, October 2005.

[RFC4195] Kameyama, W., "A Uniform Resource Name (URN) Namespace for the TV-Anytime Forum", RFC 4195, October 2005.

[RFC4198] Tessman, D., "A Uniform Resource Name (URN) Namespace for Federated Content", RFC 4198, November 2005.

[RFC5031] Schulzrinne, H., "A Uniform Resource Name (URN) for Emergency and Other Well-Known Services", RFC 5031, January 2008.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

[RFC5589] Sparks, R., Johnston, A., and D. Petrie, "Session Initiation Protocol (SIP) Call Control - Transfer", BCP 149, RFC 5589, June 2009.

[RFC5890] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", RFC 5890, August 2010.

[RFC6365] Hoffman, P. and J. Klensin, "Terminology Used in Internationalization in the IETF", BCP 166, RFC 6365, September 2011.

[TS24.615] "3GPP TS 24.615 Communication Waiting (CW) using IP Multimedia (IM) Core Network (CN) subsystem".

Authors' Addresses

Laura Liess (editor)
Deutsche Telekom AG
Heinrich-Hertz Str 3-7
Darmstadt, Hessen 64295
Germany

Phone: +49 6151 5812761
Email: laura.liess.dt@gmail.com

Roland Jesske
Deutsche Telekom AG
Heinrich-Hertz Str. 3-7
Darmstadt, Hessen 64295
Germany

Phone: +49 6151 5812766
Email: r.jesske@telekom.de

Alan Johnston
Avaya Inc.
St. Louis, MO
United States

Phone:
Email: alan.b.johnston@gmail.com

Dale R. Worley
Ariadne Internet Services, Inc.
738 Main St.
Waltham, MA 02451
US

Phone: +1 781 647 9199
Email: worley@ariadne.com

Paul Kyzivat
Huawei
United States

Email: pkyzivat@alum.mit.edu