Abstract

This document defines a new Session Description Protocol (SDP) media-level attribute, ‘content’. The ‘content’ attribute defines the content of the media stream in more detailed level than the media description line. The sender of an SDP session description can attach the ‘content’ attribute to one or more media streams. The receiving application can then treat each media stream differently (e.g., show it on a big screen or small screen) based on their content.
Table of Contents

1. Introduction ................................................. 3
2. Terminology .................................................. 3
3. Related Techniques ......................................... 3
4. Motivation for the New Content Attribute .................... 4
5. The Content Attribute ........................................ 5
6. The Content Attribute in the Offer/Answer Model .................. 6
7. Example .................................................... 6
8. Operation with SMIL .......................................... 7
9. Security Considerations ...................................... 7
10. IANA Considerations ......................................... 7
11. Acknowledges ............................................... 8
12. References .................................................. 8
    12.1. Normative References ................................ 8
    12.2. Informational References ............................... 9
Authors’ Addresses .............................................. 10
Intellectual Property and Copyright Statements ..................... 11
1. Introduction

SDP [1] is a protocol that is intended for describing multimedia sessions for the purposes of session announcement, session invitation, and other forms of multimedia session initiation. One of the most typical use cases of SDP is the one where it is used with SIP [5].

There are situations where one application receives several similar media streams which are described in an SDP session description. The media streams can be similar in a sense that their content cannot be distinguished from each other just by examining the media description lines (e.g., two video streams). The ‘content’ attribute is needed, so that the receiving application can treat each media stream appropriately based on its content.

This specification defines the SDP ‘content’ media-level attribute, which provides more information about the media stream than the ‘m’ line in an SDP session description.

2. Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in BCP 14, RFC 2119 [3] and indicate requirement levels for compliant implementations.

3. Related Techniques

The ‘label’ attribute [9] enables a sender to attach a pointer to a particular media stream. The name space of the ‘label’ attribute itself is unrestricted, so in principle it could also be used to convey information about the content of a media stream. However, in practice, this is not possible because of the backwards compatibility. Existing implementation of the ‘label’ attribute already use values from that unrestricted namespace in an application specific way. So, it is not possible to reserve parts from the ‘label’ attributes’ namespace without possible conflict with an already used, application specific labels.

It is possible to assign semantics to a media stream with the external document that uses the ‘label’ attribute as a pointer. The downside of this approach is that it requires external document. Typically this kind of mechanism would be defined for some particular use case, for example centralized conferencing.
Yet another way to attach semantics to a media stream is by using the ‘i’ SDP attribute, defined in RFC 2327 [6]. Nevertheless, values of the ‘i’ attribute are intended for human users and not for automata.

4. Motivation for the New Content Attribute

Currently, SDP does not provide any means to describe what is the content of a media stream (e.g., speaker’s image, slides, sign language) in a form that the application can understand. Of course the end user can see the content of the media stream, and read its title, but the application cannot understand what the media stream contains.

The application that is receiving multiple similar (e.g., same type and format) media stream needs, in some cases, to know what is the content of those streams. This kind of situation occurs for example in cases where presentation slides, speakers image and sign language are transported as separate media streams. It would be desirable that the receiving application could distinguish them in a way that it could handle them automatically in appropriate manner.

Figure 1: Application’s screen

The Figure 1 presents a screen of a typical communication application. The ‘content’ attribute enables the application to make the decision on where to show each media stream. From end user’s perspective, it is desirable that the user does not need to arrange media stream every time the media session starts.

The ‘content’ attribute could also be used in more complex situations. This kind of complex situation could be e.g., an application that is controlling the equipments in the auditorium.
Auditorium can have many different output channels for the video (main screen and two smaller screens) and the audio (main speakers, headsets for the participants). In this kind of environment, a lot of interaction from the end user who is operating the application would be required. So, the possibility for the application to handle the media stream without end users input is greatly emphasized.

5. The Content Attribute

This specification defines a new media-level value attribute, ‘content’. Its formatting in SDP is described by the following BNF [2]:

```
content-attribute   = "a=content:" mediacnt-tag
mediacnt-tag        = mediacnt *("", mediacnt)
mediacnt            = "slides" / "speaker" / "sl" / "main"
                     / "alt" / "user-floor" / mediacnt-ext
mediacnt-ext        = token
```

The ‘content’ attribute contains a token, which MAY be attached to a media stream by a sending application. It describes the content of the transmitted media stream to the receiving application. Many ‘content’ attribute values MAY be attached to a single media stream.

There is a set of pre-defined values for the ‘content’ attribute. Also other values can be defined in the future. The pre-defined values are:

- slides: This is a media stream that includes presentation slides. The media type can be e.g., a video stream or a set of instant message with pictures. Typical use case for this is e.g., online seminars and courses. This is similar to the ‘presentation’ role in H.239 [11].
- speaker: This is a image from the speaker. The media can be e.g., a video stream or a still image. Typical use case for this is e.g, online seminars and courses.
- sl: This means that the media stream contains sign language. The media type is a video stream. Typical use case for this is the one where the audio stream is translated into sign language.
- main: This means that the main stream is taken from the main source. Typical use case for this is e.g., a concert, where the camera is shooting the performer.
alt: This means that the main stream is taken from the alternative source. This is the alternative media stream. Typical use case for the is e.g., an event, where there is a separate ambient sound and the main sound. The alternative audio stream could be e.g., a sound of a jungle. Another example is the video of the conference room while the main is the video of the speaker. This is similar to the 'live' role in H.239.

user-floor: This indicates that a user level floor control is required. In other words, this is meant for system without any mechanism for floor control, and where a human needs to figure out whether an act of floor control, e.g., saying 'over', is needed or not. Typically this can be used with walkie-talkie type of devices, and with transcoded Text over IP (ToIP) streams originating from Telecommunications Devices for the Deaf (TDDs).

All of the values can be used with any media type. Application can make decision on how to handle a single media stream based on both media type and the value of 'content' attribute. Therefore the situation, where one value of 'content' attribute occurs more than once in a single session descriptor, is not problematic.

6. The Content Attribute in the Offer/Answer Model

This specification does not define a means to discover whether or not the peer endpoint understands the 'content' attribute, because 'content' values are informative only at the offer/answer model level. The fact that the peer endpoint does not understands the 'content' attribute, does not keep the media session from being established. The only consequence is that interaction from the receiving end user may be required.

7. Example

The following is an example of the SDP session description that uses the 'content' attribute:
8. Operation with SMIL

The values of 'content' attribute, defined in Section 5, can also be used with SMIL [10]. SMIL contains a 'param' element, which is used for describing the content of a media. However, this 'param' element provides only application specific description of media content. By using the values of the 'content' attribute, this 'param' element can also be used to describe the media content in a globally interpretable way.

Details on how to use the values of the 'content' attribute with SMIL's 'param' element are outside the scope of this specification.

9. Security Considerations

An attacker may attempt to add, modify, or remove 'content' attributes from a session description. This could result in an application behaving in a non-desirable way. So, it is strongly RECOMMENDED that integrity protection be applied to the SDP session descriptions. For session descriptions carried in SIP [5], S/MIME [7] is the natural choice to provide such end-to-end integrity protection, as described in RFC 3261 [5]. Other applications MAY use a different form of integrity protection.

10. IANA Considerations

This document defines a new 'content' attribute for SDP. It also defines initial set of values for it.
Contact name: Jani Hautakorpi Jani.Hautakorpi@ericsson.com.

Attribute name: 'content'.

Type of attribute Media level.

Subject to charset: Not.

Purpose of attribute: The ‘content’ attribute gives information from the content of the media stream to the receiving application.

Allowed attribute values: "slides", "speaker", "sl", "main", "alt", "user-floor", and a token.

The IANA is requested to create a subregistry for the ‘content’ attribute values under the Session Description Protocol (SDP) Parameters registry. The following are the initial values for the subregistry:

<table>
<thead>
<tr>
<th>Value of ‘content’ attribute</th>
<th>Reference</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>slides</td>
<td>RFC xxxx</td>
<td>Presentation slides</td>
</tr>
<tr>
<td>speaker</td>
<td>RFC xxxx</td>
<td>Image from the speaker</td>
</tr>
<tr>
<td>sl</td>
<td>RFC xxxx</td>
<td>Sign language</td>
</tr>
<tr>
<td>main</td>
<td>RFC xxxx</td>
<td>Main media stream</td>
</tr>
<tr>
<td>alt</td>
<td>RFC xxxx</td>
<td>Alternative media stream</td>
</tr>
<tr>
<td>user-floor</td>
<td>RFC xxxx</td>
<td>User level floor control req.</td>
</tr>
</tbody>
</table>

As per the terminology in RFC 2434 [4], the registration policy for new values for the ‘content’ parameter shall be ‘Specification Required’.

11. Acknowledges

Arnould van Wijk and Roni Even provided valuable ideas for this document.

12. References

12.1. Normative References


12.2.  Informational References


Authors’ Addresses

Jani Hautakorpi
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

Email: Jani.Hautakorpi@ericsson.com

Gonzalo Camarillo
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

Email: Gonzalo.Camarillo@ericsson.com
Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The Internet Society (2005). This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.