Abstract

From time to time, there are discussions to describe some session parameters specific to the codecs in a media in the application of SDP [1]. Codec specific ptime and transport address are two examples. This document analyses those requirements, and provides the evaluation for possible solutions.
## Table of Contents

1. Introduction .................................................. 3  
2. Terminology ................................................... 4  
3. Requirement Analysis .......................................... 5  
   3.1. Requirement to support codec specific ptime .......... 5  
   3.2. Requirement to support codec specific transport address 5  
   3.3. Summary .................................................. 7  
4. Evaluation of possible solutions .............................. 8  
   4.1. Extending the SDP ........................................ 8  
   4.1.1. Format ................................................ 8  
   4.2. Using grouping method .................................. 8  
   4.3. Using SDPng .............................................. 8  
5. Security Considerations ....................................... 9  
6. IANA Considerations .......................................... 10  
7. Acknowledgments .............................................. 11  
8. References .................................................... 12  
   8.1. Normative References .................................. 12  
   8.2. Informative References ................................ 12  
Appendix A. History of change ................................. 13  
Author’s Address ............................................... 14  
Intellectual Property and Copyright Statements ............. 15
1. Introduction

SDP is defined for general real-time multimedia session description purposes and is extended for the negotiation of media encodings incorporated with other protocols like SIP [3].

As per RFC2327, a session description consists of a session-level description (details that apply to the whole session and all media streams) and optionally several media-level descriptions (details that apply onto to a single media stream). In general, session-level values are the default for all media unless overridden by an equivalent media-level value.

In the media-level descriptions, each media description starts with an "m=" field, and is terminated by either the next "m=" field or by the end of the session description. A media field also has several sub-fields:

m=<media> <port> <transport> <fmt list>

The fourth and subsequent sub-fields are media formats. For audio and video, these will normally be a media payload type as defined in the RTP Audio/Video Profile. This list contains the payload formats that may be used in the session.

In application, there may have different attributes for different media formats in one media stream. One example is the number of samples per packet (ptime) for RTP audio formats, media tools may support different packetization time for different audio formats.

Another example is to provide different transport address and port for different media formats in some specific applications like transcoding gate way.

The requirement to support codec specific parameters in SDP is analyzed here, examples of the possible extension is also given in this document.

Part of the topic comes from the discussion on the mailing list. The purpose of this document is to act as a basis for discussion and hope to get the best solution from the community.
2. Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in RFC 2119 [2] and indicate requirement levels for compliant implementations.
3. Requirement Analysis

3.1. Requirement to support codec specific ptime

In VoIP, the number of samples per packet for RTP audio format is commonly called "ptime", the definition for it in RFC 2327 is like this:

\[ a=\text{ptime}:<\text{packet time}> \]

This gives the length of time in milliseconds represented by the media in a packet. This is probably only meaningful for audio data. It should not be necessary to know ptime to decode RTP or VAT audio, and it is intended as a recommendation for the encoding/packetisation of audio. It is a media attribute, and is not dependent on charset.

Although it should not be a key to decode RTP packet, any ptime should be accepted, in practice, many implementations don't support such flexibility. If the received packet is not compatible to specified mode, the quality of voice will be bad. This always cause interworking problem in deployment.

The requirement to support codec specific ptime may be summarized as follows:

Req 1: Should support different ptime value for different codec formats in one media stream.

Req 2: The content of codec specific ptime may be a list of supported values, e.g. 10 20 30.

Req 3: The content of codec specific ptime may be contain scope of supported ptime values, e.g. 10-30.

Req 4: If there is no codec specific ptime value, the media level ptime value shall be used.

3.2. Requirement to support codec specific transport address

The transport address information for media stream contains two parts: connection address and port number.

Connection address is defined like this:
c=<network type> <address type> <connection address>

It may be a session level description or media level description (in this case override the session level one for this media).

Port number is one of the sub-fields in the definition of media stream information:

m=<media> <port> <transport> <fmt list>

From this we can see that we can not provide codec specific transport address information now.

There may have requirements to support codec specific transport address information:

1) Direct transcoding function

A direct transcoding model is introduced in [4].

| A | T | B |
| a | a-b,a-c | c |

Figure 1: Direct transcoding model

In this model, when forwarding INVITE the transcoding function T can directly add his capability. The purpose is that if the called party B only support c, it will be connected to T, where as if it support a, it may still be connected to A directly.

To fulfill this function, it’s required that T can provide transport address information for the codecs added by it, and with the other media information unchanged.

For example, the SDP modified by T may be like this:
v=0
o=A 2890844526 2890844526 IN IP4 host.anywhere.com
s=
c=IN IP4 A.anywhere.com
t=0 0
m=audio 62986 RTP/AVP 0 4
a=rtpmap:0 PCMU/8000 // supported by A
a=rtpmap:4 G723/8000 // Added by T
[XXX T's address information] // Added by T

The requirement to support codec specific transport address information may be summarized as follows:

Req 1: Should support different transport address information for different codec formats in one media stream.

Req 2: The content of transport address information may contain connect address, port number or both.

Req 3: If there are no codec specific transport address information, the session level or media level one shall be used.

3.3. Summary

From the discussion above, we can see that there may have requirements for codec specific parameters. The parameter may contain ptime, transport address information or others.
4. Evaluation of possible solutions

This section evaluates three possible solutions: extending the SDP, using grouping method and using SDPng.

4.1. Extending the SDP

Based on the discussion above, an example for the extension of SDP is illustrated here.

4.1.1. Format

\texttt{a=x-codecparam:<codec> <codec specific parameters>}

This is a codec level attribute. Each codec may have several parameters each described in one line. The detail of each parameter should be defined separately.

1) ptime negotiation

\texttt{a=x-codecparam:<codec> ptime=<list of scope of codec>}

\texttt{e.g. a=x-codecparam:0 ptime=10, 20, 30}

2) transport address negotiation

\texttt{a=X-codecparam:<codec> c=<network type> <address type> <connection address>}

\texttt{a=X-codecparam:<codec> port=<port number>}

\texttt{e.g. a=x-codecparam:0 c=IN IP4 127.0.0.1}

\texttt{a=x-codecparam:0 port=12345}

4.2. Using grouping method

\texttt{RFC3388} provide the method to group different m= lines in SDP. To extend the grouping mechanism, so that we can define multiple m= lines with different codec parameters, but all of them belong to the same actual media. Further consideration needs to be done.

4.3. Using SDPng

SDPng is more structural which support the description and extention of codec specific param more easily. But the problem is that the actual deployment of SDPng is very less.
5. Security Considerations

There are no specific security issues here.
6. IANA Considerations

None.
7. Acknowledgments

Part of the information come from the discussion on ietf mailing list. Thanks for the suggestions and discussions from r.parthasarathi, christer.holmberg, Lyons,Terry and Colin Perkins.
8. References

8.1. Normative References


8.2. Informative References

Appendix A. History of change

The consideration of using grouping method and SDPng is added.
Author’s Address

Peili Xu
Huawei Technologies
Bantian
Shenzhen, Longgang  518129
China

Phone: +86 755 28780808
Email: xupeili@huawei.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 (2006). 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.