

MMUSIC	R. Gilman
Internet-Draft	Independent
Intended status: Standards Track	R. Even
Expires: May 03, 2012	Gesher Erove Ltd
	F. Andreasen
	Cisco Systems
	October 31, 2011

SDP Media Mapabilities Negotiation  
draft-ietf-mmusic-sdp-media-capabilities-12

## Abstract

Session Description Protocol (SDP) capability negotiation provides a general framework for indicating and negotiating capabilities in SDP. The base framework defines only capabilities for negotiating transport protocols and attributes. In this document, we extend the framework by defining media capabilities that can be used to negotiate media types and their associated parameters.

## 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/>.

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."

This Internet-Draft will expire on May 03, 2012.

## Copyright Notice

Copyright (c) 2011 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material

may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

## [Table of Contents](#)

- \*1. [Introduction](#)
- \*2. [Terminology](#)
- \*3. [SDP Media Capabilities](#)
  - \*3.1. [Requirements](#)
  - \*3.2. [Solution Overview](#)
  - \*3.3. [New Capability Attributes](#)
    - \*3.3.1. [The Media Format Capability Attributes](#)
    - \*3.3.2. [The Media Format Parameter Capability Attribute](#)
      - \*3.3.2.1. [Media Format Parameter Concatenation Rule](#)
    - \*3.3.3. [The Media-Specific Capability Attribute](#)
    - \*3.3.4. [New Configuration Parameters](#)
      - \*3.3.4.1. [The Media Configuration Parameter \(m=\)](#)
      - \*3.3.4.2. [The Payload Type Number Mapping Parameter \(pt=\)](#)
      - \*3.3.4.3. [The Media Type Parameter](#)
    - \*3.3.5. [The Latent Configuration Attribute](#)
    - \*3.3.6. [Enhanced Potential Configuration Attribute](#)
      - \*3.3.6.1. [Returning Capabilities in the Answer](#)
      - \*3.3.6.2. [Payload Type Number Mapping](#)
      - \*3.3.6.3. [Processing of Media-Format-Related Conventional Attributes for Potential Configurations](#)
    - \*3.3.7. [Substitution of Media Payload Type Numbers in Capability Attribute Parameters](#)

- \*3.3.8. [The Session Capability Attribute](#)
- \*3.4. [Offer/Answer Model Extensions](#)
  - \*3.4.1. [Generating the Initial Offer](#)
  - \*3.4.2. [Generating the Answer](#)
  - \*3.4.3. [Offerer Processing of the Answer](#)
  - \*3.4.4. [Modifying the Session](#)
- \*4. [Examples](#)
  - \*4.1. [Alternative Codecs](#)
  - \*4.2. [Alternative Combinations of Codecs \(Session Configurations\)](#)
  - \*4.3. [Latent Media Streams](#)
- \*5. [IANA Considerations](#)
  - \*5.1. [New SDP Attributes](#)
  - \*5.2. [New SDP Option Tag](#)
  - \*5.3. [New SDP Capability Negotiation Parameters](#)
- \*6. [Security Considerations](#)
- \*7. [Changes from previous versions](#)
  - \*7.1. [Changes from version 11](#)
  - \*7.2. [Changes from version 10](#)
  - \*7.3. [Changes from version 09](#)
  - \*7.4. [Changes from version 08](#)
  - \*7.5. [Changes from version 04](#)
  - \*7.6. [Changes from version 03](#)
  - \*7.7. [Changes from version 02](#)
  - \*7.8. [Changes from version 01](#)
  - \*7.9. [Changes from version 00](#)
- \*8. [Acknowledgements](#)

\*9. [References](#)

\*9.1. [Normative References](#)

\*9.2. [Informative References](#)

\*[Authors' Addresses](#)

## [1. Introduction](#)

Session Description Protocol (SDP) capability negotiation [\[RFC5939\]](#) provides a general framework for indicating and negotiating capabilities in SDP [\[RFC4566\]](#). The base framework defines only capabilities for negotiating transport protocols and attributes. The [\[RFC5939\]](#) document lists some of the issues with the current SDP capability negotiation process. An additional real life case is to be able to offer one media stream (e.g. audio) but list the capability to support another media stream (e.g. video) without actually offering it concurrently.

In this document, we extend the framework by defining media capabilities that can be used to indicate and negotiate media types and their associated format parameters. This document also adds the ability to declare support for media streams, the use of which can be offered and negotiated later, and the ability to specify session configurations as combinations of media stream configurations. The definitions of new attributes for media capability negotiation are chosen to make the translation from these attributes to "conventional" SDP [\[RFC4566\]](#) media attributes as straightforward as possible in order to simplify implementation. This goal is intended to reduce processing in two ways: each proposed configuration in an offer may be easily translated into a conventional SDP media stream record for processing by the receiver; and the construction of an answer based on a selected proposed configuration is straightforward.

## [2. Terminology](#)

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 [\[RFC2119\]](#) and indicate requirement levels for compliant RTP implementations.

"Base Attributes": Conventional SDP attributes appearing in the base configuration of a media block.

"Base Configuration": The media configuration represented by a media block exclusive of all the capability negotiation attributes defined in this document, the base capability negotiation document [\[RFC5939\]](#), or any other capability negotiation document. In an offer SDP, the base configuration corresponds to the actual configuration as defined in [\[RFC5939\]](#).

"Conventional Attribute": Any SDP attribute other than those defined by the series of capability negotiation specifications.

"Conventional SDP": An SDP record devoid of capability negotiation attributes.

"Media Capability": A media encoding, typically a media subtype such as PCMU, H263-1998, or T38.

### **3. SDP Media Capabilities**

The SDP capability negotiation [\[RFC5939\]](#) discusses the use of any SDP [\[RFC4566\]](#) attribute (a=) under the attribute capability "acap". The limitations of using acap for fmp and rtpmap in a potential configuration are described in [\[RFC5939\]](#); for example they can be used only at the media level since they are media level attributes. The [\[RFC5939\]](#) does not provide a way to exchange media-level capabilities prior to the actual offer of the associated media stream. This section provides an overview of extensions providing an SDP Media Capability negotiation solution offering more robust capabilities negotiation. This is followed by definitions of new SDP attributes for the solution and its associated updated offer/answer procedures [\[RFC3264\]](#)

#### **3.1. Requirements**

The capability negotiation extensions requirements considered herein are as follows.

**REQ-01:** Support the specification of alternative (combinations of) media formats (codecs) in a single media block.

**REQ-02:** Support the specification of alternative media format parameters for each media format.

**REQ-03:** Retain backward compatibility with conventional SDP. Ensure that each and every offered configuration can be easily translated into a corresponding SDP media block expressed with conventional SDP lines.

**REQ-04:** Ensure the scheme operates within the offer/answer model in such a way that media formats and parameters can be agreed upon with a single exchange.

**REQ-05:** Provide the ability to express offers in such a way that the offerer can receive media as soon as the offer is sent. (Note that the offerer may not be able to render received media prior to exchange of keying material.)

**REQ-06:** Provide the ability to offer latent media configurations for future negotiation.

**REQ-07:**

Provide reasonable efficiency in the expression of alternative media formats and/or format parameters, especially in those cases in which many combinations of options are offered.

**REQ-08:** Retain the extensibility of the base capability negotiation mechanism.

**REQ-09:** Provide the ability to specify acceptable combinations of media streams and media formats. For example, offer a PCMU audio stream with an H264 video stream, or a G729 audio stream with an H263 video stream. This ability would give the offerer a means to limit processing requirements for simultaneous streams. This would also permit an offer to include the choice of an audio/T38 stream or an image/T38 stream, but not both.

Other possible extensions have been discussed, but have not been treated in this document. They may be considered in the future. Three such extensions are:

**FUT-01:** Provide the ability to mix, or change, media types within a single media block. Conventional SDP does not support this capability explicitly; the usual technique is to define a media subtype that represents the actual format within the nominal media type. For example, T.38 FAX as an alternative to audio/PCMU within an audio stream is identified as audio/T38; a separate FAX stream would use image/T38.

**FUT-02:** Provide the ability to support multiple transport protocols within an active media stream without reconfiguration. This is not explicitly supported by conventional SDP.

**FUT-03:** Provide capability negotiation attributes for all media-level SDP line types in the same manner as already done for the attribute type, with the exception of the media line type itself. The media line type is handled in a special way to permit compact expression of media coding/format options. The line types are bandwidth ("b="), information ("i="), connection data ("c="), and, possibly, the deprecated encryption key ("k=").

### **3.2. Solution Overview**

The solution consists of new capability attributes corresponding to conventional SDP line types, new parameters for the pcfg, acfg, and the new lcfg attributes extending the base attributes from [\[RFC5939\]](#), and a use of the pcfg attribute to return capability information in the SDP answer.

Several new attributes are defined in a manner that can be related to the capabilities specified in a media line, and its corresponding rtpmap and fmtp attributes.

\*A new media attribute ("a=rmcap") defines RTP-based media capabilities in the form of a media subtype (e.g. "PCMU"), and its encoding parameters (e.g. "/8000/2"). Each resulting media format type/subtype capability has an associated handle called a media capability number. The encoding parameters are as specified for the rtpmap attribute defined in [\[RFC4566\]](#), without the payload type number part.

\*A new media attribute ("a=omcap") defines other (non RTP-based) media capabilities in the form of a media subtype only (e.g. "T38"). Each resulting media format type/subtype capability has an associated handle called a media capability number.

\*A new attribute ("a=mfcap") specifies media format parameters associated with one or more media capabilities. The mfcap attribute is used primarily to associate the formatting capabilities normally carried in the fmpm attribute. Note that media format parameters can be used with RTP and non-RTP based media formats.

\*A new attribute ("a=mscap") that specifies media parameters associated with one or more media capabilities. The mscap attribute is used to associate capabilities with attributes other than fmpm or rtpmap, for example, the rtcp-fb attribute defined in [\[RFC4585\]](#).

\*A new attribute ("a=lcfg") specifies latent media stream configurations when no corresponding media line ("m=") is offered. An example is the offer of latent configurations for video even though no video is currently offered. If the peer indicates support for one or more offered latent configurations, the corresponding media stream(s) may be added via a new offer/answer exchange.

\*A new attribute ("a=sescap") is used to specify an acceptable combination of simultaneous media streams and their configurations as a list of potential and/or latent configurations.

New parameters are defined for the potential configuration (pcfg), latent configuration (lcfg), and accepted configuration (acfg) attributes to associate the new attributes with particular configurations.

\*A new parameter type ("m=") is added to the potential configuration ("a=pcfg:") attribute and the actual configuration ("a=acfg:") attribute defined in [\[RFC5939\]](#), and to the new latent configuration ("a=lcfg:") attribute. This permits specification of media capabilities (including their associated parameters) and

combinations thereof for the configuration. For example, the "a=pcfg:" line might specify PCMU and telephone events [\[RFC4733\]](#) or G.729B and telephone events as acceptable configurations. The "a=acfg:" line in the answer would specify the configuration chosen.

\*A new parameter type ("pt=") is added to the potential configuration, actual configuration, and latent configuration attributes. This parameter associates RTP payload type numbers with the referenced RTP-based media capabilities, and is appropriate only when the transport protocol uses RTP.

\*A new parameter type ("mt=") is used to specify the media type for latent configurations.

Special processing rules are defined for capability attribute arguments in order to reduce the need to replicate essentially-identical attribute lines for the base configuration and potential configurations.

\*A substitution rule is defined for any capability attribute to permit the replacement of the (escaped) media capability number with the media format identifier (e.g., the payload type number in audio/video profiles).

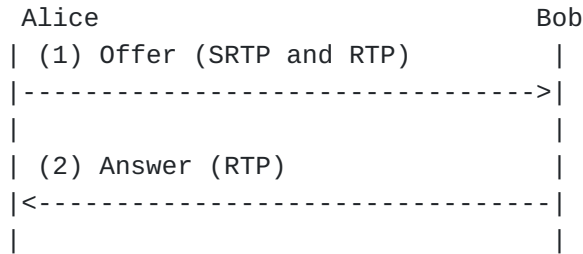
\*Replacement rules are defined for the conventional SDP equivalents of the mfcap and mscap capability attributes. This reduces the necessity to use the deletion qualifier in the a=pcfg parameter in order to ignore rtpmap, fmp, and certain other attributes in the base configuration.

\*An argument concatenation rule is defined for mfcap attributes which refer to the same media capability number. This makes it convenient to combine format options concisely by associating multiple mfcap lines with multiple media capabilities.

This document extends the base protocol extensions to the offer/answer model that allow for capabilities and potential configurations to be included in an offer. Media capabilities constitute capabilities that can be used in potential and latent configurations. Whereas potential configurations constitute alternative offers that may be accepted by the answerer instead of the actual configuration(s) included in the "m=" line(s) and associated parameters, latent configurations merely inform the other side of possible configurations supported by the entity. Those latent configurations may be used to guide subsequent offer/answer exchanges, but they are not part of the current offer/answer exchange.

The mechanism is illustrated by the offer/answer exchange below, where Alice sends an offer to Bob:





Alice's offer includes RTP and SRTP as alternatives. RTP is the default, but SRTP is the preferred one (long lines are folded to fit the margins):

```
*v=0
o=- 25678 753849 IN IP4 192.0.2.1
s=
c=IN IP4 192.0.2.1
t=0 0
a=creq:med-v0
m=audio 3456 RTP/AVP 0 18
a=tcap:1 RTP/SAVP RTP/AVP
a=rtpmap:0 PCMU/8000/1
a=rtpmap:18 G729/8000/1
a=fmtp:18 annexb=yes
a=rmcap:1,4 g729/8000/1
a=rmcap:2 PCMU/8000/1
a=rmcap:5 telephone-event/8000
a=mfcap:1 annexb=no
a=mfcap:4 annexb=yes
a=mfcap:5 0-11
a=acap:1 crypto:1 AES_CM_128_HMAC_SHA1_32 \
inline:NzB4d1BINUAvLEw6UzF3WSJ+PSdFcGdUJShpX1Zj|2^20|1:32
a=pcfg:1 m=4,5|1,5 t=1 a=1 pt=1:100,4:101,5:102
a=pcfg:2 m=2 t=1 a=1 pt=2:103
a=pcfg:3 m=4 t=2 pt=4:18
```

The required base and extensions are provided by the "a=creq" attribute defined in [\[RFC5939\]](#), with the option tag "med-v0", which indicates that the extension framework defined here, must be supported. The Base level support is implied since it is required for the extensions.

The "m=" line indicates that Alice is offering to use plain RTP with PCMU or G.729B. The media line implicitly defines the default transport protocol (RTP/AVP in this case) and the default actual configuration. The "a=tcap:1" line, specified in the base protocol, defines transport protocol capabilities, in this case Secure RTP (SAVP profile) as the first option and RTP (AVP profile) as the second option.

The "a=rmcap:1,4" line defines two G.729 RTP-based media format capabilities, numbered 1 and 4, and their encoding rate. The

capabilities are of media type "audio" and subtype G729. Note that the media subtype is explicitly specified here, rather than RTP payload type numbers. This permits the assignment of payload type numbers in the media stream configuration specification. In this example, two G.729 subtype capabilities are defined. This permits the declaration of two sets of formatting parameters for G.729.

The "a=rmcap:2" line defines a G.711 mu-law capability, numbered 2.

The "a=rmcap:5" line defines an audio telephone-event capability, numbered 5.

The "a=mfcap:1" line specifies the fmp format parameters for capability 1 (offerer will not accept G.729 Annex B packets).

The "a=mfcap:4" line specifies the fmp format parameters for capability 4 (offerer will accept G.729 Annex B packets).

The "a=mfcap:5" line specifies the fmp format parameters for capability 5 (the DTMF touchtones 0-9,\*,#).

The "a=acap:1" line specified in the base protocol provides the "crypto" attribute which provides the keying material for SRTP using SDP security descriptions.

The "a=pcfg:" attributes provide the potential configurations included in the offer by reference to the media capabilities, transport capabilities, attribute capabilities and specified payload type number mappings. Three explicit alternatives are provided; the lowest-numbered one is the preferred one. The "a=pcfg:1 ..." line specifies media capabilities 4 and 5, i.e., G.729B and DTMF, or media capability 1 and 5, i.e., G.729 and DTMF. Furthermore, it specifies transport protocol capability 1 (i.e. the RTP/SAVP profile - secure RTP), and the attribute capability 1, i.e. the crypto attribute provided. Lastly, it specifies a payload type number mapping for (RTP-based) media capabilities 1, 4, and 5, thereby permitting the offerer to distinguish between encrypted media and unencrypted media received prior to receipt of the answer.

Use of unique payload type numbers in alternative configurations is not required; codecs such as AMR-WB [\[RFC4867\]](#) have the potential for so many combinations of options that it may be impractical to define unique payload type numbers for all supported combinations. If unique payload type numbers cannot be specified, then the offerer will be obliged to wait for the SDP answer before rendering received media. For SRTP using SDES inline keying [\[RFC4568\]](#), the offerer will still need to receive the answer before being able to decrypt the stream.

The second alternative ("a=pcfg:2 ...") specifies media capability 2, i.e. PCMU, under the RTP/SAVP profile, with the same SRTP key material.

The third alternative ("a=pcfg:3 ...") offers G.729B unsecured; its only purpose in this example is to show a preference for G.729B over PCMU.

The media line, with any qualifying attributes such as fmp or rtpmap, is itself considered a valid configuration; it is assumed to be the lowest preference.

Bob receives the SDP offer from Alice. Bob supports G.729B, PCMU, and telephone events over RTP, but not SRTP, hence he accepts the potential

configuration 3 for RTP provided by Alice. Bob generates the following answer:

```
*v=0
o=- 24351 621814 IN IP4 192.0.2.2
s=
c=IN IP4 19x2.0.2.2
t=0 0
a=csup:med-v0
m=audio 4567 RTP/AVP 18
a=rtpmap:18 G729/8000
a=fmtp:18 annexb=yes
a=acfg:3 m=4 t=2 pt=4:18
```

Bob includes the "a=csup" and "a=acfg" attributes in the answer to inform Alice that he can support the med-v0 level of capability negotiations. Note that in this particular example, the answerer supported the capability extensions defined here, however had he not, he would simply have processed the offer based on the offered PCMU and G.729 codecs under the RTP/AVP profile only. Consequently, the answer would have omitted the "a=csup" attribute line and chosen one or both of the PCMU and G.729 codecs instead. The answer carries the accepted configuration in the "m=" line along with corresponding rtpmap and/or fmtp parameters, as appropriate.

Note that per the base protocol, after the above, Alice MAY generate a new offer with an actual configuration ("m=" line, etc.) corresponding to the actual configuration referenced in Bob's answer (not shown here).

### 3.3. New Capability Attributes

In this section, we present the new attributes associated with indicating the media capabilities for use by the SDP Capability negotiation. The approach taken is to keep things similar to the existing media capabilities defined by the existing media descriptions ("m=" lines) and the associated "rtpmap" and "fmtp" attributes. We use media subtypes and "media capability numbers" to link the relevant media capability parameters. This permits the capabilities to be defined at the session level and be used for multiple streams, if desired. For RTP-based media formats, payload types are then specified at the media level (see [Section 3.3.4.2](#)).

A media capability merely indicates possible support for the media type and media format(s) in question. In order to actually use a media capability in an offer/answer exchange, it MUST be referenced in a potential configuration.

Media capabilities can be provided at the session-level and/or the media-level. Media capabilities provided at the session level may be referenced in any pcfg or lcfg attribute at the media level (consistent with the media type), whereas media capabilities provided at the media

level may be referenced only by the pcfg or lcfg attribute within that media stream only. In either case, the scope of the <med-cap-num> is the entire session description. This enables each media capability to be uniquely referenced across the entire session description (e.g. in a potential configuration).

### 3.3.1. The Media Format Capability Attributes

Media subtypes can be expressed as media format capabilities by use of the "a=rmcap" and "a=omcap" attributes. The "a=rmcap" attribute MUST be used for RTP-based media whereas the "a=omcap" attribute MUST be used for non-RTP-based (other) media formats. The two attributes are defined as follows:

```
a=rmcap:<media-cap-num-list> <encoding-name>/<clock-rate>
                               [/<encoding-parms>]
```

```
a=omcap:<media-cap-num-list> <format-name>
```

where <media-cap-num-list> is a (list of) media capability number(s) used to number a media format capability, the <encoding name> is the media subtype e.g. H263-1998 or PCMU, <clock rate> is the encoding rate, and <encoding parms> are the media encoding parameters for the media subtype;. All media format capabilities in the list are assigned to the same media type/subtype. Each occurrence of the rmcap and omcap attribute MUST use unique values in their <media-cap-num-list>; the media capability numbers are shared between the two attributes and the numbers MUST be unique across the entire SDP session. In short, the rmcap and omcap attributes define media capabilities and associates them with a media capability number in the same manner as the rtpmap attribute defines them and associates them with a payload type number. Additionally, the attributes allow multiple capability numbers to be defined for the media format in question. This permits the media format to be associated with different media parameters in different configurations.

In ABNF, we have:

```

media-capability-line = rtp-mcap / non-rtp-mcap

rtp-mcap                = "a=rmcap:" media-cap-num-list
                        1*WSP encoding-name "/" clock-rate
                        ["/" encoding-params]
non-rtp-mcap            = "a=omcap:" media-cap-num-list 1*WSP format-name
media-cap-num-list      = media-cap-num-element
                        *["," media-cap-num-element]
media-cap-num-element    = media-cap-num
                        / media-cap-num-range
media-cap-num-range     = media-cap-num "-" media-cap-num
media-cap-num            = 1*10(DIGIT)
encoding-name           = token ; defined in RFC4566
clock-rate              = 1*10(DIGIT)
encoding-params         = token
format-name             = token ;defined in RFC4566

```

The encoding-name, clock-rate and encoding-params are as defined to appear in an rtpmap attribute for each media type/subtype. Thus, it is easy to convert an rmcap attribute line into one or more rtpmap attribute lines, once a payload type number is assigned to a media-cap-num (see [Section 3.3.5](#)).

The format-name is a media format description for non-RTP based media as defined for the <fmt> part of the media description ("m=" line) in [RFC4566](#). In simple terms, it's the name of the media format, e.g. "t38".

The "rmcap" and "omcap" attributes can be provided at the session-level and/or the media-level. There can be more than one rmcap plus one omcap attribute at the session or media level (i.e at most one of each at the session-level and at most one of in each media description). Each media-cap-num MUST be unique within the entire SDP record; it is used to identify that media capability in potential, latent and actual configurations, and in other attribute lines as explained below. Note that the media-cap-num values are shared between the rmcap and omcap attributes, and hence the uniqueness requirement applies to the union of them. When the media capabilities are used in a potential, latent or actual configuration, the media formats referred by those configurations apply at the media level, irrespective of whether the media capabilities themselves were specified at the session or media level. In other words, the media capability applies to the specific media description associated with the configuration which invokes it. For example:

```

*v=0
a=rmcap:1 L16/8000/1
a=rmcap:2 L16/16000/2
a=rmcap:3,4 H263-1998/90000
m=audio 54320 RTP/AVP 0

```

```
a=pcfg:1 m=1|2, pt=1:99,2:98
m=video 66544 RTP/AVP 100
a=rtpmap:100 H264/90000
a=pcfg:10 m=3 pt=3:101
```

### 3.3.2. The Media Format Parameter Capability Attribute

This attribute is used to associate media format specific format parameters with one or more media capabilities. The form of the attribute is:

```
*a=mfcap:<media-caps> <list of parameters>
```

where <media-caps> permits the list of parameters to be associated with one or more media capabilities and the format parameters are specific to the type of media format. The mfcap lines map to a single traditional SDP ftmp attribute line (one for each entry in <media-caps>) of the form

```
*a=fmt:<fmt> <list of parameters>
```

where <fmt> is the media format description defined in RFC 4566 [\[RFC4566\]](#), as appropriate for the particular media stream. The mfcap attribute MUST be used to encode attributes for media capabilities, which would conventionally appear in an ftmp attribute. The existing acap attribute MUST NOT be used to encode ftmp attributes. The mfcap attribute adheres to [\[RFC4566\]](#) attribute production rules with

```
media-format-capability = "a=mfcap:" media-cap-num-list 1*WSP
                               fmt-specific-param-list
fmt-specific-param-list = text ; defined in RFC4566
```

Note that media format parameters can be used with RTP-based and non-RTP based media formats.

#### 3.3.2.1. Media Format Parameter Concatenation Rule

The appearance of media subtypes with a large number of formatting options (e.g., AMR-WB [\[RFC4867\]](#)) coupled with the restriction that only a single ftmp attribute can appear per media format, suggests that it is useful to create a combining rule for mfcap parameters which are associated with the same media capability number. Therefore, different mfcap lines MAY include the same media-cap-num in their media-cap-num-list. When a particular media capability is selected for processing, the parameters from each mfcap line which references the particular capability number in its media-cap-num-list are concatenated together via ";", in the order the mfcap attributes appear in the SDP record, to form the equivalent of a single ftmp attribute line. This permits one

to define a separate mfcap line for a single parameter and value that is to be applied to each media capability designated in the media-cap-num-list. This provides a compact method to specify multiple combinations of format parameters when using codecs with multiple format options. Note that order-dependent parameters SHOULD be placed in a single mfcap line to avoid possible problems with line rearrangement by a middlebox.

Format parameters are not parsed by SDP; their content is specific to the media type/subtype. When format parameters for a specific media capability are combined from multiple a=mfcap lines which reference that media capability, the format-specific parameters are concatenated together and separated by ";" for construction of the corresponding format attribute (a=fmtp). The resulting format attribute will look something like the following (without line breaks):

```
a=fmtp:<fmt> <fmt-specific-param-list1>;  
          <fmt-specific-param-list2>;  
          ...
```

where <fmt> depends on the transport protocol in the manner defined in RFC4566. SDP cannot assess the legality of the resulting parameter list in the "a=fmtp" line; the user must take care to ensure that legal parameter lists are generated.

The "mfcap" attribute can be provided at the session-level and the media-level. There can be more than one mfcap attribute at the session or media level. The unique media-cap-num is used to associate the parameters with a media capability.

As a simple example, a G.729 capability is, by default, considered to support comfort noise as defined by Annex B. Capabilities for G.729 with and without comfort noise support may thus be defined by:

```
*a=rmcap:1,2 audio G729/8000  
a=mfcap:2 annexb:no
```

Media format capability 1 supports G.729 with Annex B, whereas media format capability 2 supports G.729 without Annex B.  
Example for H.263 video:

```
*a=rmcap:1 video H263-1998/90000  
a=rmcap:2 video H263-2000/90000  
a=mfcap:1 CIF=4;QCIF=2;F=1;K=1  
a=mfcap:2 profile=2;level=2.2
```

Finally, for six format combinations of the Adaptive MultiRate codec:

```
*a=rmcap:1-3 AMR/8000/1  
a=rmcap:4-6 AMR-WB/16000/1  
a=mfcap:1,2,3,4 mode-change-capability=1  
a=mfcap:5,6 mode-change-capability=2
```

```
a=mfcap:1,2,3,5 max-red=220
a=mfcap:3,4,5,6 octet-align=1
a=mfcap:1,3,5 mode-set=0,2,4,7
a=mfcap:2,4,6 mode-set=0,3,5,6
```

So that AMR codec #1, when specified in a pcfg attribute within an audio stream block (and assigned payload type number 98) as in

```
*a=pcfg:1 m=1 pt=1:98
```

is essentially equivalent to the following

```
*m=audio 49170 RTP/AVP 98
a=rtpmap:98 AMR/8000/1
a=fmtp:98 mode-change-capability=1; \
max-red=220; mode-set=0,2,4,7
```

and AMR codec #4 with payload type number 99, depicted by the potential configuration:

```
*a=pcfg:4 m=4, pt=4:99
```

is equivalent to the following:

```
*m=audio 49170 RTP/AVP 99
a=rtpmap:99 AMR-WB/16000/1
a=fmtp:99 mode-change-capability=1; octet-align=1; \
mode-set=0,3,5,6
```

and so on for the other four combinations. SDP could thus convert the media capabilities specifications into one or more alternative media stream specifications, one of which can be chosen for the answer.

### 3.3.3. The Media-Specific Capability Attribute

Media-specific attributes, beyond the rtpmap and fmtp attributes, may be associated with media capability numbers via a new media-specific attribute, mscap, of the following form:

```
a=mscap:<media caps star> <att field> <att value>
```

Where <media caps star> is a (list of) media capability number(s), <att field> is the attribute name, and <att value> is the value field for the named attribute. The media capability numbers may include a wildcard ("\*"), which will be used instead of any payload type mappings. In ABNF, we have:



```

media-specific-capability = "a=mscap:"
                           media-caps-star
                           1*WSP att-field ; from RFC4566
                           1*WSP att-value ; from RFC4566
media-caps-star            = media-cap-star-element
                           *["," media-cap-star-element]
media-cap-star-element    = media-cap-num [wildcard]
                           / media-cap-num-range [wildcard]
wildcard                  = "*"

```

Given an association between a media capability and a payload type number as specified by the pt= parameters in an lcfg or pcfg attribute line, a mscap line may be translated easily into a conventional SDP attribute line of the form

```
*a=<att field>":<fmt> <att value> ; <fmt> defined in \[RFC4566\]
```

A resulting attribute that is not a legal SDP attribute as specified by RFC4566 MUST be ignored by the receiver.

If a media capability number (or range) contains a wildcard character at the end, any payload type mapping specified for that media specific capability will be use the wildcard character instead of the payload type.

A single mscap line may refer to multiple media capabilities; this is equivalent to multiple mscap lines, each with the same attribute values (but different media capability numbers), one line per media capability.

Multiple mscap lines may refer to the same media capability, but, unlike the mfcap attribute, no concatenation operation is defined. Hence, multiple mscap lines applied to the same media capability is equivalent to multiple lines of the specified attribute in a conventional media record.

Here is an example with the rtcp-fb attribute, modified from an example in [\[RFC5104\]](#) (with the session-level and audio media omitted). If the offer contains a media block like the following (note the wildcard character),

```

*m=video 51372 RTP/AVP 98
a=rtpmap:98 H263-1998/90000
a=tcap:1 RTP/AVPF
a=rmcap:1 H263-1998/90000
a=mscap:1 rtcp-fb ccm tstr
a=mscap:1 rtcp-fb ccm fir
a=mscap:1* rtcp-fb ccm tmmbr smaxpr=120
a=pcfg:1 t=1 m=1 pt=1:98

```

and if the proposed configuration is chosen, then the equivalent media block would look like

```
*m=video 51372 RTP/AVPF 98
a=rtpmap:98 H263-1998/90000
a=rtcp-fb:98 ccm tstr
a=rtcp-fb:98 ccm fir
a=rtcp-fb:* ccm tmmbr smaxpr=120
```

#### 3.3.4. New Configuration Parameters

Along with the new attributes for media capabilities, new extension parameters are defined for use in the potential configuration, the actual configuration, and/or the new latent configuration defined in [Section 3.3.5](#).

##### 3.3.4.1. The Media Configuration Parameter (m=)

The media configuration parameter is used to specify the media encoding(s) and related parameters for a potential, actual, or latent configuration. Adhering to the ABNF for extension-config-list in [\[RFC5939\]](#) with

```
ext-cap-name = "m"
ext-cap-list = media-cap-num-list
               [*(BAR media-cap-num-list)]
```

we have

```
media-config-list = ["+" "m=" media-cap-num-list
                    [*(BAR media-cap-num-list)]
                    ; BAR is defined in RFC5939
                    ; media-cap-num-list is defined above
```

Alternative media configurations are separated by a vertical bar ("|"). The alternatives are ordered by preference, most-preferred first. When media capabilities are not included in a potential configuration at the media level, the media type and media format from the associated "m=" line will be used. The use of the plus sign ("+") is described in RFC5939.

##### 3.3.4.2. The Payload Type Number Mapping Parameter (pt=)

The payload type number mapping parameter is used to specify the payload type number to be associated with each media type in a potential, actual, or latent configuration. We define the payload type number mapping parameter, payload-number-config-list, in accordance with the extension-config-list format defined in [\[RFC5939\]](#). In ABNF:

```

payload-number-config-list = ["+"] "pt=" media-map-list
media-map-list = media-map *["," media-map]
media-map = media-cap-num ":" payload-type-number
                ; media-cap-num is defined in 3.3.1
payload-type-number = 1*3(DIGIT) ; RTP payload type number
                        / "*"      ; dummy

```

The example in [Section 3.3.7](#) shows how the parameters from the rmcap line are mapped to payload type numbers from the pcfg "pt" parameter. The use of the plus sign ("+") is described in RFC5939.

The "\*" value for payload-type-number is used in cases such as BFCP [\[RFC4583\]](#) in which the fmt list in the m-line is ignored.

A latent configuration represents a future capability, hence the pt= parameter is not directly meaningful in the lcfc attribute because no actual media session is being offered or accepted; it is permitted in order to tie any payload type number parameters within attributes to the proper media format. A primary example is the case of format parameters for the Redundant Audio Data (RED) payload, which are payload type numbers. Specific payload type numbers used in a latent configuration MAY be interpreted as suggestions to be used in any future offer based on the latent configuration, but they are not binding; the offerer and/or answerer may use any payload type numbers each deems appropriate. The use of explicit payload type numbers for latent configurations can be avoided by use of the parameter substitution rule of [Section 3.3.7](#). Future extensions are also permitted.

#### [3.3.4.3. The Media Type Parameter](#)

When a latent configuration is specified (always at the media level), indicating the ability to support an additional media stream, it is necessary to specify the media type (audio, video, etc.) as well as the format and transport type. The media type parameter is defined in ABNF as

```

media-type = ["+"] "mt=" media; media defined in RFC4566

```

At present, the media-type parameter is used only in the latent configuration attribute, and the use of the "+" prefix to specify that the entire attribute line is to be ignored if the mt= parameter is not understood, is unnecessary. However, if the media-type parameter is later added to an existing capability attribute such as pcfg, then the "+" would be useful. The media format(s) and transport type(s) are specified using the media configuration parameter ("m=") defined above, and the transport parameter ("t=") defined in [\[RFC5939\]](#), respectively.

### 3.3.5. The Latent Configuration Attribute

One of the goals of this work is to permit the exchange of supportable media configurations in addition to those offered or accepted for immediate use. Such configurations are referred to as "latent configurations". For example, a party may offer to establish a session with an audio stream, and, at the same time, announce its ability to support a video stream as part of the same session. The offerer can supply its video capabilities by offering one or more latent video configurations along with the media stream for audio; the responding party may indicate its ability and willingness to support such a video session by returning a corresponding latent configuration.

Latent configurations returned in SDP answers must match offered latent configurations (or parameter subsets thereof). Therefore, it is appropriate for the offering party to announce most, if not all, of its capabilities in the initial offer. This choice has been made in order to keep the size of the answer more compact by not requiring acap, rmcap, tcap, etc. lines in the answer.

Latent configurations may be announced by use of the latent configuration attribute, which is defined in a manner very similar to the potential configuration attribute. The latent configuration attribute combines the properties of a media line and a potential configuration. The media type (mt=) and the transport protocol(s) (t=) MUST be specified since the latent configuration is independent of any media line present. In most cases, the media configuration (m=) parameter MUST be present as well (see [Section 4](#) for examples). The lcfg attribute is a media level attribute and, like a media line, it ends the session level of the session description if it appears before any media line.

Each media line in an SDP description represents an offered simultaneous media stream, whereas each latent configuration represents an additional stream which may be negotiated in a future offer/answer exchange. Session capability attributes may be used to determine whether a latent configuration may be used to form an offer for an additional simultaneous stream or to reconfigure an existing stream in a subsequent offer/answer exchange.

The latent configuration attribute is of the form:

```
a=lcfg:<config-number> <latent-cfg-list>
```

which adheres to the [\[RFC4566\]](#) "attribute" production with att-field and att-value defined as:

```
att-field   = "lcfg"
att-value   = config-number 1*WSP lcfg-cfg-list
config-number = 1*10(DIGIT) ; defined in RFC5234
lcfg-cfg-list = media-type 1*WSP pot-cfg-list
                ; as defined in RFC5939
                ; and extended herein
```

The media-type (mt=) parameter identifies the media type (audio, video, etc.) to be associated with the latent media stream, and MUST be present. The pot-cfg-list MUST contain a transport-protocol-config-list (t=) parameter and a media-config-list (m=) parameter. The pot-cfg-list MUST NOT contain more than one instance of each type of parameter list. As specified in [\[RFC5939\]](#), the use of the "+" prefix with a parameter indicates that the entire configuration MUST be ignored if the parameter is not understood; otherwise, the parameter itself may be ignored.

Media stream payload numbers are not assigned by a latent configuration. Assignment will take place if and when the corresponding stream is actually offered via an m-line in a later exchange. The payload-number-config-list is included as a parameter to the lcfg attribute in case it is necessary to tie payload numbers in attribute capabilities to specific media capabilities.

If an lcfg attribute invokes an acap attribute that appears at the session level, then that attribute will be expected to appear at the session level of a subsequent offer when and if a corresponding media stream is offered. Otherwise, acap attributes which appear at the media level represent media-level attributes. Note, however, that rmcap, omcap, mfcap, mscap, and tcap attributes may appear at the session level because they always result in media-level attributes or m-line parameters.

The configuration numbers for latent configurations do not imply a preference; the offerer will imply a preference when actually offering potential configurations derived from latent configurations negotiated earlier. Note however that the offerer of latent configurations MAY specify preferences for combinations of potential and latent configurations by use of the sescap attribute defined in [Section 3.3.8](#). For example, if an SDP offer contains, say, an audio stream with pcfg: 1, and two latent video configurations, lcfg:2, and lcfg:3, then a session with one audio stream and one video stream could be specified by including "a=sescap:1 1,2|3". One audio stream and two video streams could be specified by including "a=sescap:2 1,2,3" in the offer. In order to permit combinations of latent and potential configurations in session capabilities, latent configuration numbers MUST be different from those used for potential configurations. This restriction is especially important if the offerer does not require cmed-v0 capability and the recipient of the offer doesn't support it. If the lcfg attribute is not recognized, the capability attributes intended to be

associated with it may be confused with those associated with a potential configuration of some other media stream.

If a cryptographic attribute, such as the SDES "a=crypto:" attribute [\[RFC4568\]](#), is referenced by a latent configuration through an acap attribute, any keying material required in the conventional attribute, such as the SDES key/salt string, MUST be included in order to satisfy formatting rules for the attribute. The actual value(s) of the keying material SHOULD be meaningless, and the receiver of the lcfig attribute MUST ignore the values.

### **3.3.6. Enhanced Potential Configuration Attribute**

The present work requires new extensions (parameters) for the pcfg attribute defined in the base protocol [\[RFC5939\]](#). The parameters and their definitions are "borrowed" from the definitions provided for the latent configuration attribute in [Section 3.3.5](#). The expanded ABNF definition of the pcfg attribute is

```
*a=pcfg: <config-number> [<pot-cfg-list>]
```

where

```
*config-number = 1*DIGIT ;defined in \[RFC5234\]  
pot-cfg-list = pot-config *(1*WSP pot-config)  
pot-config = / attribute-config-list / ;def in \[RFC5939\]  
transport-protocol-config-list / ;defined in \[RFC5939\]  
extension-config-list / ;\[RFC5939\]  
media-config-list / ; Section 3.3.4.1  
payload-number-config-list ; Section 3.3.4.2
```

Except for the extension-config-list, the pot-cfg-list MUST NOT contain more than one instance of each parameter list.

#### **3.3.6.1. Returning Capabilities in the Answer**

Potential and/or latent configuration attributes may be returned within an answer SDP to indicate the ability of the answerer to support alternative configurations of the corresponding stream(s). For example, an offer may include multiple potential configurations for a media stream and/or latent configurations for additional streams; the corresponding answer will indicate (via an acfg attribute) the configuration accepted and used to construct the base configuration for each active media stream in the reply, but the reply MAY also contain potential and/or latent configuration attributes, with parameters, to indicate which other offered configurations would be acceptable. This information is useful if it becomes desirable to reconfigure a media stream, e.g., to reduce resource consumption.

When potential and/or latent configurations are returned in an answer, all numbering MUST refer to the configuration and capability attribute numbering of the offer. The offered capability attributes need not be

returned in the answer. The answer MAY include additional capability attributes and/or configurations (with distinct numbering). The parameter values of any returned pcfg or lcfcg attributes MUST be a subset of those included in the offered configurations or those added by the answerer; values may be omitted only if they were indicated as alternative sets, or optional, in the original offer. The parameter set indicated in the returned acfg attribute need not be repeated in a returned pcfg attribute. The answerer may return more than one pcfg attribute with the same configuration number if it is necessary to describe selected combinations of optional or alternative parameters. Similarly, one or more session capability attributes (a=sescap) may be returned to indicate which of the offered session capabilities is/are supportable by the answerer (see [Section 3.3.8.](#))

Note that, although the answerer MAY return capabilities beyond those included by the offerer, these capabilities MUST NOT be used to form any base level media description in the answer. For this reason, it is advisable for the offerer to include most, if not all, potential and latent configurations it can support in the initial offer, unless the size of the resulting SDP is a concern. Either party MAY later announce additional capabilities by renegotiating the session in a second offer/answer exchange.

#### [3.3.6.2. Payload Type Number Mapping](#)

When media capabilities defined in rmcap attributes are used in potential configuration lines, the transport protocol uses RTP and it is necessary to assign payload type numbers. In some cases, it is desirable to assign different payload type numbers to the same media capability when used in different potential configurations. One example is when configurations for AVP and SAVP are offered: the offerer would like the answerer to use different payload type numbers for encrypted and unencrypted media so that it (the offerer) can decide whether or not to render early media which arrives before the answer is received. This association of distinct payload type number(s) with different transport protocols requires a separate pcfg line for each protocol. Clearly, this technique cannot be used if the number of potential configurations exceeds the number of possible payload type numbers.

#### [3.3.6.3. Processing of Media-Format-Related Conventional Attributes for Potential Configurations](#)

In cases in which media capabilities negotiation is employed, SDP records are likely to contain conventional attributes such as rtpmap, fmp, and other media-format-related lines, as well as capability attributes such as rmcap, omcap, mfcap, and mscap which map into those conventional attributes when invoked by a potential configuration. In such cases, it MAY be appropriate to employ the delete-attributes option [\[RFC5939\]](#) in the attribute configuration list parameter in order to avoid the generation of conflicting fmp attributes for a particular

configuration. Any media-specific attributes in the media block which refer to media formats not used by the potential configuration MUST be ignored.

For example:

```
*v=0
o=- 25678 753849 IN IP4 192.0.2.1
s=
c=IN IP4 192.0.2.1
t=0 0
a=creq:med-v0
m=audio 3456 RTP/AVP 0 18 100
a=rtpmap:100 telephone-events
a=fmtp:100 0-11
a=rmcap:1 PCMU/8000
a=rmcap:2 g729/8000
a=rmcap:3 telephone-events/8000
a=mfcap:3 0-15
a=pcfg:1 m=2,3|1,3 a=-m pt=1:0,2:18,3:100
a=pcfg:2
```

In this example, PCMU is media capability 1, G729 is media capability 2, and telephone-event is media capability 3. The a=pcfg:1 line specifies that the preferred configuration is G.729 with extended dtmf events, second is G.711 mu-law with extended dtmf events, and the base media-level attributes are to be deleted. Intermixing of G.729, G.711, and "commercial" dtmf events is least preferred (the base configuration provided by the "m=" line, which is, by default, the least preferred configuration). The rtpmap and fmtp attributes of the base configuration are replaced by the rmcap and mfcap attributes when invoked by the proposed configuration. If the preferred configuration is selected, the SDP answer will look like

```
*v=0
o=- 25678 753849 IN IP4 192.0.2.1
s=
c=IN IP4 192.0.2.1
t=0 0
a=csup:med-v0
m=audio 6543 RTP/AVP 18 100
a=rtpmap:100 telephone-events/8000
a=fmtp:100 0-15
a=acfg:1 m=2,3 pt=1:0,2:18,3:100
```



### 3.3.7. Substitution of Media Payload Type Numbers in Capability Attribute Parameters

In some cases, for example, when an RFC 2198 redundancy audio subtype (RED) capability is defined in an mfcap attribute, the parameters to an attribute may contain payload type numbers. Two options are available for specifying such payload type numbers. They may be expressed explicitly, in which case they are bound to actual payload types by means of the payload type number parameter (pt=) in the appropriate potential or latent configuration. For example, the following SDP fragment defines a potential configuration with redundant G.711 mu-law:

```
*m=audio 45678 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=rmcap:1 PCMU/8000
a=rmcap:2 RED/8000
a=mfcap:2 0/0
a=pcfg:1 m=2,1 pt=2:98,1:0
```

The potential configuration is then equivalent to

```
*m=audio 45678 RTP/AVP 98 0
a=rtpmap:0 PCMU/8000
a=rtpmap:98 RED/8000
a=fmtp:98 0/0
```

A more general mechanism is provided via the parameter substitution rule. When an mfcap, mscap, or acap attribute is processed, its arguments will be scanned for a payload type number escape sequences of the following form (in ABNF):

```
*ptn-esc = "%m=" media-cap-num "%" ; defined in 3.3.1
```

If the sequence is found, the sequence is replaced by the payload type number assigned to the media capability number, as specified by the pt= parameter in the selected potential configuration; only actual payload type numbers are supported - wildcards are excluded. The sequence "%" (null digit string) is replaced by a single percent sign and processing continues with the next character, if any.

For example, the above offer sequence could have been written as

```
*m=audio 45678 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=rmcap:1 PCMU/8000
a=rmcap:2 RED/8000
a=mfcap:2 %m=1%/%m=1%
a=pcfg:1 m=2,1 pt=2:98,1:0
```

and the equivalent SDP is the same as above.

### 3.3.8. The Session Capability Attribute

The session capability attribute provides a means for the offerer and/or the answerer to specify combinations of specific media stream configurations which it is willing and able to support. Each session capability in an offer or answer MAY be expressed as a list of required potential configurations, and MAY include a list of optional potential and/or latent configurations.

The choices of session capabilities may be based on processing load, total bandwidth, or any other criteria of importance to the communicating parties. If the answerer supports media capabilities negotiation, and session configurations are offered, it MUST accept one of the offered configurations, or it MUST refuse the session.

Therefore, if the offer includes any session capabilities, it SHOULD include all the session capabilities the offerer is willing to support. The session capability attribute is described by:

```
"a=sescap:" <session num> <list of configs>
```

which corresponds to the standard value attribute definition with

```
att-field      = "sescap"
att-value      = session-num 1*WSP list-of-configs
                  [1*WSP optional-configs]
session-num    = 1*10(DIGIT) ; defined in RFC5234
list-of-configs = alt-config *["," alt-config]
optional-configs = "[" list-of-configs "]"
alt-config     = config-number *["|" config-number]
                  ; config-number defined in RFC5939
```

The session-num identifies the session; a lower-number session is preferred over a higher-numbered session. Each alt-config list specifies alternative media configurations within the session; preference is based on config-num as specified in [\[RFC5939\]](#). Note that the session preference order, when present, takes precedence over the individual media stream configuration preference order.

Use of session capability attributes requires that configuration numbers assigned to potential and latent configurations MUST be unique across the entire session; [\[RFC5939\]](#) requires only that pcfg configuration numbers be unique within a media description.

As an example, consider an endpoint that is capable of supporting an audio stream with either one H.264 video stream or two H.263 video streams with a floor control stream. The SDP offer might look like the following (offering audio, two H.263 video streams and BFCP)- the empty lines are added for readability only (not part of valid SDP):

```
*v=0
o=- 25678 753849 IN IP4 192.0.2.1
```

```
s=  
c=IN IP4 192.0.2.1  
t=0 0  
a=creq:med-v0  
a=sescap:2 1,2,3,5  
a=sescap:1 1,4
```

```
m=audio 54322 RTP/AVP 0  
a=rtpmap:0 PCMU/8000  
a=pcfg:1
```

```
m=video 22344 RTP/AVP 102  
a=rtpmap:102 H263-1998/90000  
a=fmtp:102 CIF=4;QCIF=2;F=1;K=1  
i= main video stream  
a=label:11  
a=pcfg:2  
a=rmcap:1 H264/90000  
a=mfcap:1 profile-level-id=42A01E; packetization-mode=2  
a=acap:1 label:13  
a=pcfg:4 m=1 a=1 pt=1:104
```

```
m=video 33444 RTP/AVP 103  
a=rtpmap:103 H263-1998/90000  
a=fmtp:103 CIF=4;QCIF=2;F=1;K=1  
i= secondary video (slides)  
a=label:12  
a=pcfg:3
```

```
m=application 33002 TCP/BFCP *  
a=setup:passive  
a=connection:new  
a=floorid:1 m-stream:11 12  
a=floor-control:s-only  
a=confid:4321  
a=userid:1234  
a=pcfg:5
```

If the answerer understands MediaCapNeg, but cannot support the Binary Floor Control Protocol, then it would respond with (invalid empty lines in SDP included again for readability):

```
*v=0  
o=- 25678 753849 IN IP4 192.0.2.1  
s=  
c=IN IP4 192.0.2.22
```

```
t=0 0
a=csup:med-v0
a=sescap:1 1,4
```

```
m=audio 23456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=acfg:1
```

```
m=video 41234 RTP/AVP 104
a=rtpmap:100 H264/90000
a=fmtp:104 profile-level-id=42A01E; packetization-mode=2
a=acfg:4 m=1 a=1 pt=1:104
```

```
m=video 0 RTP/AVP 103
a=acfg:3
```

```
m=application 0 TCP/BFCP *
a=acfg:5
```

An endpoint that doesn't support Media capabilities negotiation, but does support H.263 video, would respond with one or two H.263 video streams. In the latter case, the answerer may issue a second offer to reconfigure the session to one audio and one video channel using H.264 or H.263.

Session capabilities can include latent capabilities as well. Here's a similar example in which the offerer wishes to initially establish an audio stream, and prefers to later establish two video streams with chair control. If the answerer doesn't understand Media CapNeg, or cannot support the dual video streams or flow control, then it may support a single H.264 video stream. Note that establishment of the most favored configuration will require two offer/answer exchanges.

```
*v=0
o=- 25678 753849 IN IP4 192.0.2.1
s=
c=IN IP4 192.0.2.1
t=0 0
a=creq:med-v0
a=sescap:1 1,3,4,5
a=sescap:2 1,2
a=sescap:3 1

a=rmcap:1 H263-1998/90000
a=mfcap:1 CIF=4;QCIF=2;F=1;K=1
a=tcap:1 RTP/AVP TCP/BFCP
```

```

m=audio 54322 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=label:11
a=pcfg:1

m=video 22344 RTP/AVP 102
a=rtpmap:102 H264/90000
a=fmtp:102 profile-level-id=42A01E; packetization-mode=2
a=label:11
a=content:main
a=pcfg:2
a=lcfg:3 mt=video t=1 m=1 a=31,32 i=3
a=acap:31 label:12
a=acap:32 content:main

a=lcfg:4 mt=video t=1 m=1 a=41,42 i=4
a=acap:41 label:13
a=acap:42 content:slides

a=lcfg:5 mt=application m=51 t=51
a=tcap:51 TCP/BFCP
a=rmcap:51 *
a=acap:51 setup:passive
a=acap:52 connection:new
a=acap:53 floorid:1 m-stream:12 13
a=acap:54 floor-control:s-only
a=acap:55 confid:4321
a=acap:56 userid:1234

```

In this example, the default offer, as seen by endpoints which do not understand capabilities negotiation, proposes a PCMU audio stream and an H.264 video stream. Note that the offered lcfg lines for the video streams don't carry pt= parameters because they're not needed (payload type numbers will be assigned in the offer/answer exchange that establishes the streams). Note also that the three rmcap, mfcap, and tcap attributes used by lcfg:3 and lcfg:4 are included at the session level so they may be referenced by both latent configurations. As per [Section 3.3](#), the media attributes generated from the rmcap, mfcap, and tcap attributes are always media-level attributes. If the answerer supports Media CapNeg, and supports the most desired configuration, it would return the following SDP:

```

*v=0
o=- 25678 753849 IN IP4 192.0.2.1
s=
c=IN IP4 192.0.2.22
t=0 0
a=csup:med-v0
a=sescap:1 1,3,4,5

```

```
a=sescap:2 1,2
a=sescap:3 1

m=audio 23456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=acfg:1

m=video 0 RTP/AVP 102
a=pcfg:2
a=lcfg:3 mt=video t=1 m=1 a=31,32

a=lcfg:4 mt=video t=1 m=1 a=41,42

a=lcfg:5 mt=application t=2
```

This exchange supports immediate establishment of an audio stream for preliminary conversation. This exchange would presumably be followed at the appropriate time with a "reconfiguration" offer/answer exchange to add the video and chair control streams.

### [3.4. Offer/Answer Model Extensions](#)

EDITOR'S NOTE: SECTION NEEDS MORE ELABORATE PROCEDURES

In this section, we define extensions to the offer/answer model defined in RFC3264 [\[RFC3264\]](#) and [\[RFC5939\]](#) to allow for media capabilities, bandwidth capabilities, and latent configurations to be used with the SDP Capability Negotiation framework.

The [\[RFC5939\]](#) provides a relatively compact means to offer the equivalent of an ordered list of alternative media stream configurations (as would be described by separate m= lines and associated attributes). The attributes acap, mscap, mfcap and rmcap are designed to map somewhat straightforwardly into equivalent m= lines and conventional attributes when invoked by a pcfg, lcfg, or acfg attribute with appropriate parameters. The a=pcfg: lines, along with the m= line itself, represent offered media configurations. The a=lcfg: lines represent alternative capabilities for future use.

#### [3.4.1. Generating the Initial Offer](#)

When an endpoint generates an initial offer and wants to use the functionality described in the current document, it should identify and define the codecs it can support via rmcap, mfcap and mscap attributes. The SDP media line(s) should be made up with the configuration to be used if the other party does not understand capability negotiations (by default, this is the least preferred configuration). Typically, the media line configuration will contain the minimum acceptable capabilities. The offer MUST include the level of capability negotiation extensions needed to support this functionality in a "creq" attribute.

Preferred configurations for each media stream are identified following the media line. The present offer may also include latent configuration (lcfg) attributes, at the media level, describing media streams and/or configurations the offerer is not now offering, but which it is willing to support in a future offer/answer exchange. A simple example might be the inclusion of a latent video configuration in an offer for an audio stream.

#### 3.4.2. Generating the Answer

When the answering party receives the offer and if it supports the required capability negotiation extensions, it should select the most-preferred configuration it can support for each media stream, and build its answer accordingly. The configuration selected for each accepted media stream is placed into the answer as a media line with associated parameters and attributes. If a proposed configuration is chosen, the answer must include the supported extension attribute and each media stream for which a proposed configuration was chosen must contain an actual configuration (acfg) attribute to indicate just which pcfg attribute was used to build the answer. The answer should also include any potential or latent configurations the answerer can support, especially any configurations compatible with other potential or latent configurations received in the offer. The answerer should make note of those configurations it might wish to offer in the future.

#### 3.4.3. Offerer Processing of the Answer

When the offerer receives the answer, it should make note of any capabilities and/or latent configurations for future use. The media line(s) must be processed in the normal way to identify the media stream(s) accepted by the answer, if any. The acfg attribute, if present, may be used to verify the proposed configuration used to form the answer, and to infer the lack of acceptability of higher-preference configurations that were not chosen. Note that the base specification [\[RFC5939\]](#) requires the answerer to choose the highest preference configuration it can support, subject to local policies.

#### 3.4.4. Modifying the Session

If, at a later time, one of the parties wishes to modify the operating parameters of a session, e.g., by adding a new media stream, or by changing the properties used on an existing stream, it may do so via the mechanisms defined for offer/answer [\[RFC3264\]](#). If the initiating party has remembered the codecs, potential configurations, and latent configurations announced by the other party in the earlier negotiation, it may use this knowledge to maximize the likelihood of a successful modification of the session. Alternatively, the initiator may perform a new capabilities exchange as part of the reconfiguration. In such a

case, the new capabilities will replace the previously-negotiated capabilities. This may be useful if conditions change on the endpoint.

## [4. Examples](#)

In this section, we provide examples showing how to use the Media Capabilities with the SDP Capability Negotiation.

### [4.1. Alternative Codecs](#)

This example provide a choice of one of six variations of the adaptive multirate codec. In this example, the default configuration as specified by the media line is the same as the most preferred configuration. Each configuration uses a different payload type number so the offerer can interpret early media.

```
*v=0
o=- 25678 753849 IN IP4 192.0.2.1
s=
c=IN IP4 192.0.2.1
t=0 0
a=creq:med-v0
m=audio 54322 RTP/AVP 96
rtpmap:96 AMR-WB/16000/1
a=fmtp:96 mode-change-capability=1; max-red=220; \
mode-set=0,2,4,7
a=rmcap:1,3,5 audio AMR-WB/16000/1
a=rmcap:2,4,6 audio AMR/8000/1
a=mfcap:1,2,3,4 mode-change-capability=1
a=mfcap:5,6 mode-change-capability=2
a=mfcap:1,2,3,5 max-red=220
a=mfcap:3,4,5,6 octet-align=1
a=mfcap:1,3,5 mode-set=0,2,4,7
a=mfcap:2,4,6 mode-set=0,3,5,6
a=pcfg:1 m=1 pt=1:96
a=pcfg:2 m=2 pt=2:97
a=pcfg:3 m=3 pt=3:98
a=pcfg:4 m=4 pt=4:99
a=pcfg:5 m=5 pt=5:100
a=pcfg:6 m=6 pt=6:101
```

In the above example, media capability 1 could have been excluded from the first rmcap declaration and from the corresponding mfcap attributes, and the pcfg:1 attribute line could have been simply "pcfg:1".

The next example offers a video stream with three options of H.264 and 4 transports. It also includes an audio stream with different audio qualities: four variations of AMR, or AC3. The offer looks something like:



```
*v=0
o=- 25678 753849 IN IP4 192.0.2.1
s=An SDP Media NEG example
c=IN IP4 192.0.2.1
t=0 0
a=creq:med-v0
a=ice-pwd:speEc3QGZiNWpVLFJhQX
m=video 49170 RTP/AVP 100
c=IN IP4 192.0.2.56
a=maxprate:1000
a=rtcp:51540
a=sendonly
a=candidate 12345 1 UDP 9 192.0.2.56 49170 host
a=candidate 23456 2 UDP 9 192.0.2.56 51540 host
a=candidate 34567 1 UDP 7 198.51.100.1 41345 srflx raddr \
192.0.2.56 rport 49170
a=candidate 45678 2 UDP 7 198.51.100.1 52567 srflx raddr \
192.0.2.56 rport 51540
a=candidate 56789 1 UDP 3 192.0.2.100 49000 relay raddr \
192.0.2.56 rport 49170
a=candidate 67890 2 UDP 3 192.0.2.100 49001 relay raddr \
192.0.2.56 rport 51540
b=AS:10000
b=TIAS:100000000
b=RR:4000
b=RS:3000
a=rtpmap:100 H264/90000
a=fmtp:100 profile-level-id=42A01E; packetization-mode=2; \
sprop-parameter-sets=Z0IACpZTBmI,aMljiA==; \
sprop-interleaving-depth=45; sprop-deint-buf-req=64000; \
sprop-init-buf-time=102478; deint-buf-cap=128000
a=tcap:1 RTP/SAVPF RTP/SAVP RTP/AVPF
a=rmcap:1-3,7-9 H264/90000
a=rmcap:4-6 rtx/90000
a=mfcap:1-9 profile-level-id=42A01E
a=mfcap:1-9 aMljiA==
a=mfcap:1,4,7 packetization-mode=0
a=mfcap:2,5,8 packetization-mode=1
a=mfcap:3,6,9 packetization-mode=2
a=mfcap:1-9 sprop-parameter-sets=Z0IACpZTBmI
a=mfcap:1,7 sprop-interleaving-depth=45; \
sprop-deint-buf-req=64000; sprop-init-buf-time=102478; \
deint-buf-cap=128000
a=mfcap:4 apt=100
a=mfcap:5 apt=99
a=mfcap:6 apt=98
a=mfcap:4-6 rtx-time=3000
a=mscap:1-6 rtcp-fb nack
```

a=acap:1 crypto:1 AES\_CM\_128\_HMAC\_SHA1\_80 \  
inline:d0RmdmcmVCspeEc3QGZiNWpVLFJhQX1cfHAWJSoj|220|1:32  
a=pcfg:1 t=1 m=1,4 a=1 pt=1:100,4:97  
a=pcfg:2 t=1 m=2,5 a=1 pt=2:99,4:96  
a=pcfg:3 t=1 m=3,6 a=1 pt=3:98,6:95  
a=pcfg:4 t=2 m=7 a=1 pt=7:100  
a=pcfg:5 t=2 m=8 a=1 pt=8:99  
a=pcfg:6 t=2 m=9 a=1 pt=9:98  
a=pcfg:7 t=3 m=1,3 pt=1:100,4:97  
a=pcfg:8 t=3 m=2,4 pt=2:99,4:96  
a=pcfg:9 t=3 m=3,6 pt=3:98,6:95  
m=audio 49176 RTP/AVP 101 100 99 98  
c=IN IP4 192.0.2.56  
a=ptime:60  
a=maxptime:200  
a=rtcp:51534  
a=sendonly  
a=candidate 12345 1 UDP 9 192.0.2.56 49176 host  
a=candidate 23456 2 UDP 9 192.0.2.56 51534 host  
a=candidate 34567 1 UDP 7 198.51.100.1 41348 srflx \  
raddr 192.0.2.56 rport 49176  
a=candidate 45678 2 UDP 7 198.51.100.1 52569 srflx \  
raddr 192.0.2.56 rport 51534  
a=candidate 56789 1 UDP 3 192.0.2.100 49002 relay \  
raddr 192.0.2.56 rport 49176  
a=candidate 67890 2 UDP 3 192.0.2.100 49003 relay \  
raddr 192.0.2.56 rport 51534  
b=AS:512  
b=TIAS:512000  
b=RR:4000  
b=RS:3000  
a=maxprate:120  
a=rtpmap:98 AMR-WB/16000  
a=fmtp:98 octet-align=1; mode-change-capability=2  
a=rtpmap:99 AMR-WB/16000  
a=fmtp:99 octet-align=1; crc=1; mode-change-capability=2  
a=rtpmap:100 AMR-WB/16000/2  
a=fmtp:100 octet-align=1; interleaving=30  
a=rtpmap:101 AMR-WB+/72000/2  
a=fmtp:101 interleaving=50; int-delay=160000;  
a=rmcap:14 ac3/48000/6  
a=acap:23 crypto:1 AES\_CM\_128\_HMAC\_SHA1\_80 \  
inline:d0RmdmcmVCspeEc3QGZiNWpVLFJhQX1cfHAWJSoj|220|1:32  
a=tcap:4 RTP/SAVP  
a=pcfg:10 t=4 a=23  
a=pcfg:11 t=4 m=14 a=23 pt=14:102

This offer illustrates the advantage in compactness that arises if one can avoid deleting the base configuration attributes and recreating them in acap attributes for the potential configurations.

#### 4.2. Alternative Combinations of Codecs (Session Configurations)

If an endpoint has limited signal processing capacity, it might be capable of supporting, say, a G.711 mu-law audio stream in combination with an H.264 video stream, or a G.729B audio stream in combination with an H.263-1998 video stream. It might then issue an offer like the following:

```
*v=0
o=- 25678 753849 IN IP4 192.0.2.1
s=
c=IN IP4 192.0.2.1
t=0 0
a=creq:med-v0
a=sescap:1 2,4
a=sescap:2 1,3
m=audio 54322 RTP/AVP 18
a=rtpmap:18 G729/8000
a=fmtp:18 annexb=yes
a=rmcap:1 PCMU/8000
a=pcfg:1 m=1 pt=1:0
a=pcfg:2
m=video 54344 RTP/AVP 100
a=rtpmap:100 H263-1998/90000
a=rmcap:2 H264/90000
a=mfcap:2 profile-level-id=42A01E; packetization-mode=2
a=pcfg:3 m=2 pt=2:101
a=pcfg:4
```

Note that the preferred session configuration (and the default as well) is G.729B with H.263. This overrides the individual media stream preferences which are PCMU and H.264 by the potential configuration numbering rule.

#### 4.3. Latent Media Streams

Consider a case in which the offerer can support either G.711 mu-law, or G.729B, along with DTMF telephony events for the 12 common touchtone signals, but is willing to support simple G.711 mu-law audio as a last resort. In addition, the offerer wishes to announce its ability to support video in the future, but does not wish to offer a video stream at present. The offer might look like the following:

```
*v=0
o=- 25678 753849 IN IP4 192.0.2.1
```

```

S=
c=IN IP4 192.0.2.1
t=0 0
a=creq:med-v0
m=audio 23456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=rmcap:1 PCMU/8000
a=rmcap:2 g729/8000
a=rmcap:3 telephone-event/8000
a=mfcap:3 0-11
a=pcfg:1 m=1,3|2,3 pt=1:0,2:18,3:100 a=lcfg:10 mt=video t=1 m=10|
11
a=rmcap:10 H263-1998/90000
a=rmcap:11 H264/90000
a=tcap:1 RTP/AVP

```

The lcfg attribute line announces support for H.263 and H.264 video (H.263 preferred) for future reference. The m-line and the rtpmap attribute offer an audio stream and provide the lowest precedence configuration (PCMU without any DTMF encoding). The rmcap lines define the media capabilities (PCMU, G729, and telephone-event) to be offered in potential configurations. The mfcap attribute provides the format parameters for telephone-events, specifying the 12 commercial DTMF 'digits'. The pcfg attribute line defines the most-preferred media configuration as PCMU plus DTMF events and the next-most-preferred configuration as G.729B plus DTMF events. If the answerer is able to support all the potential configurations, and also support H.263 video (but not H.264), it would reply with an answer like:

```

*v=0
o=- 24351 621814 IN IP4 192.0.2.2
S=
c=IN IP4 192.0.2.2
t=0 0
a=csup:med-v0
m=audio 54322 RTP/AVP 0 100
a=rtpmap:0 PCMU/8000
a=rtpmap:100 telephone-event/8000
a=fmtp:100 0-11
a=acfg:1 m=1,3 pt=1:0,3:100
a=pcfg:1 m=2,3 pt=2:18,3:100 a=lcfg:1 mt=video t=1 m=10

```

The lcfg attribute line announces the capability to support H.263 video at a later time. The media line and subsequent rtpmap and fmtp attribute lines present the selected configuration for the media stream. The acfg attribute line identifies the potential configuration from which it was taken, and the pcfg attribute line announces the potential capability to support G.729 with DTMF events as well. If, at

some later time, congestion becomes a problem in the network, either party may, with expectation of success, offer a reconfiguration of the media stream to use G.729 in order to reduce packet sizes.

## 5. IANA Considerations

### 5.1. New SDP Attributes

The IANA is hereby requested to register the following new SDP attributes:

- \*Attribute name: rmcap  
Long form name: RTP-based media capability  
Type of attribute: session-level and media-level  
Subject to charset: no  
Purpose: associate RTP-based media capability number(s) with media subtype and encoding parameters  
Appropriate Values: see [Section 3.3.1](#)
  
- \*Attribute name: omcap  
Long form name: Non RTP-based media capability  
Type of attribute: session-level and media-level  
Subject to charset: no  
Purpose: associate non RTP-based media capability number(s) with media subtype and encoding parameters  
Appropriate Values: see [Section 3.3.1](#)
  
- \*Attribute name: mfcap  
Long form name: media format capability  
Type of attribute: session-level and media-level  
Subject to charset: no  
Purpose: associate media format attributes and parameters with media format capabilities  
Appropriate Values: see [Section 3.3.2](#)
  
- \*Attribute name: mscap  
Long form name: media-specific capability  
Type of attribute: session-level and media-level  
Subject to charset: no  
Purpose: associate media-specific attributes and parameters with media capabilities  
Appropriate Values: see [Section 3.3.3](#)
  
- \*Attribute name: lcfig  
Long form name: latent configuration  
Type of attribute: media-level  
Subject to charset: no  
Purpose: to announce supportable media streams without offering them for immediate use.  
Appropriate Values: see [Section 3.3.5](#)

\*Attribute name: sescap  
Long form name: session capability  
Type of attribute: session-level  
Subject to charset: no  
Purpose: to specify and prioritize acceptable combinations of media stream configurations.  
Appropriate Values: see [Section 3.3.8](#)

## [5.2. New SDP Option Tag](#)

The IANA is hereby requested to add the new option tag "med-v0", defined in this document, to the SDP Capability Option Negotiation Capability registry created for [\[RFC5939\]](#).

## [5.3. New SDP Capability Negotiation Parameters](#)

The IANA is hereby requested to expand the SDP Capability Negotiation Potential Configuration Parameter Registry established by [\[RFC5939\]](#) to become the SDP Capability Negotiation Configuration Parameter Registry and to include parameters for the potential, actual and latent configuration attributes. The new parameters to be registered are the "m" for "media", "pt" for "payload type number", and "mt" for "media type" parameters. Note that the "mt" parameter is defined for use only in the latent configuration attribute.

## [6. Security Considerations](#)

EDITOR'S NOTE: SECTION NEEDS TO BE EXPANDED

The security considerations of [\[RFC5939\]](#) apply for this document. The addition of negotiable media encoding, bandwidth attributes, and connection data in this specification can cause problems for middleboxes which attempt to control bandwidth utilization, media flows, and/or processing resource consumption as part of network policy, but which do not understand the media capability negotiation feature. As for the initial CapNeg work, the SDP answer is formulated in such a way that it always carries the selected media encoding and bandwidth parameters for every media stream selected. Pending an understanding of capabilities negotiation, the middlebox should examine the answer SDP to obtain the best picture of the media streams being established.

As always, middleboxes can best do their job if they fully understand media capabilities negotiation.

## [7. Changes from previous versions](#)

### [7.1. Changes from version 11](#)

\*Corrected several statements implying lcfg was a session-level attribute.

\*Added non-RTP based media format capabilities ("a=omcap") and renamed "mcap" to "rmcap"

## **7.2. Changes from version 10**

\*Defined the latent configuration attribute as a media-level attribute because it specifies a possible future media stream. Added text to clarify how to specify alternative configurations of a single latent stream and/or multiple streams.

\*Improved the definition of the session capability attribute to permit both required configurations and optional configurations - latent configurations cannot be required because they have not yet been offered.

\*Removed the special-case treatment of conflicts between base-level fntp attributes and fntp attributes generated for a configuration via invoked mcap and mfcap attributes.

\*Removed reference to bandwidth capability (bcap) attribute.

\*Changed various "must", etc., terms to normative terms ("MUST", etc.) as appropriate, in [Section 3.3.5](#)[Section 3.3.6.1](#) [Section 3.3.6.3](#) and [Section 3.3.8](#)

\*Attempted to clarify the substitution mechanism in [Section 3.3.7](#) and improve its uniqueness.

\*Made various editorial changes, including changing the title in the header, and removing numbering from some SDP examples.

## **7.3. Changes from version 09**

\*Additional corrections to latent media stream example in [Section 4.3](#)

\*Fixed up attribute formatting examples and corresponding ABNF.

\*Removed preference rule for latent configurations.

\*Various spelling and other editorial changes were made.

\*updated crossreferences.

## **7.4. Changes from version 08**

The major change is in [Section 4.3](#), Latent Media Streams, fixing the syntax of the answer. All the other changes are editorial.

### **7.5. Changes from version 04**

- \*The definitions for bcap, ccap, icap, and kcap attributes have been removed, and are to be defined in another document.
- \*Corrected formatting of m= and p= configuration parameters to conform to extension-config-list form defined in [\[RFC5939\]](#)
- \*Reorganized definitions of new parameters to make them easier to find in document.
- \*Added ability to renegotiate capabilities when modifying the session ([Section 3.4.4](#)).
- \*Made various editorial changes, clarifications, and typo corrections.

### **7.6. Changes from version 03**

- \*A new session capability attribute (sescap) has been added to permit specification of acceptable media stream combinations.
- \*Capability attribute definitions corresponding to the i, c, b, and k SDP line types have been added for completeness.
- \*Use of the pcfg: attribute in SDP answers has been included in order to conveniently return information in the answer about acceptable configurations in the media stream offer.
- \*The use of the lcfig: attribute(s) in SDP answers has been restricted to indicate just which latent configuration offers would be acceptable to the answerer.
- \*A suggestion for "naive" middleboxes has been added to the Security Considerations.
- \*Various editorial changes have been made.
- \*Several errors/omissions have been corrected.
- \*The description of the mscap attribute has been modified to make it clear that it should not be used to generate undefined SDP attributes, or to "extend" existing attributes.
- \*<ms-parameters> are made optional in the mscap attribute definition.
- \*"AMR" changed to "AMR-WB" in cases in which the sample rate is 16000.



### [7.7. Changes from version 02](#)

This version contains several detail changes intended to simplify capability processing and mapping into conventional SDP media blocks.

- \*The "mcap" attribute is enhanced to include the role of the "ecap" attribute; the latter is eliminated.
- \*The "fcap" attribute has been renamed "mfcap". New replacement rules vis-a-vis fntp attributes in the base media specification have been added.
- \*A new "mscap" attribute is defined to handle the problem of attributes (other than rtpmap and fntp) that are specific to a particular payload type.
- \*New rules for processing the mcap, mfcap, and mscap attributes, and overriding standard rtpmap, fntp, or other media-specific attributes, are put forward to reduce the need to use the deletion option in the a= parameter of the potential configuration (pcfg) attribute.
- \*A new parameter, "mt=" is added to the latent configuration attribute (lcfg) to specify the media stream type (audio, video, etc.) when the lcfg is declared at the session level.
- \*The examples are expanded.
- \*Numerous typos and misspellings have been corrected.

### [7.8. Changes from version 01](#)

The documents adds a new attribute for specifying bandwidth capability and a parametr to list in the potential configuration. Other changes are to align the document with the terminolgy and attribute names from draft-ietf-mmusic-sdp-capability-negotiation-07. The document also clarifies some previous open issues.

### [7.9. Changes from version 00](#)

The major changes include taking out the "mcap" and "cptmap" parameter. The mapping of payload type is now in the "pt" parameter of "pcfg". Media subtype need to explictly defineds in the "cmed" attribute if referenced in the "pcfg"

## [8. Acknowledgements](#)

This document is heavily influenced by the discussions and work done by the SDP Capability Negotiation Design team. The following people in particular provided useful comments and suggestions to either the

document itself or the overall direction of the solution defined herein: Cullen Jennings, Matt Lepinski, Joerg Ott, Colin Perkins, and Thomas Stach.

We thank Ingemar Johansson and Magnus Westerlund for examples that stimulated this work, and for critical reading of the document.

## **9. References**

### **9.1. Normative References**

[RFC2119]	<a href="#">Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels"</a> , BCP 14, RFC 2119, March 1997.
[RFC5234]	Crocker, D. and P. Overell, " <a href="#">Augmented BNF for Syntax Specifications: ABNF</a> ", STD 68, RFC 5234, January 2008.
[RFC3264]	Rosenberg, J. and H. Schulzrinne, " <a href="#">An Offer/Answer Model with Session Description Protocol (SDP)</a> ", RFC 3264, June 2002.
[RFC4566]	Handley, M., Jacobson, V. and C. Perkins, " <a href="#">SDP: Session Description Protocol</a> ", RFC 4566, July 2006.
[RFC5939]	Andreasen, F, " <a href="#">SDP Capability Negotiation</a> ", RFC 5939, September 2010.

### **9.2. Informative References**

[RFC4568]	Andreasen, F., Baugher, M. and D. Wing, " <a href="#">Session Description Protocol (SDP) Security Descriptions for Media Streams</a> ", RFC 4568, July 2006.
[RFC4585]	Ott, J., Wenger, S., Sato, N., Burmeister, C. and J. Rey, " <a href="#">Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)</a> ", RFC 4585, July 2006.
[RFC4867]	Sjoberg, J., Westerlund, M., Lankaniemi, A. and Q. Xie, " <a href="#">RTP Payload Format and File Storage Format for the Adaptive Multi-Rate (AMR) and Adaptive Multi-Rate Wideband (AMR-WB) Audio Codecs</a> ", RFC 4867, April 2007.
[RFC5104]	Wenger, S., Chandra, U., Westerlund, M. and B. Burman, " <a href="#">Codec Control Messages in the RTP Audio-Visual Profile with Feedback (AVPF)</a> ", RFC 5104, February 2008.
[RFC4583]	Camarillo, G., " <a href="#">Session Description Protocol (SDP) Format for Binary Floor Control Protocol (BFCP) Streams</a> ", RFC 4583, November 2006.
[RFC4733]	Schulzrinne, H. and T. Taylor, " <a href="#">RTP Payload for DTMF Digits, Telephony Tones, and Telephony Signals</a> ", RFC 4733, December 2006.

### **Authors' Addresses**

Robert R Gilman Gilman Independent 3243 W. 11th Ave. Dr. Broomfield,  
CO 80020, USA EMail: [bob\\_gilman@comcast.net](mailto:bob_gilman@comcast.net)

Roni Even Even Gesher Erova Ltd 14 David Hamelech Tel Aviv, 64953  
Israel EMail: [ron.even.tlv@gmail.com](mailto:ron.even.tlv@gmail.com)

Flemming Andreasen Andreasen Cisco Systems Iselin, NJ, USA EMail:  
[fandreas@cisco.com](mailto:fandreas@cisco.com)