INTERNET-DRAFT
Expires: January 2005

J. Ott
TZI

12 July 2004

SDP Attributes for T.120 Data Conferencing
draft-ott-mmusic-sdp-t120-00.txt

Status of this memo

Abstract

This memo specifies the use of the Session Description Protocol in combination with the SDP Offer/Answer exchange to setup and teardown data conferecing sessions based upon the ITU-T T.120 Series of Recommendations, thereby particularly enabling application sharing as defined in ITU-T T.128.

1. Introduction

SDP [1] is widely used in the Internet today to describe multimedia sessions between two or more endpoints. The offer/answer exchange [2] allows two endpoints to negotiate the media streams to be established, e.g. in the context of a SIP dialog [4]. While fax and text conversation have been defined for use with SIP, the primary media types in use today are still audio and video. What is currently lacking is the support for application-sharing, shared editing, whiteboard and similar teleconferencing tools.

This memo defines a minimal set of SDP attributes to enable standardized setup of T.120 data conferencing sessions -- and particularly application sharing as defined in T.128 and implemented e.g. in NetMeeting.

While SDP is defined for use with many signaling protocols, the use of T.120 is probably only meaningful for SIP calls and conferences and hence this memo focuses on its use with SIP.

2. Overview of T.120

T.120 defines multipoint conferencing protocols and services for data applications including whiteboard (T.126), file transfer (T.127), application sharing (T.128), and text conversation (T.134, T.140), among others [5].

In T.120, communication takes place in a hierarchical structure of T.120 entities interconnected by T.120 connections. In the simplest case, this may be just two entities interconnected via a T.120 connection, in a slightly more sophisticated one a single conference bridge (MCU) may be used to interconnect more than two entities. Each point-to-point T.120 connection uses TCP as underlying transport and TPKT (RFC1006) framing on top [6]. Up to four TCP connections may be between any two T.120 entities to provide independent flow control for different transmission priorities.

The lowest layer above the point-to-point transport is the Multipoint Communication Service (MCS) [7][8]. The MCS defines a connection setup procedure that allows to associate different transport connections with the same MCS Domaina and to organize the MCS "providers" in a tree structure during connection setup. In the resulting tree, the MCS provides a multiplex for application data (using up to 64K channels) and simple means for synchronizations (tokens).

On top of MCS, the Generic Conference Control (GCC) [9] is responsible for controlling the connection setup and their association with a conference. Furthermore, GCC manages conference

resources, provides capability exchange, allows for floor control, and provides some kind of a conference-wide registry. In GCC, conferences are identified by means of an octet string that is mapped to the MCS Domain identifier. GCC allows to create and destroy

conferences as well as to inquire for, join, and leave existing ones.

T.120 data applications make use of the MCS communication platform to exchange information and of the GCC services to learn about each others existence and to find each other. In particular, GCC's capability exchange mechanism is used to discover commonly available applications and their respective features (with many tiny details being negotiable).

The purpose of this memo is to define a minimal meaningful set of SDP attributes that allows two nodes to learn about their respective T.120 capabilities and enable them to set up a single T.120 connection. Whether a single or more TCP connections are used and how those are associated if entirely left up to the respective T.120 entities, as is most of the T.120-specific capability exchange.

3. T.120 Setup Procedure

3.1. Use of the m= line

A T.120 session is based upon TCP as underlying transport. Therefore, the rules for connection-oriented media as defined in [3] MUST be followed.

The media part on the m= line MUST indicate "data", the port number indicate the port number to connect to (for one to four transport connections, again following the rules of [3]), the proto part MUST indicate "TCP" and the format MUST indicate "t120". The registered port number for T.120 is 1503.

3.2. T.120-specific Attributes

T.120 connections established within the context of an SDP offer/answer exchange may need to be associated with a SIP dialog or a SIP conference. Therefore, the media-level "confname" attribute is introduced. The content of the confname attribute MUST be copied to the ConferenceName field used by the GCC entity. For a simple SIP dialog, the confname attribute SHOULD contain the SIP From tag, To tag, and Call-ID, concatenated with the comma (",") as a separator. For a SIP dialog in a conference, the confname attribute SHOULD contain the focus URI.

T.120 defines a number of applications. To determine from the offer/answer exchange whether or not at least the target application(s) are available at a peer, the optional "t120apps" attribute is introduced. If present, this attribute SHOULD contain a list of T.120 application protocols supported by the respective peer.

The following application protocols are defined: "t126", "t127", "t128", "t134", "t136", and "t137". The fine-grained

negotiation of capabilities within these application protocols is left up to the GCC operation after setup of a T.120 connection.

3.3. Offer/Answer Operation

The offerer wishing to set up a T.120 session MUST include an m= line according to section 3.1 and include an appropriate c= line. The offerer MUST choose a new "connid" and MAY choose to specify a "setup" attribute of "active", "passive", or "actpass". The offerer MUST provide the "confname" attribute as specified in section 3.2 and SHOULD include the list of supported T.120 application protocols.

The answerer MUST examine that it supports T.120 protocol. If it does not support T.120 or does not wish to use T.120 in the present communication relationship, it MUST refure the media section by setting the port number to "0" in the answer. If the answerer supports T.120, it MUST validate the "confname" attribute. If no matching SIP dialog or focus URI can be found, the media section SHOULD be refused by setting the port number to "0" in the answer. If a matching conference is found and the "t120apps" attribute is not present, the answerer MAY decide whether to accept the session and proceed with the transport connection setup as per [3] or whether to refuse the media section. If the "t120apps" attribute is present, the answerer MUST examine its contents and match the applications listed by the offerer against its own capabilities. If the intersection of capabilities is empty, the answerer SHOULD refuse the media section. If the intersection is not empty, the answerer SHOULD return the intersecting capabilities (in the order provided by the offerer) and then proceed with the transport connection setup as per [3].

After successful establishment of a TCP connection as per [3], the respective entities MUST proceed with the T.120 connection setup as defined in [6], [8], and [9].

4. Formal SDP Attribute Specification

This section provides the formal SDP attribute specification:

```
confname      = "a=confname:" token

t120apps      = "a=t120apps:" t120appid *("," t120appid)
t120appid     = "t126" / "t127" / "t128" / "t134" /
                "t136" / "t137"
```

5. Example

In the following example, an node A wants to set up a telephone call

to a node B and use supportive data applications (whiteboard, file transfer, and application sharing). Node A expects to initiate connection setup to node B and therefore uses port number 9 [3] and

sets the setup attribute to "active".

```
c=IN IP4 10.20.30.40
m=audio 52000 UDP/AVP 96 97
a=rtpmap:96 PCMU/8000
a=rtpmap:97 L16/16000
m=data 9 TCP t120
a=setup:active
a=connid:1
a=confname:623847692,789234789,78687afded278bd@example.com
a=t120apps:t126,t127,t128
```

Node B accepts the T.120 media session in its answer. However, as node B does not support whiteboard and file transfer, it only confirms application sharing (T.128) in its answer. Node B offers the registered T.120 port number to connect to.

```
c=IN IP4 10.20.30.40
m=audio 52000 UDP/AVP 96
a=rtpmap:96 PCMU/8000
m=data 1503 TCP t120
a=setup:passive
a=connid:1
a=confname:623847692,789234789,78687afded278bd@example.com
a=t120apps:t128
```

6. IANA Considerations

This document defines two media level SDP attributes: "confname" and "t120apps". Their format is defined in Section 4. These two attributes should be registered by the IANA on

http://www.iana.org/assignments/sdp-parameters

under "att-field (at the media level only)".

7. Security Considerations

TBD.

8. Author's Addresses

Joerg Ott <jo@tzi.org>
Universitaet Bremen
MZH 5180
Bibliothekstr. 1
D-28359 Bremen

Germany
tel:+49-421-201-7028
sip:jo@tzi.org

9. Normative References

[1] Mark Handley, Van Jacobson, Colin Perkins, "SDP: Session Description Protocol," Internet Draft draft-ietf-mmusic-sdp-new-18.txt, Work in Progress, June 2004.

[2] Jonathan Rosenberg and Henning Schulzrinne, "An Offer/Answer Model with SDP," RFC 3264, June 2002.

[3] David Yon and Gonzalo Camarillo, "Connection-Oriented Media Transport in SDP," Internet Draft draft-ietf-mmusic-sdp-comedia-07.txt, Work in Progress, June 2004.

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

[5] ITU-T Recommendation T.120, "Data Protocols for Multimedia Conferencing", 1996.

[6] ITU-T Recommendation T.123, "Network Specific Data Protocol Stacks for Multimedia Conferencing", 1996.

[7] ITU-T Recommendation T.122, "Multipoint Communication Service for Audiographics and Audiovisual Conferencing Service Definition," 1993.

[8] ITU-T Recommendation T.125,."Multipoint Communication Service Protocol Specification," 1994.

[9] ITU-T Recommendation T.124, "Generic Conference Control,", 1995.

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

Full Copyright Statement

Acknowledgment

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