

# [MS-DTMF]: RTP Payload for DTMF Digits, Telephony Tones, and Telephony Signals Extensions

---

## Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft's Open Specification Promise (available here: <http://www.microsoft.com/interop/osp>) or the Community Promise (available here: <http://www.microsoft.com/interop/cp/default.mspx>). If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting [iplg@microsoft.com](mailto:iplg@microsoft.com).
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights.
- **Fictitious Names.** The example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

**Reservation of Rights.** All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

**Tools.** The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

## Revision Summary

Date	Revision History	Revision Class	Comments
04/04/2008	0.1		Initial version
04/25/2008	0.2		Updated based on feedback
06/27/2008	1.0		Updated based on feedback
08/15/2008	1.01		Updated based on feedback
12/12/2008	2.0		Updated with latest template bug fixes (redlined)
02/13/2009	2.01		Updated with latest template bug fixes (redlined)
03/13/2009	2.02		Updated with latest template bug fixes (redlined)
07/13/2009	2.03	Major	Revised and edited the technical content
08/28/2009	2.04	Editorial	Revised and edited the technical content
11/06/2009	2.05	Editorial	Revised and edited the technical content
02/19/2010	2.06	Editorial	Revised and edited the technical content
03/31/2010	2.07	Major	Updated and revised the technical content
04/30/2010	2.08	Editorial	Revised and edited the technical content
06/07/2010	2.09	Editorial	Revised and edited the technical content
06/29/2010	2.10	Editorial	Changed language and formatting in the technical content.
07/23/2010	2.10	No change	No changes to the meaning, language, or formatting of the technical content.
09/27/2010	3.0	Major	Significantly changed the technical content.
11/15/2010	3.0	No change	No changes to the meaning, language, or formatting of the technical content.
12/17/2010	3.0	No change	No changes to the meaning, language, or formatting of the technical content.

# Table of Contents

<b>1 Introduction</b>	<b>5</b>
1.1 Glossary	5
1.2 References	5
1.2.1 Normative References	5
1.2.2 Informative References	5
1.3 Protocol Overview (Synopsis)	6
1.4 Relationship to Other Protocols	6
1.5 Prerequisites/Preconditions	6
1.6 Applicability Statement	6
1.7 Versioning and Capability Negotiation	6
1.8 Vendor-Extensible Fields	7
1.9 Standards Assignments	7
<b>2 Messages</b>	<b>8</b>
2.1 Transport	8
2.2 Message Syntax	8
2.2.1 DTMF Telephony Event	8
<b>3 Protocol Details</b>	<b>9</b>
3.1 Common Details	9
3.1.1 Abstract Data Model	9
3.1.2 Timers	9
3.1.3 Initialization	9
3.1.4 Higher-Layer Triggered Events	9
3.1.5 Message Processing Events and Sequencing Rules	9
3.1.6 Timer Events	9
3.1.7 Other Local Events	9
3.2 Receiver Details	10
3.2.1 Abstract Data Model	10
3.2.2 Timers	10
3.2.3 Initialization	10
3.2.4 Higher-Layer Triggered Events	10
3.2.5 Message Processing Events and Sequencing Rules	10
3.2.6 Timer Events	10
3.2.7 Other Local Events	10
3.3 Sender Details	10
3.3.1 Abstract Data Model	10
3.3.2 Timers	10
3.3.3 Initialization	11
3.3.4 Higher-Layer Triggered Events	11
3.3.5 Message Processing Events and Sequencing Rules	11
3.3.6 Timer Events	11
3.3.7 Other Local Events	11
<b>4 Protocol Examples</b>	<b>12</b>
<b>5 Security</b>	<b>13</b>
5.1 Security Considerations for Implementers	13
5.2 Index of Security Parameters	13
<b>6 Appendix A: Product Behavior</b>	<b>14</b>

<b>7</b>	<b>Change Tracking.....</b>	<b>15</b>
<b>8</b>	<b>Index .....</b>	<b>16</b>

# 1 Introduction

This document specifies the Real-Time Transport Protocol (RTP) Payload for dual-tone multi-frequency (DTMF) Digits, Telephony Tones, and the Telephony Signals Extensions Protocol. This protocol is a proprietary extension to [\[RFC4733\]](#). This protocol describes the payload format needed to carry DTMF digits, tones, and signals in RTP packets over a network transport.

Any behavior not explicitly defined in this document is described in [\[RFC4733\]](#).

## 1.1 Glossary

The following terms are defined in [\[MS-OFCGLOS\]](#):

**dual-tone multi-frequency (DTMF)**  
**Real-Time Transport Protocol (RTP)**  
**RTP packet**  
**RTP payload**  
**RTP session**  
**Session Description Protocol (SDP)**

The following terms are specific to this document:

**MAY, SHOULD, MUST, SHOULD NOT, MUST NOT:** These terms (in all caps) are used as described in [\[RFC2119\]](#). All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

## 1.2 References

### 1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact [dochelp@microsoft.com](mailto:dochelp@microsoft.com). We will assist you in finding the relevant information. Please check the archive site, <http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624>, as an additional source.

[MS-RTP] Microsoft Corporation, "[Real-time Transport Protocol \(RTP\) Extensions](#)", June 2008.

[MS-RTPRADEX] Microsoft Corporation, "[RTP Payload for Redundant Audio Data Extensions](#)", June 2008.

[MS-SDPEXT] Microsoft Corporation, "[Session Description Protocol \(SDP\) Version 2.0 Protocol Extensions](#)", June 2008.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://www.ietf.org/rfc/rfc2119.txt>

[RFC4733] Schulzrinne, H., "RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals", RFC 4733, December 2006, <http://www.ietf.org/rfc/rfc4733.txt>

### 1.2.2 Informative References

[MS-OFCGLOS] Microsoft Corporation, "[Microsoft Office Master Glossary](#)", June 2008.

### 1.3 Protocol Overview (Synopsis)

The **Real-Time Transport Protocol (RTP)** Payload for **dual-tone multi-frequency (DTMF)** Digits, Telephony Tones, and the Telephony Signals Extensions protocol extends the protocol described in [\[RFC4733\]](#), which describes a mechanism for transmission of in-band and out-of-band telephony signals.

In-band telephony signal is where the events or tones are mixed directly into the media stream (typically audio data). Out-of-band telephony signal is where the events or tones are transmitted through a separate band.

Telephony tones represent the DTMF tones mixed into the audio signal of the media stream. Telephony events represent the different call control events (such as off-hook event or specific digit being dialed).

The scope of this protocol is limited to telephony signals using out-of-band transmission. The in-band transmission of digits and tones is not supported by this protocol.

### 1.4 Relationship to Other Protocols

This protocol relies on RTP, as described in [\[MS-RTP\]](#), as its transport mechanism. This protocol can be used to communicate signaling DTMF telephony events between clients and gateways using the **RTP payload**.

### 1.5 Prerequisites/Preconditions

This protocol is a payload of RTP; therefore, a valid **RTP session** is established between the client and the gateway.

Furthermore, because of the dynamic payload typing of the telephony events, some form of out-of-band negotiation to bind the payload type of the RTP payload to the telephony events is required. This is done using the **Session Description Protocol (SDP)**, as described in [\[MS-SDPEXT\]](#) section 3.1.5.3.

### 1.6 Applicability Statement

This protocol is applicable wherever telephony digits, tones, or signals need to be sent or consumed either by remote clients or through gateways.

### 1.7 Versioning and Capability Negotiation

**Supported Transports:** This protocol is sent using the RTP transport mechanism.

**Protocol Versions:** This protocol, as a format of an RTP payload, does not provide for versioning information within the scope of the protocol itself. However, as a part of the RTP payload, any versioning information about the RTP level applies.

**Security and Authentication Methods:** This document does not describe any security or authentication methods. Security and authentication is dependent on the security method, authentication method, or both methods used by the RTP version 2 protocol and is beyond the scope of this document.

**Localization:** None.

## **1.8 Vendor-Extensible Fields**

None.

## **1.9 Standards Assignments**

None.

## 2 Messages

### 2.1 Transport

This protocol MUST be sent using RTP, as specified in [\[MS-RTP\]](#), as its transport. This protocol assumes that a successful RTP session has been established with valid payload information.

The Session Description Protocol (SDP), as specified in [\[MS-SDPEXT\]](#), MUST be used to negotiate the payload type information.

### 2.2 Message Syntax

The structure and syntax of this protocol is defined in [\[RFC4733\]](#) section 2.3.

#### 2.2.1 DTMF Telephony Event

The DTMF Telephony Event format is defined in [\[RFC4733\]](#) section 2.3.1.



## 3 Protocol Details

### 3.1 Common Details

This protocol conforms more to the "sender-receiver" paradigm, rather than the classic "client-server" paradigm. More specifically, it is appropriate to discuss in terms of the receiver of the telephony signals and the sender of the telephony signals.

This section covers the common details between the sender and receiver. Subsequent sections provide the specifics for the sender and the receiver.

In [\[RFC4733\]](#), out-of-band negotiation of telephony signal information is required to establish a session. During this negotiation, both payload types and the clock rate of the telephony signals are negotiated as specified in [\[RFC4733\]](#) section 2.5.1.1 using SDP for out-of-band negotiation. While dynamic payload type binding is required, both the sender and receiver of message blocks conforming to this protocol MUST fix the telephony signaling information at 8000 Hz. Dynamic negotiation of the clock frequency of the DTMF payload MUST NOT be used.

Multiple payload type binding for different telephony events MUST NOT be used. There MUST be only one telephony event binding for a payload type. The payload type binding MUST be symmetrical. This means the receive payload type and send payload type MUST be the same. Asymmetrical payload type information MUST NOT be used.

This protocol supports only the telephony event. An in-band telephony tone transmission MUST NOT be used.

#### 3.1.1 Abstract Data Model

None.

#### 3.1.2 Timers

None.

#### 3.1.3 Initialization

None.

#### 3.1.4 Higher-Layer Triggered Events

None.

#### 3.1.5 Message Processing Events and Sequencing Rules

None.

#### 3.1.6 Timer Events

None.

#### 3.1.7 Other Local Events

None.

## 3.2 Receiver Details

Redundant payload support, as specified in [\[MS-RTPRADEX\]](#), MUST NOT be used.

Multiple events per RTP block MUST NOT be used.

### 3.2.1 Abstract Data Model

None.

### 3.2.2 Timers

None.

### 3.2.3 Initialization

None.

### 3.2.4 Higher-Layer Triggered Events

None.

### 3.2.5 Message Processing Events and Sequencing Rules

None.

### 3.2.6 Timer Events

None.

### 3.2.7 Other Local Events

None.

## 3.3 Sender Details

Implementation for this protocol MUST NOT generate redundant blocks, as specified in [\[MS-RTPRADEX\]](#).

The sender MUST NOT pack multiple DTMF payloads into a single **RTP packet**.

The sender MUST NOT generate a DTMF event whose duration exceeds the maximum expressible duration, as specified in [\[RFC4733\]](#) section 2.3.5.

The sender MUST NOT generate a DTMF event payload with a zero duration.

### 3.3.1 Abstract Data Model

None.

### 3.3.2 Timers

None.

### **3.3.3 Initialization**

None.

### **3.3.4 Higher-Layer Triggered Events**

None.

### **3.3.5 Message Processing Events and Sequencing Rules**

None.

### **3.3.6 Timer Events**

None.

### **3.3.7 Other Local Events**

None.

## 4 Protocol Examples

For examples of the DTMF telephony signal blocks, see [RFC4733](#) section 2.3.

## **5 Security**

### **5.1 Security Considerations for Implementers**

None.

### **5.2 Index of Security Parameters**

None.

## 6 Appendix A: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs:

- Microsoft® Office Communications Server 2007
- Microsoft® Office Communicator 2007
- Microsoft® Office Communications Server 2007 R2
- Microsoft® Office Communicator 2007 R2
- Microsoft® Lync™ Server 2010
- Microsoft® Lync™ 2010

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

Unless otherwise specified, any statement of optional behavior in this specification that is prescribed using the terms SHOULD or SHOULD NOT implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that the product does not follow the prescription.

## 7 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

## 8 Index

### A

Abstract data model  
  receiver ([section 3.1.1](#) 9, [section 3.2.1](#) 10)  
  sender ([section 3.1.1](#) 9, [section 3.3.1](#) 10)  
[Applicability](#) 6

### C

[Capability negotiation](#) 6  
[Change tracking](#) 15  
Client  
  [overview](#) 9

### D

Data model - abstract  
  receiver ([section 3.1.1](#) 9, [section 3.2.1](#) 10)  
  sender ([section 3.1.1](#) 9, [section 3.3.1](#) 10)  
[DTMF Telephony Event message](#) 8

### E

[Examples](#) 12

### F

[Fields - vendor-extensible](#) 7

### G

[Glossary](#) 5

### H

Higher-layer triggered events  
  receiver ([section 3.1.4](#) 9, [section 3.2.4](#) 10)  
  sender ([section 3.1.4](#) 9, [section 3.3.4](#) 11)

### I

[Implementer - security considerations](#) 13  
[Index of security parameters](#) 13  
[Informative references](#) 5  
Initialization  
  receiver ([section 3.1.3](#) 9, [section 3.2.3](#) 10)  
  sender ([section 3.1.3](#) 9, [section 3.3.3](#) 11)  
[Introduction](#) 5

### L

Local events  
  receiver ([section 3.1.7](#) 9, [section 3.2.7](#) 10)  
  sender ([section 3.1.7](#) 9, [section 3.3.7](#) 11)

### M

Message processing  
  receiver ([section 3.1.5](#) 9, [section 3.2.5](#) 10)

  sender ([section 3.1.5](#) 9, [section 3.3.5](#) 11)

Messages  
  [DTMF Telephony Event](#) 8  
  [transport](#) 8

### N

[Normative references](#) 5

### O

[Overview \(synopsis\)](#) 6

### P

[Parameters - security index](#) 13  
[Preconditions](#) 6  
[Prerequisites](#) 6  
[Product behavior](#) 14  
Proxy  
  [overview](#) 9

### R

Receiver  
  abstract data model ([section 3.1.1](#) 9, [section 3.2.1](#) 10)  
  higher-layer triggered events ([section 3.1.4](#) 9, [section 3.2.4](#) 10)  
  initialization ([section 3.1.3](#) 9, [section 3.2.3](#) 10)  
  local events ([section 3.1.7](#) 9, [section 3.2.7](#) 10)  
  message processing ([section 3.1.5](#) 9, [section 3.2.5](#) 10)  
  overview ([section 3.1](#) 9, [section 3.2](#) 10)  
  sequencing rules ([section 3.1.5](#) 9, [section 3.2.5](#) 10)  
  timer events ([section 3.1.6](#) 9, [section 3.2.6](#) 10)  
  timers ([section 3.1.2](#) 9, [section 3.2.2](#) 10)

References  
  [informative](#) 5  
  [normative](#) 5  
[Relationship to other protocols](#) 6

### S

Security  
  [implementer considerations](#) 13  
  [parameter index](#) 13  
Sender  
  abstract data model ([section 3.1.1](#) 9, [section 3.3.1](#) 10)  
  higher-layer triggered events ([section 3.1.4](#) 9, [section 3.3.4](#) 11)  
  initialization ([section 3.1.3](#) 9, [section 3.3.3](#) 11)  
  local events ([section 3.1.7](#) 9, [section 3.3.7](#) 11)  
  message processing ([section 3.1.5](#) 9, [section 3.3.5](#) 11)  
  overview ([section 3.1](#) 9, [section 3.3](#) 10)



sequencing rules ([section 3.1.5](#) 9, [section 3.3.5](#) 11)  
timer events ([section 3.1.6](#) 9, [section 3.3.6](#) 11)  
timers ([section 3.1.2](#) 9, [section 3.3.2](#) 10)  
Sequencing rules  
receiver ([section 3.1.5](#) 9, [section 3.2.5](#) 10)  
sender ([section 3.1.5](#) 9, [section 3.3.5](#) 11)  
Server  
[overview](#) 9  
[Standards assignments](#) 7

## T

Timer events  
receiver ([section 3.1.6](#) 9, [section 3.2.6](#) 10)  
sender ([section 3.1.6](#) 9, [section 3.3.6](#) 11)  
Timers  
receiver ([section 3.1.2](#) 9, [section 3.2.2](#) 10)  
sender ([section 3.1.2](#) 9, [section 3.3.2](#) 10)  
[Tracking changes](#) 15  
[Transport](#) 8  
Triggered events  
receiver ([section 3.1.4](#) 9, [section 3.2.4](#) 10)  
sender ([section 3.1.4](#) 9, [section 3.3.4](#) 11)

## V

[Vendor-extensible fields](#) 7  
[Versioning](#) 6