

JVoIP ED-137

Implementation of the ED-137 specification for the MizuTech Java SIP Library

Contents

About	2
Usage	2
Parameters	3
Configuration	3
ED-137 specific parameters	4
API.....	6
boolean API_ED137PTT (int line, int ptt, int pttid).....	6
boolean API_RTPHeaderExtension (int line, int profile, String extension)	7
boolean API_SetLineParameter (int line, String param, String value, int permanent)	7
boolean API_SetSDPField (int line, String field, int type).....	7
Notifications.....	7
RTPT,line,PTT-Type,SQU,PTT-ID,SCT,VF,extensionwords	8
FAQ, Tasks, Notes	8
Local audio streaming.....	8
RTP statistics	8
Target URI	8
Multiple calls.....	8
SIP headers.....	9
SDP parameters	9
Auto answer.....	9
Terms and other notes	10
Troubleshooting	10
Implementation	11
Implemented/Changes.....	11
Not implemented	11
Resources	12

About

This document will explain the [ED-137](#) specific changes, configurations, API and usage for [JVoIP](#).

ED-137 is used for air traffic management services (ground to air communication) and it is based on the existing SIP/RTP protocol (already supported by JVoIP) to migrate old TDM circuits to VoIP. The specification aims to make the SIP protocol more predictable (by adding a list of constraints and restrictions) and extends the SIP/RTP protocols with ED-137 features (such as extra SDP parameters, PTT signaling in RTP extensions, a special RTP keep-alive mechanism and prioritizations).

JVoIP can be used as a VCS to connect to radio(s) and it is compatible with the ED-137 specification prepared by EUROCAE Working Group 67. It implements most of the ED-137 specification ignoring the non-relevant parts.

The implementation is based mostly in the ED-137-B specification (EUROCAE Interoperability Standards for VoIP ATM Components ED-137/1B SIP signaling and RTP) which means that the base ED-137 protocol is also covered, but ED-137-C is also (partially) supported (both the ED-137 B and C profile are recognized and most of the ED-137-C requirements are also supported).

Please note that many of the ED-137 functionalities can be also implemented in your app instead of using the JVoIP built-in features. This might be important if something is not supported by JVoIP or the implementation doesn't fulfill your requirements. For example if you stream the local media (JVoIP not using microphone/player audio device), then you can send any RTP as you wish or you can mimic the `API_ED137PTT` by using the `API_Mute` and the `API_RTPHeaderExtension` functions.

This document describes only the ED-137 related functionalities of the JVoIP Java SIP library. For general usage see the [JVoIP main documentation](#).

Usage

The ED-137 specification is implemented behind the `ed137` feature flag.

Set the `ed137` parameter to `1` to enable the ED-137 specific behavior.

Change any other parameters after your needs.

Launch JVoIP (create a webphone object) then `start/register/call` as usually.

Use the `API_ED137PTT` API function for PTT ON/OFF.

Parse the `RTPT` notifications about the remote PTT state changes.

Example

```
//create a JVoIP instance
```

```
webphone wobj = new webphone();
```

```
//create the SIPNotifications object to catch the events from JVoIP
```

```
sipnotifications = new SIPNotifications(webphoneobj);
```

```
sipnotifications.Start(); //start receiving the notifications
```

```
//configure
```

```
wobj.SetParameter("serveraddress", "sipdomain"); //the target SIP server
```

```
wobj.SetParameter("proxyaddress", "domain or ip:port"); //server network address if different from the above serveraddress
```

```
wobj.SetParameter("transport", "0"); //set transport protocol to UDP
```

```
wobj.SetParameter("username", "callerid"); //SIP user name
```

```
wobj.SetParameter("sipusername", "auth username"); //auth username if different from the above username
```

```
wobj.SetParameter("password", "auth password"); //SIP account password
```

```
wobj.SetParameter("register", "1"); //register after startup
```

```
wobj.SetParameter("ed137", 1); //switch to ED-137 behavior
```

```
wobj.SetParameter("pttid", X); //set PTT-ID if required
```

```
wobj.SetParameter("defptt", -1); //set default PTT state (default is 0/off/not speaking if ed137 is set)
```

```
wobj.SetParameter("prefcodec", "PCMU"); //set preferred coded to PCMA, PCMU or G.729
```

```
wobj.SetParameter("enablepresence", 0); //disable presence
```

```
wobj.SetParameter("parametername", "value"); //set any other parameters after your needs
```

```
//initialize the sip stack
```

```
wobj.API_Start();
```

```
//make a call
```

```
wobj.API_Call(-1,"peer"); //peer can be a full SIP URI or only the username part (for the server part it will use the serveraddress configured above)
//...meantime you might watch for any notifications of your interests such as STATUS for call state management or RTPT for peer PTT state changes
//...wait for call connect ("CallConnect"/"InCall" or "Speaking" STATUS) before toggling PTT
//PTT ON
wobj.API_ED137PTT (-1, 1); //Normal PTT ON for the current active line
//PTT OFF
wobj.API_ED137PTT (-1, 0); //PTT OFF (not speaking) for the current active line
```

Parameters

Configuration

This is a list of suggested settings.

JVoIP might be initialized with the following parameters for ED-137 (change/add/remove after your specific requirements):

ed137=1 //must be set!

pttid=? //set if needed. Otherwise will default to 0.

register=? //set it explicitly?

startsipstack=? //set it explicitly?

prefcodec=PCMA, PCMU or G.729

mediatimeout=? //set if you need (this is different from ED-137 rtp keep-alive timeout)

autoreadial=? //you might need to change it after your needs

enablepresence=0 //disable presence

hasconference=false //disable conference

Media processing

use_fast_stun=0 //disable STUN negotiations if peer is on the same network

use_rtpstun=0 //disable STUN negotiations if peer is on the same network

use_fast_ice=0 //disable ICE negotiations if peer is on the same network

aec=0 //disable AEC (acoustic echo cancellation) if not needed

aec2=0 //disable AEC (acoustic echo cancellation) if not needed

agc=0 //disable auto gain control if not needed

plc=0 //optionally disable packet loss concealment

denoise=0 //disable noise suppression for more strictness

p2prtp=0 //disable peer to peer direct media

p2prtencrypt=0 //disable peer to peer direct media

Strictness

transport=? //set it explicitly to avoid guessing

mediaencryption=? //set to 2 if you need SRTP media encryption. See the "How to configure SIPS" FAQ point for more details.

signalingport=? //set only if you wish to use a fix local port

enable_3pcc=0 //you might disable 3pcc if not needed for sure

enableautoaccept=? //set if you need incoming calls to be auto accepted

natopenpackets=0 //you might turn off RTP NAT open packets

handleexceptions=0 //set if you wish to disable JVoIP exception handler

changesystemsettings=0 //set if you wish to avoid JVoIP changing any JVM parameters

hasincomingcallpopup=0 //disable built-in GUI

allowmizuservices=0 //disable dependencies on mizutech services

geolocation=0 //don't lookup for country

altexternpublicip=0 //don't lookup for public IP from other sources

extchecklocalip=0 //don't lookup for public IP from other sources

callretrypreject=0 //see the "How to disable redial" for more

Streaming related

useaudiodevicerecord=false //set if not using the local audio recording device

useaudiodeviceplayback=false //set if not using the local audio playback device

See the "How to send audio stream" and "How to get the audio stream" FAQ points for more details.

ED-137 specific parameters

The below parameters are related only to ED-137 and might not be mentioned in the JVoIP documentation.

ed137

(int)

Enable/disable the ED-137 behavior. This is the most important parameter to be set.

Set to 0 to ignore the ED-137 specification (act as a legacy SIP client).

Set to 1 to turn on ED-137. This will change a list of internal settings and behaviors to conform with the ED-137 specification.

Default value is 0.

When set to 1, it will also auto set also the following configurations:

- processmutedrtp = 1;
- rtpextraheadernotify = 2;
- disconed137rtptimeout = 1;
- rtpkeepaliveival = 200;
- maxforwards = 15;
- video_config = 2;
- enable_video = false;
- video = 0;
- has_video = false;
- p2prtp = 0;
- jittersize = 2;
- playing = 0;
- ringincall = 0;
- defppt = 0;
- autore dial = 2;

- usepcma = 2;
- usepcmu = 2;
- useg729 = 2;
- usegsm = 1;
- usespeex = 1;
- usespeexwb = 1;
- usespeexuwb = 1;
- useilbc = 1;
- useopus = 1;
- useopuswb = 1;
- useopusuwb = 1;
- useopuswb = 1;
- useg7221 = 1;
- useg7221uwb = 1;
- usespeex_tmp = 1;
- usespeexwb_tmp = 1;
- usespeexuwb_tmp = 1;
- useopus_tmp = 1;
- useopuswb_tmp = 1;
- useopusuwb_tmp = 1;
- useopuswb_tmp = 1;

If you need to change any of the above settings then you can change it as you wish AFTER you set the ed137 parameter (because when you set ed137 it will reset all to the above values).

If set to 0 it will NOT reset the above parameters to their default value (Since JVoIP might cache the settings, you might set all of them separately or use the API_DeSettings function or delete the config file or use the autocfgsave/resetsettings parameters).

defppt

(int)

Default PTT state for new calls.

It has the same effect like calling the API_ED137PTT for each new call (but might be faster, immediately when the endpoint starts media).

Possible values:

-1: not set (default)

0: off (not speaking)

- 1: on (Normal PTT ON)
- 2: Coupling PTT ON
- 3: Priority PTT ON
- 4: Emergency PTT ON

If the ed137 parameter is set to 1, then it is auto configured to 0.

processmutedrtp

(int)

By default JVoIP ignores packets with no voice data payload, however ED-137 requires processing of all packets for the RTP extensions.

Set this parameter to 1 or 2 to conform to the ED-137 specification.

Possible values:

0: no

1: yes (if audioplayer already exists)

2: always (create audioplayer on demand if needed)

Default is 0.

pttid

(int)

PTT-ID (4 bits: b4-b7): The PTT-ID of the transmitting device.

This ID is assigned by the transmitting radio and SHALL be defined during the SIP session. If no PTT-ID is assigned the PTT-ID value PTT-ID= 0 SHALL be used.

You can also set a different PTT-ID for each call using the third parameter of the API_ED137PTT function.

maxforwards

(int)

Max-Forwards SIP header.

Default is -1 which means auto calculated (usually 40-75).

It should be set to less than 20 after the ED-137 specification.

If the ed137 parameter is set to 1, then it is auto configured to 15.

rtpkeepaliveival

(int)

Specify default value (milliseconds) for the R2S-KeepAlivePeriod.

If the ed137 parameter is set to 1, then it is auto configured to 200.

Possible values: 0 (disable) or between 20 and 1000

rtpkeepalivemultiplier

(int)

R2S-KeepAliveMultiplier setting.

Possible values: 0 (auto) or between 2 and 50.

Default is 0 which means auto calculated/negotiated but not offered (negotiated) by default (if peer will also not send then it will default to 15).

disconed137rtptimeout

(int)

Specify if to auto disconnect the call on RTP timeout after the ED-137 specification

Possible values:

0: no (default)

1: yes (auto negotiated/calculated if VF ever received)

2: yes (auto negotiated/calculated even if VF never received)

Other: specify exact milliseconds (not negotiated)

If the ed137 parameter is set to 1, then it is auto configured to 1.

autore dial

(int)

Redial on call disconnect.

Possible values:

- 0: no (default)
- 1: if it was disconnected because rtp timeout and we were the originator
- 2: if it was disconnected because rtp timeout
- 3: if we were the originator
- 4: if we were the called
- 5: always

The redial will be performed only if there are no other call in progress and the call was not locally terminated (such as pushing the Hangup button or calling the API_Reject or API_Hangup functions).

For better control on when to redial, you might use the API_Call instead on certain conditions as required for your app logic.

If the ed137 parameter is set to 1, then it is auto configured to 2.

rtpextraheadernotify

(int)

Specify whether to receive notifications about incoming RTP extra header changes.

Possible values:

0: no notifications

1: RTPE notifications

2: RTPT notifications

If the ed137 parameter is set to 1, then it is auto configured to 2.

mainlogicwaitmultiplier

(int)

Affects the mainlogic thread wait timer.

Default value is 5 which means maximum 50 millisecond wait time (but usually below 10 milliseconds while in use and not sleeping/idle).

Set to lower value for faster responsivity (might increase battery usage).

rtpextraheader_profile

(int)

Set the RTP profile.

Default is 0 (if not using ED-137) or 103 (0x0067h with ED-137). You might set it to 359 (0x0167) for ED137B.

API

ED-137 related API's. The most important is the API_ED137PTT function. Others were added to ease the ED-137 behavior implementations.

boolean API_ED137PTT (int line, int ptt, int pttid)

Toggle ED-137 Push to talk.

Parameters:

- line: channel number (-2: all, -1: current call, 1+ call channel number)
- ptt: 0: off (not speaking), 1: on (Normal PTT ON), 2: Coupling PTT ON, 3: Priority PTT ON, 4: Emergency PTT ON
- pttid: optional PTT-ID (otherwise if default / -1 is passed then it will be loaded from global config)

Will return true if initiated successfully or false on failure.

Internally this function will set the required RTP header extension and mute/unmute the required line(s). When muted, JVoIP will send the keep-alive packets according to the ED-137 specification.

boolean API_RTPHeaderExtension (int line, int profile, String extension)

The above API_ED137PTT will set the RTP extension after the ED-137 specification. You might use this function if you need something different. Set [RTP header extension](#) for the RTP packets.

The profile will be set as the first two bytes in the extension header (might be used as a custom identifier or parameter).

The extension can be one or multiple numbers separated by semicolon (;). These numbers will be sent as 32 bit words with the RTP header extension payload.

Example: `API_RTPHeaderExtension (-1, 1, "987");` //will set the profile number to 1 and the rtp extension word to 987 for the current call

Line -2 means all lines, -1 means current active line or use 1+ means for an existing line in call.

You can set a SIP header for the next call by using line number -3 or clear the old header(s) by using line number 0.

Set the profile to 0 and the extension string to empty to clear it.

boolean API_SetLineParameter (int line, String param, String value, int permanent)

Set parameter for the current/next/all/specific line. Only some of the parameters are supported by this function (listed below).

These are advanced settings and this function might be used only if you need some specific/different behavior for a specific line only.

Otherwise, the behavior of the different lines are loaded from the global configuration (parameters), from the circumstances or can be influenced by the other API functions (try to use the other API's instead of this when possible).

Parameters:

Line: channel number.

If -2, then it will be set as global config like the API_SetParameter.

If -1, then for the active line.

If 0, then it will be applied for all next/new channel(s).

Otherwise it will be applied for the specific line.

Param: settings key name

Value: value as string (or "NULL" to clear any old preset)

Permanent

If 1 then the setting will be kept for all future lines with the same line number

If 0 then the setting will be applied only for the current or very next channel with the same line

The following parameters are supported by this function:

peer (called number),address (will set both the serveraddress and proxyaddress),serveraddress,proxyaddress,

username,authusername,password,displayname,

voicerecording,muted,holded,maxsipmessagesize,codecretry,disccode,registerinterval,maxmsgresend,srtp_suite,strictsrtp,dtmfmode,rtp_timestamp,

rtp_seq,rtp_sscc,defpayload,setfinalcodec,presenceexpiresec,notifyexpiresec,sipproto,volumein,volumeout,volumering,videocalltype,videodirection,

keepvideosdponhold,defvideopayload,serverdomainandport,serveraddr,serverport,serverip,proxyport,proxyip,realm,

customsipheader,customsdffield,customsdpmediafield,sip_uui,sessionid,discreason,usesdpip,transport,line,callid,state

All preset settings for all lines can be removed by passing 0 for the line and empty or NULL for both the key and the value.

Examples:

```
wobj.API_SetLineParameter(1,"username","mia",0); //set the username to mia for the next call on line 1
```

```
wobj.API_SetLineParameter(2,"username","mia",1); //set the username to mia for all upcoming calls on line 2
```

```
wobj.API_SetLineParameter(2,"username","NULL",1); //clear the permanently preset username from line 2
```

```
wobj.API_SetLineParameter(-1,"muted","0",0); //unmute the current call (use the API_Mute instead)
```

```
wobj.API_SetLineParameter(0,"muted","4",0); //set the default muted state both side mute for the next call
```

```
wobj.API_SetLineParameter(0,"NULL","NULL",1); //clear all the permanent settings from all lines
```

boolean API_SetSDPField (int line, String field, int type)

Set a custom SDP field (a line in the SDP body) that will be sent with all messages.

Set the type parameter to 0 for global values (before the m= line) or 1 for media values (after the m= line).

Example: `API_SetSDPField(-3, "a=fid:118.005",1);`

Multiple headers can be separated by CRLF (\r\n). You can also set this with the [customsdffield](#) and [customsdpmediafield](#) parameters.

Line -2 means all lines, -1 means current active line or use 1+ means for an existing line in call.

You can set a SIP header for the next call by using line number -3 or clear the old header(s) by using line number 0.

Notifications

Watch for the RTPT notification for the RTP extra header changes sent by the peers.

RTPT,line,PTT-Type,SQU,PTT-ID,SCT,VF,extensionwords

Reports about [RTP extra header](#) changes.

If the ed137 parameter is set to 1, then JVoIP will send RTPT notifications instead of RTPE.

It is sent only if there RTP header extensions are received and the `rtpeextraheadernotify` parameter is set to 1.

Parameters:

- line: the session endpoint line number
- PTT-Type: (3 bits: b0-2): ptt off/on types: 0: off (not speaking), 1: on (Normal PTT ON), 2: Coupling PTT ON, 3: Priority PTT ON, 4: Emergency PTT ON
- SQU: (1 bit: b3)
- PTT-ID: (4 bits: b4-7)
- SCT: (1 bit: b8): simultaneous transmission
- VF: (1 bit : b31) Visibility Flag marker bit. Set to 1 when the peer received an RTP packet from us
- extension: the whole 32 bit words as int converted to string. If there are more then one then separated by semicolon (;)

Example: RTPT, 1,1,0,0,0,1,76543

Most probably you will also need to process some other notifications such as STATUS, DTMF or CDR.

See the “Notifications” chapter and the “RTP header extension” FAQ point in the JVoIP documentation for more details.

FAQ, Tasks, Notes

Various notes and usage suggestions.

Local audio streaming

If JVoIP will not use audio device recording/playback then you will have to feed/receive the audio stream from your own app.

See the following FAQ points in the JVoIP documentation about this topic:

- How to send audio stream
- How to get the audio stream
- Using JVoIP on a server

RTP statistics

See the “Media statistics” FAQ point in the JVoIP documentation.

Target URI

You might need use specially formatted target URI as specified at the 3.3 ADDRESS FORMAT chapter in the ED-137 specification.

If the server SIP domain is different from the server network address, then set the domain as the “serveraddress” parameter and the network address as the “proxyaddress” parameter.

The full target URI can be passed also for the `API_Call()` function. Set the username part after your needs.

3.3 ADDRESS FORMAT

SIP URI for ATS: `sip:txrx.frequency.atsu@radio_site_id.local_domain`

SIP URI for user roles: `sip:user_role@atsu.centre_icao_id.local_domain`

username: `txrx.frequency.atsu` or `user_role`

serveraddress: `radio_site_id.local_domain` or `atsu.centre_icao_id.local_domain`

proxyaddress: the real server network address

Multiple calls

For multiple simultaneous calls you might use multiple webphone instances or use one instance for multiple calls.

If you will use one instance for multiple calls then you will need to handle the line management explicitly. See the “Multiple lines” FAQ point the JVoIP documentation for the details about this topic.

SIP headers

Extra SIP headers can be set with the `API_SetSIPHeader` function or with the `customsipheader` parameter.

There is no any priority header sent by default.

You might set it to emergency/normal using the `API_SetSIPHeader` function if required.

Example: `wobj.API_SetSIPHeader(-3, "Priority=normal");`

Or use the `customsipheader` if you wish to set it for all sessions.

SDP parameters

The SDP parameters are described at the 3.2.5 MESSAGE BODY chapter in the ED-137 specification.

All ED-137 parameters are optional with meaningful default values and JVoIP doesn't send them by default.

Use the `API_SetSDPField` function if you wish to set any extra parameter(s).

SDP and RTP related random notes:

SDP send

`a=X-PTT-PCMU/8000 ?`

`a=type:radio/coupling`

`a=bss:RSSI/AGC/C/N/PSD`

`a=sigtime:Signalling Info Time Period (PTT/PTT OFF time period in multiple of the packet interval (default value is 1 -> that means the Radio`

signalling info is sent in every RTP with voice packet))

`a=ptt_rep:PTT OFF Repetition (Number of PTT OFF messages (0-3) (default value is 0 -> only one PTT OFF message)`

`a=R2S-KeepAlivePeriod (Maximum time between each keep alive message (20-1000ms) Default Value = 200ms)`

`a=R2S-KeepAliveMultiplier (Number of Keep Alive messages error before Time Out of the session (2-50) Default Value = 10)`

`a=fid:Frequency ID (Frequency ID (7 character – ICAO standard presentation e.g. 118.005))`

bss header?

`a=sigtime header: PTT/PTT OFF time period`

`a=ptt_rep: number of PTT OFF messages (0-3) (default value is 0 -> only one PTT OFF message`

`a=R2S-KeepAlivePeriod: Maximum time between each keep alive message (20-1000ms) Default Value = 200ms`

`a=R2S-KeepAliveMultiplier: Number of Keep Alive messages error before Time Out of the session (2-50) Default Value = 10`

`a=fid: Frequency ID (7 character – ICAO standard presentation e.g. 118.005)`

special target URI (just set the proxy to the network address and use anything as target user / domain)

RTP header extension for PTT

RTP header extension for: PTT, keep-alive, (A/C call, quality index, time stamping info)

The Type field SHALL be 0x67h.

5.10.3 RTPTx Information field

PTT

extension type: 0x67h

type (3 bits: 0-2): ptt off/on types

SQU (1 bit: 3): always 0 | SQ OFF: 0, SQ ON: 1

PTT-ID (4 bit: 4-7): 0 or configurable

SCT (1 bit: 8): simultaneous transmission always 0 | 0/1

X (1 bit: 9): 0 | 1

extension (20 bit: 10-30)

type: (4 bits: 10-13) 0: nothing, 1: BSS, 2: climax delay

length: (4 bits: 14-17)

value: (x bits: 18-30)

VF: (1 bit: 31): set to 1 when the end point has received a RTP packet

RTP keep-alive packets

extension type: 0x67h

X=1

PT=123

CC=0

seq

timestamp: 0

SSRC: different

Auto answer

See the “Auto-answer” FAQ point in the JVoIP documentation if you wish to auto accept (certain) incoming calls.

Terms and other notes

Voice communications for air traffic management, whether it is Air-Ground (A-G) or Ground-Ground (G-G) were predominantly over TDM based Air Traffic Management (ATM) networks.

With the latest developments in EUROCAE (European Organization for Civil Aviation Equipment) ED-137 inter-operability standard, it is now possible to implement VoIP

ATM: Air Traffic Management

AOC: Airline Operational Control

The radio systems as well as the air traffic controllers with their controller working positions (CWP)

are connected with the terrestrial voice communications system (VCS).

Answerback: After a call request (DTMF code) from an aircraft to the console, the crew hears an answerback tone or message which confirms that the console system has received the call and the call will be handled by a dispatcher. Scout can present a tone or a .wav file. This feature can also be disabled.

Inbound Emergency Call: A dispatcher can receive inbound emergency calls via configured Emergency Call-in Codes. You can define three Emergency Call-in Codes per system. Dialing one of these defined codes from the field generates an emergency call at all the consoles in the system.

Instant Transmit: A dispatcher can immediately transmit to a predetermined endpoint by touching and holding the Instant Transmit pad on the screen. Each endpoint can have its own Instant Transmit pad.

The Scout VoIP console system supports a wireline connection to ED-137 radio networks through a direct IP connection to the ground radio station (GRS).

AudioRecorder/AudioQueue: ARecord_Read (read audio device) -> ARecord_Preprocess -> (volume, aec, voicerec) -> ARecord_Encode (codec) -> ARecord_Send (rtp header) -> RTPReceive (forward to all ep, ep.RTPReceive, ep.RTPSend)

usage:

VCS->Radio call

RTP header extension for real time transport of PTT signal and keep-alive

From the VCS side to the Radio side: Push to Talk (“PTT)

From the Radio side to the VCS side: A/C call, Reception Quality Information

When PTT is not active, Voice samples SHOULD NOT be transmitted in the RTP packet. The RTP Packet SHALL include only the header extension with the appropriate PTT status.

When PTT is active, the RTP packet SHALL include the PTT information in the header extension and the voice samples.

The PTT signal SHALL be transmitted by the VCS in the RTP extension field to the radio interface as soon as the controller at the VCS side has activated the PTT key at their controller working position.

PTT activating delay SHALL have a maximum value of 100ms

the deactivation message SHOULD be repeated in a short period a number of times (e.g. 3 times)

Applications without silence suppression SHALL set the marker bit to zero during periods of silence

RTP used for:

RTP header extension SHALL be used for the real time transportation of PTT signal from the VCS to the Radio equipment and A/C Call and BSS quality signals from the Radio towards the VCS.

Additionally, RTP protocol SHALL be used to control the link between the VCS and the radio using “Keep Alive” mechanism.

PTT signal from the VCS to the Transmitter Radio equipment in an unidirectional mode and also for the real time transportation of the A/C Call and BSS quality signals in the opposite side from the Receiver Radio equipment

If real time information is needed without audio, RTP protocol using Header extension only SHALL be used to convey information over the links between the VCS site and each Radio equipment

a) PTT (Push To Talk)

b) Aircraft Call (A/C call), also nominated Squelch (SQL);

c) Quality index (used for the Best Signal Signalling feature)

d) Time stamping information (used for CLIMAX application)

e) Reserved field for proprietary applications

f) Keep alive message

Troubleshooting

For testing you should always set the jvoip `loglevel` parameter to 5 (higher log levels might be useful only for RTP debugging).

Use the logs and a network capture tool (such as Wireshark) for troubleshooting.

See the “How to get logs” FAQ point in the JVoIP documentation for more details.

Possible issues:

- Issues/bugs with general ED-137 specific behavior (changes covered by the ed137 parameter)
- PTT ON/OFF behavior (muted RTP)
- SIP headers not sent/parsed after the ED-137 specification
- SDP parameters not sent/parsed after the ED-137 specification
- RTP (especially the RTP extension bits) not sent/parsed after the ED-137 specification
- Local audio streaming issues (check your implementation because this is a well tested functionality already used also by other customers)
- Multi calls / prioritization / BSS handling
- Other missing / not implemented functionalities (try to implement it externally when possible; otherwise ask us)

If you need our help, please [contact us by email](#) with the followings:

- Detailed problem description (including What you did? What happened? What should happen?)
- Detailed JVoIP log (the generated webphonelog.dat logfile, the java console output or as described at the “How to get logs” FAQ point in the documentation)
- [Wireshark](#) pcap trace especially if the problem is RTP related
- Server side log if possible (must include also the SIP signaling)

Implementation

Mizutech checklist about the ED-137 specification (done / pending / not planned tasks).

Implemented/Changes

- ED-137 usage instructions (this document)
- **ed137** global config to change a list of internal settings behaviors (maxforwards=15, etc)
- ed137 global config to change a list of behaviors
- restrict **SIP headers** according to the ED-137 specification (3.2.4.3 SUPPORTED HEADER FIELDS / 3.2.4.3.1 User Agent Request Headers)
- remove extra X- headers
- avoid advertising or using SIP methods not listed in the ED-137 specification, such as PRACK
- default **codec restriction**: Use only the audio codecs specified in ED-137: PCMU, PCMA and G.729 only by default
- fix frame rate / voice packetization (20 msec or one frame, avoid using high frame rates)
- Max-Forwards should be less then 20 (maxforwards parameter)
- Extra parameters: pttid, rtpkeepaliveival, rtpkeepalivemultiplier, disconed137rtptimeout
- Subject: radio SIP header
- API for PTT on/off: **API_ED137PTT API**
- natopenpackettype parameter change (option 3 for ED-137)
- IPv6: implemented for both the signaling and the media
- Real Time Session Supervision
- ED-137 specific keep-alive packets (payload: 123, timestamp: 0, ssrc: different, rtp extension)
- Keep Alive: RTP packets with no voice payload one in every x msec (200 by default, configurable and/or negotiated by the peer)
- send **RTP keep-alive packets** when PTT is off (like muted with rtpkeepaliveival)
- send RTP keep-alive even when the call is muted or holded
- process RTP extensions
- changed RTP NAT open packets to ED-137 keep-alive packets
- set VF bit at runtime if RTP was received
- extract VF bit at runtime if RTP was received: lastpeermediarec
- disconnect the call on RTP timeout (like mediatimeout, but using lastpeermediarec)
- auto redial on call disconnect
- **RTPT notifications**
- new **API_SetLineParameter** to easily preconfigure the different lines
- special target URI (set the proxy to the network address and use anything as target user / domain)
- change default jittersize to 2
- processmutedrtp setting to process RTP packets with zero payload size
- document all strictness related parameters
- test ED-137 behavior including signaling, RTP, RTP extension, PTT ON/OFF, keep-alive, etc. Analyze using Wireshark.

Not implemented

- G.728 (not planned)

- IsMultiCalls? (Improvement might be needed; check if recording/playing?)
- Send some SDP headers by default? (a=R2S-KeepAlivePeriod / a=R2S-KeepAliveMultiplier / etc)
- Send and process extension extensions? (BSS, CLIMAX); Best Signal Selection (BSS), Climax time delay (CLD) ...auto switch to best BSS call? or it will be handled externally?
- Add extension also to rtp keepalive packets? (WriteRTPFirst)
- SUBSCRIBE/NOTIFY with WG67 KEY-IN Event Package (the ability to send such SUBSRIBE and send notifications about NOTIFY)
- Handle call prioritization? (a priority radio call shall not be interrupted)
- Handle PTT stream priority: playback only the highest priority call/stream if multiple calls?
- Rx/Tx path requirement? (not needed; must be implemented at the radio side and/or in hardware):
 - The radio receiver SHALL produce a $-10\text{ dbm0} \pm 1\text{ dB}$ audio output level when a 30% AM modulated RF signal is applied at the RF input.
 - The radio transmitter SHALL produce a 30% AM modulated RF output signal, when a $-10\text{ dBm0} \pm 1\text{ dB}$ audio input level
- RTPT notification change min/max interval (rtpeextraheadernotify)

Resources

JVoIP homepage: <https://www.mizu-voip.com/Software/SIPSDK/JavaSIPSDK.aspx>

JVoIP documentation: <https://www.mizu-voip.com/Portals/0/Files/JVoIP.pdf>

ED-137 specification: http://startrinity.com/voip/Resources/ED-137_Interoperability_Standards_for_VoIP_ATM_Components_Part_1_Radio.pdf

Contact email: info@mizu-voip.com