**MsmlTool Guide**

MsmlTool is a JAVA based test application for passing and receiving MSML messages to and from the Powermedia XMS.

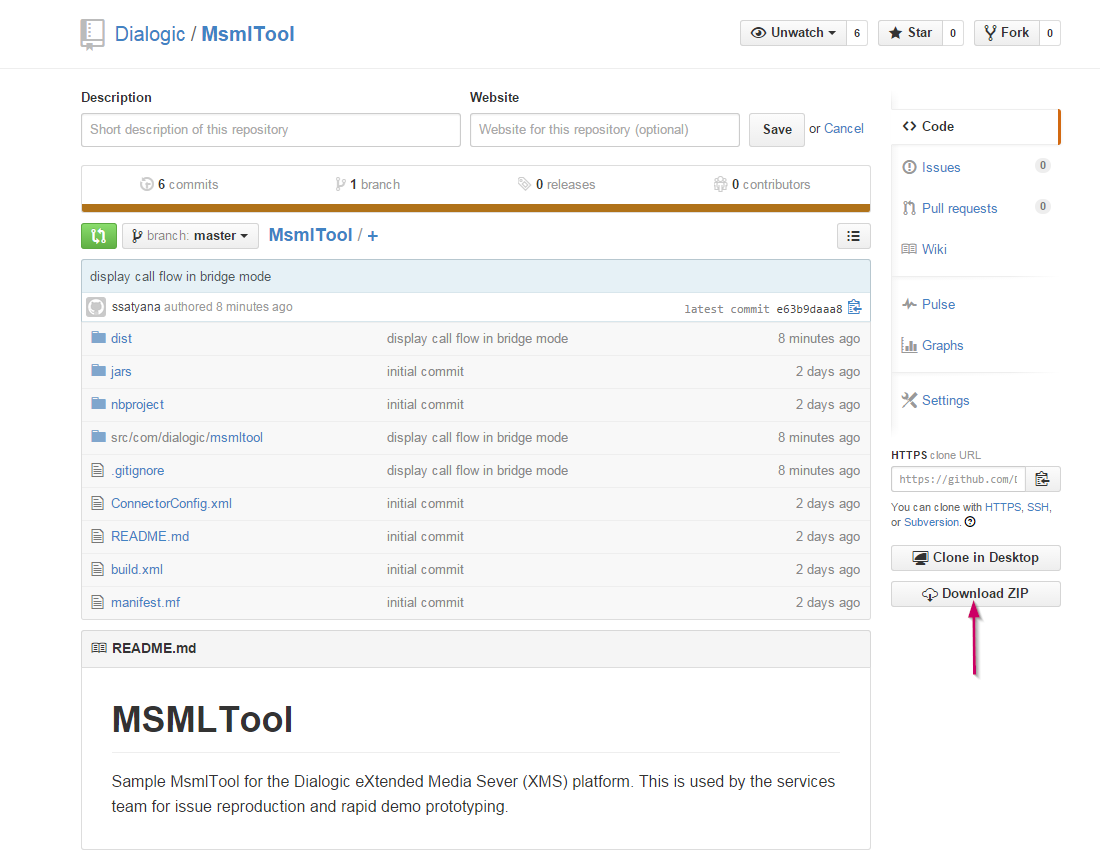
**Prerequisites**

* JAVA 1.8

**Procedure**

To use MsmlTool, download the zip file from Github.

<https://github.com/Dialogic/MsmlTool>



Unzip the downloaded file and then go to the top level directory where you will see the /dist directory. From the top level directory run the tool as follows:

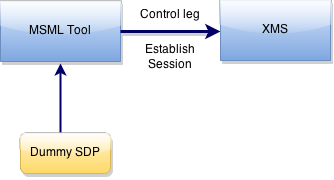
* java –jar dist/MsmlTool.jar

C:\Users\ssatyana\Desktop\PROJECT-1\MsmlToolDocumentation\cmdLine.PNG

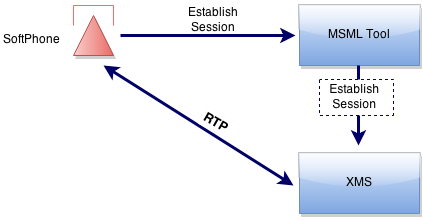
**Call modes**

There are 2 types of mode: Direct Media-less mode and the Bridged mode.

1. Direct Media-less Mode: Use MsmlTool to call the XMS directly. This mode establishes a SIP session by sending in a dummy SDP. This mode can be used to test your XMS signaling, controlling etc.

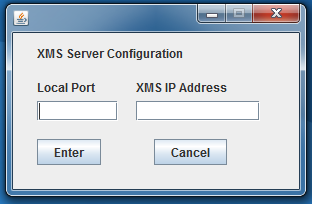


1. Bridged Mode: Caller calls in to the MsmlTool; MsmlTool then makes an outbound call to XMS. MsmlTool acts as a back to back UA which bridges the call. In this mode RTP is peer to peer and you can use the MsmlTool’s control window to exchange MSML content.



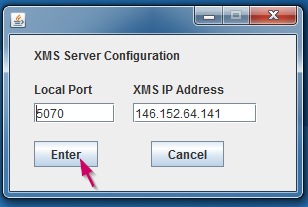
**GUI**

When you run the MsmlTool jar, it opens up the first screen allowing the user to enter the configuration details.

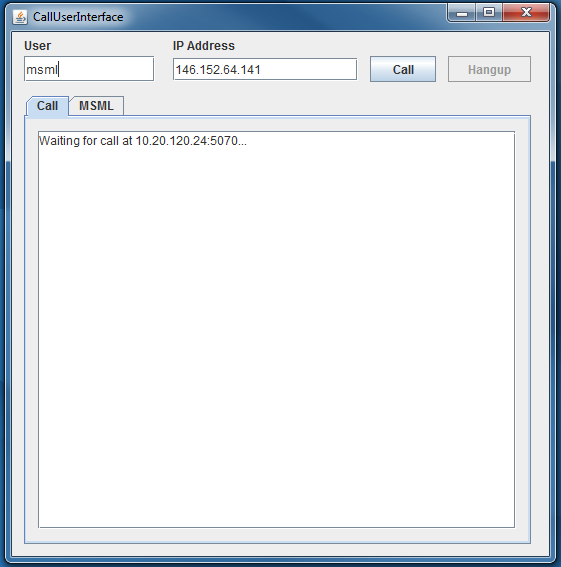


The window consists of 2 fields:

1. Local Port – This is the port you want to listen to for the inbound calls.
2. XMS IP Address – This is your XMS IP Address.



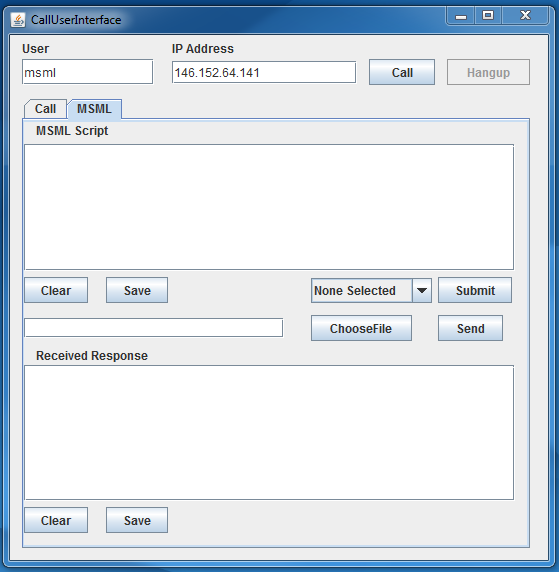
After entering the configuration details, hit enter. This will open another window which allows the user to make calls and send MSML content.



**Call User Interface Components**

This section explains the various buttons, text fields of the Call GUI.

1. User Text Field – The user name to pass in to the Request URI.
2. IP Address Text Field – The XMS Address to call to.
3. Call Button – Makes a call to the entered address.
4. Hangup Button – Hangs up the call.
5. Call Tab – Displays the call flow messages.
6. MSML Tab – Used to send MSML scripts/content (see below).



MSML Tab Components:

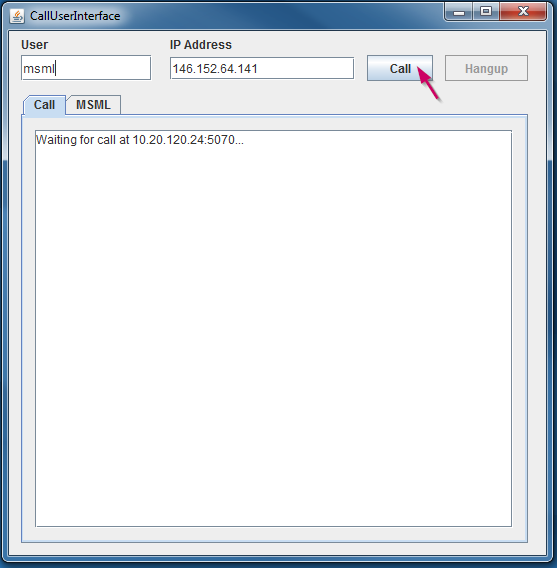
* MSML Script Text area – Used to enter the MSML content.
* Received Response Text area – Displays the responses received by XMS.
* Clear Button – Clears the text area.
* Save Button – Saves the contents of the text area to a file.
* Drop down menu – Consists of sample MSML scripts to choose from.
* Submit Button – Submit the selected sample script. This populates the MSML script text area with the selection. (**UPDATED**: Submit button has been removed in the latest code, Selecting from drop down menu will populate the MSML script text area)
* ChooseFile Button – Used to choose a file consisting of MSML script.
* Send button – Sends the MSML script to XMS

**Basic operation**

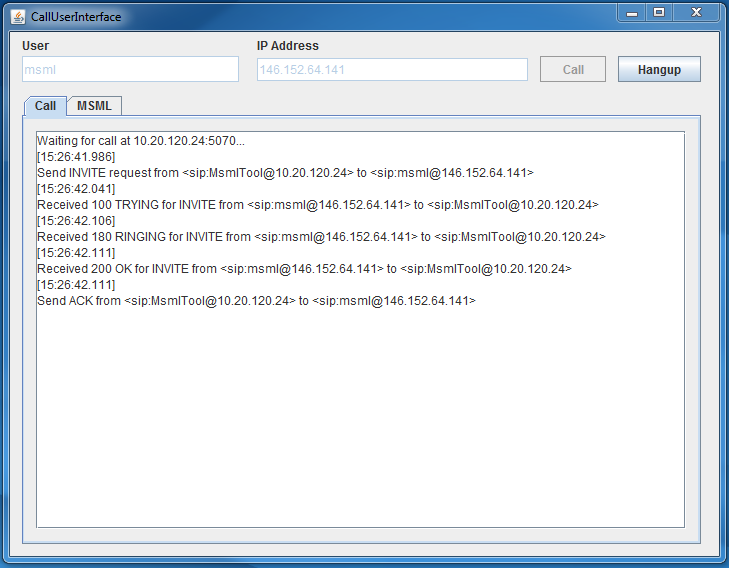
This section describes the basic flow to create calls using the media-less and bridged mode.

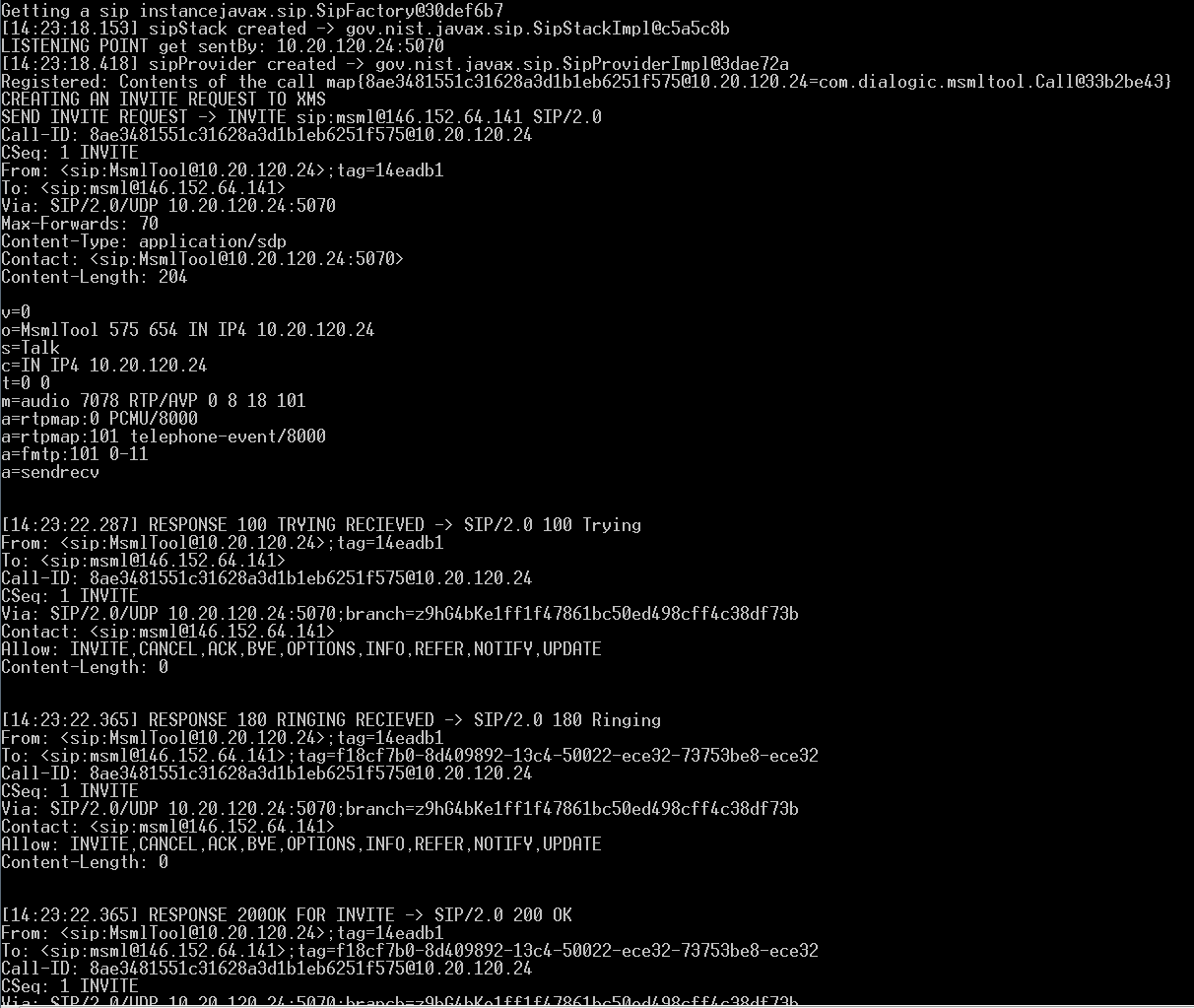
1. Media-less mode

* Once the call GUI is launched, make a call to XMS by clicking the call button.



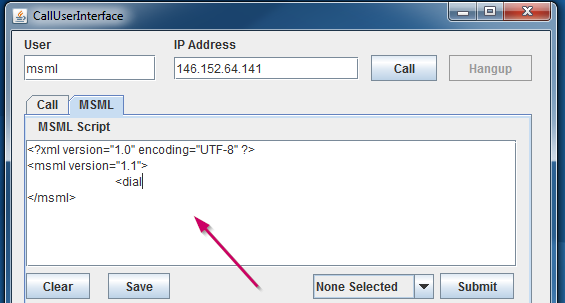
* After clicking the call button, the call tab shows a series of request and responses exchanged between the MsmlTool and the XMS. Detailed request/response messages can be seen on the cmd prompt.



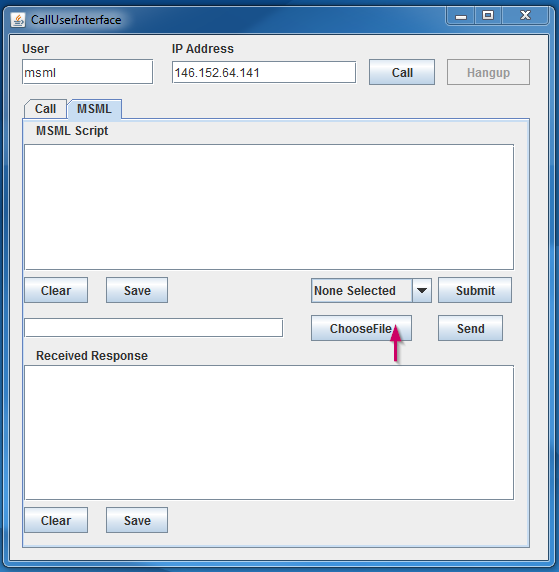


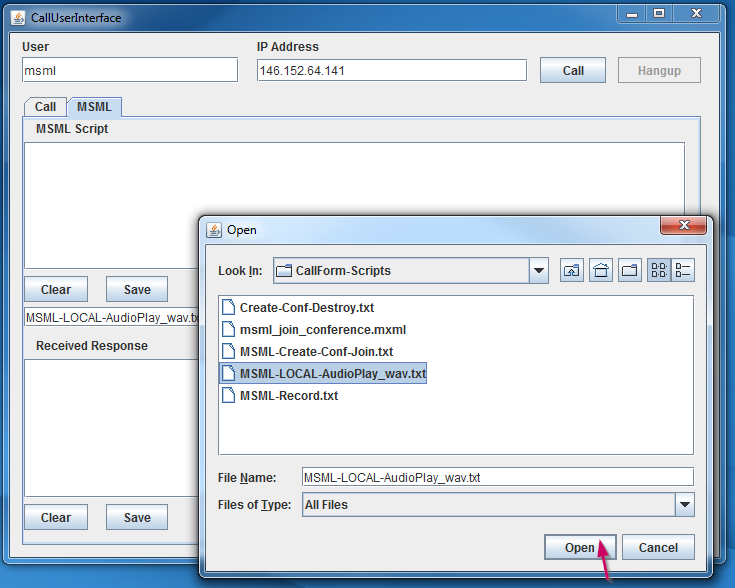
* Now the sip session has been established. Use the MSML tab to send MSML content. This can be done either by selecting a file from the local system consisting of MSML script, typing inside the MSML script text area or by selecting sample scripts from the drop down menu. After entering the MSML script click send.

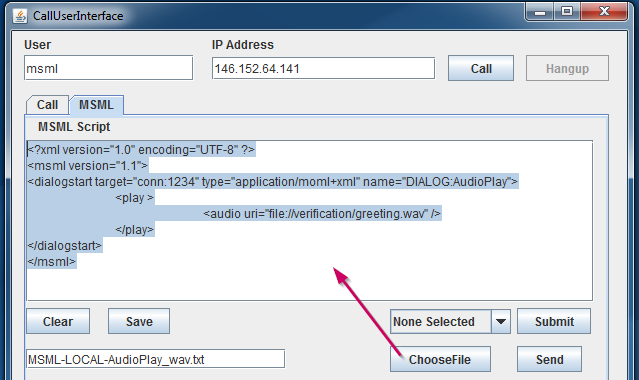
1. Typing to the MSML Script text area



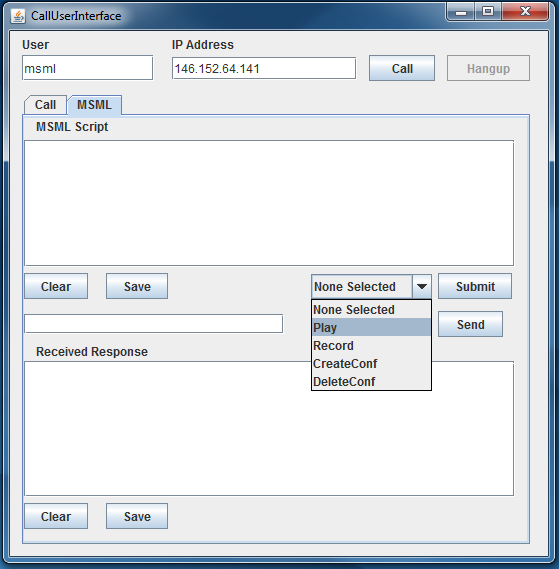
1. Choosing a file: Click on the ChooseFile button. This opens up your local directory, select file and open it. The contents of the file gets populated to the MSML text area.



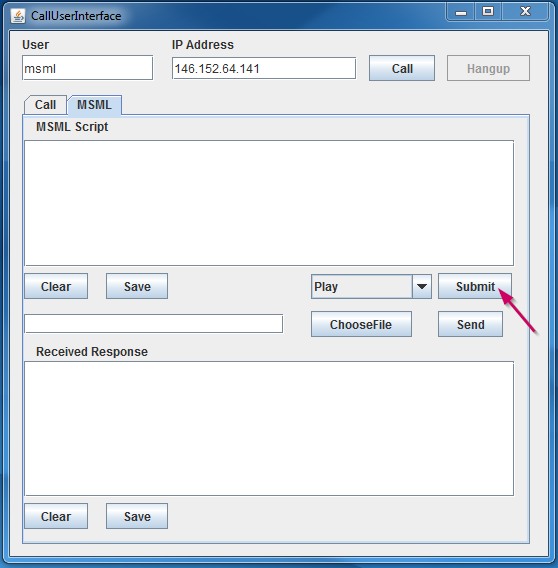


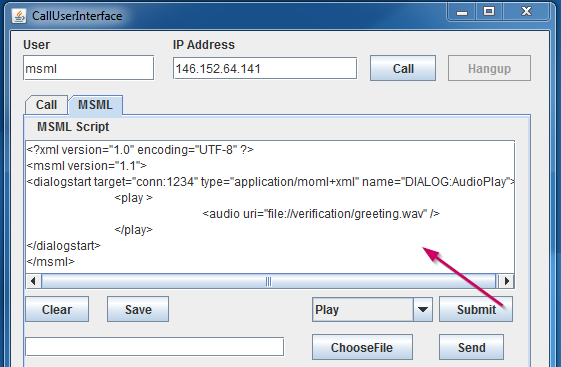


1. Choosing MSML script from the drop down menu.

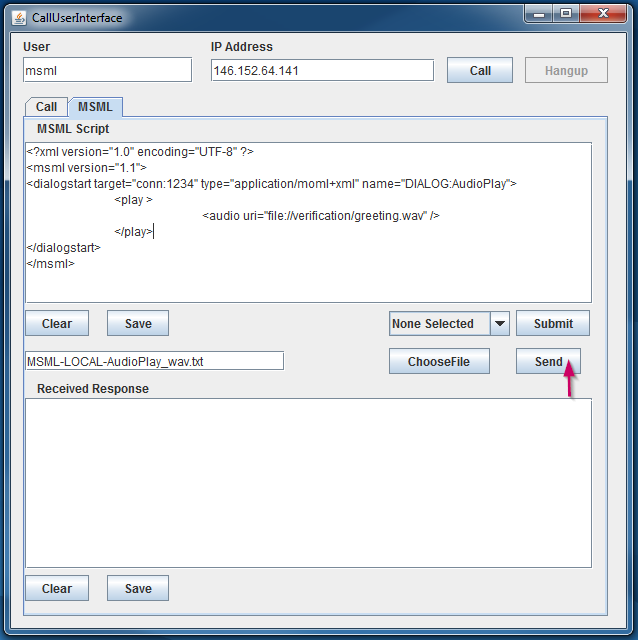


(**UPDATED**: Submit button has been removed in the latest code, Selecting from drop down menu will populate the MSML script text area)

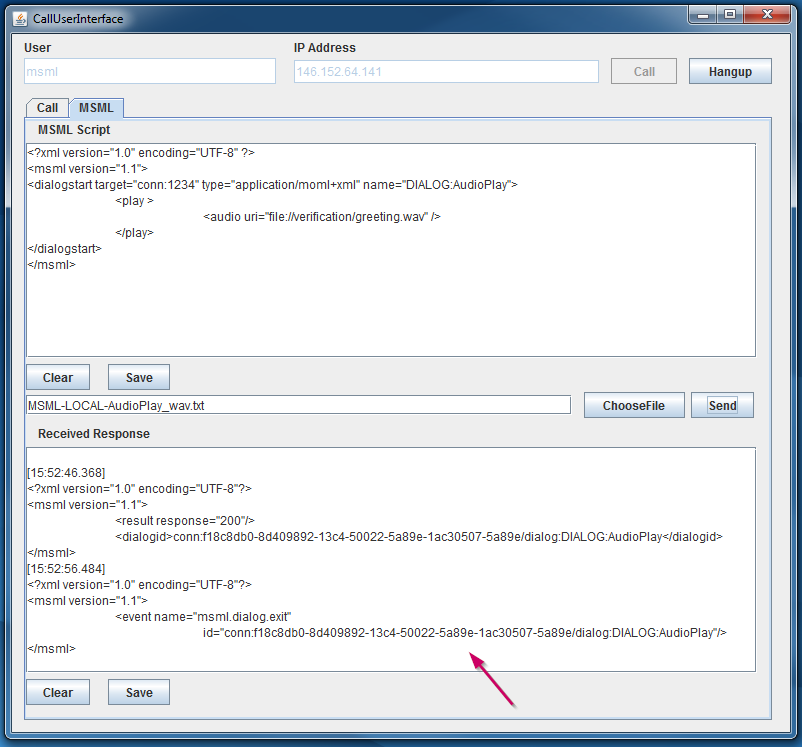




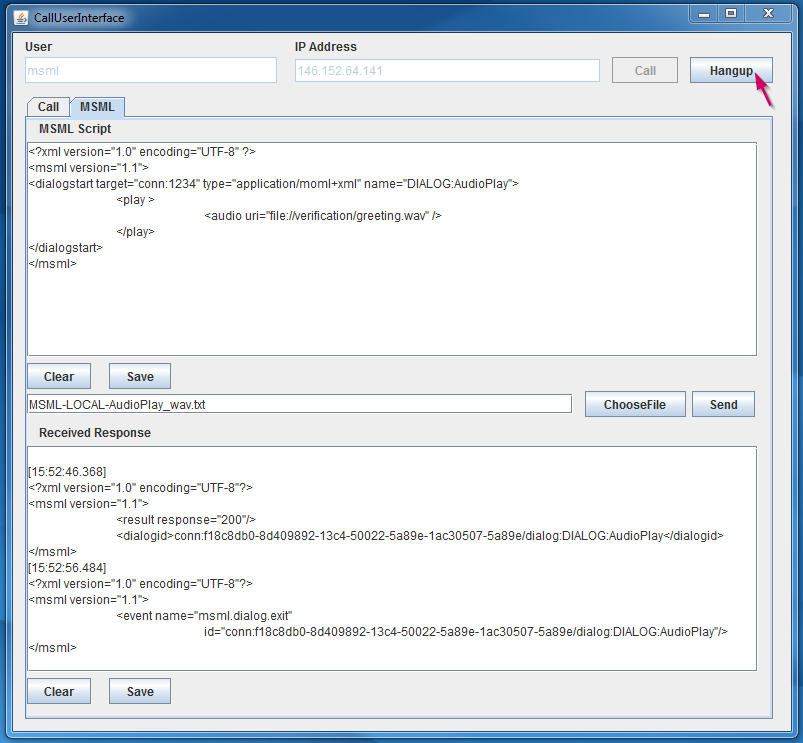
* After selecting the MSML content using either of the above methods, click on send button.



* The MsmlTool creates an info message and sends in the MSML content with this info message to XMS. The response received by the XMS gets displayed in the “Received Response” text area.

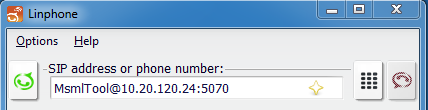


* Use the Hangup button to cut the call. Once this button is clicked the MsmlTool creates a bye request to XMS and terminates the call.

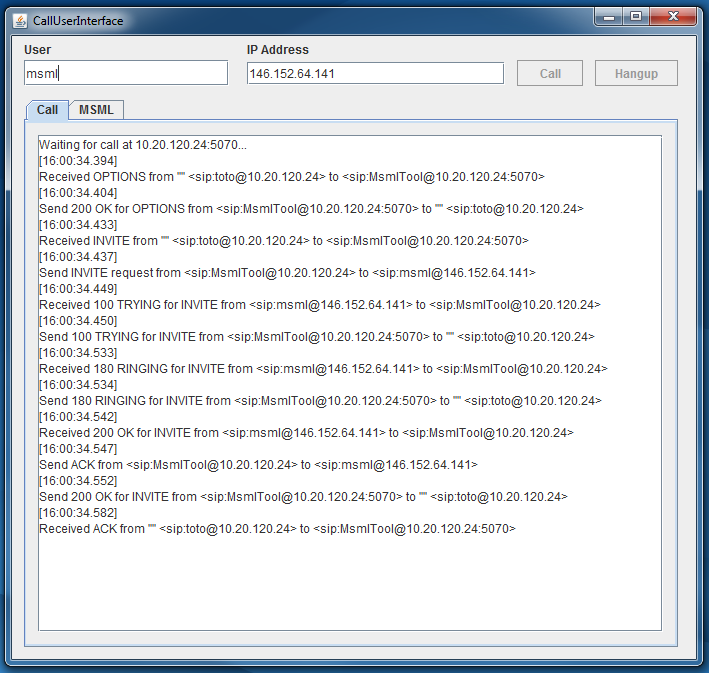


1. Bridged mode

* In order to accomplish this, you need to use a softphone (ex.linphone) to call the MsmlTool.
* Once the Call GUI is up, make a call to the MsmlTool using a softphone.
* The MsmlTool listens to this incoming request and creates an outbound call to XMS thus creating a sip session.



* Shows the request/responses between the Softphone->MsmlTool and MsmlTool->XMS



* Once the call is set up, use the MSML window to exchange info messages between the MsmlTool and XMS.

**Version 2.0**

This section includes all the updates made to MsmlTool in the year 2016. MsmlTool can now be used to test against MRB.

The following are the updates; each of these will be explained in detail later.

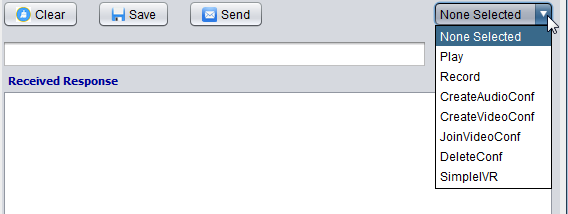
1. Modified the code to use Client Library components.
2. Supports Re-invite
3. Add new, update existing MSML Scripts
4. Updates to user interface view

**Updates Explanation**

|  |  |
| --- | --- |
| **Update** | **Description** |
| Modified code to use Client Library components | The functionality of the MsmlTool remains the same. Some of the classes (call and connector) have been removed and now we use the Client library jar classes.  XMSClientLibrary.jar has been added to jars folder |
| Supports Re-invite | This feature is to support the re-invite sent by MRB in case of failover. When one of XMS nodes failes, calls will be moved to another XMS, during this event MRB sends a re-invite. |
| MSML Scripts | Existing script have been updated to include more attributes. New scripts are added to the drop down list in GUI.  Refer Page 19 for the scripts |
| GUI | New icons are added to button. Functionality remains the same. |
| MRB call flow | When called to MRB, user has to use the port number along with the MRB IP address |

MSML Scripts

|  |  |
| --- | --- |
| **Type** | **Script** |
| Play | <?xml version="1.0" encoding="UTF-8" standalone="yes"?>  <msml version="1.1" xmlns:ns2="http://www.dialogic.com/DialogicTypes">  <dialogstart target="conn:1234" type="application/moml+xml" name="Play">  <group topology="parallel">  <play>  <media>  <audio uri=<file://verification/greeting.wav> format="audio/wav;codec=L16" audiosamplerate="16000" audiosamplesize="16"/>  </media>  <playexit>  <exit namelist="play.end play.amt"/>  </playexit>  </play>  <collect>  <pattern digits="#">  <send event="TermkeyRecieved" target="source" namelist="dtmf.digits dtmf.len dtmf.end"/>  <send event="terminate" target="play"/>  </pattern>  </collect>  </group>  </dialogstart>  </msml> |
| Record | <?xml version="1.0" encoding="UTF-8" standalone="yes"?>  <msml version="1.1" xmlns:ns2="http://www.dialogic.com/DialogicTypes">  <dialogstart target="conn:1234" type="application/moml+xml" name="Record">  <group topology="parallel">  <record beep="true" audiodest="file://recorded/Test.wav" format="audio/wav;codec=L16" audiosamplerate="16000" audiosamplesize="16" maxtime="10s">  <recordexit>  <exit namelist="record.end record.len"/>  </recordexit>  </record>  <collect>  <pattern digits="#">  <send event="TermkeyRecieved" target="source" namelist="dtmf.digits dtmf.len dtmf.last"/>  <send event="terminate" target="record"/>  </pattern>  </collect>  </group>  </dialogstart>  </msml> |
| Create conference + join (Audio) | <msml version="1.1">  <createconference name="XMSConference\_1" deletewhen="nomedia" mark="1" term="true">  <audiomix id="mix12345"/>  </createconference>  <join id1="conf:XMSConference\_1" id2="conn:1234" mark="2">  <stream media="audio"/>  </join>  </msml> |
| Create Conference (Video, 4 panel) | <?xml version="1.0" encoding="UTF-8" standalone="yes"?>  <msml version="1.1" xmlns:ns2="http://www.dialogic.com/DialogicTypes">  <createconference name="XMSConference\_1" deletewhen="nomedia" term="true" mark="1">  <videolayout>  <root size="VGA"/>  <region id="1" left="0" top="0" relativesize="1/2"/>  <region id="2" left="50%" top="0" relativesize="1/2"/>  <region id="3" left="0" top="50%" relativesize="1/2"/>  <region id="4" left="50%" top="50%" relativesize="1/2"/>  </videolayout>  </createconference>  </msml> |
| Join video conference | <?xml version="1.0" encoding="UTF-8" standalone="yes"?>  <msml version="1.1" xmlns:ns2="http://www.dialogic.com/DialogicTypes">  <join id1="conn:1234" id2="conf:XMSConference\_1" mark="1">  <stream media="audio"/>  <stream display="1" dir="from-id1" media="video"/>  <stream dir="to-id1" media="video"/>  </join>  </msml> |
| Delete conference | <?xml version="1.0" encoding="UTF-8" standalone="yes"?>  <msml version="1.1" xmlns:ns2="http://www.dialogic.com/DialogicTypes">  <destroyconference id="conf:XMSConference\_1" mark="1" />  </msml> |
| SimpleIVR | <?xml version="1.0" encoding="UTF-8" standalone="yes"?>  <msml version="1.1" xmlns:ns2="http://www.dialogic.com/DialogicTypes">  <dialogstart target="conn:1234" type="application/moml+xml" name="Collect">  <group topology="parallel">  <play barge="true" cleardb="true">  <media>  <audio uri="file://verification/play\_menu.wav"/>  </media>  <playexit>  <send event="starttimer" target="collect"/>  </playexit>  </play>  <collect cleardb="true" fdt="20s" idt="2s" starttimer="true">  <pattern digits="#">  <send event="termKey" target="source" namelist="dtmf.digits dtmf.len dtmf.end"/>  </pattern>  <pattern digits="x"/>  <noinput>  <send event="noinput" target="source" namelist="dtmf.digits dtmf.len dtmf.end"/>  </noinput>  <nomatch>  <send event="nomatch" target="source" namelist="dtmf.digits dtmf.len dtmf.end"/>  </nomatch>  <dtmfexit>  <send event="dtmfexit" target="source" namelist="dtmf.digits dtmf.len dtmf.end"/>  </dtmfexit>  </collect>  </group>  </dialogstart>  </msml> |



GUI icons

