Configuration

Configuration file: vxml.conf

The configuration file is /etc/asterisk/vxml.conf. It is editable by typing:

```
# vi /etc/asterisk/vxml.conf
```

The vxml.conf file is divided in serveral sections to set different group of parameters and functions:

```
; commented text starts with a “;”

[general]
...

[control]
...

[license]
...

[account1]
...

[account2]
...

[accountn]
...
```

NOTE:

Please note that before configuring the Asterisk Module, you should make a backup copy of vxml.conf. Vxi upgrade process keep the current configuration but some specific updates can be mandatory for future releases.

General Section [general]

The following section covers general functions of VoiceXML Asterisk module.

Key	Values	Default	Purpose
msqkey*	decimal number	1973	Message queue ID.
msqfactor*	value in Kbytes	10	Size by VoiceXML channel allocated for the message queue.

Key	Values	Default	Purpose
msqfirsttimeout*	value in milisecondes	100	First timeout, the next ones are set to 0.5s, setting lower value can increase the reactivity of the browser.
msqtimeout*	value in secondes	30	Timeout when the module not receive commands from the VoiceXML browser (can stop the Asterisk if autoexit set to yes).
msqsize*	auto or value in Kbytes	auto	Size of the message queue or if auto msqfactor x number of VoiceXML channels.
msqlock*	yes/no	no	Force to wait and disable the timeouts.
debug	yes/no/full	no	Enable or disable the logs from the Asterisk module and the VoiceXML Browser.
paramseparator*	character	,	Separator for Asterisk application parameters.
systemname	auto/value string name	(empty)	Empty not change the systemname (used in the uniqueID for the CDR), auto use the hostename of the server or set a string to define the system name.
autoanswer	yes/no/prompt/recognize/ringing/	yes	yes, answer the call with the vxml() execution, no never answer the call you need to call the Answer() application from the VoiceXML, **prompt** answer before first prompt/audio play, recognize answer before first recognize action, ringing no answer send a single audio frame to cancel the ringing tone.
autohangup	yes/no/soft/exit	no	yes execute hangup() a the end of the vxml() execution, soft execute softhangup(), exit execute hangup() and set the variable VXML_EXIT with the VoiceXML exit value.

Key	Values	Default	Purpose
autoexit	yes/no	no	Stop the Asterisk with return code -1, if VXI interpreter no answer (msq timeouts).
monitor	yes/no/old/1..100	no	yes execute mixmonitor() (or monitor() if old) before starting the VoiceXML session. You can set a random value (using a %).
monitordirectory	directory	mixmonitor directory	Set the mixmonitor() monitor() directory or use the default one (/var/lib/asterisk/monitor).
monitorformat	wav/gsm/pcm/ulaw/alaw	wav	Set the format for the mixmonitor recording.
defaulttimeout	value in milisecondes	5000	Default timeout if not set in the VoiceXML syntax.
defaultinterdigittimeout	value in milisecondes	3000	Default interdigit timeout if not set in the VoiceXML syntax.
priorityevents	yes/no	yes	If using single DTMF events and DTMF inputs/grammars, yes force to check the first DTMF to the DTMF events as soon as it is receive, no will wait full entring or a timeout.
tdd*	yes/no	no	yes enable TDD () to send and receive text messages.
keepcontext*	yes/no	yes	yes force to keep the VoiceXML channel during the consultation transfer.
video	yes/no	no	refer to Lience options
speech	yes/no/automatic/emulation	no	refer to Lience options
video	yes/no	no	refer to Lience options
dialer			refer to Lience options
rtmp			refer to Lience options
voicexml			refer to Lience options
messaging*	yes/no/database	no	yes allows to control the messaging counters as local context value. database use the Asterisk internal database to store the counters values.
priority*	value priority	0	if set us the priority at the end od the vxml() execution.

Key	Values	Default	Purpose
removeprompts*	yes/no	yes	yes allows remove the prompts after their play or processing. no is for debugging.
autoreload	yes/no	no	yes Can for to automatically reload the configuration if the configuration file date has changed.
recordsendaudio	yes/no	no	yes force the VoiceXML module to send audio silence during the recording. Some operators can close the call is there is no audio traffic during the recording.
recordsendvideo	yes/no	no	yes forec the VoiceXML module to send back to the user its own video during the video recording (as an echo test).
recordsilence	yes/no/value miliseconds	yes(300)	yes allows to record this time before the VAD detection. no force to record all the audio without waiting for the voice detection. You can set a value in milisecondes.
recordtotext*	url	empty	If record type="text/plain", use this URL to convert speech to text (as Google)
recorddirectory	directory	/tmp/cacheContent	Default temporary directory for recording
recordrewind	yes/no	yes	Strip off the last 1/4 second of the audio records.
threshold	dsp/default/number	default	Set the default threshold for silence detection.
videoupdate	yes/no	yes	Request a video update before recording.
audiosilence*	no/file	no	Play an audio file during silences.
cachedirectory	directory	/tmp/cacheContent	Cache directory for audio memory, MRCPsynth, record, builtin grammars…

Key	Values	Default	Purpose
ttsdirectory	directory	/tmp/cacheContent/	TTS cache directory, used to purge old files (greater the maxage parameter).
videosilence*	no/file	no	Play an video file during silences.
wavcodec	pcm/gsm		Codec use for record
videocodec*	h263/h263p		Deprecated
dialformat	string	%s	Format string used to generate outgoing calls (allows to set the channel type and peer for example)
dialformatvideo	string	%s	Format string used to generate outgoing video calls (allows to set the channel type and peer for example)
dialnumbersonly	yes/no	no	Filter to suppress non numeric charaters
nomaskerade	yes/no	no	For video calls add /nj for video calls
videoprofile	string	empty	Set the variable VXML_VIDEO_CODEC used by the defaukt transcoder dialplan
blindapplication	transfer/	none	To generate a SIP REFER, set a <transfer bridge="false" .../> and this parameter to "transfer"
localformat	spawn/bridge/nooptimization/jitter	spawn	Set the mode to transfer to local the dialplan
localextension	string	vxml	Set the default extension for local transfers
conferenceformat	string	empty	Specific parameter for the conference application.
celudanurl*			deprecated
speechprovider	unimrcp/verbio/lumenvox/string	empty	Set the Speech module (ASR) provider.
speechdirectory	directory	/tmp/cacheContent	Directory to generate temporary grammars (buitins for example).
speechscore	1-100	20	Minimal score to forward the speech result o the interpreter.
speechfactor	-x/+x	0	Parameter to multiply or divise the result from the speech engine.
speechforcescore*	0-100	0	Force a speech score (debug only)
speechforceresult*	string	empty	Force a speech value (debug only)

Key	Values	Default	Purpose
speechforcename*	yes/no	no	Add a 'V' for the grammar name, only for old Vestec engines
speechhotwordscore	0-100	0	If score is lower than this value, the module restart the recognition
speechbargein	yes/no/forced/enhanced	no	Force the prompt bargein mode for the speech.
speechmaxresults	number	1	Max results returned when it use Asterisk speech module.
speechtimeout			
speechmaxtimeout			
speechconcatenate	yes/no	no	Concatenate multiple speech results for Verbio.
speechrecord	yes/no	no	Record Speech sessions (wav) and results (txt) (for debug)
speechrecorddirectory	directory	/tmp/cacheContent	Directory to generate temporary speech recordings.
speechunload	yes/no	yes	Disable the Speech unload (some ASR engine not support to set empty grammars contents to free the loaded grammars.
speechurl*	url	empty	not implemented
speechbeepstart*	file	empty	not implemented
speechbeepstop*	file	empty	not implemented
cli	yes/no	no	Allows to execute CLI commands form the VoiceXML syntax (audio src="cli:command")
urldecode	yes/no	no	Decode VoiceXML first URL.
cdrupdate	yes/no	no	Allows to modify all the CDR fields
cdrexit	yes/no	no	Use the VoiceXML exit value to set the CDR userfiled
cdrdial	yes/no	no	Generate a CDR with the transfered call.
cdroverwrite	yes/no	no	Overwrite the local/remote call numbers (with the values managed the VoiceXML VXML_LOCAL, VXML_REMOTE…)
cdrparam	yes/no	no	Use the VXML_PARAM to set the CDR userfield

Key	Values	Default	Purpose
cachetimeout	number	60	Execute the cache process each x seconds.
cachehour	number	-1	Execute the cache process at a definided hour.
cachemaxage	number	-1	Remove cache file older than maxage seconds.
cachemaxremoves	number	-1	Max removed files for each execution process.
loadlibrary*	library		Deprecated
loadcli	string	empty	Execute a specific CLI command when the module is loaded.
originatedelay	number	10000	Time to wait before dial in miliseconds.
originatedelay2	number	10000	Time to wait before dial the second time n miliseconds.
requesturl*	URL	empty	URL for the billing HTTP/API.
countermax*	number		Deprecated
countermin*	number		Deprecated
counterval*	number		Deprecated
audiowait	file	empty	Plays this file and waits for a free VoiceXML channel.
mrcpsynthparams	string	empty	Add extra parameters for the MRCPsynth application (volume, keep session...)

License Section [license]

The following section covers license information and keys of Vxi. To obtain a commercial valid license key for your copy of VoiceXML browser (for use more than one session), you should run asterisk and get your code by typing the following commands:

```
*CLI> vxml show license
```

Default License section (no key) allow to run any VoiceXML application with one free port.

Example:

```
[license]
max=1
key=
```

Example:

```
[license]
```

```
max=30
key=yourkey
```

Key	Values	Default	Purpose
max	number of channels	1	Max number of VoiceXML channels for this license.
texttospeech	yes/no	yes	Allows to use Text to Speech prompts.
video	yes/no	no	Allows to use Video options.
speech	yes/no	no	Allows to use ASR/Speech options.
dialer	yes/no	no	Allows to use the VoiceXML with external Dialer.
voicexml	yes/no	yes	A allows to use the VoiceXML browser execution.
expiration	date YYMMDD		Optional expiration date of the license
key	License key		Key after purchasing a official commercial license. If not set or wrong key, you have 1 trial VoiceXML channel.

Control Section [control]

The control section describes the control functions that allow you to configure DTMF commands that stream a file with fast forward, pause, reverse, restart, etc. To enable a control during a prompt, you must first set the VoiceXML property name to equal “control, with a value of “yes.”

```
<property name=”control” value=”yes” />
```

This works only with GSM and WAV files - in all other cases the property is ignored.

Key	Values	Default	Purpose
forward	dtmf	#	Fast-forward when this DTMF digit is received.
reverse	dtmf	empty	Rewind when this DTMF digit is received.
stop	dtmf	0123456789	Stop when this DTMF digit is received.
pause	dtmf	empty	Pause playback when this DTMF digit is received.
restart	dtmf	empty	Restart playback when this DTMF digit is received.
skipms	number	5000	This is the number of milliseconds to skip when rewinding or fast-forwarding.

Account Section [account_]

The Vxi VoiceXML browser is designed to manage hosted VoiceXML services and share different applications thru different Inbound / Outbound phone lines. This will allow you to manage customers’ accounts for VoiceXML hosted applications and control easily the capacity you need to assign to each one.

To create accounts, you need to add an [accountX] section (where “X” is a number 1-100) in the bottom part of the /etc/asterisk/vxml.conf file. Please find here three added accounts to manage different port capacity per application/URI:

Examples:

```
[account1]
name=voiceportal1
local=0099090
url=http://www.i6net.com/vxml/voiceportal1.vxml
max=25

[account2]
name=helloworld
local=0099090
url=http://www.i6net.com/vxml/helloword/index.vxml
max=10

[account3]
name=localservice
local=0099090
url=http://localhost/vxml/test.vxml
max=51
```

The account properties are defined within sections. Each account has a specific section. The sections are named and numbered from account1 to account99 (maximun 100 accounts). The example below shows the functions in each account section.

```
[account1]
name=example
number=*3
url=http://…/index.vxml
urlvideo=http://…/index.vxml
max=5
speech=no
dialformat=SIP/%s
dialformatvideo=SIP/%s
```

Key	Values	Default	Purpose
name			
number			
url			
urlvideo			
max			
dialformat			
dialformatvideo			
dialnumbersonly			
force	y/n/video/audio/ulaw/alaw/flash/ringing/vxml/h263/h263p/h264		y=Video / n=audio
videoprofile			
speech	y/n/emulation/automatic/debug		
speechprovider			
speechscore			
speechfactor			
speechforcescore			

Key	Values	Default	Purpose
speechforcename			
mark			
durationlimit			
maxtime			
param			
application			
app			
wait			
volumetx			
volumerx			
audiodenied			
audiowait			

LogsSection [logs] (deprecated)

Key	Values	Default	Purpose
enable			
statsfilename			
logsdirectory			
split			

Account Section [account_]

The Vxi VoiceXML browser is designed to manage hosted VoiceXML services and share different applications thru different Inbound / Outbound phone lines. This will allow you to manage customers' accounts for VoiceXML hosted applications and control easily the capacity you need to assign to each one.

To create accounts, you need to add an [accountX] section (where "X" is a number 1-100) in the bottom part of the /etc/asterisk/vxml.conf file. Please find here three added accounts to manage different port capacity per application/URI:

Examples:

```
[account1]
name=voiceportal1
local=0099090
url=http://www.i6net.com/vxml/voiceportal1.vxml
max=25

[account2]
name=helloworld
local=0099090
url=http://www.i6net.com/vxml/helloword/index.vxml
max=10
```

```
[account3]
name=localservice
local=0099090
url=http://localhost/vxml/test.vxml
max=51
```

The account properties are defined within sections. Each account has a specific section. The sections are named and numbered from account1 to account99 (maximun 100 accounts). The example below shows the functions in each account section.

```
[account1]
name=example
number=*3
url=http://…/index.vxml
urlvideo=http://…/index.vxml
max=5
speech=no
dialformat=SIP/%s
dialformatvideo=SIP/%s
```

** name= {string}**

This function indicates the name or designation of the account. This reference can be used to identify the account used by the VXML application, Vxml(name of the account). This allows you to execute a VoiceXML session corresponding to the account values.

number= {called number}

This function allows the identification of an account with the called number, signaling information from the ANI function, CALLERID(). You can use the “*”, the wildcard character, to specify a substring such as *03 or 014612* , that the number function will contain. You can start the number with “_” to use Asterisk Diaplan Patterns too (see : http://www.voip-info.org/wiki/view/Asterisk+Dialplan+Patterns).The function can also be used to identify the account, like the name, Vxml(number of the account). You can start the number with “@” to match the caller number to the account number (use the same syntax as the called number There is a way to)

url= {voicexml URL}

This function indicates the VoiceXML URL of the account.

urlvideo= {voicexml URL}

This function defines the Video VoiceXML URL of the account. The call is identified as a video call by the set codecs or the function CHANNEL(transfercapability)=VIDEO.

max= {0…120}

This indicates the maximum number of sessions allowed to this account. If there are not enough sessions then the VoiceXML application will generate an error.

dialformat= {application(]/%s[)}

This is similar to the general function, but for the account only. If not set, use the general value.

dialformatvideo= {application(]/%s[)}

This function has the same purpose as the dialformat, only for video sessions.

force= {video/audio}

If set to video, set the Transfercapability to VIDEO (and enable the h324m processing). If set to audio, the Vxml application execute directly the account URL and bypass the redirection execution (parameter with '@').

speech= {yes/no/emulation/automatic}

This speech function is as the general function, but for the account only. If not set, use the general value.

speechprovider= {lumenvox/verbio}

This speech function is as the general function, but for the account only. If not set, use the general value.

speechscore= {0..100}

This speech function is as the general function, but for the account only. If not set, use the general value.

speechforcedscore= {0..100}

This speech function is as the general function, but for the account only. If not set, use the general value.

mark= {string/@local/@remote/@id/@param}

Set a string mark in the VoiceXML browser traces. The session ID and this string will be added to the channel number column (3rd) in the traces (Example : ...|33|... → ...|33_1_user1|...). Four redirection exist :

```
@remote : caller number
@local : called number
@id : VoieXML id parameter value
@param : VoiceXML parameter value
```

durationlimit= {0...}

This indicates the maximum duration in seconds for the session using this account. The call will be hangup after this maximal duration.

Configuration example

Example:

```
; VoiceXML Configuration
;
[general]
wav_codec=gsm
videosilence=;silence
audiosilence=;silence
debug=4
video=yes

[license]
max=100
key=…

[account1]
name=Test1
url=http://host.i6net.com/vxml/links/vxml/index.php
max=1

[account2]
name=Test2
url=http://www.mdfactory.com/mpm/camp/index.vxml
max=3
dialformat=SIP/%s@voztele-out
```

To assign an extension to a VXML account just follow this example, where we are assigning the previous account to three extensions number in your /etc/asterisk/extension.conf asterisk:

```
[default]

exten => 981001001,1,Vxml(Test1)
exten => 981001002,1,Vxml(Test2)
exten => 981001003,1,Vxml(Test3)
```

NOTE: When you update your vxml.conf file, remember to refresh configuration making a command “vxml reload” in your CLI*> prompt. If you have added SIP, PRI or new extensions you must launch “sip reload”, “extensions reload”, “dialplan reload” or reload asterisk/openvxi processes. Use the command “vxml show accounts” to dump your accounts.

```
CLI*> extensions reload
CLI*> vxml reload
CLI*> vxml show accounts
```

Example:

Add extensions to the Asterisk dial plan /etc/asterisk/extensions.conf, if you want to set the URL in the dialplan:

```
exten => 888,1,Answer
exten => 888,n,Wait(3)
exten => 888,n,Vxml(file:///root/example.vxml)
exten => 888,n,Hangup
```

You can create and edit the file /root/example.vxml with the GNU text editor, VI, for example.

This example will work if you have text-to-speech configured. If not, use a pre-recorded wav or gsm file to replace the “Hello world!” text by an <audio> tag. For more information, see the format extensions supported by Asterisk.

```
<?xml version="1.0"?>
<vxml version = "2.0" xmlns="http://www.w3.org/2001/vxml">
 <form>
  <block><audio src="hello.wav"/></block>
 </form>
</vxml>
```

Save the file in the same directory as the VoiceXML script (relative reference in this example). Reload the extensions configuration with:

```
*CLI> extensions reload
```

Call the service by calling:

```
SIP:888@<your server address>
```

From:
https://wiki.voximal.com/ - **Voximal documentation**

Permanent link:
https://wiki.voximal.com/doku.php?id=vxi_installation_guide:configuration:start&rev=1433843817

Last update: **2017/07/28 22:29**