VoiceGenie 7.2.2

Speech Resource Management

System Reference Guide

About Genesys

Genesys Telecommunications Laboratories, Inc., a subsidiary of Alcatel-Lucent, is 100% focused on software for call centers. Genesys recognizes that better interactions drive better business and build company reputations. Customer service solutions from Genesys deliver on this promise for Global 2000 enterprises, government organizations, and telecommunications service providers across 80 countries, directing more than 100 million customer interactions every day. Sophisticated routing and reporting across voice, e-mail, and web channels ensure that customers are quickly connected to the best available resource—the first time. Genesys offers solutions for customer service, help desks, order desks, collections, outbound telesales and service, and workforce management. Visit *www.genesyslab.com* for more information.

Each product has its own documentation for online viewing at the Genesys Technical Support website or on the Documentation Library DVD, which is available from Genesys upon request. For more information, contact your sales representative.

Notice

Although reasonable effort is made to ensure that the information in this document is complete and accurate at the time of release, Genesys Telecommunications Laboratories, Inc., cannot assume responsibility for any existing errors. Changes and/or corrections to the information contained in this document may be incorporated in future versions.

Your Responsibility for Your System's Security

You are responsible for the security of your system. Product administration to prevent unauthorized use is your responsibility. Your system administrator should read all documents provided with this product to fully understand the features available that reduce your risk of incurring charges for unlicensed use of Genesys products.

Trademarks

Genesys, the Genesys logo, and T-Server are registered trademarks of Genesys Telecommunications Laboratories, Inc. All other trademarks and trade names referred to in this document are the property of other companies. The Crystal monospace font is used by permission of Software Renovation Corporation, *www.SoftwareRenovation.com.*

Technical Support from VARs

If you have purchased support from a value-added reseller (VAR), please contact the VAR for technical support.

Technical Support from Genesys

If you have purchased support directly from Genesys, please contact Genesys Technical Support at the following regional numbers:

Region	Telephone	E-Mail
North America and Latin America	+888-369-5555 or +506-674-6767	*support@genesyslab.com*
Europe, Middle East, and Africa	+44-(0)-127-645-7002	*support@genesyslab.co.uk*
Asia Pacific	+61-7-3368-6868	*support@genesyslab.com.au*
Japan	+81-3-6361-8950	*support@genesyslab.co.jp*

Prior to contacting technical support, please refer to the Genesys Technical Support Guide for complete contact information and procedures.

Ordering and Licensing Information

Complete information on ordering and licensing Genesys products can be found in the Genesys 7 Licensing Guide.

Released By

Genesys Telecommunications Laboratories, Inc. *www.genesyslab.com*

Document Version: 3-2009

Table of Contents

Chapter

1 Introduction

This guide serves as the system reference manual for the VoiceGenie 7 Speech Resource Management (SRM) product. It is intended to provide a complete reference for all aspects related to the configuration, Metrics and alarming of the SRM system – including both the SRM Client and the SRM Server.

The following table gives definitions of some acronyms that are used throughout this document:

Acronyms	Full Definitions
ASR	Automated Speech Recognition (Engines/Technologies)
CLC	Command-Line Console— A command-line interface that can be used to query information and issue commands
MRCP	Media Resource Control Protocol— Adopted by the VoiceGenie Media Platform to control ASR and TTS resources In VG7.2.1, both MRCP v1 and MRCP v2 are supported.
SRM	Speech Resource Management— A component integrated into the VoiceGenie Media Platform to provide Speech Recognition and Synthesis functionalities to the application developers
SMC	System Management Console— A web-based tool for administering clusters of VoiceGenie VoiceXML Platforms
OA&M	Operation, Administration and Management
TTS	Text-to-Speech (Engines/Technologies)

The following sections may contain references to terminology that has become obsolete since the last release, NeXusPoint 6.4.*x*. The following table shows the mapping between these terms:

Historical Terms	New Terms
PhoneWeb Software/ NeXusPoint 6.4.*x* Software	VoiceGenie 7 Software
Cluster Management Platform (CMP)	OA&M Framework
Voice Resource Manager (VRM)	Speech Resource Management (SRM)
VoiceGenie Management Console (VMC)	System Management Console (SMC)

Chapter

2 SRM Client Configuration

The SRM client has a few configuration parameters that define its behaviors. They can be modified via the CMP SMC component through Call Manager configuration.

The following table lists these configuration parameters:

Parameter Name	Description
vrm.client.dll	This configuration parameter defines the location of the SRM Client library to be used by the Media Platform.
vrm.client.grammar.path	This parameter specifies the location of the built-in grammars residing on the VoiceGenie platform.
vrm.client.tmp.path	This parameter specifies the location of the temporary directory used by the media platform. It must match the *PW_TMP* entry in the *voicexml.cfg* file used by the VoiceXML Interpreter.

Parameter Name	Description
vrm.client.vggrammarbase	This parameter specifies the base-URL for translation of grammars residing under the subdirectory *vrm.client.tmp.path.* For example, in a Linux platform, if: *vrm.client.tmp.path = /usr/local/phoneweb/tmp/* *vrm.client.vggrammarbase = /vggrammarbase/tmp* the file */usr/local/phoneweb/tmp/index.txt* would be translated by the SRM client into: *http://205.150.90.166/vggrammarbase/tmp/index.txt* where *205.150.90.166* is the IP address of the MP. These two options allow the temporary grammar generated by the media platform to be fetched by an offboard server. The web server defined in the "HTTP Access to Grammars" section of the *SRM User's Guide* is used provide the hotkey grammars, so this configuration item must work together with the configuration defined in the web server.
vrm.client.timeout	This parameter is the timeout value used by the SRM client to wait for a response from the MRCP server, for both the VoiceGenie SRM Server and the native MRCP servers. If a response to an MRCP request has is not received within this timeout period, then the request is deemed to have failed.
vrm.ping.frequency	This parameter defines, in milliseconds, the frequency in which the SRM client pings each of its servers. The MRCP *DESCRIBE* method is used as a ping message for each of the MRCP Servers provisioned.
vrm.ping.timeout	This parameter defines, in milliseconds, the timeout period for which we would be waiting for ping response from a MRCP server. If a ping response is not heard back from the server within this timeout, the SRM Client would consider the MRCP server to have become unavailable, and it would then disconnect from the server and periodically re-try connection to the MRCP server again.
vrm.client.max.noinput.timeout	This parameter sets the value, in milliseconds, for the noinput timeout header that is sent to an MRCP engine. This should be set this to a large value, as the VoiceGenie Media Platform handles the no input timer. Default value: *90* (seconds)
vrm.client.modules	This parameter lists the MRCP client protocol modules installed in the platform. In VG7.2.1, the value of this parameter can be any combination of MRCP v1 and MRCP v2.

Parameter Name	Description
vrm.client.MRCPV2.dll	In VG7.2.1, this parameter configures the MRCP v2 client library name and path.
vrm.client.MRCPV1.dll	In VG7.2.1, this parameter configures the MRCP v1 client library name and path
vrm.client.mrcpv2.prefix	In VG7.2.1, this parameter is used by the SIP stack. The specified prefix allows the SIP stack to choose a SIP port for the MRCP v2 client.
mrcpv2client.sip.transport.0	In VG7.2.1, this parameter, together with vrm.client.mrcpv2.prefix, specifies the SIP port used by the MRCP v2 client. Note that the "mrcpv2client" must be the prefix specified by vrm.client.mrcpv2.prefix.
vrm.client.mrcpv2.maxopensocket	In VG7.2.1, this parameter specifies the maximum-allowed sockets opened for MRCP sessions.
vrm.client.mrcpv2.earlynomatch	In VG7.2.1, setting the value of this parameter `TRUE` tells the MRCPv2 server not to wait for the end of speech before processing the collected speech to match active grammars.
vrm.client.ping.disable	In VG7.2.1, this parameter controls the MRCP v2 client sending OPTIONS as a ping message to an MRCP v2 server. Setting the value of this parameter to `TRUE` will disable ping to all MRCP v2 servers.
stack.transport.type	This parameter is the protocol type for the MRCP v1 stack. Possible value: `RTSP` Default: `RTSP`
stack.connection.type	This parameter is the type of the MRCP v1 stack handling. Possible value: `client` Default: `client`
stack.connection.timeout	This parameter is the connection timeout for the MRCP stack to establish a TCP connection to the server. Default: `10000`
stack.trace.debug	This parameter specifies whether to enable the stack debug message. Possible values: `TRUE, FALSE` Default: `TRUE`

Parameter Name	Description
stack.socket.onesend	This parameter indicates whether to send a complete TCP message in one send request. Possible values: *TRUE, FALSE* Default: *TRUE*
vrm.client.grammar.path	This parameter specifies the location of the built-in grammars residing on the VoiceGenie platform. Default (Linux/Solaris): */usr/local/phoneweb/grammar/* Default (Windows): *C:\VoiceGenie\mp\grammar*
vrm.client.tmp.path	This parameter specifies the location of the temporary directory used by the media platform. It must match the *PW_TMP* entry in the *voicexml.cfg* file used by the interpreter. Default (Linux/Solaris): */usr/local/phoneweb/tmp/* Default (Windows): *C:\VoiceGenie\mp\tmp*
vrm.client.universals.uri	This parameter gives the URI convention that the NextGen VXMLI uses to specifiy the universals grammars. The default value should be set to *vrm.client.universals.uri = builtin:grammar/universals.* Default: *builtin:grammar/universals*
vrm.client.logmetrics	This parameter enables collection of MRCP message timing data. Possible values: *FALSE, TRUE* Default: *true*

Chapter

3 SRM Client Speech Resource Configuration Parameters

The following table contains the full list of the parameters that may be set in the SRM client Speech Resource provisioning. Note that all parameters should have a `vrm.client` prefix:

Parameter Name	Description
SendVGParams	If this parameter is set to `true`, the SRM Client will send VG SRM server vendor-specific parameters in the MRCP messages it sends. This parameter should be set to `true` for all the MRCP Native engines that are hosted by VG SRM Server. Default value: `false`
DefineGrammarSerial	This parameter is used to send the Define Grammar Request serially—that is, the SRM client waits for the `DEFINE-GRAMMAR` reply from the server before sending the `DEFINE-GRAMMAR` request for the next grammar. This may not be as efficient, but some MRCP engines do not accept simultaneous grammar define requests. Default value: `false` Currently, it is required to be set to `true` for Telisma Telispeech MRCP ASR server and Loquendo MRCP ASR server.
TelispeechRecognitionBargein	This parameter is set to `true` to supports Telisma's recognition barge-in capability.

Parameter Name	Description
UniqueGramID	This parameter is used to send a unique Grammar ID for each grammar request, across different channels. Default value: *false* Currently, this parameter must be set to *true* for NUANCE MRCP ASR server.
NuanceTranslateGrammar	For support of legacy application, when this parameter is set to *true,* the Nuance 7 grammars are translated into Nuance 8 grammars. It is not recommended, however, for new applications to be developed by using Nuance 7 grammars. Because the nuance built-in grammars are in Nuance 7 format, this parameter should be set to *true* for a Nuance MRCP ASR server. Default value: *false*
NuanceTranslateBuilt-inTo	By setting this parameter to *static,* the Media platform will translate the URI of built-in#rule to a Nuance-desirable URI to support Nuance static grammar. Default value (Nuance Engine): *static*
InsertXmlNs	For a Nuance MRCP server, set the value of this parameter to *http://www.w3.org/2001/06/grammar.* Default: Not set
NotEscapeApos	Set this parameter to *true* for a Nuance MRCP server. Default value: *false*
SendSWMSParams	When this parameter is set, the SRM client will send the OSR 3.0 parameters by using the SWMS 3.0 [the MRCP server on top of OSR 3.0 engine] convention. It should be set to *true* when the VRM client is connected to the native MRCP OSR3.0 server directly. This is not required if you are connected to an OSR 3.0 engine via the VG SRM Server. Default value: *false*
NoDuplicatedGramURI	This parameter is used to work around the problem for some engines that cannot accept duplicated URIs in the same recognition session. Default value: *false* Currently, this parameter must be set to *true* for SSFT SWMS OSR server.

Parameter Name	Description
SkipRecognitionTimeout	For some engines, it does not accept the Recognition-Timeout header of MRCP. Setting this parameter to `true` would cause the SRM client not to send this header. Default value: `false` Currently, this parameter must be set to `true` for NUANCE MRCP ASR server.
SkipPromptDone	When this parameter is set to `true`, the SRM client will not send the *REGOGNITION-START-TIMER* MRCP method to the MRCP engines. Some engines do not handle this method well. Default value: `false`
SkipSetMRCPParams	When this parameter is set to `true`, SRM client will not send any non-vendor-specific MRCP parameters in the MRCP *SET-PARAM* method. This helps increase the efficiency in the overall system. Default value: `false`
NLSMLEncoding	When this parameter is defined, it is the encoding type in which the NLSML will be returned. This is used for ASR engines that provide its NLSML result in an encoding other than UTF-8, and it does not indicate its encoding type in the XML header. The default value for this parameter is that it is not defined.
SendGrammarContent	This parameter is used to send the *DEFINE-GRAMMAR* with Grammar content, instead of a URI, for inline grammars. For a TTY/TDD engine, this parameter must be set to `true`.
DisableHotWord	When you set this parameter, the platform will treat recognition-based barge-in as speech-based barge-in. This parameter should be set to `true` for every ASR server that does not support recognition-based barge-in.
IBMHotWord	This parameter must be set to `true` for the IBM ASR Speech Resource in order to do hot-word recognition with an IBM ASR engine.
HotKeyBasePath	This parameter is the HTTP fetchable location for hotkey grammars. The IP address is the VG platform IP address. The web server defined in the “HTTP Access to Grammars” section of the *SRM User’s Guide* is used to provide the hotkey grammars, so this configuration item must work together with the configuration defined in the web server.

Parameter Name	Description
HotKeyLocalPath	This parameter is the local path for the hotkey grammars on the VoiceGenie Media Platform. The SRM Client will translate this address, using the *HotKeyBasePath*, to the appropriate URIs to be sent to the ASR servers. For each ASR engine, its value should be *$VGROOT/grammar/<engine name>/hotkey.*
PassThruTTSPort	When this parameter is set to *true*, the Media Platform will filter out the received TTS RTP packets that are not sent from the source RTP port number when the TTS session is first established. Default value: *false* Currently, this parameter must be set to *true* for SSFT RealSpeak MRCP TTS server.
ConnectPerSetup	When this parameter is set to *true*, the SRM Client will create a new connection to the ASR or TTS server per MRCP session setup. Default value: *false* Currently, this parameter must be set to *true* for IBM ASR and TTS. It must be set to true for NSS MRCP v2 server in VG7.2.1.
SendLoggingTag	When this parameter is set to *true*, the SRM Client will set the *logging-tag* parameter in the first *SET-PARAMS* method to the unique Call ID of the call. Default value: *false* Currently, this parameter must be set to *true* for IBM ASR and TTS.
TTSInsertVoiceTag	When this parameter is set to *true*, the SRM Client will use the voice name defined in the *TTSENGINE* property to add the voice tag around the TTS, so as to tell the TTS engine what voice to use. For example, if the *<property name = "TTSENGINE" value = "NATUALVOICE_CRYSTAL"/>* property is set, the TTS will be surrounded with the *<voice name="CRYSTAL"> .... </voice>* SSML tags. Default value: *false* Currently, this parameter must be set to *true* for NaturalVoices TTS.

Parameter Name	Description
ReverseGrammarOrder	When this parameter is set to `true,` the SRM Client will reverse the order of the list of grammars that is sent in the *DEFINE-GRAMMAR* and the *RECOGNIZE* requests to the ASR server. Default value: *false* Currently, this parameter must be set to *true* for IBM ASR and TTS.
SpeechMarkerEncoding	If a TTS MRCP server does not have the content-encoding header to specify the Speech Marker name encoding in a SPEECH-MARKER event, the VG SRM client will use the value specified by this parameter as the encoding of the Speech Marker name. Default value: *UTF-8*
TransportProtocol	In VG7.2.1, this parameter specifies the MRCP protocol used by the Speech Resource. Its value can be MRCPv1 or MRCPv2 and is case-sensitive.

Chapter

4 SRM Server Configuration

4.1 TTS Client Provisioning

The top-level parameter is *TTS_CLIENT_LIST.* Using this parameter, the SRM server may specify a comma-delimited list of TTS clients to start. An example of a top-level parameter is as follows:

TTS_CLIENT_LIST = SPEECHIFY_TOM, SPEECHIFY_JILL

For each *<TTS_CLIENT_NAME>* in the list, the following parameters may be specified:

Parameter Name	Description
TTS_CLIENT.<TTS_CLIENT_NAME>. TYPE	This may be one of the following values: *a* – this entry is for an alias *n* – this entry is for a normal TTS client
TTS_CLIENT.<TTS_CLIENT_NAME>. NAME	This is the name of the TTS client. This is used to uniquely identify a type of client.
TTS_CLIENT.<TTS_CLIENT_NAME>. VIRTUAL_DIR	This is the "virtual directory" recognized by the SRM Server. If the suffix of a resource URI of an incoming MRCP message matches this parameter, then a client of this type is used. For example, for the following the resource URI of: *rtsp://10.0.0.12/speechify_synthesizer* *speechifiy_synthesizer* is the *VIRTUAL_DIR.* When used in conjunction with the SRM Client, this suffix must match the URI portion of the *Servers* parameter.
TTS_CLIENT.<TTS_CLIENT_NAME>. EXE	This is the full path name to the executable for the client.

Parameter Name	Description
TTS_CLIENT.<TTS_CLIENT_NAME>. PROFILE	This is the configuration file to be used by the TTS Client. For more information about how to configure each of the TTS clients, please refer to the sections below.
TTS_CLIENT.<TTS_CLIENT_NAME>. NUM_HOST	This is the number of TTS clients to start.
TTS_CLIENT.<TTS_CLIENT_NAME>. PARAMETER	This is the list of parameters to be appended to the command line when starting the TTS client. Typically, this parameter has value: `-i <imt-cmp-config-file> -o imt` where `<imt-cmp-config-file>` is a configuration file used to control the logging behaviour of the TTS client. Each TTS client has its own `imt-cmp-config-file.` In order to support TFCI make sure the `-r` parameter is not used for the above value.
TTS_CLIENT.<TTS_CLIENT_NAME>. ENV_VAR_FILE	This is an optional file, used to specify environment variables to be set before starting the client. Each line of the environment variable file has the format: `env_var = value`

4.2 ASR Client Provisioning

The top-level parameter is *ASR_CLIENT_LIST.* Using this parameter, the SRM server may specify a comma-separated list of ASR clients to start. An example of a top-level parameter is as follows:
ASR_CLIENT_LIST = SPEECHWORKS

For each *<ASR_CLIENT_NAME>* in the list, the following parameters may be specified:

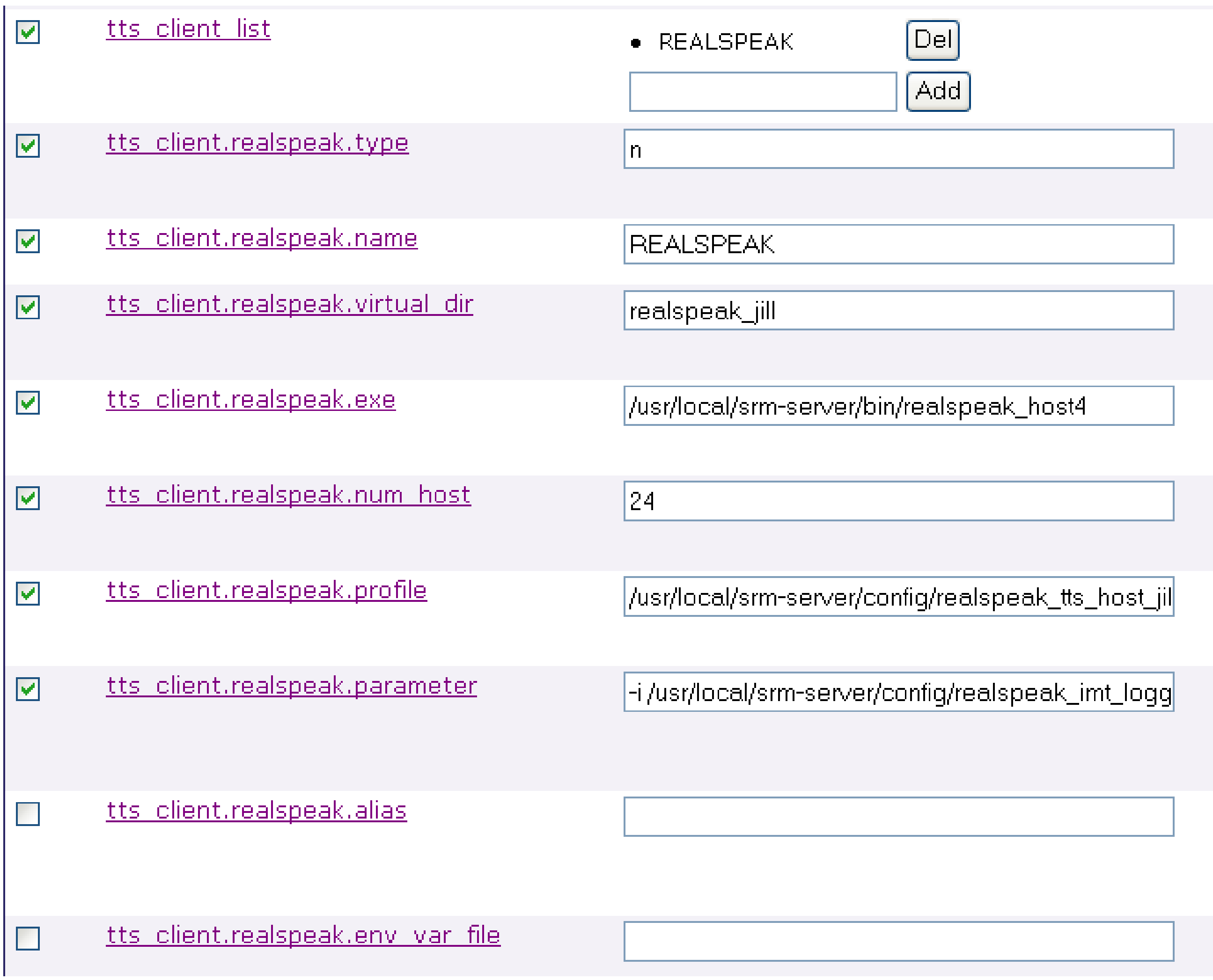
Parameter Name	Description
ASR_CLIENT.<ASR_CLIENT_NAME>. TYPE	This may be one of the following values: *a* – this entry is for an alias *n* – this entry is for an ASR client, returning results in the VoiceGenie proprietary format. Phonetics, BBN, and Watson clients are this type of client. *x* – this entry is for an ASR client which can return results in NLSML format. The client for OSR is this type of client.

Parameter Name	**Description**
ASR_CLIENT.<ASR_CLIENT_NAME>. NAME	This is the name of the ASR client. This is used to uniquely identify a type of client.
ASR_CLIENT.<ASR_CLIENT_NAME>. VIRTUAL_DIR	This is the "virtual directory" recognized by the SRM Server. If the suffix of a resource URI of an incoming MRCP message matches this parameter, then a client of this type is used. For example, for the following the resource URI of: `rtsp://10.0.0.12/spwx_recognizer` `spwx_recognizer` is the `VIRTUAL_DIR.` When used in conjunction with the SRM Client, this suffix must match the `vrm.client.resource.uri` portion in the SRM client Speech Resource provision.
ASR_CLIENT.<ASR_CLIENT_NAME>. EXE	This is the full path name to the executable for the client.
ASR_CLIENT.<ASR_CLIENT_NAME>. PROFILE	This is the configuration file to be used by the ASR Client. For more information about how to configure each of the ASR clients, please refer to sections below.
ASR_CLIENT.<ASR_CLIENT_NAME>. NUM_CLIENT	This is the number of ASR client processes to start.
ASR_CLIENT.<ASR_CLIENT_NAME>. NUM_THREADS	This is the number of threads each ASR client should start. For BBN, Phonetics, and OSR, we should start 1 client with *n* threads, where *n* is the number of ASR resources required. For Watson, we should start n clients with 1 thread each.
ASR_CLIENT.<ASR_CLIENT_NAME>. CONTEXT	This is a required parameter for each client, and it specifies a file, whish contains some default ASR parameters used for the initial recognition session when the ASR session is first requested.
ASR_CLIENT.<ASR_CLIENT_NAME>. PARAMETER	This is the list of parameters to be appended to the command line when starting the ASR client. This is optional, and varies for different ASR clients.
ASR_CLIENT.<ASR_CLIENT_NAME>. ENV_VAR_FILE	This is an optional file, used to specify environment variables to be set before starting the client. Each line of the environment variable file has the format: `env_var = value`

The following screen capture illustrates the GUI part that allows user to add a new TTS engine. The user can type a new TTS engine name in the blank box of the `tts_client_list` parameter. Clicking the *Add* button beside the blank box will create the set of parameters to define the new engine. The user must fill in the right parameter values in the respective boxes. Selecting the *Update* button at the bottom of the GUI is required to have the changes take effect.

To delete a TTS engine, the user can select the *Del* button beside the TTS engine in the *tts_client_list* section. Selecting the *Update* button at the bottom of the GUI is required to have the changes take effect.

It is similar to add and delete an ASR engine.



4.3 ASR/TTS Client Related Configuration

These parameters are used to control where the temporary files are created, and how long they will last in the file system:

Parameter Name	Description
TEMPORARY_PATH	This is the temporary path in the file system used to temporarily store some information by the ASR or TTS clients. For example, the save utterance files are stored in this *TEMPORARY_PATH.*
TEMPORARY_PATH_URI	This is the HTTP fetchable location for the *TEMPORARY_PATH.* The IP address of this URI is the IP address of the SRM Server. The web server defined in the "Application Server" subsection under the "SRM Server" Section of the *SRM User's Guide* is used make the files under *TEMPORARY_PATH* available. This configuration item must work together with the configuration defined in the web server.
SAVE_TMP_FILES	This parameter is used to indicate whether the SRM server should keep the temporary files after the ASR/TTS sessions have completed.

These parameters are used to control how quickly the TTS and ASR clients are restarted after the SRM server has detected they have unexpectedly disconnected (most likely due to an unintended termination of the ASR/TTS client or ASR/TTS server). The time between restarts increases by *RESPAWN_INTERVAL_BAASE* each time a client cannot be restarted successfully.

Parameter Name	Description
RESPAWN_INTERVAL_BASE	When an ASR or TTS client has restarted for the n^{th} consecutive time, it will be restarted in *RESPAWN_INTERVAL_BASE* * ($n - 1$) seconds, up to a maximum of *MAX_RESPAWN_TIMEOUT*
RESPAWN_CHECK_INTERVAL	After a TTS/ASR client has restarted, if it hasn't prematurely terminated after *RESPAWN_CHECK_INTERVAL* seconds, the TTS/ASR client is considered successfully restarted and the counter n where n is the number of times the client has restarted is reset.
MAX_RESPAWN_TIMEOUT	This is the maximum amount of time, in seconds, between which the clients are restarted. This is intended so the re-spawn interval does not grow without bound

These parameters are used for ASR clients:

Parameter Name	**Description**
ASR_MANAGER.CLIENT_PORT	This is the port number on which the ASR clients make their socket connections to the SRM Server. It is specified in the format: */dev/tcp/local/9600* where *9600* is the actual port number.
ASR_MANAGER.MAX_PENDING_ CONNECTIONS	This specifies the length of the listen socket queue for *ASR_MANAGER.CLIENT_PORT*
ASR_MANAGER.ONE_CLIENT_PER_ CHANNEL	This flag is used when a call session requires two different types of clients. When this flag is set to ***true***, the client of the previous type is released before allocating a new client.
ASR_MANAGER.SPAWN_CLIENT_ INTERVAL	This is the duration in milliseconds between starting two clients when the SRM server initially starts up. A delay is put here so that we don't overload the machine where the SRM server is.

These parameters are used for TTS clients:

Parameter Name	**Description**
TTS_MANAGER.CLIENT_PORT	This is the port number on which the TTS clients make their socket connections to the SRM Server. It is specified in the format: */dev/tcp/local/9004* where *9004* is the actual port number.
TTS_MANAGER.MAX_PENDING_ CONNECTIONS	This specifies the length of the listen socket queue for *TTS_MANAGER.CLIENT_PORT*
TTS_MANAGER.SPAWN_CLIENT_ INTERVAL	This is the duration in milliseconds between starting two clients when the SRM server initially starts up. A delay is put here so the machine where the SRM server is not overloaded.
TTS_MANAGER.MAX_STRING_LEN	This is the maximum length of a TTS request such that the SRM server would not use a file to forward the TTS request to the client. If the TTS request is less than this size, then the entire request will be sent within the socket message itself. Default value: *2048*

Parameter Name	Description
TTS_MANAGER.RESET_INTERVAL	If this value is set to *n,* each TTS client will terminate itself after having started for *n* milliseconds and if it is not currently doing any synthesis. When set to `0,` the TTS client will not terminate itself. Default value: *0*
TTS_MANAGER.SEARCH_INTERVAL	This is the number of milliseconds between checking each of the TTS clients to determine it is necessary to restart the client because it has been running for longer than *TTS_MANAGER.RESET_INTERVAL* milliseconds. When this is set to `0,` the engine will not restart a client because it has been running for too long. Note that a client will be restarted only if it is not taking any calls. Also note that at each search interval time a maximum of only one TTS client is restarted. This is done to avoid losing too many TTS resources and minimize CPU usage overhead. Default value: *0*

4.4 Other Configuration

The rest of the following parameters are for configuring the SRM Server in terms of how it accepts requests from the clients:

Parameters	Description
VRMServer.RM.RequestTimeOut	This is the time to wait in milliseconds before returning a timeout error back to the client that made the request to the SRM server. Default value: *5000*
VRMServer.RM.VRMProtocolModules	Semi-colon delimited list of names of .so files of protocol modules. For this version of SRM Server, this should be set to: */usr/local/srm/lib/MRCPProtocolModule.so* (Linux) *C:\VoiceGenie\srm-server\lib\mrcpprotocolmodule.dll* (Windows)

Parameters	**Description**
VRMServer.RM.VRMEngineModules	Semi-colon delimited list of names of .so files of engine modules. For example: */usr/local/srm-server/lib/VRMLegacyEngine.so;/usr/local/srm-server/lib/RtspTtsEngine.so;/usr/local/srm-server/lib/VRMTTYEngine.so* (Linux) *C:\VoiceGenie\srm-server\lib\vrmlegacyengine.dll;C:\VoiceGenie\srm-server\lib\rtsptts.dll;C:\VoiceGenie\srm-server\lib\TTYEngine.dll* (Windows)
VRMServer.RM.MRCP.Standard	This can be set to either *standard* or *extended.* This is used to indicate whether the SRM server should accept VG-specific vendor-specific parameters in the MRCP messages.
VRMServer.RM.OpenIdleTimeOut	Time in milliseconds to release the session if there is no subsequent request after SETUP message. Default Value: *900000*
VRMServer.DataSrvHdlr.StaleTimeout	When the VoiceGenie CMP component requests a status update from the SRM server, the SRM server returns the status from an internal cache of data kept specifically for sending to the CMP. If the data is older than the timeout specified in this parameter (kept in milliseconds), the SRM Server would not use the cache data, instead it would query the data from various components and return the queried data.
VRMServer.MRCPPM.VRMServerIP	The IP address of the protocol module
VRMServer.MRCPPM.VRMServerPort	Port number that the protocol module can use for incoming requests. When used in conjunction with the SRM Client, this suffix must match the port portion of the *Servers* parameter.
VRMServer.MRCPPM.VRMRestrict Connections	This could be set to *true* or *false.* If *false,* the protocol module will accept connections from any computer. If *true,* the protocol module will only accept connections from the list of valid IP addresses

Parameters	Description
VRMServer.MRCPPM.VRMValidIps	A semi-colon delimited list of IP addresses that we are permitted to accept connections from. This value is only used when *VRMRestrictConnections* is set to *true.*
VRMServer.MRCPPM.FailedSession Timeout	The length of time (in ms) after a session has been flagged as “failed” to wait before removing the failed session from the automatic error response list.
NotPingClient	This parameter indicates if the MRCP client connected to the SRM server has implemented the *DESCRIBE* as ping mechanism. If not, this flag should be set to *true.* By default, the flag is *false.*
MaxClientIdleTimeOut	Time in seconds to close a socket to a client if there is no client ping message from the client to the server. If not set default is *300* seconds The number will not have effect if *VRMServer.RM.NotPingClient* is set to *true*
MaxSessionIdleTimeOut	Time in seconds to close a session if there is no session activity. If not set ,the default value is *3600* seconds
PingCheckFrequency	Time in seconds to check the session and client ping message. If not set the default is *60* seconds

Chapter

5 ASR/TTS Engines Specific Configuration

5.1 Phonetics Configuration

The configuration files for Phonetics are all located in `/usr/local/srm-server/config/`. There are 3 files needed for the configuration: `profile.phonetic`, `phonetics.context`, and `phoneticclnt_imt_logger.cfg`. The `phonetics.context` file is more for internal use. We don't recommend user making any changes on it. Note that these files cannot be modified via the OA&M Framework, but rather must be modified on the VoiceGenie machines directly.

The following are the configuration parameters for `profile.phonetic:`

Parameter	Description
SERVER_NAME	Phonetic Gateway server name (or IP Address)
SERVER_PORT	Phonetic Gateway server port number. This is the parameter `pdsgateway` in the service file on the Phonetics Gateway
SECOND_SERVER_NAME	Backup GW name (IP Address)
SECOND_SERVER_PORT	Backup GW Port
USER_NAME	This is the user name we send to the Phonetics Gateway as the user of the Phonetics Gateway. We typically set this to `PW.`
BUILD_NUMBER	This is the Phonetics Operator Build number. It must match the number of the Phonetics server being connected to.
SERVER_TYPE	This is the Phonetics server type we're connecting to. It should always be set to `PSRT_GWCLIENT`

Parameter	Description
RECEIVE_TIMEOUT	This is the timeout we have for waiting for a reply from the Phonetics server. If the server doesn't reply within this time, we would consider the request to have failed.
CONNECT_TIMEOUT	This is the socket timeout we use to connect to Phonetics Gateway.
EXIT_VSE_DISCONNECT	Process should restart once it detects VSE disconnected.
MAX_CANDIDATE	Maximum number of candidates to be returned.
EXIT_VSE_DISCONNECT	A flag indicates if Phonetic Client should exit after it detect a VSE socket disconnection. If not set the default value is *true*

The following are the parameters used to configure the *phoneticclnt_imt_logger.cfg:*

Parameter	Description
mtinternal.jitter_log	Defines the logging period in terms of number of received packets. If less than *1,* Jitter logging is turned off. Jitter logging will be disabled if variable frame size codec is used for received packets.
mtinternal.max_concurrent_savedata	If specified as an integer n, and *mtinternal.transmit_savedata* or *mtinternal.receive_savedata* is enabled, then only a maximum of n concurrent files will be open for writing data. Default value is *-1,* which would place no limit.
mtinternal.max_sessions	Defines the maximum *MTInternal* sessions Default is *400*
mtinternal.receive_max_size	Defines the maximum packet sample size that will be notified to the receiver. Note that this number is applied to all codecs with fixed frame size. It will be rounded down to the nearest multiple of the codec frame size. This parameter will be disabled when variable frame size codec is used. Set to *-1* to disable the limit.
mtinternal.receive_min_size	Defines the minimum packet sample size that will be notified to the receiver. Note that this number is applied to all codecs with fixed frame size. It will be rounded down to the nearest multiple of the codec frame size. This parameter will be disabled when variable frame size codec is used. Set to *-1* to disable the limit.

Parameter	Description
mtinternal.receive_rate_alarm	If greater than *0,* minor alarm is generated if the transmission rate of incoming packets is slower the real time by the specified delay in milliseconds. This alarm will be disabled if variable frame size codec is used for received packets.
mtinternal.receive_savedata	If specified, received data is saved under the specified directory.
mtinternal.rtp_max_port	The maximum port range for RTP sockets in *MTInternal*
mtinternal.rtp_min_port	The minimum port range for RTP sockets in *MTInternal*
mtinternal.transmit_interval	Defines a constant transmission interval in milliseconds. If set to *0,* packets will be sent as soon as data arrives.
mtinternal.transmit_max_size	Defines the maximum data size in bytes that can be sent. Note that this number is applied to all codecs with fixed frame size. It will be rounded down to the nearest multiple of the codec frame size. This parameter will be disabled when variable frame size codec is used. Set to *-1* to disable the limit.
mtinternal.transmit_min_size	Defines the minimum data size in bytes that can be sent. Note that this number is applied to all codecs with fixed frame size. It will be rounded down to the nearest multiple of the codec frame size. This parameter will be disabled when variable frame size codec is used. Set to *-1* to disable the limit.
mtinternal.transmit_rate	When *mtinternal.transmit_interval* is non-zero, this parameter specifies the maximum number of packets to be sent for each transmission interval. Set to *0* to turn off this restriction.
mtinternal.transmit_rate_alarm	If greater than *0,* minor alarm is generated if the transmission rate of outgoing packets is slower the real time by the specified delay in milliseconds. This alarm will be disabled if variable frame size codec is used for transmitted packets.
mtinternal.transmit_savedata	If specified, utterance is saved under the specified directory.
cmp.log_file	The name of the log file for this process. Use full path. default is: */usr/local/srm-server/logs/log.phoneticclnt*
cmp.pid_option	This parameter will indicate if you wish to create a new log file every time the process restarts Default value is *FALSE*

Parameter	Description
cmp.size_option	This parameter determines whether the log files are rotated by size or time. Set to *TRUE* to roll over by size, *FALSE* to roll over by time. Default value is *TRUE*
cmp.rollover_size	If *cmp.size_option* is set to *TRUE* this arameter should be specified. The size limit (in megabytes) for roll over. Default value is *10*
cmp.num_rollover_files	If *cmp.size_option* is set to *TRUE* this arameter should be specified. The number of files to roll through, i.e they are overwritten Default value is *5*
cmp.rollover_mins	If *cmp.size_option* is set to *FALSE* this parameter determines how often the files are rolled over. The value is the time interval (in minutes) to roll over Default value is *1440*
cmp.rollover_time	If *cmp.size_option* is set to *FALSE* and *cmp.rollover_mins* is not specified, then this parameter determines when the files are rolled over. The time of day (using a 24 hour clock, values from *0:00* to *23:59*) to roll over Default value is *4:00*
cmp.log_sinks	The sinks that will be used by this component. Possible sinks are: *FILE, SYSLOG, EMAIL* For this component, we should use *FILE*
cmp.email	This parameter is only used if the *EMAIL* sink is specified By default this parameter is commented out. The value should be people's e-mail address
cmp.trace_flag	This flag determines if logs at level *log_5* (tracing/debugging) should be logged It overrides the setting in *cmp.log_5* Default value is *FALSE*
cmp.metrics	The log mask for metrics data, default is to log metrics to a metrics sink and send some upstream Default value is *0*
cmp.log_0	The log mask for data logged at log level *0.* Please consult VoiceGenie before change the default value.

Parameter	Description
cmp.log_1	The log mask for data logged at log level `1.` The log mask for data logged at log level `0.` Please consult VoiceGenie before change the default value.
cmp.log_2	The log mask for data logged at log level `2.` The log mask for data logged at log level `0.` Please consult VoiceGenie before change the default value.
cmp.log_3	The log mask for data logged at log level `3.` The log mask for data logged at log level `0.` Please consult VoiceGenie before change the default value.
cmp.log_4	The log mask for data logged at log level `4.` The log mask for data logged at log level `0.` Please consult VoiceGenie before change the default value.
cmp.log_5	The log mask for data logged at log level `5.` The log mask for data logged at log level `0.` Please consult VoiceGenie before change the default value.

5.2 RealSpeak Configuration

A different speechify client configuration file is required to be used for each of the RealSpeak voices. Each of the voices would require their own entry in the `TTS_CLIENT_LIST`, as described in 4.1 TTS Client Provisioning.

Notes: A Linux RealSpeak 4.0.4 server, ttsserver, will terminate if a wrong formated User Dictionary is loaded.

To generate a RealSpeak 4.0.4 user dictionary supported by Linux, please follow the following steps:

Read Appendix C in the *Telecom RealSpeak/Host Programmer's Guide* coming with the RealSpeak 4.0.4 installation to understand the steps to create and use user dictionaries.

The tool that generates RealSpeak User Dictionary, `rsude.exe`, is only installed in a Windows version of RealSpeak. This tool provides online help documentation for detailed instructions.

After the user dictionary is generated in `rsude.exe`, please Save As Binary LSB UCS-4 Format (`*.bdc`) to avoid this problem.

For configuration files of all the RealSpeech voices are located in `/usr/local/srm-server/config/`. There are 2 files needed for the configuration: `realspeak_tts_host_<voice_name>.cfg`, and `realspeak_imt_logger.cfg`. The same `realspeak_imt_logger.cfg` file can be shared across all the different voices. Note that these files cannot be modified via the OA&M Framework, but rather they must be modified on the VoiceGenie machines directly.

The following are the configuration parameters for *realspeak_tts_host_<voice_name>.cfg:*

Parameter	Description
IP_ADDRESS	The machine name or IP address of the machine with the TTS Engine for the TTS client to connect to.
PORT	The port number of the TTS Engine instance with the required voice installed. Note this port must match with the port number specified by the RealSpeak server configuration *file-swittsclient.cfg.*
AUDIOFORMAT	Audio format. Valid values are *alaw* or *ulaw*
ROOT_DICT	Absolute path of root dictionary
MAIN_DICT	Absolute path of main dictionary
ABBREVIATION_DICT	Absolute path of the abbreviation dictionary
CONTENT_TYPE	This is the content type to be sent to the Speechify engine in the *SWIttsSpeak()* function call. This value defaults to *application/synthesis+ssml.* This parameter should not be changed except when using Japanese, in which case it should be set to one of the following: *CONTENT_TYPE=charset=UTF-8* *CONTENT_TYPE=charset=Shift-JIS* *CONTENT_TYPE=charset=EUC*

The following are the parameters used to configure the *realspeak_imt_logger.cfg:*

Parameter	Description
mtinternal.jitter_log	Defines the logging period in terms of number of received packets. If less than *1,* Jitter logging is turned off. Jitter logging will be disabled if variable frame size codec is used for received packets.
mtinternal.max_concurrent_savedata	If specified as an integer *n,* and *mtinternal.transmit_savedata* or *mtinternal.receive_savedata* is enabled, then only a maximum of n concurrent files will be open for writing data. Default value is *-1,* which would place no limit.

Parameter	Description
mtinternal.max_sessions	Defines the maximum `MTInternal` sessions Default is `400`
mtinternal.receive_max_size	Defines the maximum packet sample size that will be notified to the receiver. Note that this number is applied to all codecs with fixed frame size. It will be rounded down to the nearest multiple of the codec frame size. This parameter will be disabled when variable frame size codec is used. Set to `-1` to disable the limit.
mtinternal.receive_min_size	Defines the minimum packet sample size that will be notified to the receiver. Note that this number is applied to all codecs with fixed frame size. It will be rounded down to the nearest multiple of the codec frame size. This parameter will be disabled when variable frame size codec is used. Set to `-1` to disable the limit.
mtinternal.receive_rate_alarm	If greater than `0,` minor alarm is generated if the transmission rate of incoming packets is slower the real time by the specified delay in milliseconds. This alarm will be disabled if variable frame size codec is used for received packets.
mtinternal.receive_savedata	If specified, received data is saved under the specified directory.
mtinternal.rtp_max_port	The maximum port range for RTP sockets in `MTInternal`
mtinternal.rtp_min_port	The minimum port range for RTP sockets in `MTInternal`
mtinternal.transmit_interval	Defines a constant transmission interval in milliseconds. If set to `0,` packets will be sent as soon as data arrives.
mtinternal.transmit_max_size	Defines the maximum data size in bytes that can be sent. Note that this number is applied to all codecs with fixed frame size. It will be rounded down to the nearest multiple of the codec frame size. This parameter will be disabled when variable frame size codec is used. Set to `-1` to disable the limit.
mtinternal.transmit_min_size	Defines the minimum data size in bytes that can be sent. Note that this number is applied to all codecs with fixed frame size. It will be rounded down to the nearest multiple of the codec frame size. This parameter will be disabled when variable frame size codec is used. Set to `-1` to disable the limit.
mtinternal.transmit_rate	When `mtinternal.transmit_interval` is non-zero, this parameter specifies the maximum number of packets to be sent for each transmission interval. Set to `0` to turn off this restriction.

Parameter	Description
mtinternal.transmit_rate_alarm	If greater than *0*, minor alarm is generated if the transmission rate of outgoing packets is slower the real time by the specified delay in milliseconds. This alarm will be disabled if variable frame size codec is used for transmitted packets.
mtinternal.transmit_savedata	If specified, utterance is saved under the specified directory.
cmp.log_file	The name of the log file for this process. Use full path. default is: */usr/local/srm-server/logs/log.realspeak_host4*
cmp.pid_option	This parameter will indicate if you wish to create a new log file every time the process restarts Default value is *FALSE*
cmp.size_option	This parameter determines whether the log files are rotated by size or time. Set to *TRUE* to roll over by size, *FALSE* to roll over by time. Default value is *TRUE*
cmp.rollover_size	If *cmp.size_option* is set to *TRUE* this arameter should be specified. The size limit (in megabytes) for roll over. Default value is *10*
cmp.num_rollover_files	If *cmp.size_option* is set to *TRUE* this arameter should be specified. The number of files to roll through, i.e they are overwritten Default value is *5*
cmp.rollover_mins	If *cmp.size_option* is set to *FALSE* this parameter determines how often the files are rolled over. The value is the time interval (in minutes) to roll over Default value is *1440*
cmp.rollover_time	If *cmp.size_option* is set to *FALSE* and *cmp.rollover_mins* is not specified, then this parameter determines when the files are rolled over. The time of day (using a 24 hour clock, values from *0:00* to *23:59*) to roll over Default value is *4:00*
cmp.log_sinks	The sinks that will be used by this component. Possible sinks are: *FILE, SYSLOG, EMAIL* For this component, we should use *FILE*

Parameter	Description
cmp.email	This parameter is only used if the *EMAIL* sink is specified By default this parameter is commented out. The value should be people's e-mail address
cmp.trace_flag	This flag determines if logs at level *log_5* (tracing/debugging) should be logged It overrides the setting in *cmp.log_5* Default value is *FALSE*
cmp.metrics	The log mask for metrics data, default is to log metrics to a metrics sink and send some upstream Default value is *0*
cmp.log_0	The log mask for data logged at log level *0.* Please consult VoiceGenie before change the default value.
cmp.log_1	The log mask for data logged at log level *1.* The log mask for data logged at log level *0.* Please consult VoiceGenie before change the default value.
cmp.log_2	The log mask for data logged at log level *2.* The log mask for data logged at log level *0.* Please consult VoiceGenie before change the default value.
cmp.log_3	The log mask for data logged at log level *3.* The log mask for data logged at log level *0.* Please consult VoiceGenie before change the default value.
cmp.log_4	The log mask for data logged at log level *4.* The log mask for data logged at log level *0.* Please consult VoiceGenie before change the default value.
cmp.log_5	The log mask for data logged at log level *5.* The log mask for data logged at log level *0.* Please consult VoiceGenie before change the default value.

5.3 Rhetorical Configuration

For each Rhetorical Voice, a different Rhetorical client configuration file need to used for each of the voices. Each of the voices would require their own entry in the *TTS_CLIENT_LIST,* as described in 4.1 TTS Client Provisioning.

For configuration files for all the Rhetorical voices are located in */usr/local/srm-server/config/.* There are 3 files needed for the configuration: *rvoice_tts_host_<voice_name>.cfg, rvoice.env* and *rvoice_imt_logger.cfg.* The same *rvoice_imt_logger.cfg* file can be shared across all the different voices. Note that these files cannot be modified via the CMP, but rather they need to be modified on the VoiceGenie machines directly.

The following are the configuration parameters for *rvoice_tts_host_<voice_name>.cfg.* Note that the configuration file should have no spaces before or after the equals (=) signs. Also, the configuration file should be in Unix ASCII format only (ie. Windows edited files with *CTRL+M*'s in them are invalid):

Parameter	Description
voice	This is the name of the required voice to be used which must be running on at least one of the servers listed in the *servers* configuration line.
servers	This is a semicolon delimited list of IP addresses that have rVoice Servers running on them. Each server is followed by a colon then the port number that the rVoice Server is listening on. For example: *servers=rvoiceserver1:1314;rvoiceserver2:1314; 172.0.0.1:1315*
timeout	This is the time in milliseconds that the rVoice Client will attempt to contact a server before giving up.
init_timeout	This is the time in milliseconds that the rVoice Client will attempt to establish valid contact with available servers on start-up before giving up.
logdir	This is the directory where the Rhetorical Client log files should be written
loglevel	This controls the amount of logs to be generated. Unless we're debugging problems with the Rhetorical Client, this should be set to *0000.*

The following are the parameters used to configure the *rvoice_imt_logger.cfg:*

Parameter	Description
mtinternal.jitter_log	Defines the logging period in terms of number of received packets. If less than *1,* Jitter logging is turned off. Jitter logging will be disabled if variable frame size codec is used for received packets.

Parameter	Description
mtinternal.max_concurrent_savedata	If specified as an integer *n,* and *mtinternal.transmit_savedata* or *mtinternal.receive_savedata* is enabled, then only a maximum of n concurrent files will be open for writing data. Default value is *-1,* which would place no limit.
mtinternal.max_sessions	Defines the maximum *MTInternal* sessions Default is *400*
mtinternal.receive_max_size	Defines the maximum packet sample size that will be notified to the receiver. Note that this number is applied to all codecs with fixed frame size. It will be rounded down to the nearest multiple of the codec frame size. This parameter will be disabled when variable frame size codec is used. Set to *-1* to disable the limit.
mtinternal.receive_min_size	Defines the minimum packet sample size that will be notified to the receiver. Note that this number is applied to all codecs with fixed frame size. It will be rounded down to the nearest multiple of the codec frame size. This parameter will be disabled when variable frame size codec is used. Set to *-1* to disable the limit.
mtinternal.receive_rate_alarm	If greater than *0,* minor alarm is generated if the transmission rate of incoming packets is slower the real time by the specified delay in milliseconds. This alarm will be disabled if variable frame size codec is used for received packets.
mtinternal.receive_savedata	If specified, received data is saved under the specified directory.
mtinternal.rtp_max_port	The maximum port range for RTP sockets in *MTInternal*
mtinternal.rtp_min_port	The minimum port range for RTP sockets in *MTInternal*
mtinternal.transmit_interval	Defines a constant transmission interval in milliseconds. If set to *0,* packets will be sent as soon as data arrives.
mtinternal.transmit_max_size	Defines the maximum data size in bytes that can be sent. Note that this number is applied to all codecs with fixed frame size. It will be rounded down to the nearest multiple of the codec frame size. This parameter will be disabled when variable frame size codec is used. Set to *-1* to disable the limit.
mtinternal.transmit_min_size	Defines the minimum data size in bytes that can be sent. Note that this number is applied to all codecs with fixed frame size. It will be rounded down to the nearest multiple of the codec frame size. This parameter will be disabled when variable frame size codec is used. Set to *-1* to disable the limit.

Parameter	Description
mtinternal.transmit_rate	When *mtinternal.transmit_interval* is non-zero, this parameter specifies the maximum number of packets to be sent for each transmission interval. Set to *0* to turn off this restriction.
mtinternal.transmit_rate_alarm	If greater than *0,* minor alarm is generated if the transmission rate of outgoing packets is slower the real time by the specified delay in milliseconds. This alarm will be disabled if variable frame size codec is used for transmitted packets.
mtinternal.transmit_savedata	If specified, utterance is saved under the specified directory.
cmp.log_file	The name of the log file for this process. Use full path. default is: */usr/local/srm-server/logs/log.rhetorical*
cmp.pid_option	This parameter will indicate if you wish to create a new log file every time the process restarts Default value is *FALSE*
cmp.size_option	This parameter determines whether the log files are rotated by size or time. Set to *TRUE* to roll over by size, *FALSE* to roll over by time. Default value is *TRUE*
cmp.rollover_size	If *cmp.size_option* is set to *TRUE* this arameter should be specified. The size limit (in megabytes) for roll over. Default value is *10*
cmp.num_rollover_files	If *cmp.size_option* is set to *TRUE* this arameter should be specified. The number of files to roll through, i.e they are overwritten Default value is *5*
cmp.rollover_mins	If *cmp.size_option* is set to *FALSE* this parameter determines how often the files are rolled over. The value is the time interval (in minutes) to roll over Default value is *1440*
cmp.rollover_time	If *cmp.size_option* is set to *FALSE* and *cmp.rollover_mins* is not specified, then this parameter determines when the files are rolled over. The time of day (using a 24 hour clock, values from *0:00* to *23:59*) to roll over Default value is *4:00*

Parameter	Description
cmp.log_sinks	The sinks that will be used by this component. Possible sinks are: `FILE, SYSLOG, EMAIL` For this component, we should use `FILE`
cmp.email	This parameter is only used if the `EMAIL` sink is specified By default this parameter is commented out. The value should be people's e-mail address
cmp.trace_flag	This flag determines if logs at level `log_5` (tracing/debugging) should be logged It overrides the setting in `cmp.log_5` Default value is `FALSE`
cmp.metrics	The log mask for metrics data, default is to log metrics to a metrics sink and send some upstream Default value is `0`
cmp.log_0	The log mask for data logged at log level `0.` Please consult VoiceGenie before change the default value.
cmp.log_1	The log mask for data logged at log level `1.` The log mask for data logged at log level `0.` Please consult VoiceGenie before change the default value.
cmp.log_2	The log mask for data logged at log level `2.` The log mask for data logged at log level `0.` Please consult VoiceGenie before change the default value.
cmp.log_3	The log mask for data logged at log level `3.` The log mask for data logged at log level `0.` Please consult VoiceGenie before change the default value.
cmp.log_4	The log mask for data logged at log level `4.` The log mask for data logged at log level `0.` Please consult VoiceGenie before change the default value.
cmp.log_5	The log mask for data logged at log level `5.` The log mask for data logged at log level `0.` Please consult VoiceGenie before change the default value.

5.4 OSR 3.0 Configuration

OSR 3.0 legacy integration is only supported in Linux.

The configuration files for the OSR 3.0 are all located in `/usr/local/srm-server/config/`. There are 4 files needed for the configuration: `speechworks2.cfg`, `osr.context`, `osr.env` and `spwx_imt_logger.cfg`. For more information about the `osr.context` and the `spwx_imt_logger.cfg` files, please refer to 4.2 ASR Client Provisioning. Note that these files cannot be modified via the OA&M Framework, but rather must be modified on the VoiceGenie machines directly.

The following are the parameters used to configure the `speechworks2.cfg`:

Parameter	**Description**
RunAsOSRClient	This parameter controls whether it runs as OSR client or all-in-one mode. If OSR is a client/server setup, this should be `TRUE` for the robustness reason. When this is set to `true`, the OSR client will call the `SWIrecRecognizerCreate()` and the `SWIrecRecognizerDestroy()` functions at the start and the end of each phone call
WebAddressOfTmpDir	This is used for off-board OSR server that can not access the grammars in `tmp` directory. If `RunAsOSRClient` equals to `TRUE`, this has to be set. This works in conjunction with the web server configured in the “Application Server” subsection under the “SRM Server” section of the *SRM User’s Guide*, to provide an offboard OSR server with grammars found in the SRM Server
IMTConfigFile	This specifies the location of config file of the CMP logger
MaxNumOfLineLogWaveform	`MaxNumOfLineLogWaveform` controls the maximum number of concurrent calls that can save waveform. On production systems, the platform operator would want to keep this value to a small number so that waveform logging won’t affect system performance too mauch. This needs to work with the voicexml property `swirec_suppress_waveform_logging` set to `0`, which can be set in `defaults.vxml` or application page. If `MaxNumOfLineLogWaveform` equals to `0`, there is no limit at all (i.e. all lines will log waveform data). Default value: `0`

Parameter	Description
SupportServerSelection	`SupportServerSelection` controls whether or not application can select server with client/server setup. To support this, `SWIsvcServerSelectionMode` should be set to `explicit` in the `User.xml` on the client side. Default value: `false`
NomatchOnInvalidSlotValue	`NomatchOnInvalidSlotValue` controls which event the platform will throw if there is no valid slot value pair. If it is set to `TRUE,` a `nomatch` event will be thrown. Otherwise, an `error.grammar.asr` event will be thrown. Default value: `false`
AudioBufferThreshold	This parameter is used to control the minimum number of bytes of audio data to deliver to the endpointer. Default value: `800`
ResetEpAcousticStatePerUtterance	`ResetEpAcousticStatePerUtterance` controls whether or not the platform will reset endpointer acoustic state per utterance. In most cases, it should be `false.` Default value: `false`
AllocateEPLicensePerCall	This controls whether or not to allocate endpointer license on a per-call basis. This facilitate better license sharing between servers, but it may lead to more delays in the endpointing of the first recognition session. Default value: `false`
LegacyResultFormat	This controls whether the OSR3 Client should return the result in the proprietary VG ASR result format. This should always be set to `false` Default value: `false`

Chapter

6 Metrics Entries

Label	Description

asr_audioready
Logged by: SRMS
Level: all
	Audio Channel Established
	This is logged when the ASR client is ready to receive audio.
	The format is:
	`asr_audioready <Client>|<CLIENT_ID>|<STATE>|<Dest>|<DEST>`
	`<Client>`
	It is a lable.
	`<CLIENT_ID>`
	It is an integer representing an ASR client
	`<STATE>`
	The state of the srmserver when this metric is logged.
	`<Dest>`
	It is a lable.
	`<DEST>`
	This is the IP address and port number of the RTP destination in format of `dst_rtp=IP address:port`
	Example:
	`asr_audioready Client 0 [LOADING] - Dest: dst_rtp=10.0.0.114:30000`

Label	Description
asr_bargein Logged by: SRMS Level: all	**Start Of Speech Is Detected** This is logged when a start-of-speech is detected by the ASR. **The format is:** *asr_bargein <Client>|<CLIENT_ID>|<STATE>|<Info>|<INFO>* *<Client>* It is a lable. *<CLIENT_ID>* It is an integer representing an ASR client *<STATE>* The state of the srmserver when this metric is logged. *<Status>* It is a lable. *<STATUS>* This is an integer representation whether the request was successful or not. *0* always represents successful. *<Info>* It is a lable. *<INFO>* This logs the recognition result, and failure reason representing extra info sent back to the Media Platform. **Example:** *asr_bargein Client 0 [RECOGNIZING] - Info: _bargein_*

Label	Description
asr_done Logged by: SRMS Level: all	**Recognition Completed** This is logged when a recognition has been completed. **The format is:** *asr_done <Client>|<CLIENT_ID>|<STATE>|<Result>|<RESULT>* *<Client>* It is a lable. *<CLIENT_ID>* It is an integer representing an ASR client *<STATE>* The state of the srmserver when this metric is logged. *<Status>* It is a lable. *<STATUS>* This is an integer representation whether the request was successful or not. *0* always represents successful. *<Result>* It is a lable. *<RESULT>* This logs the recognition result sent back to the Media Platform. **Example:** *asr_done Client 0 [RECOGNIZING] - Result: <?xml version='1.0'?><result><interpretation grammar="session:_gram1" confidence="93"><input mode="speech">hello</input><instance><X confidence="93">valueX</X><SWI_meaning>{X:valueX}</SWI_meaning></instance></interpretation></result>*

Label	Description
asr_fail Logged by: SRMS Level: all	**Recognition Failed** This is logged when a recognition cannot be completed, either due to a nomatch, or due to a failure in the recognizer. **The format is:** *asr_fail <Client>|<CLIENT_ID>|<STATE>|<Info>|<INFO>* *<Client>* It is a lable. *<CLIENT_ID>* It is an integer representing an ASR client *<STATE>* The state of the srmserver when this metric is logged. *<Status>* It is a lable. *<STATUS>* This is an integer representation whether the request was successful or not. *0* always represents successful. *<Reason>* It is a lable. *<REASON>* This logs the recognition failure reason representing extra info sent back to the Media Platform. **Example:** *asr_fail Client 0 [RECOGNIZING] - Reason: -914*

Label	Description
asrload_begin Logged by: SRMS Level: all	**Load Grammar Request Sent** This is logged when a grammar load request is sent to the ASR client. **The format is:** *asrload_begin <Client>|<CLIENT_ID>|<STATE>|<Context>|<CONTEXT_FILE>* *<Client>* It is a lable. *<CLIENT_ID>* It is an integer representing an ASR client *<STATE>* The state of the srmserver when this metric is logged. *<Context>* It is a lable. *<CONTEXT_FILE>* This is the path of temporary asr context file used to store grammar and recognition information. **Example:** *asrload_begin Client 0 [USED] - Context: /usr/local/srm-server/tmp/00020023-101D5B3D-asr/asrcontext.00020023-101D5B3D-asr.0*

Label	Description
asrload_end Logged by: SRMS Level: all	**Load Grammar Completed** This is logged when loading grammars has finished. **The format is:** *asrload_end <Client>|<CLIENT_ID>|<STATE>|<Status>|<STATUS>* *<Client>* It is a lable. *<CLIENT_ID>* It is an integer representing an ASR client *<STATE>* The state of the srmserver when this metric is logged. *<Status>* It is a lable. *<STATUS>* This is an integer representation whether the request was successful or not. *0* always represents successful. **Example:** *asrload_end Client 0 [LOADING] - Status: 0*
asrstart_begin Logged by: SRMS Level: all	**Recognition Request Sent** This is logged when the ASR starts to perform recognition. **The format is:** *asrstart_begin <Client>|<CLIENT_ID>|<STATE>|<Status>|<STATUS>* *<Client>* It is a lable. *<CLIENT_ID>* It is an integer representing an ASR client *<STATE>* The state of the srmserver when this metric is logged. **Example:** *asrstart_begin Client 0 [LOADING]*

Label	Description
asrstart_end Logged by: SRMS Level: all	**Engine Started Recognition** This is logged when the ASR has started. **The format is:** *asrstart_end <Client>|<CLIENT_ID>|<STATE>* *<Client>* It is a lable. *<CLIENT_ID>* It is an integer representing an ASR client *<STATE>* The state of the srmserver when this metric is logged. *<Status>* It is a lable. *<STATUS>* This is an integer representation whether the request was successful or not. *0* always represents successful. **Example:** *asrstart_end Client 0 [STARTING] - Status: 0*

Label	Description
asrstop_begin Logged by: SRMS Level: all	**Stop Recognition Request Sent** This is logged when the ASR is stopped by the Media Platform. **The format is:** *asrstop_begin <Client>|<CLIENT_ID>|<STATE>|<Status><STATUS>|<Reply>|<REPLY>* *<Client>* It is a lable. *<CLIENT_ID>* It is an integer representing an ASR client *<STATE>* The state of the srmserver when this metric is logged. *<Reply>* It is a lable. *<REPLY>* This indicates if a reply messge to the stop requst is reqired. *true:* reply required; *false:* no reply reqired. **Example:** *asrstop_begin Client 0 [RECOGNIZING] - Reply: true*

Label	Description
asrstop_end Logged by: SRMS Level: all	**Engine Started Recognition** This is logged when the ASR has started. **The format is:** *asrstop_end <Client>|<CLIENT_ID>|<STATE>* *<Client>* It is a lable. *<CLIENT_ID>* It is an integer representing an ASR client *<STATE>* The state of the srmserver when this metric is logged. *<Status>* It is a lable. *<STATUS>* This is an integer representation whether the request was successful or not. *0* always represents successful. **Example:** *asrstop_end Client 0 [STOPPING] - Status: 0*
asrunload_begin Logged by: SRMS Level: all	**Free ASR Resource Request Sent** This is logged when an ASR resource is being freed up. **The format is:** *asrunload_begin <Client>|<CLIENT_ID>|<STATE>* *<Client>* It is a lable. *<CLIENT_ID>* It is an integer representing an ASR client *<STATE>* The state of the srmserver when this metric is logged. **Example:** *asrunload_begin Client 0 [USED]*

Label	Description
asrunload_end Logged by: SRMS Level: all	**Free ASR Resource Completed** This is logged when an ASR resource has been freed up. **The format is:** *asrunload_end <Client>|<CLIENT_ID>|<STATE>|<Status>|<STATUS>* *<Client>* It is a lable. *<CLIENT_ID>* It is an integer representing an ASR client *<STATE>* The state of the srmserver when this metric is logged. *<Status>* It is a lable. *<STATUS>* This is an integer representation whether the request was successful or not. *0* always represents successful. **Example:** *asrunload_end Client 0 [UNLOAD] - Status: 0*

Label	Description
tts_done Logged by: SRMS Level: all	**A TTS Session Completed** This is logged when a TTS request is completed. **The format is:** *tts_done <Host>|<HOST_NAME>|<STATE>|<Status>|<STATUS>* *<Host>* It is a lable. *<HOST_NAME>* It represents a TTS client *<STATE>* The state of the srmserver when this metric is logged. *<Status>* It is a lable. *<STATUS>* This is an integer representation whether the request was successful or not. *0* always represents successful. **Example:** *tts_done Host H0009 [SPEAKING] - Status: 0*

Label	Description
tts_play Logged by: SRMS Level: all	**Start A TTS Request** This is logged when a TTS request is sent. **The format is:** *tts_play <Host>|<HOST_NAME>|<STATE>|<Status>|<STATUS>* *<Host>* It is a lable. *<HOST_NAME>* It represents a TTS client *<STATE>* The state of the srmserver when this metric is logged. *<Status>* It is a lable. *<STATUS>* This is an integer representation whether the request was successful or not. *0* always represents successful. **Example:** *tts_play Host H0009 [SPEAK] - Status: 0*

Label	Description
ttsstop_end Logged by: SRMS Level: all	**A TTS Session Is Stopped** This is logged when the TTS synthesis has been stopped. **The format is:** *ttsstop_end <Host>|<HOST_NAME>|<STATE>|<Status>|<STATUS>* *<Host>* It is a lable. *<HOST_NAME>* It represents a TTS client *<STATE>* The state of the srmserver when this metric is logged. *<Status>* It is a lable. *<STATUS>* It is an integer representation whether the request was successful or not. *0* always represents successful **Example:** *ttsstop_end Host H0009 [IDLE] - Status: 0*

Label	Description
ttsstop_engin Logged by: SRMS Level: all	**Sending Request To Stop A TTS Session** This is logged when the TTS synthesis is requested to be stopped. **The format is:** *ttsstop_engin <Host>|<HOST_NAME>|<STATE>|<Reason>|<REASON>* *<Host>* It is a lable. *<HOST_NAME>* It represents a TTS client *<STATE>* The state of the srmserver when this metric is logged. *<Reason>* It is a lable. It is optional *<REASON>* It is the reason why a stop request was issued. It is optional. If it appears, it is one of *reset*, *session closed*, *session disconnected* and *barge-in-occurred.* **Example:** *ttsstop_begin Host H0009 [SPEAKING]*

Chapter

7 SNMP Traps

Name	OID	Type	Description
Started	.1.3.6.1.4.1.7469.3.9.12.1.1.1	Scalar	The time the server was started
ASRDisplayName	.1.3.6.1.4.1.7469.3.9.13.1.1.x	Tabular	The name of the engine
ASRURI	.1.3.6.1.4.1.7469.3.9.13.1.2.x	Tabular	The URI used to access the engine
ASRAvailable	.1.3.6.1.4.1.7469.3.9.13.1.3.x	Tabular	The number of available clients
ASRTotal	.1.3.6.1.4.1.7469.3.9.13.1.4.x	Tabular	The total number of clients
ASRPeak	.1.3.6.1.4.1.7469.3.9.13.1.5.x	Tabular	The lowest number of clients that were available at any given
ASRDied	.1.3.6.1.4.1.7469.3.9.13.1.6.x	Tabular	The number of clients that died unexpectedly
ASRFailed	.1.3.6.1.4.1.7469.3.9.13.1.7.x	Tabular	The number of sessions that ended with a failure code
ASRSuccess	.1.3.6.1.4.1.7469.3.9.13.1.8.x	Tabular	The number of successfully completed sessions
TTSDisplayName	.1.3.6.1.4.1.7469.3.9.13.1.9.x	Tabular	The name of the engine
TTSURI	.1.3.6.1.4.1.7469.3.9.13.1.10.x	Tabular	The URI used to access the engine
TTSAvailable	.1.3.6.1.4.1.7469.3.9.13.1.11.x	Tabular	The number of available clients
TTSTotal	.1.3.6.1.4.1.7469.3.9.13.1.12.x	Tabular	The total number of clients
TTSPeak	.1.3.6.1.4.1.7469.3.9.13.1.13.x	Tabular	The lowest number of clients that were available at any given time
TTSDied	.1.3.6.1.4.1.7469.3.9.13.1.14.x	Tabular	The number of clients that died unexpectedly

Name	OID	Type	Description
TTSFailed	.1.3.6.1.4.1.7469.3.9.13.1.15.x	Tabular	The number of sessions that ended with a failure code
TTSSuccess	.1.3.6.1.4.1.7469.3.9.13.1.16.x	Tabular	The number of successfully completed sessions
PMName	.1.3.6.1.4.1.7469.3.9.13.1.17.x	Tabular	The name of the protocol module
PMOpenMin	.1.3.6.1.4.1.7469.3.9.13.1.18.x	Tabular	The minimum amount of time an `open` request took
PMCloseMin	.1.3.6.1.4.1.7469.3.9.13.1.19.x	Tabular	The minimum amount of time a `close` request took
PMStopMin	.1.3.6.1.4.1.7469.3.9.13.1.20.x	Tabular	The minimum amount of time a `stop` request took
PMSetParamsMin	.1.3.6.1.4.1.7469.3.9.13.1.21.x	Tabular	The minimum amount of time a `set params` request took
PMGetParamsMin	.1.3.6.1.4.1.7469.3.9.13.1.22.x	Tabular	The minimum amount of time a `get params` request took
PMLoadGrammarMin	.1.3.6.1.4.1.7469.3.9.13.1.23.x	Tabular	The minimum amount of time a `load grammar` request took
PMRecognizeMin	.1.3.6.1.4.1.7469.3.9.13.1.24.x	Tabular	The minimum amount of time a `recognize` request took
PMPromptDoneMin	.1.3.6.1.4.1.7469.3.9.13.1.25.x	Tabular	The minimum amount of time a `prompt done` request took
PMSpeakMin	.1.3.6.1.4.1.7469.3.9.13.1.26.x	Tabular	The minimum amount of time a `speak` request took
PMPauseMin	.1.3.6.1.4.1.7469.3.9.13.1.27.x	Tabular	The minimum amount of time a `pause` request took
PMResumeMin	.1.3.6.1.4.1.7469.3.9.13.1.28.x	Tabular	The minimum amount of time a `resume` request took
PMControlMin	.1.3.6.1.4.1.7469.3.9.13.1.29.x	Tabular	The minimum amount of time a `control` request took
PMBargeInMin	.1.3.6.1.4.1.7469.3.9.13.1.30.x	Tabular	The minimum amount of time a `barge in occurred` request took
PMOpenMax	.1.3.6.1.4.1.7469.3.9.13.1.31.x	Tabular	The maximum amount of time an `open` request took

Name	OID	Type	Description
PMCloseMax	.1.3.6.1.4.1.7469.3.9.13.1.32.x	Tabular	The maximum amount of time a `close` request took
PMStopMax	.1.3.6.1.4.1.7469.3.9.13.1.33.x	Tabular	The maximum amount of time a `stop` request took
PMSetParamsMax	.1.3.6.1.4.1.7469.3.9.13.1.34.x	Tabular	The maximum amount of time a `set params` request took
PMGetParamsMax	.1.3.6.1.4.1.7469.3.9.13.1.35.x	Tabular	The maximum amount of time a `get params` request took
PMLoadGrammarMax	.1.3.6.1.4.1.7469.3.9.13.1.36.x	Tabular	The maximum amount of time a `load grammar` request took
PMRecognizeMax	.1.3.6.1.4.1.7469.3.9.13.1.37.x	Tabular	The maximum amount of time a `recognize` request took
PMPromptDoneMax	.1.3.6.1.4.1.7469.3.9.13.1.38.x	Tabular	The maximum amount of time a `prompt done` request took
PMSpeakMax	.1.3.6.1.4.1.7469.3.9.13.1.39.x	Tabular	The maximum amount of time a `speak` request took
PMPauseMax	.1.3.6.1.4.1.7469.3.9.13.1.40.x	Tabular	The maximum amount of time a `pause` request took
PMResumeMax	.1.3.6.1.4.1.7469.3.9.13.1.41.x	Tabular	The maximum amount of time a `resume` request took
PMControlMax	.1.3.6.1.4.1.7469.3.9.13.1.42.x	Tabular	The maximum amount of time a `control` request took
PMBargeInMax	.1.3.6.1.4.1.7469.3.9.13.1.43.x	Tabular	The maximum amount of time a `barge in occurred` request took
PMOpenAvg	.1.3.6.1.4.1.7469.3.9.13.1.44.x	Tabular	The average amount of time an `open` request took
PMCloseAvg	.1.3.6.1.4.1.7469.3.9.13.1.45.x	Tabular	The average amount of time a `close` request took
PMStopAvg	.1.3.6.1.4.1.7469.3.9.13.1.46.x	Tabular	The average amount of time a `stop` request took
PMSetParamsAvg	.1.3.6.1.4.1.7469.3.9.13.1.47.x	Tabular	The average amount of time a `set params` request took
PMGetParamsAvg	.1.3.6.1.4.1.7469.3.9.13.1.48.x	Tabular	The average amount of time a `get params` request took

Name	OID	Type	Description
PMLoadGrammarAvg	.1.3.6.1.4.1.7469.3.9.13.1.49.x	Tabular	The average amount of time a `load grammar` request took
PMRecognizeAvg	.1.3.6.1.4.1.7469.3.9.13.1.50.x	Tabular	The average amount of time a `recognize` request took
PMPromptDoneAvg	.1.3.6.1.4.1.7469.3.9.13.1.51.x	Tabular	The average amount of time a `prompt done` request took
PMSpeakAvg	.1.3.6.1.4.1.7469.3.9.13.1.52.x	Tabular	The average amount of time a `speak` request took
PMPauseAvg	.1.3.6.1.4.1.7469.3.9.13.1.53.x	Tabular	The average amount of time a `pause` request took
PMResumeAvg	.1.3.6.1.4.1.7469.3.9.13.1.54.x	Tabular	The average amount of time a `resume` request took
PMControlAvg	.1.3.6.1.4.1.7469.3.9.13.1.55.x	Tabular	The average amount of time a `control` request took
PMBargeInAvg	.1.3.6.1.4.1.7469.3.9.13.1.56.x	Tabular	The average amount of time a `barge in occurred` request took

Chapter

8 Alarms

8.1 SRM Client

8.1.1 MRCP v2 Client

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
061003E9	CRIT	Invalid engine type specified	Any ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration; each engine must have a type, which must listed in *`vrm.client.enginetypelist`*
061003EA	CRIT	Invalid engine URI specified	Any ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration
061003EB	CRIT	Invalid engine entry in config file	Any ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration; platform must configure at least one server or backup server

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
061003EC	CRIT	Invalid IP or port for the engine	Any ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration and ASR/TTS server configuration
061003ED	CRIT	Engine list is empty	Any ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration
061003EE	CRIT	EROR when parsing engine info	Any ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration
061003EF	CRIT	Engine-type list is missing or empty	Any ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration
061003F0	CRIT	EROR creating stack	Any ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration
061003F1	CRIT	EROR initializing engine-type map	Any ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
061003F2	CRIT	EROR initializing stack	Any ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Software error	Report to VG
061003F3	CRIT	EROR initializing request manager	Any ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Software error	Report to VG
061003F4	CRIT	EROR initializing connection manager	Any ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Software error	Report to VG
061003F5	CRIT	EROR initializing stack handler	Any ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Software error	Report to VG
061003F6	CRIT	EROR reading vrmclient provision file	Any ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration in `cm_provision.dat`
061007D1	EROR	Failed to obtain information about file	Any ASR/TTS will fail for current call	Software error	Report to VG
061007D2	EROR	After stripping header, grammar-file size is less than 0	Any ASR will fail for current call	Software error	Report to VG

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
061007D3	EROR	Unable to find grammar file	Any ASR will fail for current call	Software error	Report to VG
061007D4	EROR	Unable to locate information in grammar file	Any ASR will fail for current call	Software error	Report to VG
061007D5	EROR	Failed to allocate memory	Any ASR/TTS will fail for current call	Server error	Check system resources
061007D6	EROR	Failed to read from grammar file	Any ASR will fail for current call	System error	Check server disk
061007D7	EROR	Failed to connect to server	Any ASR/TTS will fail until ASR/TTS server is up, no need to restart VG Platform	Configuration error	Check VRM Client configuration; check if server is up
061007D8	EROR	Failed to find info about server	Any ASR/TTS will fail until ASR/TTS server is up, no need to restart VG Platform	Configuration error	Check VRM Client configuration
061007D9	EROR	Invalid configuration for parameter	Any ASR/TTS will fail until configuration is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration
061007DA	EROR	Unable to read grammar base path	Any ASR/TTS will fail until configuration is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
061007DB	EROR	Unable to get information for all servers	Any ASR/TTS will fail until problem is fixed and VGPlatform is restarted	Software error	Report to VG
061007DC	EROR	Unable to look up connection information for specified vrm engine	Any ASR/TTS will fail until configuration is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration
061007DD	EROR	EROR when storing info about session	Any ASR/TTS will fail for current call	Software error	Report to VG
061007DE	EROR	EROR when changing state of session	Any ASR/TTS will fail for current call	Software error	Report to VG
061007DF	EROR	Received invalid timer event in client library	Any ASR/TTS will fail for current call	Software error	Report to VG
061007E0	EROR	EROR occurred while trying to remove info about session	Any ASR/TTS will fail for current call	Software error	Report to VG
061007E1	EROR	Client timed out while waiting for response	Any ASR/TTS will fail for current call	Server error; network error	Check server connections
061007E2	EROR	Received unknown message ID in message	Any ASR/TTS will fail for current call	Software error	Report to VG
061007E3	EROR	Received timeout for unknown event	Any ASR/TTS will fail for current call	Software error	Report to VG
061007E4	EROR	EROR looking up request type for session	Any ASR/TTS will fail for current call	Software error	Report to VG

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
061007E5	EROR	EROR removing timer event	Any ASR/TTS will fail for current call	Software error	Report to VG
061007E6	EROR	Received EROR code in response message from server	Any ASR/TTS will fail for current call	Server error; network error	Check server connections; check ASR/TTS Engine software, if it is running
061007E7	EROR	EROR when removing request from request manager	Any ASR/TTS will fail for current call	Software error	Report to VG
061007E8	EROR	Unable to find request in internal structures	Any ASR/TTS will fail for current call	Software error	Report to VG
061007E9	EROR	Received unexpected socket disconnect	Any ASR/TTS will fail for current call	Server error; network error	Check server connections; check ASR/TTS Engine software, if it is running
061007EA	EROR	Received invalid audio codec	Any ASR/TTS will fail for current call	Configuration error	Check VRM Client configuration
061007EB	EROR	Unable to send request	Any ASR/TTS will fail for current call	Server error; network error	Check server connections; check ASR/TTS Engine software, if it is running
061007EC	EROR	EROR occurred while processing message; most likely, memory-allocation EROR	Any ASR/TTS will fail for current call	Server error	Check system resources
061007ED	EROR	Method that should not be called was called	Any ASR/TTS will fail for current call	Software error	Report to VG
061007EF	EROR	Lost a connection to ASR/TTS server	Any ASR/TTS will fail for current call	Server error; network error	Check server connections; check ASR/TTS Engine software, if it is running

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
06100BB9	WARN	Received EROR message in recognize session	Any ASR/TTS will fail for current call	Server error; network error	Check server connections; check ASR/TTS Engine software, if it is running
06100BBA	WARN	Successfully reconnected to server	None		
06100BBB	WARN	Received speak complete before speak response	None		
06100BBC	WARN	NLSML format is not correct	Any ASR/TTS will fail for current call	Software error	Report to VG
06100BBD	WARN	Decoding EROR failed	Any ASR/TTS will fail for current call	Software error	Report to VG
06100BBE	WARN	Grammar file does not exist	Any ASR/TTS will fail for current call	Server error	Check system resources
06100BBF	WARN	EROR is encountered when reading grammar file	Any ASR/TTS will fail for current call	Server error	Check system resources
06101389	INFO	VRM Client sends TTS request	None		
0610138A	INFO	VRM Client receives SPEAK-COMPLETE	None		
0610138B	INFO	VRM Client uses default TTS/ASR engine	None		
0610138C	INFO	VRM Client service-quality analyst metrics	None		

8.1. 2 MRCP v2 Client

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
06A003E9	CRIT	Failed to load MRCP client modules	MRCP v1 or MRCP v2 ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration to ensure existence of dll specified by following three parameters in callmgr.cfg: *vrm.client.modules* *vrm.client.mrcpv1.dll* *vrm.client.mrcpv2.dll*
06B003E9	CRIT	Failed to load MRCP client module	MRCP v2 ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration to ensure existence of dll specified by *vrm.client.mrcpv2.dll*
061003EA	CRIT	Invalid engine type specified	MRCP v2 ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Configuration error	Check the Speech Resource provision data to ensure the right type is specified; the valid resource type is either primary or backup
061003EB	CRIT	EROR creating stack	Any ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
061003EE	CRIT	EROR initializing MRCP v2 Resource manager	Any MRCP v2 ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Software error	Report to VG
061003EF	CRIT	Failed to establish MRCP v2 session due to configuration issues	Any MRCP v2 ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration
06A007D1	EROR	Value of vrm.client.modules is not correctl	MRCP v1 or MRCP v2 ASR/TTS will fail until ASR/TTS is fixed and VGPlatform is restarted	Configuration error	Check VRM Client configuration
06A007D1	EROR	MRCP v1 and MRCP v2 Close Session Request failed to be sent	No impact on session	Software error	Report to Genesys
06A007D2	EROR	Stop request failed to be sent	Not able to stop speech server	Software error	Report to Genesys
06A007D3	EROR	Log failed to send to speech server	No log in to speech server	Software error	Report to Genesys
06A007D3	EROR	Setting of speech-recognition parameters on MRCP v1 or MRCP v2 server failed	Speech-recognition request will fail	Software error	Report to Genesys

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
06A007D4	EROR	Failed to send recognize request to MRCP v1 or MRCP v2 server	Speech-recognition request will fail	Software error	Report to Genesys
06A007D5	EROR	Failed to send start-input-timers request to MRCP v2 server or recognition-start-timers to MRCP v1	Possible impact on recognition performance	Software error	Report to Genesys
06A007D9	EROR	Failed to send Speak request to MRCP v1 or MRCP v2 server	TTS session will fail	Software error	Report to Genesys
06A007E1	EROR	Failed to send OpenSession request to MRCP v1 or MRCP v2 server	TTS or ASR session will fail	Software error	Report to Genesys
06A007E2	EROR	Speech Resource is not MRCP v1 or MRCP v2 server	TTS or ASR session will fail	Configuration error	Check VRM Client configuration; check “asrengine” and “ttsengine” property values in VoiceXML application
06A007E3	EROR	Failed to get Speech Resource provision handler	Adding or removing TTS or ASR resources will fail	Software error	Report to Genesys
06A007E3	EROR	Failed to send media-redirect request in MRCP v1 configuration	Fail for capability-based routine in MRCP v1 with MRCP proxy	Software error	Report to Genesys
06B007E4	EROR	EROR occurred while trying to remove info about MRCP v2 session	Any ASR/TTS will fail for current call	Software error	Report to Genesys

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
06B007E5	EROR	Unable to get MRCP v2 session information	Any MRCP v2 ASR/TTS will fail for current call	Software error	Report to Genesys
06B007E6	EROR	MRCP v2 session disconnect has been detected	MRCP v2 TTS/ASR session will fail	Server error	Check system resources
06B007E7	EROR	MRCP v2 server socket-system error has been detected	MRCP v2 TTS/ASR session will fail	Server error	Check system resources
06B007E8	EROR	Failed to access MRCPv2 session-type data	MRCP v2 TTS/ASR session will fail	Software error	Report to Genesys
06B007E9	EROR	Failed to find MRCP v2 resource for TTS/ASR request	ASR/TTS request will fail as no resource	Configuration error	Check VRM Client configuration
06B007EA	EROR	MRCP v2 session failed change state	MRCP v2 session will fail	Software error	Report to Genesys
06B007EB	EROR	Failed to add MRCP v2 session	MRCP v2 session will fail	Software error	Report to Genesys
06B007EC	EROR	Failed to initiate request manager	Any ASR/TTS will fail until problem is fixed and VGPlatform is restarted	Software error	Report to Genesys
06B007ED	EROR	Unable to initiate MRCP v2 Stack handler	Any ASR/TTS will fail until configuration is fixed and VGPlatform is restarted	Software error	Report to Genesys
06B007EE	EROR	Unable to read grammar base path	Any ASR/TTS will fail for current call	Configuration error	Check Speech Resource Provision parameter value of: *vrm.client.HotKeyBasePath*

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
06B007EF	EROR	MRCP v2 client failed to send MRCP request	Any ASR/TTS will fail for current call	Software error	Report to Genesys
06B007F1	EROR	MRCP v2 client RequestMgr failed to start its thread during initialization	Any ASR/TTS will fail until problem is fixed and VGPlatform is restarted	Software error	Report to Genesys
06B007F2	EROR	MRCP v2 client failed to send SIP message to MRCP v2 server	Any ASR/TTS will fail for current call	Configuration error; server error	Check configuration; report to Genesys
06B007F3	EROR	MRCP v2 client failed to handle responses from MRCP v2 server	Any ASR/TTS will fail for current call	Software error	Report to Genesys
06B007F4	EROR	Response from MRCP v2 indicated failure to corresponding MRCP request	Any ASR/TTS will fail for current call	Server error; network error; application error	Check server; check application; report to Genesys
06B007F6	EROR	Error removing timer event	Any ASR/TTS could fail for current call	Software error	Report to Genesys
06B007F7	EROR	Error removing outstanding connection data	May have memory leak	Software error	Report to Genesys
06B007F8	EROR	Received unknown message ID in MRCP message	Any ASR/TTS could fail for current call	Software error	Report to Genesys
06B007F9	EROR	Failed to remove outstanding define-grammar request from request manager	Any ASR/TTS could fail for current call	Software error	Report to Genesys

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
06B007FA	EROR	Connecting to MRCP v2 server failed	Any ASR/TTS could fail for current call	Network/Server error; software error	Check network; check server; report to Genesys
06B007FB	EROR	Failed to retrieve MRCP v2 resource information	Any ASR/TTS could fail for current call	Software error	Report to Genesys
06B007FC	EROR	Incorrect audio codec received from user	Any ASR/TTS could fail for current call	Software error	Report to Genesys
06B007FD	EROR	Failed to establish MRCP v2 session with server	Any ASR/TTS will fail for current call	Server error; network error; software error	Check server connections; check ASR/TTS engine software, if it is running; report to Genesys
06B007FE	EROR	Failed to remove session data	May have memory leak	Software error	Report to Genesys
06B007FF	EROR	Received invalid timer event in client library	Any ASR/TTS will fail for current call	Software error	Report to Genesys
06B00800	EROR	Received timeout for unknown event	Any ASR/TTS will fail for current call	Software error	Report to Genesys
06B00801	EROR	Client timed out while waiting for response	Any ASR/TTS will fail for current call	Server error; network error	Check server; check network
06B00802	EROR	Error encountered when handling hotkey grammar	Any ASR/TTS will fail for current call	Configuration error; software error	Check configuration; report to Genesys
06B00803	EROR	Failed to create SIP user agent for MRCP v2	Any ASR/TTS will fail for current call	Configuration error; system error; software error	Check configuration; check system memory; report to Genesys
06B00804	EROR	Failed to initialize SIP session	Any ASR/TTS will fail for current call	System error; software error	Check system memory; report to Genesys

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
06B00805	EROR	SIP error is received	Any ASR/TTS may fail for current call	Network error; server error; system error; software error	Check server; check network; check system memory; report to Genesys
06B00BB9	WARN	NLSML format is not correct	Any ASR/TTS will fail for current call	Software error	Report to VG
06B00BBA	WARN	Received speak complete before speak response	None		
06B00FA1	INFO	Indicates default TTS/ASR engines for MRCP v2	None		
06B00FA2	INFO	Cannot retrieve local host IP address	Inline grammar will not be loaded	Configuration error; system error	Check system-network configuration
06B00FA3	INFO	Log TTS speak request	None		
06B00FA4	INFO	Log TTS speak-complete response	None		
06B00FA5	INFO	Cannot retrieve local host IP address	Hotkey grammar will not be loaded	Configuration error; system error	Check system-network configuration

8.3 SRM Server

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
063003E8	CRIT	EROR reading configuration file	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	Configuration error	Check server configurations

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
063003E9	CRIT	EROR opening socket	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	Server error; network error	Check server connections
063003EA	CRIT	Unable to initialize OA&M	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	Software error	Report to VG
063003EB	CRIT	Unable to initialize license manager	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	License is missing or expired	Verify Software Configuration or License
063003EC	CRIT	Unable to load at least one protocol module	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	Configuration error	Check server configurations
063003ED	CRIT	Unable to load at least one engine module	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	Configuration error	Check server configurations
063007D0	CRIT	Engine had internal failure	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	Software error	Check ASR/TTS engine, if it is up and running
064003E8	CRIT	EROR reading configuration file	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	Configuration error	Check server configurations

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
064003E9	CRIT	EROR opening socket	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	Server error; network error	Check server connections
060007D1	EROR	New failed for memory allocation	MRCP Interface is not functioning correctly	Software error	Software and hardware restart
060007D2	EROR	Invalid configuration setup	MRCP Interface is not functioning correctly	Software error	Check server configurations
060007D3	EROR	Operation while uninitialized	MRCP Interface is not functioning correctly	Software error	Report to VG
060007D4	EROR	Unable to construct malformed message	MRCP Interface is not functioning correctly	Software error	Report to VG
060007D5	EROR	Unable to parse malformed message	MRCP Interface is not functioning correctly	Software error	Report to VG
060007D6	EROR	Received invalid request	MRCP Interface is not functioning correctly	Software error	Report to VG
060007D7	EROR	Socket error	MRCP Interface is not functioning correctly	Software error	Check ASR/TTS software; check network connections
06300BB8	EROR	Invalid config parameter	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	Configuration error	Check server configurations
06000BB9	WARN	Unexpected socket event	MRCP Interface is not functioning correctly	Software error	Check ASR/TTS software; check network connections.

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
06200BB8	WARN	Invalid config parameter	MRCP Interface is not functioning correctly	Configuration error	Check server configurations
06300BB9	WARN	Invalid audio codec in sdp	ASR/TTS will fail for current call	Configuration error	Check ASR/TTS engine configuration
06300BBA	WARN	Invalid media line in sdp	ASR/TTS will fail for current call	Configuration error	Check ASR/TTS engine configuration
06300BBB	WARN	Invalid origin line in sdp	ASR/TTS will fail for current call	Configuration error	Check ASR/TTS engine configuration
06300BBC	WARN	RML is stopping or pointer is invalid	ASR/TTS will fail for current call	Software error	Report to VG
06300BBD	WARN	Received invalid header in message	Any ASR/TTS will fail for current call	Software error	Report to VG
06300BBE	WARN	Computer with IP not in restricted IP list attempted to connect to server	Any ASR/TTS will fail from computer with restricted IP; after configuration change, VG Platform must be restarted	Configuration error	Check VRM Server configuration
06300BBF	WARN	Failed to create directory	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	System error	Check server disk
06300BC0	WARN	Requested grammar not found	Any ASR/TTS will fail for current call	Configuration error	Check VRM Server/ASR Server configuration

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
06300BC1	WARN	Received request in wrong state	Any ASR/TTS will fail for current call	Software error	Report to VG
06300BC2	WARN	Client has disconnected from engine	Any ASR/TTS will fail for calls to ASR/TTS engine disconnected until server is up; not necessary to restart VG Platform, as it will recover itself	Server error; network error	Check server connections; check ASR/TTS Engine software, if it is running
06300BC3	WARN	Failed to send message on socket	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	Server error; network error	Check server connections
06300BC4	WARN	Failed to create/write to file	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	System error	Check server disk
06300BC5	WARN	Received malformed result from legacy client	Any ASR/TTS will fail for current call	Software error	Report to VG
06300BC6	WARN	Received invalid session ID	Any ASR/TTS will fail for current call	Software error	Report to VG
06300BC7	WARN	Received invalid event in engine	Any ASR/TTS will fail for current call	Software error	Report to VG
06300BC8	WARN	Environment variable is not set; set it to an appropriate value	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	Configuration error	Check VRM Server configuration

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
06300BC9	WARN	Received malformed message from legacy client	Any ASR/TTS will fail for current call	Software error	Report to VG
06300BCA	WARN	Attempted to use invalid client ID	Any ASR/TTS will fail for current call	Software error	Report to VG
06300BCB	WARN	Received invalid/unknown message from client	Any ASR/TTS will fail for current call	Software error	Report to VG
06300BCC	WARN	Unable to create/open socket	Any ASR/TTS will fail for calls to ASR/TTS engine disconnected until server is up; not necessary to restart VG Platform, as it will recover itself	Server error; network error	Check server connections; check ASR/TTS Engine software, if it is running
06300BCD	WARN	Received incorrect initialization status from legacy client	Any ASR/TTS will fail from computer with restricted IP; after configuration change, VG Platform must be restarted	Server error; network error	Check server connections; check ASR/TTS Engine software, if it is running
06300BCE	WARN	Tried to free license, but no license was allocated to session	None	Software error	Report to VG

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
06300BCF	WARN	Engine attempted to update number of resources to negative number	None	Software error	Report to VG
06300BD0	WARN	Attempted to use URI that is not registered with server	Any ASR/TTS will fail for current call	Configuration error	Check VRM Client configuration
06300BD1	WARN	Failed to register data point with data service	Any ASR/TTS will fail for current call	Configuration error	Check VRM Server/CMP configuration
06300BD2	WARN	Open session idle timeout occurred	Any ASR/TTS will fail for current call	Server error; network error	Check server connections
06300FBB	NOTICE	Reconnected to client	None	Server error; network error	Check server connections
06400BB8	WARN	Invalid config parameter	MRCP Interface is not functioning correctly	Configuration error	Check server configurations
06400BBF	WARN	Failed to create directory	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	System error	Check server disk
06400BC0	WARN	Requested grammar not found	Any ASR/TTS will fail for current call	Configuration error	Check VRM Server/ASR Server configuration
06400BC1	WARN	Received request in wrong state	Any ASR/TTS will fail for current call	Software error	Report to VG

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
06400BC2	WARN	Client has disconnected from engine	Any ASR/TTS will fail for calls to ASR/TTS engine disconnected until server is up; not necessary to restart VG Platform, as it will recover itself	Server error; network error	Check server connections; check ASR/TTS Engine software, if it is running
06400BC3	WARN	Failed to send message on socket	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	Server error; network error	Check server connections
06400BC4	WARN	Failed to create/write to file	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	System error	Check server disk
06400BC5	WARN	Received malformed result from legacy client	Any ASR/TTS will fail for current call	Software error	Report to VG
06400BC6	WARN	Received invalid session ID	Any ASR/TTS will fail for current call	Software error	Report to VG
06400BC7	WARN	Received invalid event in engine	Any ASR/TTS will fail for current call	Software error	Report to VG
06400BC8	WARN	Environment variable is not set; set it to an appropriate value	Any ASR/TTS will fail until problem is fixed and VG Platform is restarted	Configuration error	Check VRM Server configuration

Alarm#	Level	Definition and Possible Message/Info	Impacts	Causes	Detailed Recommended Actions
06400BC9	WARN	Received malformed message from legacy client	Any ASR/TTS will fail for current call	Software error	Report to VG
06400BCA	WARN	Attempted to use invalid client ID	Any ASR/TTS will fail for current call	Software error	Report to VG
06400BCB	WARN	Received invalid/unknown message from client	Any ASR/TTS will fail for current call	Software error	Report to VG
06400BCC	WARN	Unable to create/open socket	Any ASR/TTS will fail for calls to ASR/TTS engine disconnected until server is up;. not necessary to restart VG Platform, as it will recover itself	Server error; network error	Check server connections; check ASR/TTS Engine software, if it is running
06400BCD	WARN	Received incorrect initialization status from legacy client	Any ASR/TTS will fail from computer with restricted IP; after configuration change, VG Platform must be restarted	Server error; network error	Check server connections; check ASR/TTS Engine software, if it is running

8.3 RTSP-TTS Engine Error Events

The following error events will be thrown by the application, in case of an error:

Description of Error	Error Returned by Stream Server	MRCP Response to Media Platform	Error Thrown to VXML application
URI cannot be found on RTSP server	DESCRIBE REPLY 404	SPEAK REPLY with 405	error.tts.noresource

Description of Error	Error Returned by Stream Server	MRCP Response to Media Platform	Error Thrown to VXML application
RTSP server has problem during stream setup or URI cannot be found on the server	SETUP REPLY 404	SPEAK REPLY with 405	error.tts.noresource
URI as specified in VXML page is malformed	URI of PLAY request is malformed (that is, IP address cannot be determined or is not valid rtsp URI)	SPEAK REPLY with 408	error.tts.badtext
Errors happen during RTSP streaming, and RTSP server reports error	PLAY REPLY 500 (Internal Server Error)	SPEAK REPLY with 407	error.tts
RTSP server disconnects from RTSP client	Socket Error happening while Playing	SPEAK-COMPLETE with completion cause 004	error.tts
RTSP client has not received audio data for configurable amount of time	Audio Data Receive timeout	SPEAK-COMPLETE with completion cause 004	error.tts
Cannot connect to RTSP server	Initial Socket Connection Error when PLAY request is received	SPEAK REPLY with 407	error.tts
Internal error within VoiceGenie software	Any MTInternal Error	SPEAK REPLY with 407 or SPEAK-COMLPETE with completion cause 004 if state is SPEAKING	error.tts

8.4 TTY Error Events

The following error events will be thrown by the application, in case of an error:

Description of Error	MRCP Response to Media Platform	Error Thrown to VXML Application
URI cannot be found on SRMServer	SETUP REPLY with 404	error.asr.noresource

Description of Error	MRCP Response to Media Platform	Error Thrown to VXML Application
Maximum number of sessions exceeded	SETUP REPLY with 407	error.asr.noresource
Grammar error has occurred	DEFINE-GRAMMAR REPLY with 407	error.grammar.asr
Internal errors happen during any step in *RECOGNITION*	MRCP Reply with 407 status code; RECOGNITION-COMPLETE with 006 completion clause	error.asr

Revision History

Version	Date	Change Summary	Author/Editor
1	March 11, 2005	Initial release	Alex Lee Johnson Tse Lin Chen Andrew Ho
1.1	March 29, 2005	Revised version	Andrew Ho
1.2	April 13, 2005	Final Revision for VoiceGenie 7 Release	Andrew Ho
1.3	December 13, 2005	Updated to include new SRM Client parameters	Rakesh Tailor
1.4	September 5, 2006	Updates for 7.1	Monti Ghai
1.6	September 21, 2007	Updates for 7.2	Lin Chen
1.7	March 4, 2008	Updates for 7.2.1	Lin Chen
1.8	March 16, 2009	Updated to fix ER **205953016**	Lin Chen