VXML Interaction Server (VIS) configuration guide version 6.3

This page was not added to the PDF due to the following tag(s): article:topic-guide

Generated on December 26, 2019

This article applies from version 6.3.0 to 6.3.0

Configuring the VXML Interaction Server (toolkit.properties)

Overview

The configuration information for the VXML Interaction Server is stored in a toolkit.properties file. The toolkit.properties file contains settings that are used to determine routing decisions within the system. The following toolkit.properties template files are located in the Sample_Configuration_*x.x.x.xxx*.zip file on the VXML Interaction Server download location.

- aacc.toolkit.properties
- avp.h323.toolkit.properties (select this when H.323 inbound or outbound is used)
- avp.sip.toolkit.properties (select this when SIP inbound and outbound are used)
- cic.toolkit.properties
- genesys.vxmlidriver.toolkit.properties
- gvp.toolkit.properties
- ivg.avaya.toolkit.properties (select this for either IVG Avaya integrations or IVG Genesys integrations)
- ucce.toolkit.properties
- uccx.toolkit.properties
- voxeo.toolkit.properties
- vtop.toolkit.properties

To configure the VXML Interaction Server, perform the following steps:

1. In Windows systems, create a VirtualHold folder on the same server and same drive that the Apache Tomcat Web server is installed on.
2. In Linux systems, create /etc/VirtualHold folder on the same server and same drive that the Apache Tomcat Web server is installed on.
3. Copy the appropriate toolkit.properties template file to the newly created VirtualHold folder.
4. Rename the template file to **toolkit.properties**.
5. Open the toolkit.properties file in a text editor.
6. Edit the file for your specific configuration based on the comments contained in the sample file.
7. **Save** and **Close** the file.

Configuring Prompt Recorder and Grammar Type Usage

A new setting, com.virtualhold.toolkit.promptrecorder.enabled, must be added to this file to indicate whether or not Prompt Recorder is being used in this environment. Configuring this setting to **true** when the promptrecorder.properties

file is read in when a call is created. Configure this to **false** when the Prompt Recorder is not utilized.

The com.virtualhold.toolkit.grammar.type=*grammar_file_type* property identifies the grammar file to be used (with VIS 6.2.1 or later).

The allowable settings for this property are:

- **inline** - An inline grammar file is used.
- **external** - An external grammar file is used. This is the default setting.

License Management

In Virtual Hold (Callback) Versions 8.3 and later, License Management for Callback includes monitoring and control of the number of simultaneous VXML Interaction Server (VIS) voice sessions. Refer to the Installing Virtual Hold Licenses topic for detailed licensing procedures.

Optional: Changing Location of toolkit.properties File

If you want to store the file in a different location, you have two choices: create an environment variable, or configure the Apache Tomcat Web server to identify the new location. The default location is C:/VirtualHold.
For the environment variable, perform the following steps on the server that contains the VXML Interaction Server:

1. Right-click **My Computer** and select **Properties**.
2. (Windows 7) In the *System* window, click **Advanced system settings** in the left pane.
3. In the *System Properties* window, select the Advanced tab, then click **Environment Variables ...**.
4. Under System Variables, click **New...** to add a new environment variable.
5. Enter the Variable name as **VHT_VIS_PATH**.
6. Enter the Variable value as the path to the toolkit.properties file.
7. Click **OK** twice, then close the *System* window.
8. Restart the server.

Or, on the server that contains Apache Tomcat, perform the following steps:

1. Open the **Apache Tomcat Java Properties** and select the Java tab.
2. Select the Advanced tab, then click **Environment Variables ...**.
3. Enter the path to the toolkit.properties file in the Java Options text window.
4. Click **Apply**.

For Linux systems using Apache Tomcat:

1. Enter **/tomcat/bin** in command prompt.
2. Create a file named setenv.sh if not already done (refer here for more information).

3. Open the setenv.sh file.
4. Edit the file with a new location.
    a. Add **export JAVA_OPTS="-DVHT_VIS_PATH=***/var/lib/sample/***"** to the file where */var/lib/sample/* is your specific location.
5. Put the toolkit.properties file in the location specified in the setenv.sh file.
6. Restart Apache Tomcat.

Alternative to Use of Colon Character in KVP Data

The use of the colon character in KVP data is not supported in integrations using the Genesys VXML or AVP IDriver. However, it is possible to configure VIS to automatically prefix destination numbers with tel: achieving the same end result. Use the following procedure to configure VIS:

1. Remove all colon characters from the KVP data.
2. Location the toolkit.property file within VIS.
3. Set the VH_TransferDestination variable to the destination number. (VH_TransferDestination=12345 for example)
4. Ensure the com.virtualhold.transferprefix option is present and set to tel:. (com.virtualhold.toolkit.transferprefix=tel: for example)

Configuring Toolkit.properties for External Media Files

Overview

The VXML Interaction Server can play voice prompts hosted on other media servers. These external media servers are configured in the toolkit.properties file.
Below is a sample toolkit.properties file for external media files.

```
#Total number of media servers where voice files willl be played from
external.mediaserver.count=1

#Media server url configuration
#This url is resolved by the voice browser so localhost will not work
#For additional entries, just increase the number on the end of the property name
#For example, external.mediaserver.2
external.meddiaserver.1=http://media_server_address:media_server_port/voices/

#Distribution method options
#failover - Goestop to bottom attempting to fetvh the media file and will play from the
#location where it first finds the media file
#balanced - Load balances the media file fetching from the media servers, will failover
#if needed
external.mediaserver.distribution=failover

#Media server filure logging options
#none - will not log anny media server failures
#first - will log the first time that a media server error occurs
#all - will log a media server error every time it occurs
external.meddiaserver.logging=first
```

Note: The external media settings are case insensitive.

Number of media servers: (external.mediaserver.count=1)

- When two or more media servers are used, change **external.mediaserver.count=1** to **external.mediaserver.count=X** (where X is the number of servers).

Path to each server: (external.mediaserver.1=http://[path to server])

- Change this to point to the directory where the voice files are stored on the first server.
- Add additional lines for **external.mediaserver.2**, etc. as needed (one line per server).

  Note: Both HTTP and HTTPS are supported.

Balanced or failover mode: (external.mediaserver.distribution=)

- balanced: Individual prompts will be played from all media servers in a balanced fashion. (The distribution will be equally spread among all servers.)
- failover: All prompts will be played from the first server only. If the first server is not available, the second server in the list will be used. If the second is not available, the third will be used, and so on. (This setting is only effective if multiple servers are used.)

  Note: Balanced mode will automatically handle a failover scenario. If one server fails, the other servers are used and the unavailable server is skipped.
  In systems using Apache Tomcat, the Apache Tomcat log indicates which media servers are configured, but does not log the individual prompts. Prompts are logged in the browser log for GVP, CVP, or AVP (or other voice platform used). Both logs include the path to the server.

Failure log level: (external.mediaserver.logging=)

- none: A failure message will only be logged for the external media files if all servers fail.
- first: A failure message will only be logged for the first server to fail. Subsequent failures for that server are not logged. The other servers are logged upon start, but not fail.
- all: All possible success and failure messages will be logged.

  Note: When using Apache Tomcat, refer to Media Server Logging in Apache Tomcat for sample log messages.
  If this setting does not exist, the log level will default to "first".

Configuring Language and Custom Media Fields in EyeQueue

Overview

You must configure the Language and Custom Media fields in the **Adv Settings** > **Language Settings** section of EyeQueue for each segment to contain the correct language and custom folder name.

1. In the Select a Segment field, choose the segment.
2. Select the Language option, then enter the name for the language used by this segment exactly as shown here:
   - Arabic
   - Danish
   - Dutch
   - English
   - EUPortuguese
   - FrenchCanadian
   - German
   - Hebrew
   - Italian
   - Korean
   - MandarinChinese
   - Norwegian
   - SpanishNA
   - Swedish
   - Turkish

Important:
Language support varies depending on your version of VIS. Refer to the Virtual Hold Compatibility and Integration Matrix for language details.

3. Select the Custom Media option, then enter the folder name from the media server containing customized voice prompts for this segment.

   Note: If the Custom Media value is blank or is not a valid folder name, the VXML Interaction Server will use the files from the Default folder for the specified language.
   If the Language value is blank or is not a valid language name, the VXML Interaction Server will use the files from the Default folder for English.

4. Click **Submit**.

The language and custom media settings will will take effect immediately. No restart of Virtual Hold is required.

Language-Specific Features

VXML Interaction Server-supported IVRs build sentences that are spoken to callers by using special callflow logic to piece recorded prompts together. For some languages, the expressing of entered digits, EWT time values and scheduled time values is better handled by capturing or using prerecorded language-specific values in five additional directories located in the Default directory. These directories are:

- ENTERED_DIGITS - Contains the numbers zero through nine, and will be used for phone numbers and extensions.
- EWT_MINUTES - Contains the numbers zero through fifty-nine and will be used to play minutes that need to be spoken for EWT.
- EWT_HOURS - Contains the numbers zero through twenty-three and will be used to play hours that need to be spoken for EWT.
- SCHED_MINUTES - Contains the numbers zero through fifty-nine and will be used to play any minutes that need to be spoken for time.
- SCHED_HOURS - Contains the numbers zero through twenty-three and will be used to play any hours that need to be spoken for time.

These additional directories are currently employed for the VIS Arabic language only. Be sure to replace these prerecorded voice prompts when creating new Brands.

Customizing external media files

Overview

any clients use customized sets of voice prompts, instead of the standard voice files provided by Virtual Hold. The custom voice prompts must be organized using a specific folder structure so that the VXML Interaction Server will recognize them and play them correctly.
You can stage the voice files on any machine as you copy and arrange them. Then you will deploy the files to the media server(s).
You can use the Voices_x.x.x.xxx.zip file for Version 5.x.x provided in the Virtual Hold Download Center as a starting point, or you can create a new parent folder to hold the voice files. This documentation assumes you are starting with the voices folder.
The voices (parent) folder must contain a subfolder for each language used.

> **Note:**
> Language support varies depending on your version of VIS. Refer to the Virtual Hold Compatibility and Integration Matrix for language details.

These folders must be named:

- VHT_Arabic
- VHT_Danish
- VHT_Dutch
- VHT_English
- VHT_EUPortuguese
- VHT_FrenchCanadian
- VHT_German
- VHT_Hebrew
- VHT_Italian
- VHT_Korean
- VHT_MandarinChinese
- VHT_Norwegian
- VHT_SpanishNA
- VHT_Swedish
- VHT_Turkish

Each language folder must contain a subfolder named Default, and optionally extra subfolders containing custom files. The VXML Interaction Server will look for a custom voice file first. If a custom file cannot be found, it will use the files from the Default folder.
If you are using the voices folder from Virtual Hold with no changes, you can skip the steps below. Continue with the

deployment steps listed in the Deploying External Media Files topic in the VXML Interaction Server Installation Guide.
To set up custom prompts for VXML Interaction Server

1. Navigate to the voices folder. Open the subfolder for the first language.
2. You will see a Default folder. This folder contains the standard set of voice prompts. We recommend making a copy of this folder under a different name (such as VHT_Defaults). This will preserve the files in case you need to use them later for troubleshooting.
3. Delete the contents of the Default folder and copy the files for the first custom call script here. Be sure this folder contains the full set of voice files.

   Note: The Default folder must also contain these four additional files:
   - .library
   - ContactNumber.grxml
   - ContactNumber.regex
   - recordtone.wav
4. To set up a second call script, create a new folder (at the same level as Default) and copy the custom prompts for the second call script into it.
5. Repeat step 4 for the remaining call scripts.

   Note: The new folders only need to contain those files which differ from the default. However, you may choose to copy entire prompt sets into the new folders. This will make the copying process simpler, but will result in more .wav files being stored on the media server(s).
   Any major variation (such as male/female voice or US English/British English) must contain the full prompt set in its folder. It must also include blank .wav files to replace any files that are present in the default set that should not be played. This is to avoid mixing genders or languages. Refer to the **Sample Voice Project** below.
6. Make sure each subfolder under [first language] contains an empty .library file (size 0 KB). If needed, copy the .library file from the Default folder and paste it into the others. This file is used by VIS to help determine if there are Custom Media voice prompts to be used or if Default for the language should be used.
7. Repeat steps 1-6 for other supported languages.
8. Deploy the voices folder and its contents to the media server(s). Refer to Deploying External Media Files.
9. Configure EyeQueue to contain the correct language and custom folder name. Refer to Configuring Language and Custom Media Fields in EyeQueue.

The changes will take effect immediately. No restart is needed.

Sample Voice Project

A client has six call scripts in English: four in US English and two in British English. The voice project should be set up in \voices\VHT_English as follows:

- The default set of US English prompts in "Default"
- Custom prompts only in US English in "VHTCallScript_2", "VHTCallScript_3", and "VHTCallScript_4"
- Full prompt sets in British English in "VHTCallScript_5" and "VHTCallScript_6"

If any prompt is missing from either 5 or 6, it will be played from "Default", which will be US English instead of British English.

Configuring the VXML Interaction Server for Monitoring by System Management

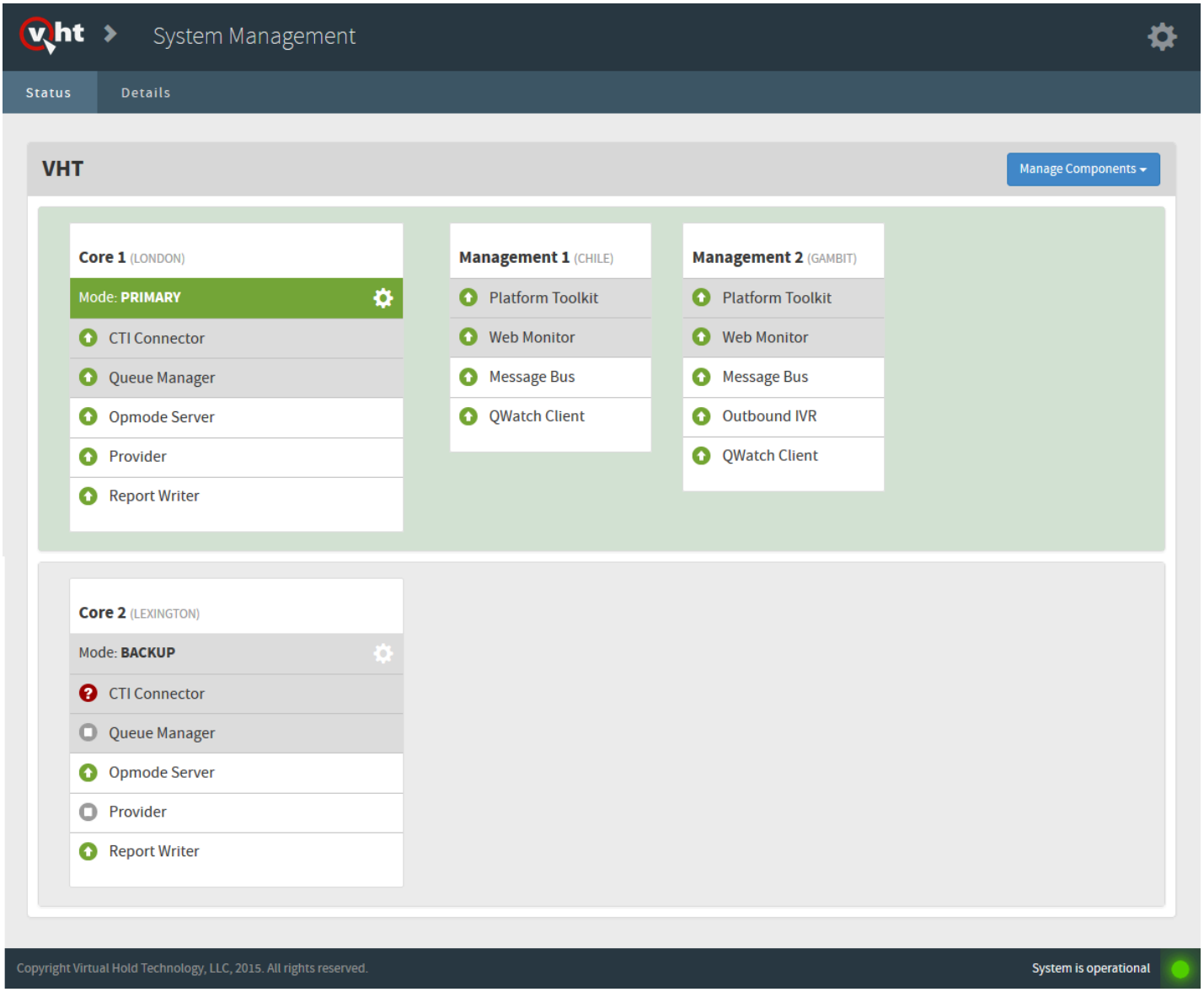
Overview

In Virtual Hold Version 8.1 and above, the VXML Interaction Server (VIS) can be monitored by the System Management interface. This allows administrators to see its status (started, stopped, or unknown) along with the status of other components of Virtual Hold.

When Virtual Hold is installed or updated, it will automatically install a Web Monitor component on each Management Instance. This Web Monitor can be configured to monitor instances of VIS by manually editing the site.config file for the Notification Server.

The System Management user interface will display each VIS instance as a component. If VIS is installed on a Management Instance, it will display inside the appropriate Management Instance container. If VIS is installed outside of a Management Instance, it will appear in its own container.

The following image shows System Management monitoring an instance of VIS installed outside of a Management Instance.



For instructions on configuring System Management to monitor VIS, refer to System Management Notification Server Configuration.

Configuring Time Zone Selection Support

Overview

The VXML Interaction Server (VIS), by default, uses the local time of the standalone or active Queue Manager server when scheduling callbacks. This behavior can cause problems for callers asking for a callback based on the time in another time zone.

Starting with Version 5.12.0, VIS can be configured to provide selection of the time zone to which the callback should be sent. This requires configuration of the segment variables table (dbo.SegmentVariables) within the Callback SQL Server database.

VIS provides time zone selection for the configured time zones, including the automatic adjustment of Daylight Savings Time (DST).

> **Note:**
>
> Ensure the appropriate custom branded time zone selection voice prompt files used by this feature are recorded and available before enabling this feature.

VIS version 6.3 or later includes time zone selection prompts in the following languages.

- Arabic
- Danish
- Dutch
- English
- European Portuguese
- Canadian French
- German
- Hebrew
- Italian
- Korean
- Mandarin Chinese
- Norwegian
- Spanish
- Swedish
- Turkish

Note:

Language support varies depending on your version of VIS. Refer to the Virtual Hold Compatibility and Integration Matrix for language details.

VIS uses the Java™ Platform Standard Edition TimeZone class to calculate a time zone offset and then adjust for daylight savings time (DST) automatically. The TimeZone class also handles time zone ID's and unique names in the form of *area*!*location* (America!New_York for example) where *area* is the name of a continent/ocean and *location* is the name of a specific location within the area (usually a city). Use of this class allows customers to configure time zone options from as broad as US!Eastern to as specific as America!Indiana!Winamac.

Limitations

The following limitations affect Callback times:

- If the *timezone_default* variable is not configured, time zone selection is not offered and callback times are based on local server time.
- If the *timezone_default* variable is properly configured without at least one alternate selectable time zone, time zone selection is still not offered and callback times are based on the default time zone.

Configuration

To configure and enable time zone selection:

1. Locate the Callback SQL Server database.
2. Open the dbo.SegmentVariables table.
3. For each incoming extension that will provide time zone selection, add the appropriate segment variables and corresponding values.

Segment Variable Name	Description	Value
timezone.1 timezone.2 timezone.3 timezone.4 timezone.5 timezone.6 timezone.7 timezone.8 timezone.9	Selectable time zones (at least one required in order to offer alternate time zones). Configured time zones do not have to start with timezone.1 or be consecutive. **Notes:** • Addition of all nine time zone variables is not required for the time zone selection feature to function. • Only the default and correctly configured time zones are offered.	Refer to Selectable Worldwide Time Zones.
timezone.default	Default time zone (required). This time zone is announced to the caller and is used if another is not selected.	
askcallerfortimezone	Determines if the time zone selection feature is allowed for this segment (required).	**TRUE** **FALSE**

4. Save and close the database.
5. Ensure the single Core instance (in Standalone systems) or active Core instance (in High Availability systems) is restarted for the time zone setting to take effect.

Selectable Worldwide Time Zones

Attached to this topic is the complete list of selectable time zones. This file (vis_selectable_time_zones.pdf) lists the exact time zone names, descriptions, time offset values, and daylight savings time applicability (Y or N). The exact time zone name, as given in column one (US!Eastern for example) should be set as the time zone segment variable value.

Required Conditions

The Time Zone Selection feature functions correctly only if ALL of the following conditions are met:

1. The askcallerfortimezone segment variable is set to **TRUE**.
2. The default time zone segment variable is added (spelled correctly), and properly configured (contains supported value).
3. At least one time zone segment variable (timezone.x) is added (spelled correctly), and properly configured (contains supported value) in order to offer alternate time zones.

Reported Callback Times

Callback time values are managed and reported by Queue Manager according to the time zone where the containing standalone or active Core instance resides. The Global Snapshot report within Dashboards also lists callback times according to the time zone where the containing standalone or active Core instance reside, and not the time zone selected in the VIS Callflow. For example:

- A callback scheduled for 5.00 PM US!Eastern is reported by a Queue Manager located in US!Pacific as 2.00 PM. (5 PM EST = 10 PM UTC = 2 PM PST)

Example Configuration

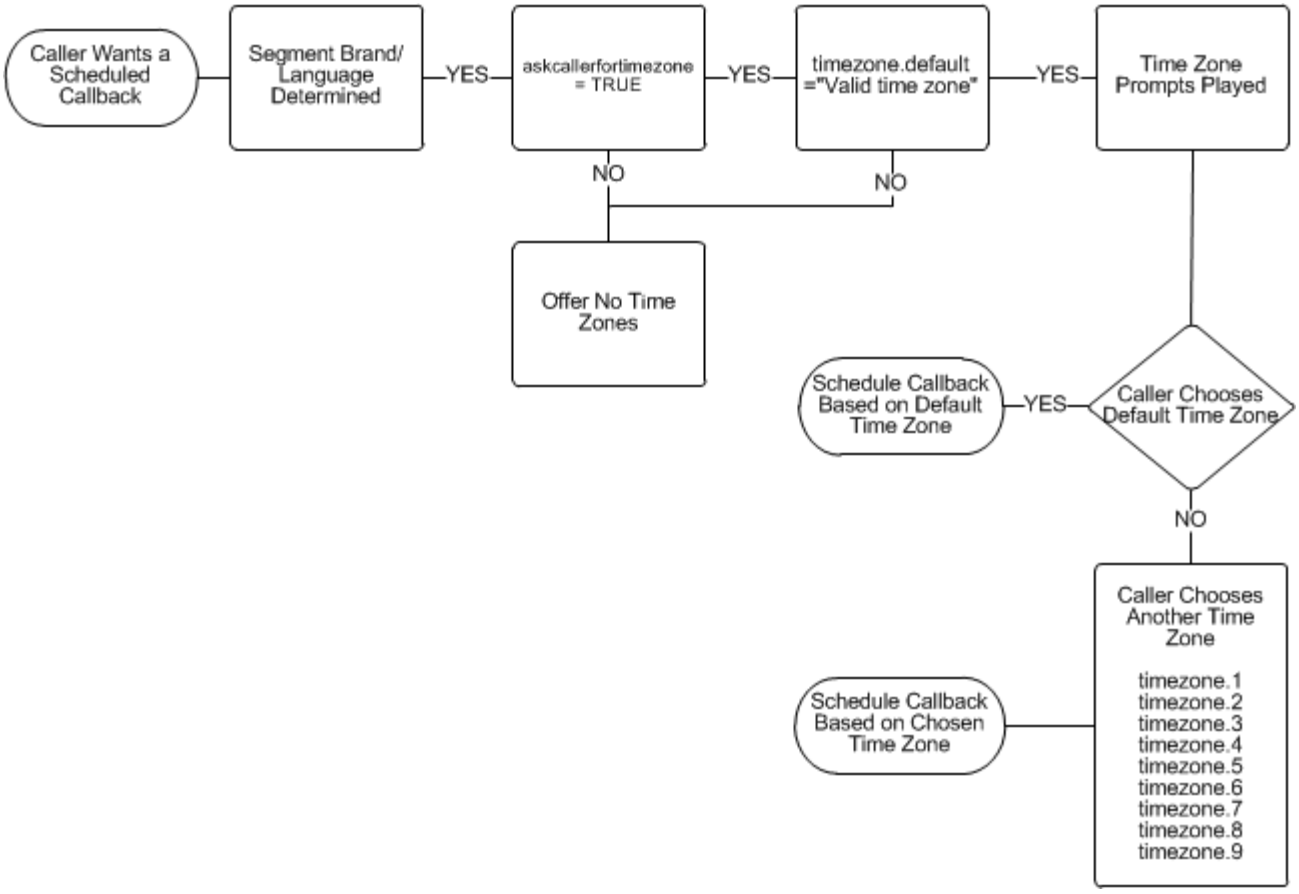
The following figure represents a possible time zone configuration for Incoming Extension 2. In this example:

- The Time Zone Selection feature is active.
- All callbacks are scheduled using the time associated with the Eastern time zone of the United States unless the caller requests other options.
- When selecting another (non-default) time zone, callers are offered the Atlantic time zone within Canada and the Eastern, Central, Mountain, and Pacific time zones within the United States.

	SegmentVariableId	IncomingExtensionsId	Name	Value
▶	1	1	ROUTEDESTINATION	58846
	12	2	timezone.1	Canada!Atlantic
	15	2	timezone.2	US!Eastern
	16	2	timezone.3	US!Central
	17	2	timezone.4	US!Mountain
	18	2	timezone.5	US!Pacific
	19	2	timezone.6	*NULL*
	20	2	timezone.7	*NULL*
	21	2	timezone.8	*NULL*
	22	2	timezone.9	*NULL*
	23	2	timezone.default	US!Eastern
	24	2	askcallerfortimezone	TRUE
*	*NULL*	*NULL*	*NULL*	*NULL*

Sequence of Events

The following figure depicts the sequence of events that take place for this feature.



An overview of the sequence is:

1. Brand/language for the segment is determined.
2. The segment variable askcallerfortimezone is verified to be TRUE. If it is not, this feature is not offered.
3. The segment variable timezone.default is verified to contain the correct name of a time zone. If it does not, this feature is not offered.
4. Time zone selection prompts played.
5. If the caller chooses the default time zone, the callback is scheduled at the appropriate time based on this time zone.
6. If the caller does not choose the default time zone, an alternate time zone is selected from the time zone options configured for the segment and the callback is scheduled at the appropriate time based on this time zone.