Abstract

Interaction Speech Recognition is the Automatic Speech Recognition (ASR) engine for Customer Interaction Center® (CIC). This document provides an overview of the feature, the standards that it uses, and procedures for enabling and configuring the feature.
Copyright and Trademark Information

Interactive Intelligence, Interactive Intelligence Customer Interaction Center, Interaction Administrator, Interaction Attendant, Interaction Client, Interaction Designer, Interaction Tracker, Interaction Recorder, Interaction Mobile Office, Interaction Center Platform, Interaction Monitor, Interaction Optimizer, and the “Spirograph” logo design are registered trademarks of Interactive Intelligence, Inc. Customer Interaction Center, EIC, Interaction Fax Viewer, Interaction Server, ION, Interaction Voicemail Player, Interactive Update, Interaction Supervisor, Interaction Migrator, and Interaction Screen Recorder are trademarks of Interactive Intelligence, Inc. The foregoing products are ©1997-2017 Interactive Intelligence, Inc. All rights reserved.

Interaction Dialer and Interaction Scripter are registered trademarks of Interactive Intelligence, Inc. The foregoing products are ©2000-2017 Interactive Intelligence, Inc. All rights reserved.

Messaging Interaction Center and MIC are trademarks of Interactive Intelligence, Inc. The foregoing products are ©2001-2017 Interactive Intelligence, Inc. All rights reserved.

Interaction Director is a registered trademark of Interactive Intelligence, Inc. e-FAQ Knowledge Manager and Interaction Marquee are trademarks of Interactive Intelligence, Inc. The foregoing products are ©2002-2017 Interactive Intelligence, Inc. All rights reserved.

Interaction Conference is a trademark of Interactive Intelligence, Inc. The foregoing products are ©2004-2017 Interactive Intelligence, Inc. All rights reserved.

Interaction SIP Proxy and Interaction EasyScripter are trademarks of Interactive Intelligence, Inc. The foregoing products are ©2005-2017 Interactive Intelligence, Inc. All rights reserved.

Interaction Gateway is a registered trademark of Interactive Intelligence, Inc. Interaction Media Server is a trademark of Interactive Intelligence, Inc. The foregoing products are ©2006-2017 Interactive Intelligence, Inc. All rights reserved.

Interaction Desktop is a trademark of Interactive Intelligence, Inc. The foregoing products are ©2007-2017 Interactive Intelligence, Inc. All rights reserved.

Interaction Process Automation, Deliberately Innovative, Interaction Feedback, and Interaction SIP Station are registered trademarks of Interactive Intelligence, Inc. The foregoing products are ©2009-2017 Interactive Intelligence, Inc. All rights reserved.

Interaction Analyzer is a registered trademark of Interactive Intelligence, Inc. Interaction Web Portal and IPA are trademarks of Interactive Intelligence, Inc. The foregoing products are ©2010-2017 Interactive Intelligence, Inc. All rights reserved.

Spotability is a trademark of Interactive Intelligence, Inc. ©2011-2017. All rights reserved.

Interaction Edge, CaaS Quick Spin, Interactive Intelligence Marketplace, Interaction SIP Bridge, and Interaction Mobilizer are registered trademarks of Interactive Intelligence, Inc. Interactive Intelligence Communications as a Service℠ and Interactive Intelligence CaaS℠ are trademarks or service marks of Interactive Intelligence, Inc. The foregoing products are ©2012-2017 Interactive Intelligence, Inc. All rights reserved.

Interaction Speech Recognition and Interaction Quality Manager are registered trademarks of Interactive Intelligence, Inc. Bay Bridge Decisions and Interaction Script Builder are trademarks of Interactive Intelligence, Inc. The foregoing products are ©2013-2017 Interactive Intelligence, Inc. All rights reserved.

Interaction Collector is a registered trademark of Interactive Intelligence, Inc. Interaction Decisions is a trademark of Interactive Intelligence, Inc. The foregoing products are ©2013-2017 Interactive Intelligence, Inc. All rights reserved.

Interactive Intelligence Bridge Server and Interaction Connect are trademarks of Interactive Intelligence, Inc. The foregoing products are ©2014-2017 Interactive Intelligence, Inc. All rights reserved.

The veryPDF product is ©2000-2017 veryPDF, Inc. All rights reserved.

This product includes software licensed under the Common Development and Distribution License (6/24/2009). We hereby agree to indemnify the Initial Developer and every Contributor of the software licensed under the Common Development and Distribution License (6/24/2009) for any liability incurred by the Initial Developer or such Contributor as a result of any such terms we offer. The source code for the included software may be found at http://wpflocalization.codeplex.com.

A database is incorporated in this software which is derived from a database licensed from Hexasoft Development Sdn. Bhd. ("HDSB"). All software and technologies used by HDSB are the properties of HDSB or its software suppliers and are protected by Malaysian and international copyright laws. No warranty is provided that the Databases are free of defects, or fit for a particular purpose. HDSB shall not be liable for any damages suffered by the Licensee or any third party resulting from use of the Databases.

Other brand and/or product names referenced in this document are the trademarks or registered trademarks of their respective companies.

DISCLAIMER

INTERACTIVE INTELLIGENCE (INTERACTIVE) HAS NO RESPONSIBILITY UNDER WARRANTY, INDEMNIFICATION OR OTHERWISE, FOR MODIFICATION OR CUSTOMIZATION OF ANY INTERACTIVE SOFTWARE BY INTERACTIVE, CUSTOMER OR ANY THIRD PARTY EVEN IF SUCH CUSTOMIZATION AND/OR MODIFICATION IS DONE USING INTERACTIVE TOOLS, TRAINING OR METHODS DOCUMENTED BY INTERACTIVE.

Interactive Intelligence, Inc.
7601 Interactive Way
Indianapolis, Indiana 46278
Telephone/Fax (317) 872-3000
www.ININ.com
Table of contents

Interaction Speech Recognition

Table of contents

Introduction to Interaction Speech Recognition .................................................. 5

Table of contents ................................................................................................. iii

Interaction Speech Recognition ........................................................................... 1

Configure Interaction Speech Recognition......................................................... 10

Configure Interaction Attendant to use Interaction Speech Recognition .......... 13

Interaction Speech Recognition grammars ....................................................... 19

Troubleshooting ................................................................................................. 36

Enable diagnostic logging for Interaction Speech Recognition through Interaction Administrator .......................................................... 37

Enable diagnostic recordings for Interaction Speech Recognition through Interaction Media Server ................................................................. 37

Warnings on usage of diagnostic recordings and diagnostic logs for Interaction Speech Recognition ................................................................. 36

Enable diagnostic recordings for Interaction Speech Recognition through Interaction Administrator ................................................................. 36

Enable diagnostic logging for Interaction Speech Recognition through Interaction Administrator ................................................................. 37

Adjust audio detection sensitivity ....................................................................... 12

Ensure appropriate licenses are installed ................................................................ 10

Create users in Interaction Administrator ............................................................ 13

Create and enable the company directory speech recognition capability ................ 13

Ensure speech recognition for company directory in Interaction Attendant ............ 14

Use Interaction Speech Recognition for Interaction Attendant operations and menu navigation ................................................................. 15

Using the company directory speech recognition capability .................................. 13

Add an operation through Interaction Attendant .................................................. 15

Add keywords and phrases to an Interaction Attendant operation .......................... 15

Add additional grammar files for preloading ....................................................... 17

Types of grammars .............................................................................................. 19

Buildin grammars ............................................................................................... 19

Custom grammars .............................................................................................. 21

Preloaded grammars ........................................................................................... 21

VoiceXML grammars ......................................................................................... 21

Grammar syntax .................................................................................................. 22

Example ABNF grammar file .............................................................................. 22

Example GrXML grammar file ........................................................................... 22

Best practices ...................................................................................................... 24

Design grammars and prompts simultaneously .................................................. 24

Remove ambiguity .............................................................................................. 24

Duplicate tokens .................................................................................................. 24

Use SISR tags for operations .............................................................................. 25

Use grammar weights and probabilities .............................................................. 26

Use filler rules or garbage rules to catch unimportant words in responses ............... 27

Reference builtin grammars instead of recreating functionality ............................ 29

Do not try to address all possible responses ....................................................... 31

Identify and fix problems after deployment ....................................................... 32

Testing grammars .............................................................................................. 34

Test grammar validity ....................................................................................... 34

Analyze functionality ......................................................................................... 34

Use custom grammars with Interaction Speech Recognition ......................... 35

Windows Event Log contains entries for invalid grammars ............................... 36

Dial by Name not working .................................................................................. 36

Audio problems with Interaction Speech Recognition ....................................... 36
Limitations ................................................................. 39
Change Log............................................................... 40
Introduction to Interaction Speech Recognition

Interaction Speech Recognition is a native Automatic Speech Recognition (ASR) engine for Customer Interaction Center and was introduced in CIC 4.0 SU4. Interaction Speech Recognition can recognize utterances (speech patterns), convert those utterances to data, and send the data to Customer Interaction Center, which can then take specific actions, such as directing a call to a specific extension, playing additional prompts, and many others.

Interaction Speech Recognition is an integrated feature of Customer Interaction Center. As such, Interaction Speech Recognition does not have an installation program nor can it reside on a separate host server.

Support for Interaction Speech Recognition is incorporated in the following Customer Interaction Center products:

- Customer Interaction Center Server
- Interaction Media Server
- Interaction Attendant
- Interaction Administrator
- Interaction Designer (handlers)

For more information on these products, see the CIC Documentation Library at http://help.inin.com.

This document contains the following sections:

- **Introduction to Interaction Speech Recognition** – This section provides an overview of Interaction Speech Recognition, its features and capabilities, and how it works with other components of Customer Interaction Center.

- **Configure Interaction Speech Recognition** – This section provides the procedures for licensing and enabling Interaction Speech Recognition.

- **Configure Interaction Attendant to use Interaction Speech Recognition** – This section provides the optional procedures of configuring Interaction Attendant to use Interaction Speech Recognition for recognizing responses from callers for Interactive Voice Response (IVR) menu prompts.

- **Interaction Speech Recognition grammars** – This section provides definitions of speech recognition grammars, examples, and best practices in grammar development, should you need to create your own custom grammars for extended IVR functionality.

- **Troubleshooting** – This section provides information and corrective procedures for problems that you experience while using Interaction Speech Recognition.
# Interaction Speech Recognition requirements

| Software | • Customer Interaction Center server 4.0 Service Update 4 or later  
• Interaction Media Server 4.0 Service Update 4 or later  
• Interaction Administrator (installed through IC Server Manager Applications 4.0 SU4 or later package) |
| --- | --- |
| Customer Interaction Center licenses | • I3_SESSION_MEDIA_SERVER_RECO  
• I3_FEATURE_SPEECH_RECOGNITION  
• I3_LICENSE_VoiceXML_SESSIONS (required only if you want to integration Interaction Speech Recognition with VoiceXML functionality)  
• For any language that you want to use with Interaction Speech Recognition, you must purchase and install an Interaction Analyzer language license. |
| Note: | For more information about viewing your Customer Interaction Center license, including product and feature licenses, see the "License Management" topic in *Interaction Administrator Help.* |
| Languages | English, Australia (en-AU)  
English, United Kingdom (en-GB)  
English, United States (en-US)  
French, Canada (fr-CA)  
French, France (fr-FR)  
Italian, Italy (it-IT)  
Mandarin Chinese, China (zh-CN)  
Spanish, Spain (es-ES)  
Spanish, United States (es-US) |
| Configuration | To enable automatic dialing of users through speech recognition, you must define one or more users in the company directory |
| Standards | • Interaction Speech Recognition uses grammars that conform to the Speech Recognition Grammar Specification (SRGS) standard.  
• Interaction Speech Recognition has limited support for the Semantic Interpretation for Speech Recognition (SISR) v1.0 standards |
Overview of the Interaction Speech Recognition process

1. A call is received by Customer Interaction Center.

2. Interaction Attendant, VoiceXML, or custom handlers, based on information with the call, selects a prompt to play to the caller.

3. The Customer Interaction Center server sends grammars and prompts to use for the input operation. The Customer Interaction Center server then directs Interaction Media Server to play a prompt to the caller and wait for input.

4. Interaction Media Server plays the prompt to the caller and waits for a response.

5. The caller responds to the prompt through speech (an utterance) or by pressing keys on the telephone keypad.

6. Interaction Speech Recognition recognizes the response.

7. Interaction Speech Recognition returns the recognition data to Customer Interaction Center.

8. Interaction Attendant, custom handlers, or the VoiceXML interpreter processes the data and takes an associated action, such as playing another prompt or sending the call to a workgroup queue.
## Details of the Interaction Speech Recognition process

<table>
<thead>
<tr>
<th>Interaction Media Server performance</th>
<th>An Interaction Speech Recognition session requires approximately the equivalent processing resources as a single, two-party call that is recorded and transcoded.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interaction Media Server selection</td>
<td>When a call requires audio, Customer Interaction Center selects an Interaction Media Server to provide audio based on Media Server Selection Rules and the location in which the call was received or placed. If the call requires Interaction Speech Recognition, the processing for speech recognition occurs on the same Interaction Media Server that is providing audio for the call. For more information about Media Server Selection Rules, see Interaction Media Server Technical Reference.</td>
</tr>
<tr>
<td>Grammar preloading and caching</td>
<td>Large or complex grammars with hundreds or thousands of entries can delay responsiveness if Interaction Media Server must download and compile those grammars during a call. For this reason, Interaction Speech Recognition supports preloading of grammar files. When Interaction Media Server starts and connects to the Customer Interaction Center server, it downloads (through HTTP), compiles, and caches in memory the grammar files specified in the Preloaded Grammars tab of the Configuration dialog box of either the Interaction Speech Recognition object or the parent Recognition container in Interaction Administrator. The recognition subsystem of the Customer Interaction Center server also compiles and caches these grammars. You can also preload grammars in custom handlers through the Reco Register Preloaded Grammars tool in Interaction Designer. If you change or add a preloaded grammar through custom handlers or Interaction Administrator, Interaction Media Server automatically downloads, compiles, and caches the new grammar. Interaction Media Server caches non-preloaded grammars for Interaction Speech Recognition when used during a call.</td>
</tr>
<tr>
<td>Interoperability and customization</td>
<td>By creating custom handlers through Interaction Designer, you can create a solution that accomplishes your goals. For more information on Interaction Designer, see Interaction Designer Help. If you want to use Interaction Speech Recognition in combination with VoiceXML, you must have a dedicated host for your VoiceXML interpreter. You must also purchase VoiceXML licenses for Customer Interaction Center. For more information about VoiceXML in a Customer Interaction Center environment, see VoiceXML Technical Reference.</td>
</tr>
<tr>
<td>Ports, sessions, and licensing</td>
<td>Interactive Intelligence recommends that you purchase enough licenses for Interaction Speech Recognition ports to equal the number of licensed Interactive Voice Response (IVR) ports. This equality ensures that Interaction Speech Recognition is able to support all IVR sessions. Each time a Reco tool step is called through a custom Customer Interaction Center handler or when a VoiceXML document requires Interaction Speech Recognition, a new Reco (representing the recognition subsystem of Customer Interaction Center) session port license is used. That port license is released when the Reco Release tool step is called through the handler.</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
</tbody>
</table>
Configure Interaction Speech Recognition

This section contains procedures for enabling Interaction Speech Recognition and configuring additional capabilities.

Ensure appropriate licenses are installed

1. Start Interaction Administrator.
2. On the tool bar, select the License button.

![Interaction Administrator](image)

The License Management dialog box appears.

3. On the Licenses tab, ensure that the following licenses are installed and have the appropriate number of sessions:
   - **I3_LICENSE_VoiceXML_SESSIONS** (if applicable)
   - **I3_SESSION_MEDIA_SERVER_RECO**
4. Select the Features tab.
5. On the Features tab, ensure that the following licenses are install and have the appropriate number of sessions:
   - **I3_FEATURE_ANALYZER_LANGUAGE_NN** (*NN* represents the language code of two to five characters)
   - **I3_FEATURE_SPEECH_RECOGNITION**
6. When you have verified the existence of the licenses and available number of ports, select the Close button.

   If you do not have the necessary licenses or number of ports, contact your sales representative.
Enable Interaction Speech Recognition

Once you have installed the Interaction Speech Recognition feature license, you need to enable the system.

1. Start Interaction Administrator.
2. In the left pane, expand the System Configuration container.
3. Under the System Configuration container, expand the Recognition container.
4. In the Recognition container, select the Interaction Speech Recognition object.
5. In the right pane, double-click the Configuration item.
6. When the ASR Engine Configuration dialog box appears, select the Enabled check box.
7. Click OK.

**Important!**
Do not change the Engine Integration Module (EIM) file specified in the EIM Module dll box.
Adjust audio detection sensitivity

You can adjust Interaction Speech Recognition's sensitivity to input noise by altering the value of the audio detection sensitivity property. By default, the value is set at 0.5, but you can change it to any value between 0.0 and 1.0.

The default value of 0.5 is recommended as it is tuned to be robust enough to ignore spurious noises yet at the same time be able to trigger on true speech. A value of 0.0 will configure the system to be the least sensitive to noise while a value of 1.0 would make it highly sensitive to quiet input. More specifically, a value of 0.0 will cause Interaction Speech Recognition to treat spurious noise as silence and thus prevent it from triggering a speech event on such noise. At the other end of the spectrum, a value of 1.0 could possibly cause Interaction Speech Recognition to trigger a speech event on any noise.

To adjust the audio detection sensitivity:

1. Start Interaction Administrator.
2. In the left pane, expand the System Configuration container.
3. Under the System Configuration container, expand the Recognition container.
4. In the Recognition container, select the Interaction Speech Recognition object.
5. In the right pane, double-click the Configuration item
6. When the ASR Engine Configuration dialog box appears, select the Properties tab, and click the Add button.
7. When the Add Property dialog box appears, type sensitivity in the Name text box and click the Add Value button.
8. In the Add Property Value dialog box enter a value between 0.0 and 1.0
9. To complete the operation, click OK three times—one to close each of the open dialog boxes and enable the audio detection sensitivity setting.
Configure Interaction Attendant to use Interaction Speech Recognition

You can configure Interaction Attendant, the Interactive Voice Response (IVR) product from Interactive Intelligence, to work in conjunction with Interaction Speech Recognition. Integrating these two products provides you with the following capabilities:

- Enable callers to speak the name of the Customer Interaction Center user to whom they want to communicate
- Enable callers to speak IVR menu item selections or operations

By default, Interaction Attendant enables the company directory speech recognition capability; however, you must configure Interaction Attendant to use Interaction Speech Recognition before the feature will function.

Create and enable the company directory speech recognition capability

Customer Interaction Center automatically builds a grammar from the users that you define through Interaction Administrator. Customer Interaction Center can automatically preload this grammar and transfer it to Interaction Media Servers where the processing for Interaction Speech Recognition occurs. This feature enables a caller to say the name of the person with whom the caller wants to communicate. Customer Interaction Center then transfers the caller to the recognized user. This feature is commonly named the company directory.

The following sections provide the procedures for you to enable and configure the company directory speech recognition capability with Interaction Administrator and Interaction Attendant. These procedures assume that you have already enabled Interaction Speech Recognition as specified in the procedure, Enable Interaction Speech Recognition.

Create users in Interaction Administrator

1. Open Interaction Administrator.
2. In the left pane, expand the People container.
3. Under the People container, select the Users object.
4. In the right pane, right-click an open area and select New from the resulting shortcut menu.
   The Entry Name dialog box appears.
5. In the Entry Name dialog box, enter the name of a new Customer Interaction Center user.
   You can also use the button to the right of the text box to select users from available Microsoft Windows Server network domains.
6. Select the OK button.
   The User Configuration dialog box appears.
7. Configure the necessary options for the new user.
   For more information about configuring Customer Interaction Center users, see the "User Configuration" topic in Interaction Administrator Help.
8. In the User Configuration dialog box, select the OK button.
   The new user is added to the list of users in the Interaction Administrator window.
9. Repeat this procedure for all users that you want to add to the company directory.
Ensure speech recognition for company directory in Interaction Attendant

1. Open Interaction Attendant.

2. In the left pane of the **Interaction Attendant** window, select a profile for which you want to enable selection of a Customer Interaction Center user through speech recognition.
   
   The right pane displays the configuration interface for the selected profile.

3. In the right pane, in the **Node Characteristics** group, select the **Configure Speech Recognition** button.
   
   The **Speech Recognition Configuration** dialog box for the selected profile appears.

4. In the **Speech Recognition Configuration** dialog box, enable the **Enable speech recognition for the company directory** check box.

   **Important!**

   If you want to use Interaction Speech Recognition for other IVR operations, such as recognizing spoken menu items, you must enable the **Enable speech recognition on menus** check box. However, this feature changes the default prompts that are associated with the **Enable speech recognition for company directory** feature. If you choose to enable speech recognition on menus and also leave it enabled for the company directory, Interaction Attendant will not prompt callers to speak the name of the Customer Interaction Center user with whom they want to communicate. If you want to keep the prompting of the caller to supply the name of the user, you must modify the main greeting prompt to present the option to the caller.

5. In the **Speech engine** list box, select the **Interaction Speech Recognition** item.

   **Note:**

   If the **Interaction Speech Recognition** item is not present in the **Speech engine** list box, you must first enable **Interaction Speech Recognition** in Interaction Administrator. For instructions, see [Enable Interaction Speech Recognition](#).

6. In the **Speech Recognition Configuration** dialog box for the profile, select the **OK** button.

7. On the menu bar of the **Interaction Attendant** window, select **File > Publish**.
Use Interaction Speech Recognition for Interaction Attendant operations and menu navigation

You can use Interaction Attendant and Interaction Speech Recognition to enable callers to execute operations through speech. By default, Interaction Attendant does not enable speech recognition for selecting menu items and for operations. Use the following procedures to enable and configure speech recognition for menu item selection and operations. The following procedures assume that you have already enabled Interaction Speech Recognition as specified in the procedure, Enable Interaction Speech Recognition.

Enable speech recognition for operations and menu navigation in Interaction Attendant

1. Open Interaction Attendant.
2. In the left pane of the Interaction Attendant window, select the profile for which you want to enable menu item selection through speech recognition.
3. In the right pane, select the Configure Speech Recognition button.
   The Speech Recognition Configuration dialog box appears.
4. Enable the Enable speech recognition on menus check box.
5. Ensure that the Speech engine list box is set to Interaction Speech Recognition.
6. Select the OK button.
7. Using the Profile Greeting area in the right pane of the Interaction Attendant window, edit the main greeting prompt to inform callers that they can communicate with a Customer Interaction Center user by speaking the name of the user.

Add an operation through Interaction Attendant.

1. Open Interaction Attendant.
2. In the left pane of the Interaction Attendant window, select the profile that contains the schedule for which you want to enable callers to speak to select a menu item.
3. Select a schedule in the selected profile for which you want to enable callers to speak to select a menu item.
4. Add a new operation to the schedule by selecting Insert > New Operation > <operation> on the menu bar.

Add keywords and phrases to an Interaction Attendant operation

After you have added a new operation to a schedule, you must enter the keywords and phrases that a caller can speak to select to cause selection of that operation.
Note:
You can add operations to Interaction Attendant schedules without assigning them to keypad digits. Those operations can then be executed through only speech recognition.

1. In the left pane of the Interaction Attendant window, select the operation that you added.
2. In the right pane, select the Configure Speech Recognition button.
   The Speech Recognition Configuration dialog box appears.

3. In the Language list box, ensure that the correct language is selected.
4. In the top text box, enter a keyword or phrase that a caller must speak to execute the operation.
5. Select the Add button.
   The specified keyword or phrase is added to the lower text box.
6. Select the OK button.
7. If required by the selected operation, provide any additional configuration in the right pane of the Interaction Attendant window, such as selecting a workgroup queue to receive a callback request.

For more information on adding keywords and phrases for speech recognition in Interaction Attendant, see the "Add Speech Recognition keywords or phrases for Inbound Call Operations" topic in Interaction Attendant Help.
Add additional grammar files for preloading

If you already possess or have created large grammar files that you use for an Interactive Voice Response (IVR) system and they conform to the Speech Recognition Grammar Specification (SRGS) standard, you can preload those grammars for use with Interaction Speech Recognition.

For more information about speech recognition grammars, see Interaction Speech Recognition grammars.

**Important!**

It is not necessary to preload small or simple grammars as they do not require significant time to transfer and compile. Adding grammars for preloading causes additional memory usage on Interaction Media Server.

1. Start Interaction Administrator.
2. In the left pane, expand the **System Configuration** container.
3. In the **System Configuration** container, expand the **Recognition** container.
4. In the **Recognition** container, select the **Interaction Speech Recognition** object.

**Note:**

If you want a grammar to be available to all speech recognition engines within Customer Interaction Center and not just Interaction Speech Recognition, select the **Recognition** container and double-click the **Configuration** item in the right pane of the **Interaction Administrator** window.

5. In the right pane, double-click the **Configuration** item.

   The **ASR Engine Configuration** (Interaction Speech Recognition object) or **Recognition Configuration** (Recognition object) dialog box appears.

6. Select the **Preloaded Grammars** tab.

7. On the **Preloaded Grammars** tab, select the **Add** button.

   The **Add Preloaded Grammar** dialog box appears.
8. In the following controls, enter the necessary information:

<table>
<thead>
<tr>
<th>Name (required)</th>
<th>Enter a name for this grammar, which the Reco tools in handlers can use to identify it. For more information on the Reco tools in handlers, see Interaction Designer Help.</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI (required)</td>
<td>Enter the path of the grammar to load. If necessary, you can use the browse button located to the right of this control to explore the file system and specify the file.</td>
</tr>
</tbody>
</table>
| MIME Type (optional) | Enter the Multipurpose Internet Mail Extension (MIME) format of the grammar:  
  • For grammars in the GrXML format, use application/srgs+xml.  
  • For grammars in the ABNF format, use application/srgs. |
| Mode (required) | Select voice. |

9. In the Add Preloaded Grammar dialog box, select the OK button. The new grammar appears in the list box of the Preloaded Grammars tab of the ASR Engine Configuration or Recognition Configuration dialog box.

10. In the ASR Engine Configuration or Recognition Configuration dialog box, select the OK button.
Interaction Speech Recognition grammars

To recognize speech, Interaction Speech Recognition uses grammars that conform to the Speech Recognition Grammar Specification (SRGS) standard developed by the World Wide Web Consortium (W3C) international standards organization. A grammar is a text file in a special format that contains a set of words that a speech recognition engine can interpret from an utterance, such as the names of people in a company. An utterance is the speech that a caller provides in response to a prompt.

Important!
Interaction Speech Recognition supports only voice grammars. It does not support DTMF grammars. Support for recognizing DTMF tones is provided, by default, through the combination of Interaction Media Server and Customer Interaction Center.

For example, one builtin (default) grammar in Interaction Speech Recognition is the Boolean builtin grammar, which specifies the words that are associated with affirmative and negative utterances.

Types of grammars

This section contains information on the different types of speech recognition grammars that you can use with Interaction Speech Recognition.

Builtin grammars

Builtin grammars are a set of grammars that a speech recognition product supports by default. These grammars recognize a basic set of words that are necessary in nearly all speech recognition tasks.

Interaction Speech Recognition supports the following types of words in its builtin grammars:

<table>
<thead>
<tr>
<th>Builtin grammar type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boolean</td>
<td>This builtin grammar type contains the words Interaction Speech Recognition can identify for affirmative and negative utterances. For the &quot;yes&quot; and &quot;no&quot; words, Interaction Speech Recognition returns the values of true and false, respectively.</td>
</tr>
</tbody>
</table>
| Date                 | This builtin grammar type contains the words that Interaction Speech Recognition can identify as a calendar date, including the month, day, and year. Interaction Speech Recognition returns the date as a string of digits in the format of yyyymmdd.  
  - yyyy represents the year (optional).  
  - mm represents the month.  
  - dd represents the numeric day.  
  If a year is not specified, Interaction Speech Recognition inserts a question mark (?) for each digit. |
| Time                 | This builtin grammar type contains the words that Interaction Speech Recognition can identify as a specific time as it is measured by a clock, such as 10:35 AM. Interaction Speech Recognition returns the time as a five-character string in the format of hhmmf. |
• **hh** represents the hour.
• **mm** represents the minute.
• **f** represents the time format:
  — **p** = PM, such as "seven PM"
  — **a** = AM, such as "seven AM"
  — **h** = 24-hour format, such as "nineteen hundred"
  — **?** = Ambiguous, such as "seven o'clock," which does not provide context

<table>
<thead>
<tr>
<th>Digits</th>
<th>This builtin grammar type contains the words that Interaction Speech Recognition can identify as a string of individual digits, not to be confused with a number where the placement of a digit has context. For example, a series of digits could represent an account number, a support case, or some other numeric identifier, such as 0045833154. Numbers, in the case of builtin grammars, represent amounts, such as 2,157 (two-thousand-one-hundred-fifty-seven). The digits builtin grammar type also supports the word &quot;double&quot; when callers reference two identical digits in succession such as &quot;double 7&quot; (77).</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Note:</strong> At this time, the digits builtin grammar type does not support usage of the <strong>minlength</strong>, <strong>maxlength</strong>, or <strong>length</strong> parameters. If you require a specific number of digits, you must add the functionality of verifying the length of the returned string of digits in your processing system, such as custom handlers or VoiceXML. Interaction Speech Recognition returns the string of digits.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Currency</th>
<th>This builtin grammar type contains the words that Interaction Speech Recognition can identify as being related to monetary amounts. Interaction Speech Recognition returns a string in the following ISO4217 international standard format:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>UUUmm.nn</strong></td>
<td><strong>UUU</strong> = Currency type, such as USD</td>
</tr>
<tr>
<td><strong>Note:</strong> The en-us currency builtin grammar type of Interaction Speech Recognition recognizes only &quot;dollars&quot; and &quot;cents&quot;.</td>
<td></td>
</tr>
<tr>
<td><strong>mmm</strong> = The number of whole units of the currency type</td>
<td></td>
</tr>
<tr>
<td><strong>nn</strong> = The number of partial units of a whole unit</td>
<td></td>
</tr>
</tbody>
</table>

| Number | This builtin grammar type contains the words that Interaction Speech Recognition can identify as a complete numerical value; not a series of digits. It also recognizes words that indicate a positive or negative value, such as "plus," "positive," "minus," and "negative." For example, if a caller says, "one-thousand-two-hundred-thirty-four," Interaction Speech Recognition returns 1234. If the caller says a word that indicates a positive or negative value, Interaction Speech Recognition includes an addition symbol (+) or |-
hyphen (-), respectively.

| Phone     | This built-in grammar type contains the words that Interaction Speech Recognition can identify as a telephone number, including an optional extension. For example, if a caller says, "eight, eight, eight, five, five, five, one, two, three, four, extension five, six, seven," Interaction Speech Recognition returns 8005551234x567. A caller can also say numbers in place of digits, such as "eight hundred" (800) and "forty-seven" (47). |

### Custom grammars

Custom grammars are grammars that you create for one or more reasons:

- The built-in grammars do not provide enough recognition entries for all situations.
- The needs of the business require a specific set of words to be recognized, such as names of products, names of subsidiary companies, industry-specific words (medical, technology, security, agriculture, government, and so on), or names of cities, states, or countries.

You create custom grammars as text files that specify the words and optional words that Interaction Speech Recognition will recognize. All custom grammars use the formats and syntax specified by the Speech Recognition Grammar Specification (SRGS) standard. For more information, see Grammar syntax.

**Tip:**

When creating a custom grammar, ensure that you do not create circular references that result in infinite processing. However, when necessary, recursive grammars can provide a great deal of efficiency in your custom grammars.

### Preloaded grammars

In some instances, grammars are required to be very large or very complex. Downloading, compiling, and processing these large or complex grammars can impact performance, both in audio communications and with the resources of the processing system. For these reasons, you can specify that the processing systems will preload those grammars. Preloaded grammars are downloaded, compiled, and cached by processing systems outside of an active speech recognition session. When a speech recognition session then requires use of one of these grammars, the processing system already has the grammar in memory and can recognize utterances very quickly.

You do not need to preload all grammars as each preloaded grammar occupies memory of the processing systems. In most situations, the grammar files are small or simple enough that downloading, compilation, and processing occurs without impacting the session or the performance of the processing system.

Interactive Intelligence recommends that you preload grammars only if testing of a non-preloaded grammar causes performance problems, such as delays in responsiveness and audio quality issues.

### VoiceXML grammars

VoiceXML is a W3C standard for using text files to facilitate interactive voice conversations between people and computers. VoiceXML uses text-based files with specialized eXtensible Markup Language (XML) elements that can define words for Text-to-Speech (TTS), speech recognition, grammars, conversation management, and playing recorded audio.

If you use Interaction Speech Recognition with a VoiceXML system, your VoiceXML files can include or reference grammars that Interaction Speech Recognition will use. VoiceXML grammars also use the Speech Recognition Grammar Specification (SRGS) standard.

When Interaction Speech Recognition recognizes a speech pattern based on the supplied grammar, it returns that data to VoiceXML interpreter.
Grammar syntax

Speech Recognition Grammar Specification (SRGS) consists of two syntaxes for defining grammars:

- Augmented Backus-Naur Form (ABNF).
- Grammar Extensible Markup Language (GrXML)

Interaction Speech Recognition supports both syntaxes in grammar files.

SRGS uses Semantic Interpretation for Speech Recognition (SISR) v1.0 for semantic interpretation.

Example ABNF grammar file

The ABNF syntax uses symbols, words, and operators; much like a scripting or programming language.

```plaintext
// Header
#ABNF 1.0;
language en-US;
mode voice;

// Root rule
root $yesorno;

// Rules
$yes = yes | correct | yeah | affirmative;
$no = no | nope | negative;
$yesorno = $yes | $no;
```

To ensure that you can easily identify ABNF grammars when browsing a file system, use the .gram file name extension.

Example GrXML grammar file

The GrXML syntax uses elements and pairs of elements defined within angle bracket characters (< >).

```xml
<!--Header-->
<?xml version="1.0"?>
<grammar xmlns="http://www.w3.org/2001/06/grammar"
xml:lang="en-US"
root="yesorno"
mode="voice">

<!--Rules-->
  <rule id="yesorno">
    <one-of>
      <item><ruleref uri="#yes" /></item>
      <item><ruleref uri="#no" /></item>
    </one-of>
  </rule>
  <rule id="yes">
    <one-of>
      <item>yes</item>
      <item>correct</item>
      <item>yeah</item>
      <item>affirmative</item>
    </one-of>
  </rule>
  <rule id="no">
    <one-of>
      <item>no</item>
      <item>nope</item>
    </one-of>
  </rule>
</grammar>
```

Interaction Speech Recognition Technical Reference
To ensure that you can easily identify GrXML grammars when browsing a file system, use the `.grxml` file name extension.

For more information about the SRGS specification and how to create grammars, see [http://www.w3.org/TR/speech-grammar/](http://www.w3.org/TR/speech-grammar/).
Best practices

The following sections provide some recommendations from Interactive Intelligence regarding the development of grammars for use with Interaction Speech Recognition.

Design grammars and prompts simultaneously

Interactive Intelligence recommends that you design and develop grammars and prompts simultaneously. Using this method ensures that all prompts have expected utterances and associated actions. It also ensures that you do not have rules in grammars and operations that are never executed.

Remove ambiguity

In grammars, ambiguity refers to entry of a word that does not provide enough context as to its meaning. When a caller speaks the ambiguous word, the grammar does not have enough details or does not cause the system to request the necessary clarification. For example, if you prompt a caller to say the name of the city from which he or she is calling, ensure that you account for cities in the United States having the same name in multiple states, such as “Springfield”. Ensure that your grammars provide you with the exact information that you seek, including necessary details.

Other examples of ambiguity in grammars are homonyms and homophones. Homonyms are different words that have the same spelling and pronunciation, such as "change". Homophones are different words that do not have the same spelling but do have the same pronunciation, such as "red" and "read", and "sale" and "sail".

You can find lists of homonyms and homophones on various Internet websites.

Duplicate tokens

Ensure that your grammars do not include the same token more than once. The following examples display how a token is replicated in two separate rules.

ABNF example of duplicate token

```
#ABNF 1.0;
language en-us;
mode voice;
tag-format <semantics/1.0>;
root $Example;

$Example = $Yes1 | $Yes2;

$Yes1 = (yes [sir|maam] | yeah);
$Yes2 = (yes | okay);
```

GrXML example of duplicate token

```
<?xml version="1.0"?>
<grammar xmlns="http://www.w3.org/2001/06/grammar" 
xml:lang="en-US"
root="Example"
mode="voice">
    <rule id="Example">
        <one-of>
            <item><ruleref uri="#Yes1"/></item>
            <item><ruleref uri="#Yes2"/></item>
        </one-of>
    </rule>
</grammar>
```
In the example above, it is possible for the `yes` token to match both the `$Yes1` and `$Yes2` rules. Creating duplicate tokens can also result from merging multiple grammars into a single grammar.

**Use SISR tags for operations**

Semantic Interpretation for Speech Recognition (SISR) is a [W3C](http://www.w3.org) standard for defining a result for an utterance within an SRGS grammar. The result is the data that you want Interaction Speech Recognition to return, regardless if callers said different words within the rule or one or more extra words.

The following examples display use of SISR tags for operations.

**ABNF example of SISR tag**

```abnf
#ABNF 1.0;
language en-us;
mode voice;
tag-format <semantics/1.0>;
root $Operator;

$Operator =
[Please] [transfer [me] [to] | call | dial] [an] [the] operator
{out.action="transfer"; out.user="operator"};
```

**GrXML example of SISR tag**

```xml
<?xml version="1.0"?>
<grammar xmlns="http://www.w3.org/2001/06/grammar"
xml:lang="en-US"
root="Operator"
mode="voice">

<rule id="Operator">
  <item repeat="0-1">Please</item>
  <one-of>
    <item repeat="0-1">transfer</item>
    <item>
      <one-of>
        <item repeat="0-1">me</item>
        <item repeat="0-1">to</item>
      </one-of>
      <item>call</item>
    </item>
  </one-of>
</rule>
```

Interaction Speech Recognition Technical Reference
In these example grammars, the `Operator` rule allows many optional words to be present in a request to speak with an operator:

- "Please transfer me to an operator."
- "Operator."
- "Call an operator."
- "Dial the operator."

All of these utterances result in Interaction Speech Recognition returning data that is defined and populated in the last line of the example defining the rule variable, its properties (`action` and `user`), and its assigned values ("transfer" and "operator"). The receiving system can then act based on the data that Interaction Speech Recognition returns. You can also define rule variables, properties, and default values within a separate rule in the grammar and change the values within other rules in the same grammar.

For more information about SISR with examples in ABNF and GrXML, see [http://www.w3.org/TR/semantic-interpretation/](http://www.w3.org/TR/semantic-interpretation/).

### Use grammar weights and probabilities

SRGS grammars support the usage of *weights* to specify the preference of matching a recognition of one response over another. For example, if your grammar prompted callers to speak the city from which they are calling, those city names may exist in multiple states, regions, or countries. By assigning weights to multiple instances of a word or phrase (`token`), you are specifying that one of them is more likely to be spoken over another. If you do not assign weight values to multiple instances of a word or phrase within a rule, all definitions of that word or phrase share the same likelihood of occurrence.

**Important!**

You should only add weights in your grammars if tuning efforts have provided evidence that they are necessary. Entering estimations of weights can result in decreased recognition confidence levels of accurate responses.

The following examples provide recognitions for the city of "Springfield". In these examples, "Springfield" could indicate more than one state. Using weights in your grammar enables you to prefer "Springfield, Missouri" over "Springfield, Illinois" as a match to "Springfield".

#### ABNF example of weight

```abnf
#ABNF 1.0;
language en-us;
mode voice;
tag-format <semantics/1.0>;
root $Cities;

$Cities =
  (/2.0/ springfield [missouri] {out.city="Springfield"; out.state="Missouri"}) |
  (/1.2/ springfield [illinois] {out.city="Springfield"; out.state="Illinois"}) |
  dallas [texas] {out.city="Dallas"; out.state="Texas"};
```

#### GrXML example of weight

```xml
<?xml version="1.0"?>
```
Interaction Speech Recognition Technical Reference

Use filler rules or garbage rules to catch unimportant words in responses

In SRGS grammars, you use *filler* rules or *garbage* rules to separate the unimportant, contextual words and phrases from the important words and phrases (*tokens*) that result in specific data and actions.

**Filler rules**

Filler rules in grammars are when you define the optional words and phrases that a caller might say. Since you define common unimportant words in filler rules, the confidence level of the recognition is usually higher. However, the larger a grammar grows in size, the lower the confidence levels might descend without further tuning. If you want to use filler rules in grammars, ensure that you include only those unimportant rules that a majority of IVR callers will use. Defining filler rules with unimportant words that callers say very infrequently or not at all impacts the efficiency and accuracy of speech recognition.

**Garbage rules**

Garbage rules are a part of the SRGS *special rules*—along with NULL and VOID—in the SRGS specification. Garbage rules use a single entity to ignore unimportant words that may or may not occur within a grammar rule. While you can use garbage rules to account for a wide number of possible responses, garbage rules generally decrease speech recognition confidence levels as it is more difficult for the speech engine to discern what speech does not match a defined token.

**Important!**

While filler rules and garbage rules can provide benefits, Interactive Intelligence recommends that you use these types of rules sparingly. For filler rules, define only unimportant words and phrases that occur frequently in responses so as to not needlessly inflate grammar sizes. For garbage rules, do not use them as filters for large numbers of unimportant words as recognition confidence levels will decrease.

The following grammar examples define filler rules or garbage rules to use in the rule for recognizing a city of departure, as when reserving passage on a commercial flight. The important part of the grammar is the city of departure. The filler rules and garbage rules are for any other commonly-spoken words that are unimportant in the response, as a caller might say in the following examples:

- "I would like to depart from *Springfield, Missouri*, please."
• "I'm leaving from Springfield."
• "I am departing from Springfield."
• "Leave Springfield."

ABNF example of filler rules

```
#ABNF 1.0;
language en-us;
mode voice;
tag-format <semantics/1.0>;
root $DepartureCity;

$DepartureCity =
  $PreFiller
  (springfield [missouri] {out.city="Springfield"; out.state="Missouri"} |
dallas [texas] {out.city="Dallas"; out.state="Texas"})
  $PostFiller;

$PreFiller =
  [ ([i would | id] like to (leave | depart) from) | ([i am | im](leaving | departing) from) ];

$PostFiller =
  [ please ];
```

GrXML example of filler rules

```
<?xml version="1.0"?>
<grammar xmlns="http://www.w3.org/2001/06/grammar"
xmllang="en-US"
root="cities"
mode="voice">
  <rule id="cities">
    <ruleref uri="#filler"/>
    <one-of>
      <item>
        Springfield
        <item repeat="0-1">Missouri</item>
        <tag>out.city="Springfield";out.state="Missouri";</tag>
      </item>
      <item>
        Dallas
        <item repeat="0-1">Texas</item>
        <tag>out.city="Dallas";out.state="Texas";</tag>
      </item>
    </one-of>
    <ruleref uri="#filler"/>
  </rule>
  <rule id="filler">
    <one-of>
      <item repeat="0-1">I would</item>
      <item repeat="0-1">Id</item>
      <item repeat="0-1">I am</item>
      <item repeat="0-1">Im</item>
    </one-of>
    <item repeat="0-1">like to (leave | depart) from</item>
    ([i am | im](leaving | departing) from)
  </rule>
</grammar>
```
 Interaction Speech Recognition Technical Reference

ABNF example of garbage rules

```abnf
#ABNF 1.0;
language en-us;
mode voice;
tag-format <semantics/1.0>;
root $DepartureCity;

$DepartureCity =
  $GARBAGE
  ((springfield [missouri]) {out.city="Springfield"; out.state="Missouri";}) |
  (dallas [texas] {out.city="Dallas"; out.state="Texas";})
$GARBAGE;
```

GrXML example of garbage rules

```xml
<?xml version="1.0"?>
<grammar xmlns="http://www.w3.org/2001/06/grammar"
  xml:lang="en-US"
  root="cities"
  mode="voice">
  <rule id="cities">
    <ruleref special="GARBAGE"/>
    <one-of>
      <item>
        Springfield
        <item repeat="0-1">Missouri</item>
        <tag>out.city="Springfield"; out.state="Missouri";</tag>
      </item>
      <item>
        Dallas
        <item repeat="0-1">Texas</item>
        <tag>out.city="Dallas"; out.state="Texas";</tag>
      </item>
    </one-of>
    <ruleref special="GARBAGE"/>
  </rule>
</grammar>
```

Reference builtin grammars instead of recreating functionality

Builtin grammars provide speech recognition in an Interactive Voice Response (IVR) application for basic responses; however, you do not have control over the data that builtin grammars return. For this purpose, you may need to create your own custom grammars to provide that information, such as specifying account numbers, identification numbers, or other specific information in a certain format with a set amount of data. In custom grammars, you can refer to builtin grammars that already provide the speech recognition functionality that is required.

For example, if you wanted callers to speak the number of animals seen during a recent hike, you could create a custom grammar to specify that it uses the builtin number grammar appended with any entries that you define in the custom grammar. Instead of recreating the functionality of number recognition in the custom grammar, you can reference the `number` builtin grammar, as displayed in the following example grammar:
ABNF builtin grammar reference example

```abnf
#ABNF 1.0;
language en-us;
mode voice;
tag-format <semantics/1.0>;
root $wildlife;

$wildlife=
  $number $type {out.number=rules.number; out.type=rules.type;};

$type=
  ({bird | birds} {out="birds";} |
   fish {out="fish";} |
   {mammal | mammals} {out="mammals";} |
   {animal | animals} {out="animals";});

$number=
  <$<builtin:grammar/number>>;
```

GrXML builtin grammar reference example

```xml
<?xml version="1.0"?>
<grammar xmlns="http://www.w3.org/2001/06/grammar"
xmllang="en-US"
root="wildlife"
mode="voice">

    <rule id="wildlife">
        <ruleref uri="#number" />
        <ruleref uri="#type" />
        <tag>out.number=rules.number;</tag>
        <tag>out.type=rules.type;</tag>
    </rule>

    <rule id="type">
        <one-of>
            <item>
                <one-of>
                    <item>bird</item>
                    <item>birds</item>
                </one-of>
                <tag>out="birds";</tag>
            </item>
            <item>fish <tag>out="fish";</tag></item>
            <item>
                <one-of>
                    <item>mammal</item>
                    <item>mammals</item>
                </one-of>
                <tag>out="mammals";</tag>
            </item>
            <item>
                <one-of>
                    <item>animal</item>
                    <item>animals</item>
                </one-of>
                <tag>out="animals";</tag>
            </item>
        </one-of>
    </rule>
</grammar>
```
Do not try to address all possible responses

When developing custom grammars, avoid including all of the possible words that a caller could speak. Instead, limit the words to the ones that callers most commonly use. Small grammars provide more accuracy than large grammars. Grammars that contain all possible words do not provide any benefit if the majority of those words are rarely or never spoken.

The following sections provide some guidelines in determining what words to include in your custom grammars.

Ensure that prompts are guiding the caller to a limited set of responses

While this task is not related to the actual development of grammars, it is very relevant as prompts influence what responses callers will speak. For example, if you want to know if a caller is calling from a business or a residence (a simple choice), the prompt should provide those choices, as displayed in the following example:

"Are you calling from a residence or a business?"

For this prompt, callers will likely response with "residence" or "business".

If the prompt does not provide choices, the responses could vary drastically:

"From where are you calling?"

Does this prompt expect a geographical location, a social location (home, office), or some other information? This vague prompt can cause any number of responses:

- "New York"
- "In my car"
- "From my house"
- "ABC Company"
- "From my couch"
- "My cell phone"
- "The corner of 10th and Main"

While you had an idea of the responses you wanted, failure to consider how the prompt could be perceived by callers could require large, complex grammars to account for the various types of responses that they provide. Even with those large, complex grammars, you still may not receive the simple information that you seek: business or residence.
Avoid defining all possible synonyms

Synonyms are different words that have the same or similar meaning. Using the example from the previous section where the expected responses are "residence" or "business", do not enter all of the possible words that have similar meaning as the expected responses. The following table displays some examples of synonyms for "residence" or "business":

<table>
<thead>
<tr>
<th>&quot;Residence&quot; synonyms</th>
<th>&quot;Business&quot; synonyms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home</td>
<td>Office</td>
</tr>
<tr>
<td>House</td>
<td>Headquarters</td>
</tr>
<tr>
<td>Apartment</td>
<td>Corporation</td>
</tr>
<tr>
<td>Condominium</td>
<td>Factory</td>
</tr>
<tr>
<td>Condo</td>
<td>Store</td>
</tr>
<tr>
<td>Townhouse</td>
<td>Restaurant</td>
</tr>
<tr>
<td>Dwelling</td>
<td>Firm</td>
</tr>
<tr>
<td>Domicile</td>
<td>Hospital</td>
</tr>
<tr>
<td>Estate</td>
<td>School</td>
</tr>
</tbody>
</table>

If you are not able to provide choices in your prompts, your grammars should only contain the most common responses to the prompt. Do not include words that are rarely spoken in the responses to a prompt unless you are tuning grammars and are testing optional words to increase accuracy.

Focus on expected responses to the prompt

Your custom grammars should contain words that relate only to a prompt. For example, if a caller has navigated to a prompt of "Please say the digits of your account number," your grammar should not attempt to cover the rare instance of a caller saying, "I just want to talk to a live person." If you enable the grammar to recognize these words in responses to this prompt, you would likely need to include those words in all responses to all prompts. This mistake grows your grammars needlessly, which results in lower overall recognition accuracy.

The purpose of your Interactive Voice Response (IVR) system is to enable callers to navigate menus by speaking responses to prompts, not to simulate an artificial intelligence that can process and act on any speech.

Identify and fix problems after deployment

After an initial deployment of Interaction Speech Recognition, test your system to ensure that Interaction Speech Recognition recognizes caller responses correctly. After testing, you may need to improve your grammars to account for the following items:

- Unexpected responses, including commonly occurring optional words that you did not include originally
  - If the unexpected responses are a new interpretation that you want to act upon, include those responses in new rules.
  - If the unexpected responses are words that you did not include for an existing rule, add those responses as additional or optional words in that rule.

- Variations or alternate pronunciations in caller responses to a prompt
  - Ensure that the variation or alternate pronunciation is a common occurrence in caller responses. Do not add to the grammar for one instance of the problem.
  - Add the variation or alternate pronunciation to the rules that already exist. Do not create new rules for the variation that would return the same semantic interpretation as an existing rule.
• Misrecognized or missed responses
  — Ensure that you are allowing enough time for the caller to provide a complete response.
  — Ensure that the call had enough audio clarity to facilitate recognition of the response. A call with low audio quality significantly impacts speech recognition.
  — Adjust your systems that receive interpretations to not act if the confidence level of an interpretation is below a certain value.
  — Analyze your grammars to determine if the rules could be modified to increase confidence levels, such as making a word of a phrase optional instead of required. For example, in a grammar in ABNF format, change `technical support representative` to `[technical] support representative` or `[technical] support ([rep] | [representative] | [person]).`

• Identify and correct ambiguity in grammars
  For information on ambiguity, see Remove ambiguity.

• Poor audio quality issues based on listening to recordings
  — Ensure that no excessive noise—background noise in the caller's environment or noise introduced in the lines of communication—is present. Excessive noise can cause missed responses, inaccurate recognitions, and low confidence levels.
  — Test your IVR grammars using codecs with higher bandwidths, such as G.711. Low-bandwidth codecs reduce dynamic audio ranges, which can result in decreased recognitions and lower confidence levels.
  — Disable silence suppression. Silence suppression is a telephony feature where audio is not transmitted for one party while the other party is speaking or playing audio. Silence suppression could cause clipping of audio where a party speaks and the feature is not deactivated quickly. This situation could result in the first portion of a response to not be received and analyzed by Interaction Speech Recognition.
Testing grammars

When you create a grammar, you should test it to ensure that it is valid and that it is functioning as intended. The following sections provide information on how to test grammars and the data that can help you determine its level of functionality.

Test grammar validity

You can test grammars for validity with the following command line program on the Customer Interaction Center server:

D:\I3\IC\Server\RecoGrammarValidatorU.exe (default install location)

To specify a grammar for the program to parse, supply the full path as displayed in the following example:

D:\I3\IC\Server\RecoGrammarValidatorU.exe D:\I3\IC\Resources\<file name>.<extension>

If the program parses the grammar successfully, the following message appears in the program output in the command prompt window:

Grammar successfully parsed:

Note:
The *RecoGrammarValidatorU.exe* program does not test for duplicated tokens or ambiguity.

Analyze functionality

When you deploy a grammar for testing or usage, you should analyze its performance to ensure that it is performing as needed. The following list provides some guidelines to assist you in analyzing grammars:

- Collect and analyze logs and recordings each day.
- Create a process for analyzing calls logs that is repeatable. A consistent process ensures that the data you collect is valid.
- Create a spreadsheet or database for recording and tracking the speech recognition data. You should, at a minimum, track the following items:
  - Call ID and time
  - The grammars used in each call
  - Utterance recording
  - Utterance transcription
  - Utterance decoded
  - Recognition properties, such as confidence level
  - Recognition results
- Analyze a significant number of calls and identify trends before making any changes.
Use custom grammars with Interaction Speech Recognition

You can use custom grammars with Interaction Speech Recognition through multiple methods:

- Preload the grammar through Interaction Administrator
- The Reco Register Grammar tool in custom handlers through Interaction Designer
- VoiceXML documents

Interactive Intelligence recommends that you store your custom grammars in the Resources directory of the Customer Interaction Center server. By default, this directory uses the following path:

D:\I3\IC\Resources
Troubleshooting

If you encounter problems using Interaction Speech Recognition, the following information can direct you to determining the cause.

Windows Event Log contains entries for invalid grammars

Customer Interaction Center creates an entry in the Windows Event Log if it preloads an invalid grammar.
To ensure that a preloaded grammar is valid, run the grammar validation tool.
For more information, see Test grammar validity.

Dial by Name not working

Verify that the Disable Speech Recognition check box is disabled in the Interaction Attendant profile.

Audio problems with Interaction Speech Recognition

If you experience audio problems with Interaction Speech Recognition, you can enable diagnostic recordings that Interactive Intelligence support representatives can review to assist in determining the cause of the problem.

There are three Interaction Media Server properties that you can use to capture diagnostic recordings for Interaction Speech Recognition:

- **RecognizerDiagnosticRecording** – This property will create .wav files for each interaction. These files will only be available on the Media Server – they will not be fetched by the CIC Server. This property can only be set through the Media Server Web interface.

- **recognizerDiagnosticLog** – This property will create .wav and .xml files containing audio data along with the corresponding events. The CIC server will then fetch these files and store them in its media storage location. This property can only be set through the ASR Engine Configuration dialog in Interaction Administrator

- **AsrDiagnosticRecording** – This property will create three .wav files for echo analysis. These files will only be available on the Media Server – they will not be fetched by the CIC Server. This property can only be set through the Media Server Web interface.

  **Caution!**

  Enable the **AsrDiagnosticRecording** property only when directed to do so by an Interactive Intelligence Support representative. ASR Diagnostic Recordings have a large impact on the performance of Interaction Media Server. These impacts reduce the number of resources that Interaction Media Server can use to service both the number of calls and the features used in those calls.

  For more information about these properties, see the "Interaction Media Server Config-Properties page" topic in Interaction Media Server Technical Reference.

Warnings on usage of diagnostic recordings and diagnostic logs for Interaction Speech Recognition

**RecognizerDiagnosticRecording** and **AsrDiagnosticRecording**

- Diagnostic recordings for Interaction Speech Recognition are large .wav files, which can exhaust data storage capabilities if diagnostic recordings are created indefinitely. Ensure that you create diagnostic
recordings for a brief period of time. You should enable diagnostic recordings only when directed to do so by an Interactive Intelligence support representative.

- If you enable diagnostic recordings for Interaction Speech Recognition through the Interaction Media Server web interface, the resulting .wav files are created on the data storage device of Interaction Media Server and will remain there until you physically remove them.

recognizerDiagnosticLog

- If you enable diagnostic logs for Interaction Speech Recognition through the Interaction Speech Recognition ASR Engine Configuration dialog box in Interaction Administrator, every Interaction Media Server connected to the Customer Interaction Center server creates diagnostic logs (.wav and .xml files). Customer Interaction Center then moves all diagnostic log files to itself. Ensure that you do not continue creating diagnostic logs through this method for an extended period of time as it could result in significant network bandwidth usage and exhaustion of the data storage capabilities of the Customer Interaction Center server.

Enable diagnostic recordings for Interaction Speech Recognition through Interaction Media Server

1. Open a web browser window and navigate to the Uniform Resource Locator (URL) address of the Interaction Media Server on which you want to enable and create diagnostic recordings.
2. When prompted, supply an administrative user name and password.
   The Status-About page of the Interaction Media Server web interface appears.
3. In the upper right corner, select the Settings icon.
4. On the left side of the page, select the Properties tab.
5. In the unlabeled text box at the bottom of the left column, enter the following text:
   RecognizerDiagnosticRecording
6. To the right of the text box, select the Add button.
   The RecognizerDiagnosticRecording property is added to the list of displayed properties.
7. To the right of the RecognizerDiagnosticRecording property, select true in the list box.
8. At the bottom of the page, select the Apply button.

Enable diagnostic logging for Interaction Speech Recognition through Interaction Administrator

Caution!
Performing this procedure enables diagnostic recordings for Interaction Speech Recognition on all Interaction Media Servers that are connected to this Customer Interaction Center server. Ensure that you have the necessary network bandwidth to accommodate the transfer of all diagnostic recording files from each Interaction Media Server to the Customer Interaction Center server. Ensure that the Customer Interaction Center server has enough free data storage space to accommodate all of the diagnostic recording files from all Interaction Media Servers.

1. Open Interaction Administrator.
2. If prompted, supply an administrative user name and password.
3. In the left pane of the Interaction Administrator window, expand the System Configuration container.
4. In the System Configuration container, expand the Recognition container.
5. In the Recognition container, select Interaction Speech Recognition.
6. In the right pane, double-click the **Configuration** item. The **ASR Engine Configuration** dialog box appears.

7. In the **ASR Engine Configuration** dialog box, select the **Properties** tab.

8. On the **Properties** tab, select the **Add** button. The **Add Property** dialog box appears.

   ![Add Property dialog box](image1)

9. In the Name box, enter the following text: **recognizerDiagnosticLog**

10. Select the **Add Value** button. The **Add Property Value** dialog box appears.

   ![Add Property Value dialog box](image2)

11. In the **Value** box, enter **true** and select the **OK** button.

12. In the **Add Property** dialog box, select the **OK** button.

    The **recognizerDiagnosticLog** property is displayed in the **Properties** tab.

13. In the **ASR Engine Configuration** dialog box, select the **OK** button.
Limitations

Interaction Speech Recognition currently has the following limitations:

- Limited support of the Semantic Interpretation for Speech Recognition (SISR) standard
- Diagnostic recordings for Interaction Speech Recognition are not transferred to the Customer Interaction Center server if the process of terminating of the recording exceeds 5 seconds.
- Interaction Speech Recognition does not currently support hotword barge-in. A hotword barge-in is an utterance spoken by a caller during the playing of a prompt that matches a defined word or phrase in a loaded grammar. When this utterance occurs, the current prompt stops and Interaction Speech Recognition returns the data associated with the match of the utterance.
# Change Log

The following changes have been made to this document since release.

<table>
<thead>
<tr>
<th>Date</th>
<th>Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>September 24, 2013</td>
<td>Initial Release</td>
</tr>
<tr>
<td>October 4, 2013</td>
<td>Removed inaccurate admonishment regarding user credentials</td>
</tr>
<tr>
<td>August 6, 2014</td>
<td>Updated documentation to reflect changes required in the transition from version 4.0 SU# to CIC 2015 R1, such as updates to product version numbers, system requirements, installation procedures, references to Interactive Intelligence Product Information site URLs, and copyright and trademark information.</td>
</tr>
<tr>
<td>December 18, 2014</td>
<td>Added information about the audio detection sensitivity property.</td>
</tr>
<tr>
<td>April 20, 2015</td>
<td>• Added information about the recognizerDiagnosticLog property to the Troubleshooting section.</td>
</tr>
<tr>
<td></td>
<td>• Updated Copyright and Trademarks page.</td>
</tr>
<tr>
<td></td>
<td>• Updated the document to reflect the CIC 2015R3 version.</td>
</tr>
<tr>
<td>July 1, 2015</td>
<td>• Updated cover page to reflect new color scheme and logo.</td>
</tr>
<tr>
<td></td>
<td>• Updated copyright and trademark information.</td>
</tr>
<tr>
<td></td>
<td>• Updated the document to reflect the CIC 2015R4 version.</td>
</tr>
<tr>
<td>October 9, 2015</td>
<td>Updated the document to reflect the CIC 2016 R1 version.</td>
</tr>
<tr>
<td>February 4, 2016</td>
<td>• Added a link to the CIC Documentation Library at Help.inin.com.</td>
</tr>
<tr>
<td></td>
<td>• Updated copyright and trademark information</td>
</tr>
<tr>
<td></td>
<td>• Updated the document to reflect the CIC 2016 R2 version.</td>
</tr>
<tr>
<td>June 2, 2016</td>
<td>• Added Europe – French (fr-FR) as a supported language in Interaction Speech Recognition requirements.</td>
</tr>
<tr>
<td></td>
<td>• Minor edits</td>
</tr>
<tr>
<td>December 6, 2016</td>
<td>Added Europe – Spanish (es-ES) as a supported language in Interaction Speech Recognition requirements.</td>
</tr>
<tr>
<td>May 24, 2017</td>
<td>Added language support for Mandarin Chinese and Italian.</td>
</tr>
</tbody>
</table>