



A MITEL
PRODUCT
GUIDE

MiContact Center Enterprise

Speech Recognition Applications Script Manager User Guide

Release 9.8

Document Version 1.0

August 2025

Notices

The information contained in this document is believed to be accurate in all respects but is not warranted by **Mitel Networks™ Corporation (MITEL®)**.

The information is subject to change without notice and should not be construed in any way as a commitment by Mitel or any of its affiliates or subsidiaries. Mitel and its affiliates and subsidiaries assume no responsibility for any errors or omissions in this document. Revisions of this document or new editions of it may be issued to incorporate such changes. No part of this document can be reproduced or transmitted in any form or by any means - electronic or mechanical - for any purpose without written permission from Mitel Networks Corporation.

Trademarks

The trademarks, service marks, logos and graphics (collectively "Trademarks") appearing on Mitel's Internet sites or in its publications are registered and unregistered trademarks of Mitel Networks Corporation (MNC) or its subsidiaries (collectively "Mitel") or others. Use of the Trademarks is prohibited without the express consent from Mitel. Please contact our legal department at legal@mitel.com for additional information. For a list of the worldwide Mitel Networks Corporation registered trademarks, please refer to the website: <http://www.mitel.com/trademarks>.

®,™ Trademark of Mitel Networks Corporation

© Copyright 2025, Mitel Networks Corporation All rights reserved

INTRODUCTION

This document contains the tutorials to create scripts using components provided in the Automatic Speech Recognition (ASR) component library. Some components in the call and media control library are also covered in this document. The tutorial provides a sample script that uses the voice recognition components in the ASR component library.

BEFORE USING THE TUTORIAL

The sample scripts used in this document is installed on the MiCC Enterprise server, in the <install dir>\ScriptManager\SampleScripts folder.

To use ASR, the Nuance Speech Server must be installed. In addition, the licenses for ASR grammar (for natural language) and licenses for ASR language need to be included. The grammar files YesNo.grxml and upolice.grxml need to be copied to the Nuance Speech Server (e.g. to the C:\grammars\grxml folder).

GRAMMARS

YESNO.GRXML

```
<?xml version="1.0" encoding="utf-8"?>
<grammar xmlns=http://www.w3.org/2001/06/grammar
  xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
  xsi:schemaLocation=http://www.w3.org/2001/06/grammar
  http://www.w3.org/TR/speech-grammar/grammar.xsd
  xml:lang="en-US" version="1.0"
  root="YesNo"
  mode="voice">

  <rule id="YesNo" scope="public">
    <item>
      <one-of>
        <item> <ruleref uri="#Yes"/> <tag> out.color=rules.Yes; </tag> </item>
        <item> <ruleref uri="#No"/> <tag> out.color=rules.No; </tag> </item>
      </one-of>
    </item>
  </rule>

  <rule id="Yes" scope="public">
    <tag> out.answer="yes"; </tag>
    <one-of>
      <item> yes </item>
      <item> yep </item>
      <item> sure </item>
      <item> yea </item>
    </one-of>
  </rule>

  <rule id="No" scope="public">
    <tag> out.answer="no"; </tag>
```

```
<one-of>
  <item> no </item>
  <item> nope </item>
  <item> nah </item>
</one-of>
</rule>

</grammar>
```

UPOLICE.GRXML

```
<?xml version="1.0" encoding="UTF-8"?>
<grammar xmlns=http://www.w3.org/2001/06/grammar
  xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
  xsi:schemaLocation=http://www.w3.org/2001/06/grammar
  http://www.w3.org/TR/speech-grammar/grammar.xsd
  xml:lang="en-US" version="1.0"
  root="INCIDENT"
  mode="voice">

  <rule id="INCIDENT" scope="public">
    <item>
      <one-of>
        <item> <ruleref uri="#theft"/> <tag> out.color=rules.theft; </tag> </item>
        <item> <ruleref uri="#murder"/> <tag> out.color=rules.murder; </tag> </item>
      </one-of>
    </item>
  </rule>

  <rule id="theft" scope="public">
    <tag> out.answer="theft"; </tag>
    <one-of>
      <item> theft </item>
      <item> burglary </item>
      <item> robbery </item>
    </one-of>
  </rule>

  <rule id="murder" scope="public">
    <tag> out.answer="murder"; </tag>
    <one-of>
      <item> murder </item>
      <item> homicide </item>
    </one-of>
  </rule>

</grammar>
```

TUTORIAL RECOGNITION

The following is a tutorial handling Recognition.

RECOGNITION.MFD

The script presented in Figure 1 demonstrates the reception of an incoming voice call, and playing of a voice message to prompt the user for speech input and branches to different block based on the user input. This tutorial emphasizes the use of the **AllocateResources**, **Recognize** and **NLBranch** components. Double click on **Recognition.mfd** to open the following script:

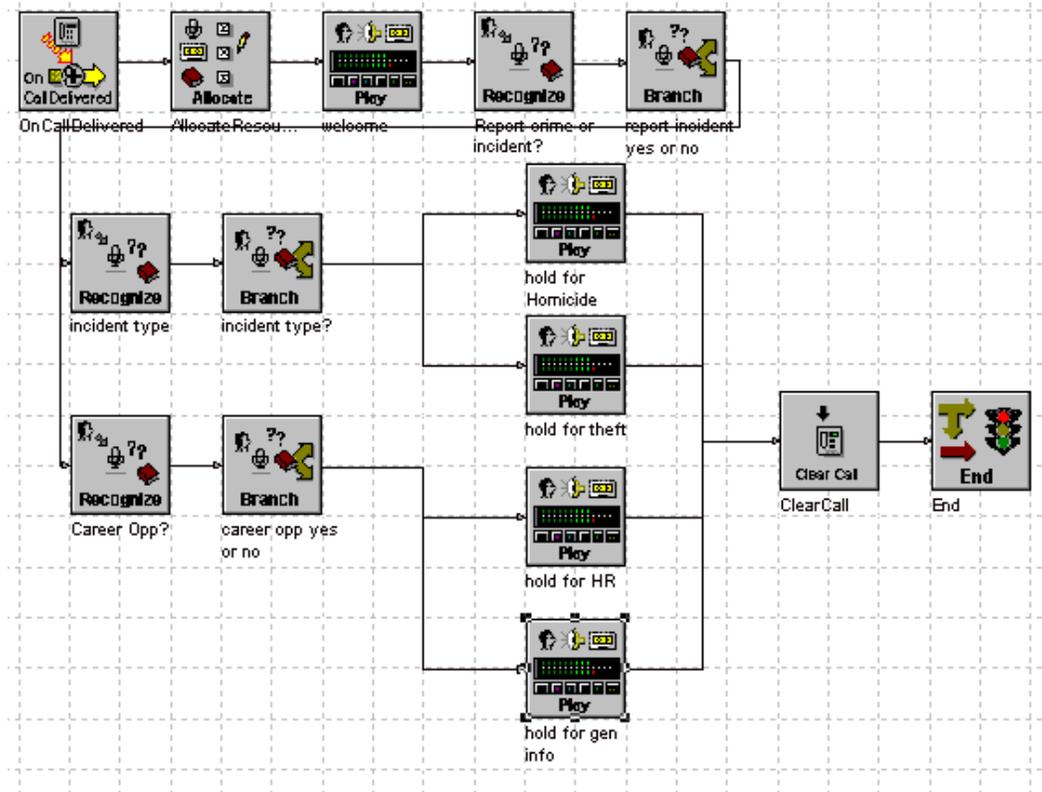


Figure 1: Recognition sample script

This script provides incoming call handling to a police department using voice recognition. This script:

- Receives an incoming call
- Allocates play and recognition resources
- Prompts the user to select Yes or No
- State the type of incident to be reported

The first block placed in the Event-Driven section is the **OnCallDelivered** block. The second block is the **AllocateResource** block, where resources are reserved to handle the call.

ALLOCATERESOURCE

See Figure 2 for an overview of the settings tab of the Allocate Resource Properties window.

For normal incoming call handling, select **Allocate Resource and Answer Call**, see Figure 2.

Select which **Resources** to add, see Figure 2. For playback of a voice prompt, a player resource must be added. For tone detection, a tone detector resource must be selected. For voice recognition, an automatic speech recognition resource must be selected.

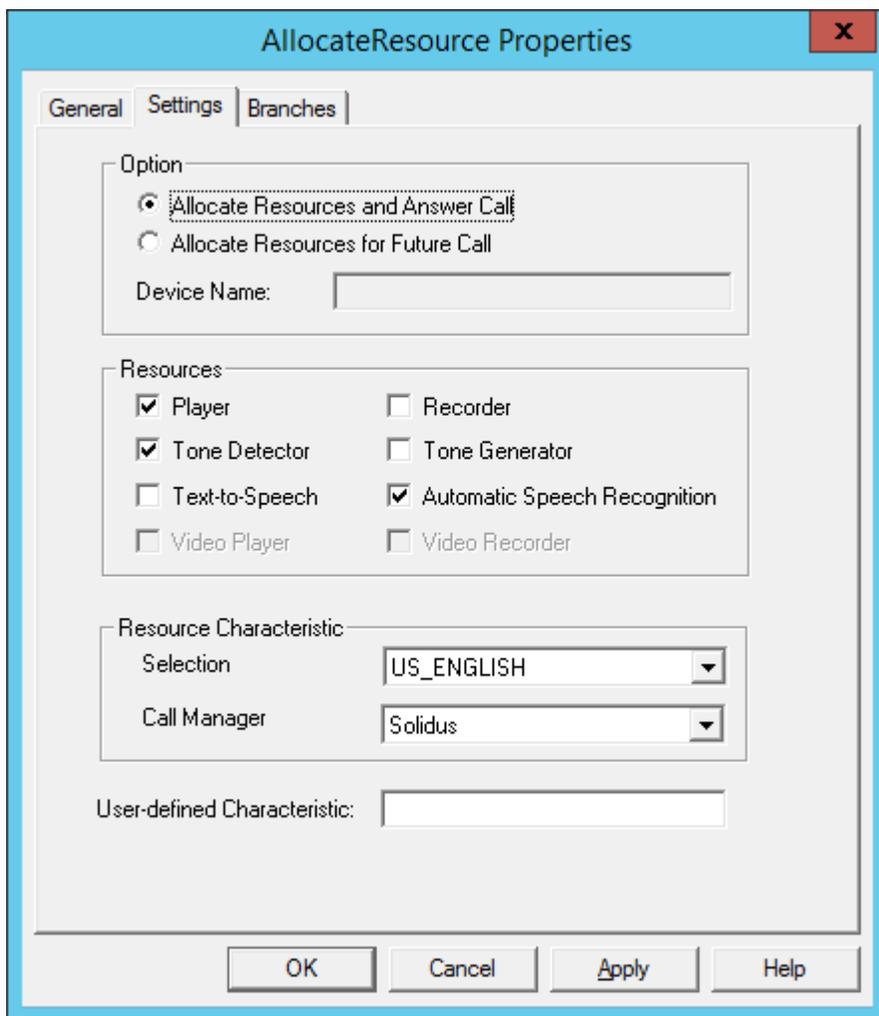


Figure 2: Settings tab of the Allocate Resource Properties dialog

RECOGNIZE

The settings tab of the Recognize Properties dialog is shown in Figure 3.

Error Condition	No. of Retries	1st Message	2nd Message	3rd Message
No Response:	2	@Msg3		
Not Recognized:	2	@Msg4		
Max. Speech:	2	@Msg4		
Unexpected Digit:	0			
Recognition Too Slow:	2	@Msg4		
Speech Too Early:	2	@Msg4		

Figure 3: Settings tab of Recognize Properties

Messages

The *Recognize* block plays the messages specified in the Messages input. Optionally, the Messages can be configurable variables. In this example, the message variable Msg13, which contains message 313, will be played in this block.

Confidence level

The confidence level is a number ranging from 0 to 100. The High and Low confidence settings will determine the branch result of the recognition. If the recognition score comes back within the values between the low and high confidence level, the branch result will take on the "Success". If the recognition score comes back higher than the high confidence level, the branch result will take

on the “Success with High Confidence”. If the recognition score comes back lower than the low confidence level, the branch result will take on the “Recognition failed”.

Vocabulary

The vocabulary input specifies the name and location of the grammar. For example, "C:\grammars\grxml\YesNo.grxml" is used because it is the name of the grammar file in the example above.

Barge in

The Barge in option changes the behavior of the block. It enables the caller to interrupt the play message either by saying something or by entering a digit.

Logging

The Logging option in the ASR Recognize block is used to enable/disable ASR recording. If Logging is enabled in a Recognize block, ASR recording will work only with Nuance 8.5.

If Nuance 9.0 is used, ASR recording is not supported in a normal Recognize block. ASR recording is supported in a VXML Execute block, provided that the “recordutterance” property is set to “true” for the specific VoiceXML document being interpreted.

Flush digits

If the Flush digits option is selected, any digit entered before this block will be discarded. If the Flush digits option is not set and a digit is entered before the block execution, none of the initial play messages will be played. When a digit that is entered ends the execution of the recognize block, the branch takes on the “Unexpected digits”. Then the entered digit can be retrieved by using the GetDigits block.

Max. Results

The Max. Results input specifies the maximum number of top choices that is to be stored in the recognized result.

Max. Retries

The Max. Retries input specifies maximum number of retries on recognition of Error Conditions in all failure cases.

Recognize result

The recognize result input is to specify an output variable of type SpeechResult object to hold the recognize result. In this case the variable @result will hold the recognize result.

No. of Result

The No. of Result input is to specify an output variable of type integer to hold the number of recognized records that are stored in the recognize result.

Error Condition

The Error Condition input is to specify the appropriate messages to remind the caller to retry. Also, it is used to specify the number of retries time.

NLBRANCH

The NLBranch block searches for an expected slot name and value in a recognize result object, see Figure 4.

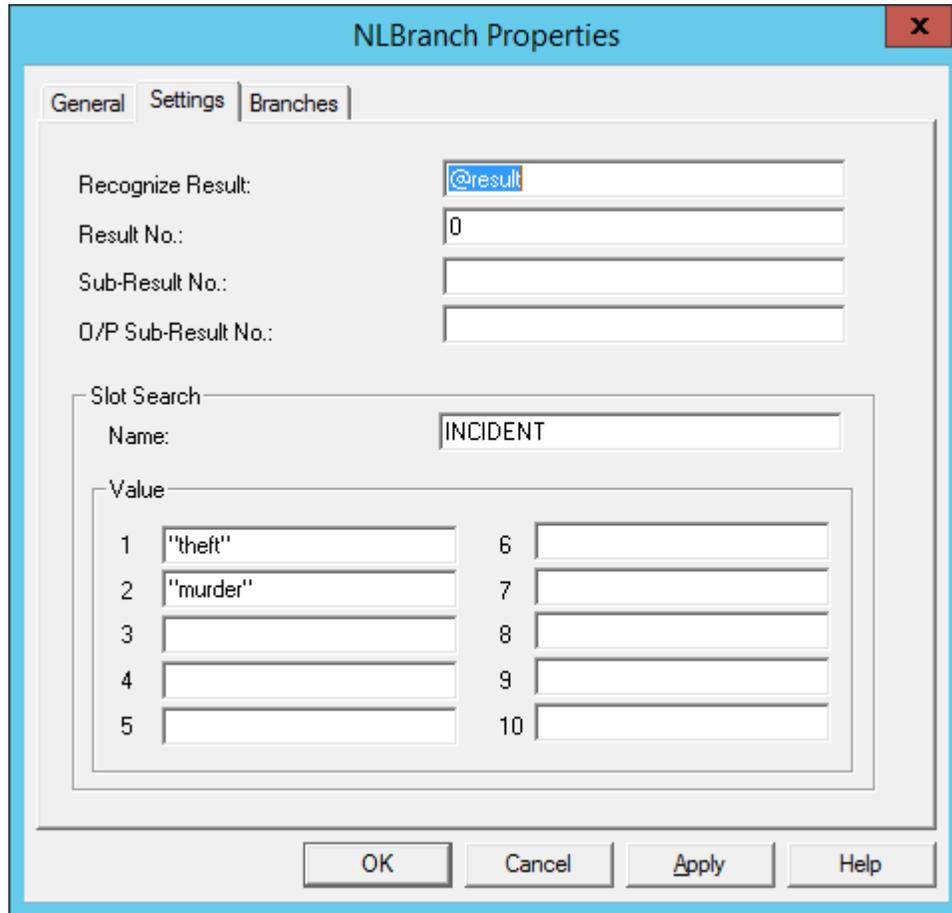


Figure 4: Settings tab of NL Branch Properties dialog

Recognize Result

The Recognize Result input is to specify the recognize result variable which is an output variable in the previous recognize block.

Result No

Sometimes, there can be more than one recognized record stored in the recognize object. As a result, the recognize object contains an attribute "NumOfResult" that holds the number of recognized records. The recognize records start indexing at 0. The first choice result or guess is

stored at index 0. The Result No. input in the NLBranch block indicates which recognize record the block should examine.

Sub-Result no

A recognize record can in turn contain one or more natural language interpretation indicated by slot name, slot value and a score of confidence.

Sub-Result no. input is to indicate the specific slot in the recognize record to be included in the search. If this number is left empty, all sub-results or slots in a recognize record will be examined, as in this case. The specific slot can be indicated by an index number starting from zero for the first interpretation.

Slot Search Name

The slot search name indicates the name of the slot to search for. This name is taken from the grammar definition.

Slot Value

The slot value indicates the value to search for when the slot name has been found.

The branch will take on the “search n found” path, where n is 1 to 10, if the slot search name and slot value are found. The branch will take on the “found” path if only the slot search name is found and not the slot value. The branch will take on the “Not Found” path when neither the slot search name nor the slot value is found.

VOICE PROMPTS

The voice prompts and play messages presented in Table 1 must be configured in the OAS or TAS server before running the script.

Table 1: Voice prompts and play messages

SCRIPT MANAGER VARIABLE	MESSAGE ID	SAMPLE VOICE PROMPT DESCRIPTION
Msg1	301	“Welcome to the police headquarters.”
Msg2	302	“Are you interested in a career within the police force?”
Msg3	303	“Sorry. I could not understand you.”
Msg4	304	“I could not hear you.”
Msg5	305	“Do you want to report a crime?”
Msg6	306	“You will be transferred to homicide.”
Msg7	307	“You will be transferred to theft.”
Msg3	308	“Please repeat.”

SCRIPT MANAGER VARIABLE	MESSAGE ID	SAMPLE VOICE PROMPT DESCRIPTION
Msg9	309	"Please say Murder, Homicide, Robbery, Theft or Burglary."
Msg6, Msg7	310	"Please hold."
Msg11	311	"Please hold for general information."
Msg12	312	"Please hold for an HR
Msg13	313	"What kind of crime do you want to report?"