

VoiceSaver® IVR

Interactive Voice Response
GUI Application Generator

March 20, 2019

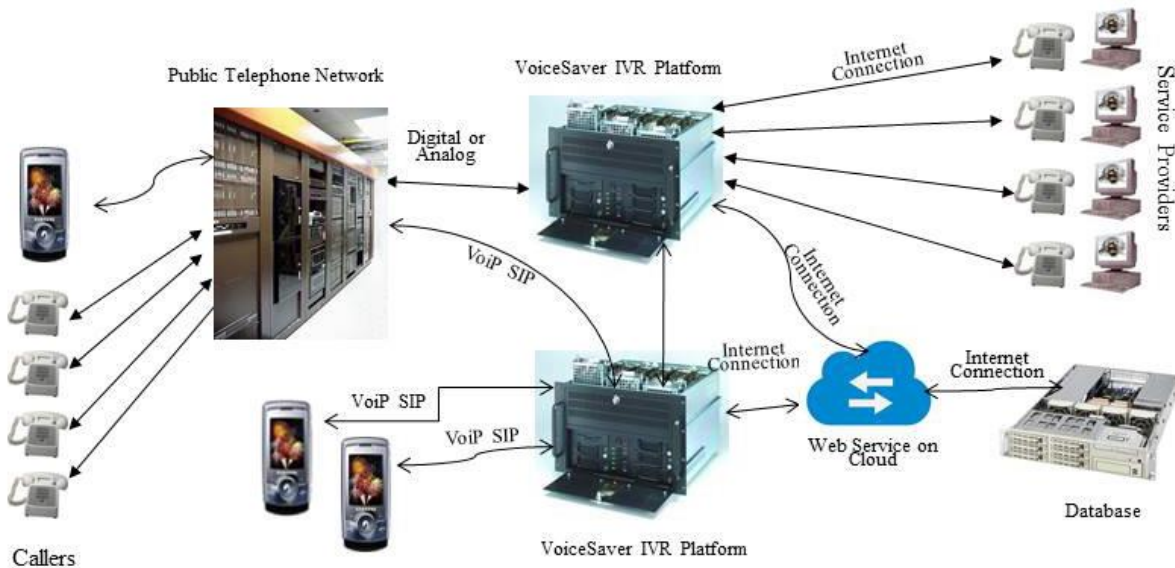
DRAFT March 20, 2019

Introduction

Interactive Voice Response (IVR) software is a technology that automates interaction with telephone callers via phone and SMS. Whether you need to dial out or send an SMS to remind a customer about an outstanding payment or upcoming appointment, provide order information, or warn them of an emergency, VoiceSaver IVR is the right solution. Our powerful software allows companies to turn their vision of an interactive call or messaging flow into reality.

The VoiceSaver® IVR saves time, minimizes caller authentication times, proactively anticipates needs, remembers caller preferences, and generates exceptional results for callers.

The following diagram shows the general topology of how VoiceSaver® IVR fits into your telephony/messaging network.



DRAFT

VoiceSaver® IVR

The VoiceSaver® IVR allows a system integrator or user to implement and make changes to a call flow in real time. Its user-friendly, drag & drop design allows changes quickly without fear of making a mistake. As a system integrator, this allows you to implement customized call flow requirements for your clients in a short period of time. This document shows you some of the capabilities of the VoiceSaver® IVR system.

VoiceSaver® IVR may be run on the following platforms:

1. Microsoft® Windows 10
2. Microsoft® Windows 2012-2016
3. Linux OS
4. Amazon Web Services (AWS)

VoiceSaver® IVR uses the following technologies:

1. Dialogic® Time Division Multiplexed (TDM) based systems.
2. Dialogic® Analog based systems.
3. Asterisk® PBX
4. Cisco® Voice Gateways
5. Twilio® network.
6. Synway - China

VoiceSaver® IVR command blocks support for the following fundamental services:

1. Receive a phone call.
2. Answer a phone call.
3. Make outbound calls.
4. Transfer a phone call to another destination.
5. Play voice file.
6. Record voice to a file.
7. Receive Dual Tone Multi-frequency (DTMF).
8. Generate DTMF
9. Receive Short Message Service (SMS)
10. Send SMS
11. Receive and email.
12. Send an email.
13. Send a message to a WhatsApp user.
14. Send a request to a web service.
15. Receive information from a web service.

VoiceSaver® IVR Command Blocks

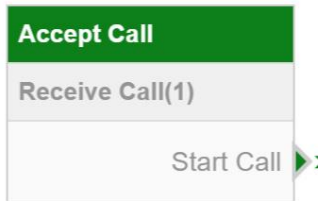
The following menu of command blocks are available to you within the VoiceSaver® IVR :



The following call flow example produced by VoiceSaver® IVR receives a call, plays the hours of operation, and disconnects:



Accept Call



The Accept Call command block is activated when the call flow receives a new telephone call:

By double clicking on the Accept Call area of the command block, the following data entry form appears:

Accept Call (1)		
	Save Block Settings	Delete Block
Block Name	<input type="text" value="Receive Call"/>	
OnHangupGoto	<input type="text" value="HANG_UP(Good Bye)"/>	
Next Block	<input type="text" value="PLAY(Hours of Operatio)"/>	

In the Block Name field, you should enter a descriptive name for this block. Then, tell the system which block to proceed to if the caller Hangs Up (OnHangupGoto) or if the caller does nothing (Next Block).

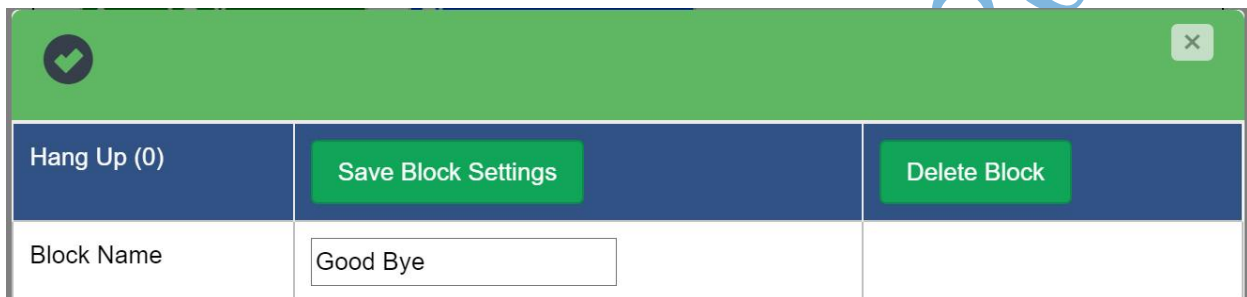
In this case, if the caller hangs up, the system will hang up and play “Good Bye.” If the caller does nothing, it will continue to play the Hours of Operation.

Hang Up



The Hang Up command block disconnects the call, stops playing any messages, stops recording, and stops any other functions that may be in progress.

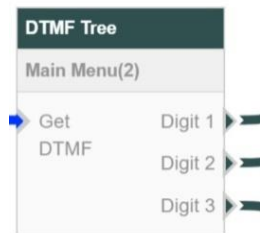
By double clicking on the Hang Up area of the command block, the following data entry form appears:



Hang Up (0)		Save Block Settings	Delete Block
Block Name	<input type="text" value="Good Bye"/>		

The form allows you to enter the Block Name. In the example above, it is set to “Good Bye”.

DTMF Tree Command Block



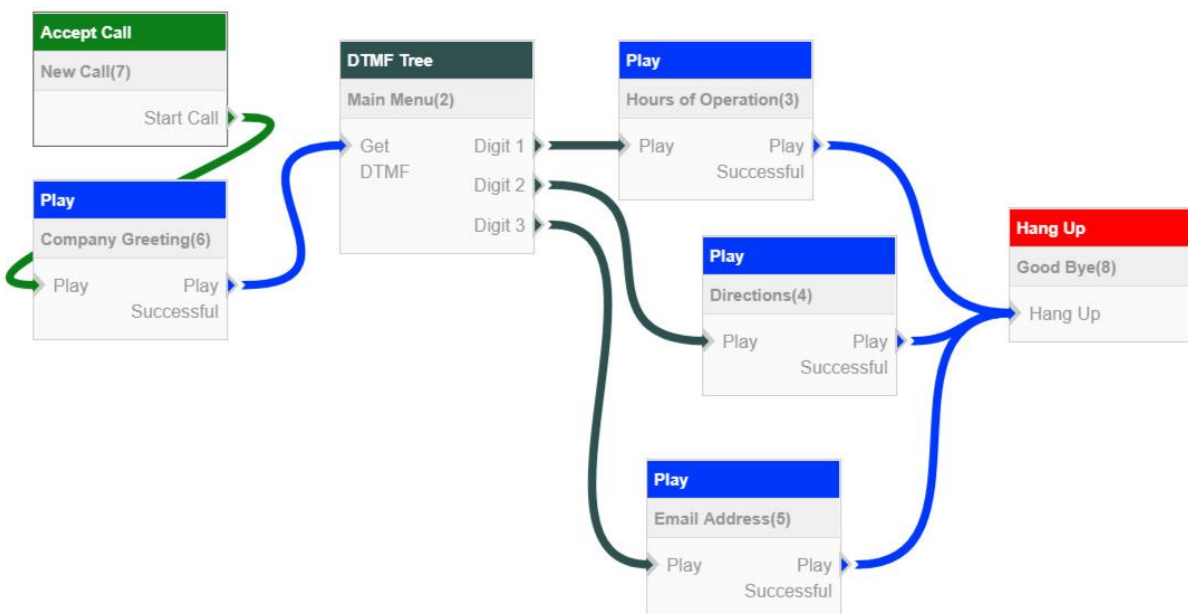
The DTMF Tree command block receives a DTMF digit from the caller and based on that key branches to different command blocks.

Click on the DTMF tree area of the command block to bring up the data entry form. In the form below, a caller can press 1 to listen to the Hours of Operation, press 2 for Directions, and press 3 to listen to the email address of the organization.

DTMF Tree (2)	Save Block Settings	Delete Block	
Block Name	<input type="text" value="Main Menu"/>		
OnHangupGoto	<input type="text" value="HANG_UP(Good Bye)"/>		
Error Allowed	<input type="text" value="3"/>		
Error-Play/Next	<input type="text" value="Not Set"/>	<input type="text" value="Not Set"/>	
Timeout-Play/Next	<input type="text" value="Not Set"/>	<input type="text" value="Not Set"/>	
Invalid Digit-Play/Next	<input type="text" value="Not Set"/>	<input type="text" value="Not Set"/>	
Store Variable	<input type="text"/>		
Speech Recognition Grammar	<input type="text" value="Do not use Speech Gra"/>		
Digit 1	<input type="text" value="PLAY(Hours of Operati"/>	<input type="text" value="1"/>	
Digit 2	<input type="text" value="PLAY(Directions)"/>	<input type="text" value="2"/>	
Digit 3	<input type="text" value="PLAY(Email Address)"/>	<input type="text" value="3"/>	
Digit 4	<input type="text" value="Not Set"/>	<input type="text" value="4"/>	
Digit 5	<input type="text" value="Not Set"/>	<input type="text" value="5"/>	
Digit 6	<input type="text" value="Not Set"/>	<input type="text" value="6"/>	
Digit 7	<input type="text" value="Not Set"/>	<input type="text" value="7"/>	
Digit 8	<input type="text" value="Not Set"/>	<input type="text" value="8"/>	
Digit 9	<input type="text" value="Not Set"/>	<input type="text" value="9"/>	
Digit 0	<input type="text" value="Not Set"/>	<input type="text" value="0"/>	
Digit *	<input type="text" value="Not Set"/>	<input text"="" type="text" value="Not Set"/>	<input type="text" value="#"/>

DTMF Tree Call Flow Example

This is what the company menu example above looks like after the DTMF Tree Command Block is completed:



In the above example, the system receives a new call and plays the Company Greeting message. Then it asks the caller to “Press 1 for Hours of Operation, press 2 for Directions, and press 3 for the email address of the Company”. The caller presses a digit on his phone and the system based on the digit entered sends the flow to one of the Play blocks. After playing the information, the flow jumps to the Hang Up block. The call disconnects.

You may modify the data in command blocks forms to allow a caller to press a key to repeat the information and handle the error conditions. However from a design and expression point of view, it is a good idea to implement the sunny conditions first and then add the error handling and abnormal conditions.


Get DTMF



The Get DTMF command block reads the DTMF digits the caller enters from his/her telephone keypad.

After the digits have been entered, the control goes to the next block specified in the Get DTMF form.

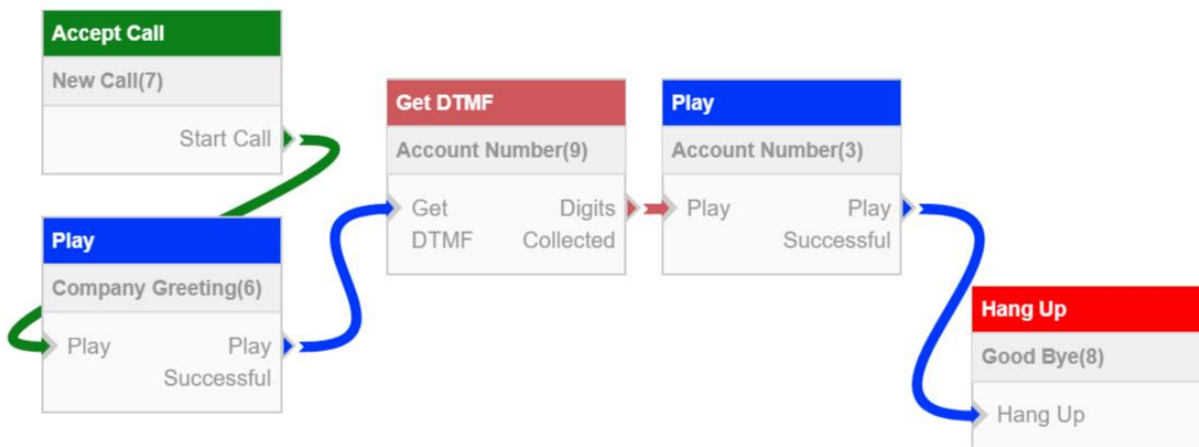
Click the area under Get DTMF to open the following form:

Get DTMF (9)	Save Block Settings	Delete Block
Block Name	<input type="text" value="Account Number"/>	
OnHangupGoto	<input type="text" value="HANG_UP(Good Bye)"/>	
Digits to Get	<input type="text" value="Enter Manually"/>	<input type="text" value="6"/>
Speech Recognition Grammar	<input type="text" value="Do not use Speech Grar"/>	
Store Variable	<input type="text"/>	
Next Block	<input type="text" value="PLAY(Account Number)"/>	
Error Allowed	<input type="text" value="3"/>	
Error-Play/Next	<input type="text" value="Not Set"/>	<input type="text" value="HANG_UP(Good Bye)"/>
Timeout-Play/Next	<input type="text" value="Not Set"/>	<input type="text" value="HANG_UP(Good Bye)"/>
Minimum # Digits	<input type="text" value="Enter Manually"/>	<input type="text" value="4"/>
Min-Digit-Play/Next	<input type="text" value="Not Set"/>	<input type="text" value="PLAY(Company Greetin)"/>

The form allows to you set various settings, such as the minimum number of digits, the amount of time to wait, how many errors to allow, etc. The block above is accepting an account number from the caller with a minimum of 4 digits and maximum of 6. If the caller makes more than 3 errors, the system will hang up.

Get DTMF Call Flow Example

The following example shows the use of the Get DTMF command block. The call flow asks the caller to enter his/her account number and plays it back to him/her. If the caller does not enter anything, the flow goes to the HANG_UP command block and disconnects.



NOTE: Many of the conditions are not connected by arrows to the Hang Up command block even though they may result in a Hang Up condition. We have done this intentionally to prevent a cluttered screen.

Send DTMF



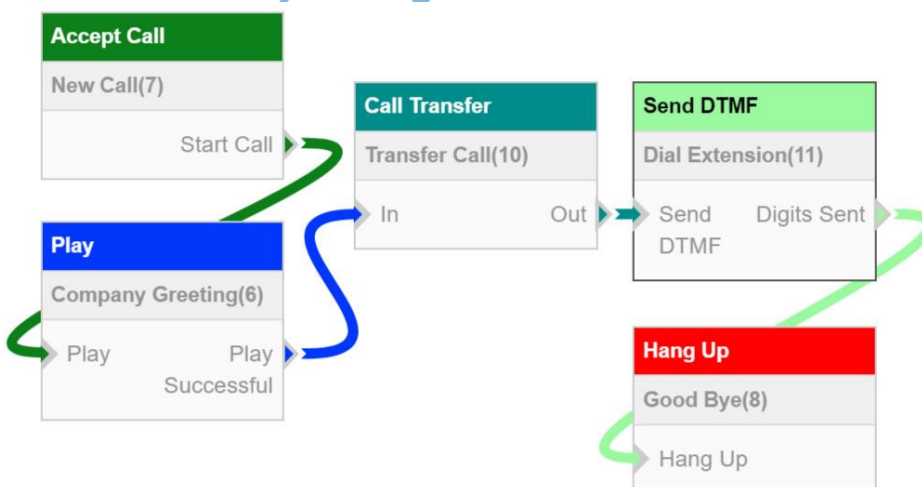
The Send DTMF command block generates DTMF digits. The uses are for making outbound calls or entering the extension number once a PBX has answered a transferred or dialed call.

Double-click the Send DTMF area of the command. The following form appears:

Send DTMF (11)	Save Block Settings	Delete Block
Block Name	<input type="text" value="Dial Extension"/>	
OnHangupGoto	<input type="text" value="Not Set"/>	
Digits to Send	<input type="text" value="ExtensionNo"/>	
Next Block	<input type="text" value="HANG_UP(Good Bye)"/>	

In the form above, the digits to dial are stored in a variable called “ExtensionNo”. Once the system executes this command, it generates the DTMF sounds based on the data in the “ExtensionNo” variable.

The example below shows how the Send DTMF command block is used: The inbound call is transferred to a phone number and then Send DTMF generates the DTMF digits to transfer to a specific extension.



Compare



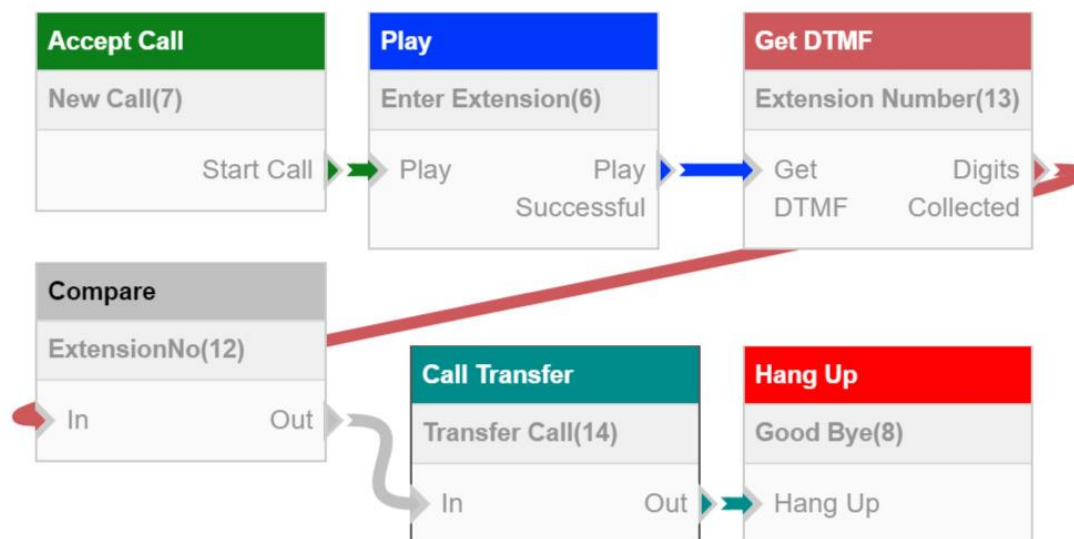
The Compare command block allows you to take an action based on the value of a specific variable. You can add as many conditions as you need.

In the example below, we are checking to see if the extension number equals 4225. If so, the system will play “Thank you” and transfer the call to the appropriate extension.

Compare (1)		Save Block Settings	Delete Block
Block Name	No Name		
OnHangupGoto	HANG_UP(Good Bye)		
Default Next Block	HANG_UP(Good Bye)		
Error-Play/Next	Not Set	CALL_TRANSFER(No t	

	Order	Case Insensitive	Variable	Operand	Value	Play	Next
Delete	1	<input type="checkbox"/>	ExtensionNo	=	4225	Thank You	CALL_TRANSFER(No t
Add							

Below, you can see how Compare fits into the overall call flow:



Play



The Play command block allows you to play a voice prompt to the caller. The file can be a pre-recorded message or it can use our text to speech engine that will play any message on the fly.

The Play block form is below. The example below plays the date and time which is stored in the variable Date_Time. The Play Type is set to “Interruptible” which means that the caller is able to press a DTMF key while the message is playing.

Play (2)	Save Block Settings	Delete Block
Block Name	<input type="text" value="Date and Time"/>	
OnHangupGoto	<input type="text" value="HANG_UP(Good Bye)"/>	
Next Block	<input type="text" value="HANG_UP(Good Bye)"/>	
Error-Play/Next	<input type="text" value="Not Set"/>	<input type="text" value="CALL_TRANSFER(No T"/>
Error Repeats Allowed	<input type="text" value="3"/>	
Store Variable	<input type="text" value="Date_Time"/>	
Play Type	<input type="text" value="Interruptible"/>	
Speech Recognition Grammar	<input type="text" value="Do not use Speech Grar"/>	

Order	Play Type	Value To Play	Type
1	<input type="text" value="Play Variable"/>	<input type="text" value="Date_Time"/>	<input type="text" value="Date/Time"/>

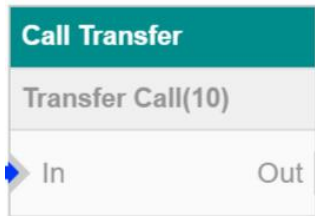
Play Block Options

The Play command allows you to select how a variable should be played. If you have currency it can play as One Thousand Two Hundred dollars. If the variable has data in the date format: 03/14/2019 11:34AM and you indicate that variable is a Date/Time, the system plays out “March Fourteenth Two Thousand Nineteen Eleven Thirty Four AM.” If you selected “Date with Day of Week/Time,” the system would prepend the word “Thursday” to the message.

Play (0)	Save Block Settings	Delete Block
Block Name	<input type="text" value="Date and Time"/>	
OnHangupGoto	<input type="text" value="HANG_UP(Good Bye)"/>	
Next Block	<input type="text" value="HANG_UP(Good Bye)"/>	
Error-Play/Next	<input type="text" value="Not Set"/>	<input type="text" value="HANG_UP(Good Bye)"/>
Error Repeats Allowed	<input type="text" value="3"/>	
Store Variable	<input type="text" value="General_Data"/>	
Play Type	<input type="text" value="Interruptible"/>	
Speech Recognition Grammar	<input type="text" value="Do not use Speech Grar"/>	
		<ul style="list-style-type: none"> Sentence (Words Broken Up) Date Date with Day of Week Time Date/Time Date with Day of Week/Time Number Payout Letters/Number Shortcut Date Shortcut Date with Day of Week Shortcut Time Sentence (Play as Whole) 10 Digit Phone Number Shortcut Date with Year Sentence (Words Broke
Order	Play Type	Value To Play
1	<input type="text" value="Play Variable"/>	<input type="text" value="General_Data"/>

DR

Call Transfer



The Call Transfer command block allows you to transfer a phone call to an extension or external phone number.

In the example below, the Call Transfer block directs the IVR to transfer the call to phone number “7322901900.” If the called party answers the call, the system will play “Your appointment is scheduled for” and disconnect the call. You can also set the Caller ID / ANI to use, set the Hold Music to play during the transfer, and decide whether or not to record the call.

Block Name	<input type="text" value="Transfer Call"/>	
OnHangupGoto	<input type="text" value="Not Set"/>	
Choose Transfer Type	<input type="text" value="Normal Transfer"/>	
Number to Transfer to	<input type="text" value="Manually Enter Number"/>	<input type="text" value="7322901900"/>
ANI to Use	<input type="text" value="Manually Enter ANI to U"/>	<input type="text" value="7671"/>
Hold Music/Message	<input type="text" value="Free Music"/>	
Variable to Store transfer status into	<input type="text" value="ExtensionNo"/>	
Default Next Block	<input type="text" value="Not Set"/>	
Record Call Settings	<input type="text" value="Do Not Record The Call"/>	
Setup Trigger to a Web Service	<input type="text" value="Disable"/>	

	Order	Status	Play	Next
<input type="button" value="Delete"/>	1	<input type="text" value="Live Answer"/>	<input type="text" value="Your Appointment is sch"/>	<input type="text" value="HANG_UP(Good Bye)"/>

Call Transfer Options

The Call Transfer command block allows you to set the type of transfer that will take place. For example, you can transfer the call to a voice mail box and take a message by selecting “Transfer to Voicemail Box.” You can also transfer to another IVR or send a screen transfer.

Choose Transfer Type	<div style="border: 1px solid black; padding: 2px;"> Transfer to Voicemail Bc ▾ Normal Transfer Transfer to Voicemail Box Screen Transfer Transfer to Inbound IVR Use Caller ANI ▾ </div>	
Number to Transfer to		3442
ANI to Use		

Once the call is transferred, the system may encounter No Answer, Busy, Answering Machine, or Operator Intercepts. You can instruct the system how to handle each scenario. In the example below, if the called party answers the call, the system will play “Thank you.”

Record Call Settings		Do Not Record The Call ▾	
Setup Trigger to a Web		▾	
	<div style="border: 1px solid black; padding: 2px;"> Live Answer Answering Machine Busy No Answer Operator Intercept Fax Live Answer ▾ </div>		
	Order	Play	Next
Delete	1	Thank You ▾	Not Set

Record File



The Record File command block allows you to record voice from the phone call and store it into a file on the IVR system.

The form below shows how the system will choose the file name and decide what to do after the file is recorded. The file name is dynamic based on the variable Message_File_Name to allow each file to have a unique name.

Record File (0)	Save Block Settings	Delete Block
Block Name	<input type="text" value="Message"/>	
OnHangupGoto	<input type="text" value="HANG_UP(Good Bye)"/>	
Recording Filename	<input type="text" value="Message_File_Name"/>	
Next Block	<input type="text" value="HANG_UP(Good Bye)"/>	
Error Allowed	<input type="text" value="3"/>	
Error-Play/Next	<input type="text" value="Not Set"/>	<input type="text" value="HANG_UP(Good Bye)"/>

Web Services



One of the most powerful features of VoiceSaver® IVR is the ability to call external Web Services or applications to process code intense or database intense activities using a programming language that is more familiar to your developers.

The Web Services form below shows how the system would make a request to Microsoft Dynamic’s API using the function “getcustomer()” and sending the variables “id” and “pin.”

The result from the service response is placed in the variable “Collection_Agency.” This data variable can be used in the future within the IVR flow.

Web Service (3)	Save Block Settings	Delete Block
Block Name	<input type="text" value="Microsoft"/>	
OnHangupGoto	<input type="text" value="HANG_UP(Good Bye)"/>	
Next Block	<input type="text" value="PLAY(Activation Key No)"/>	
Error-Play/Next	<input type="text" value="Not Set"/>	<input type="text" value="HANG_UP(Good Bye)"/>
Hold Music/Message	<input type="text" value="Not Set"/>	

Map IVR variables to End Point IVR variables	
Please select an End Point	Please select a Call
<input type="text" value="Microsoft Dynamic"/>	<input type="text" value="getcustomer()"/>

End Point / Call Parameters	Value Assignment	Value
id	<input type="text" value="IVR Variable"/>	<input type="text" value="ExtensionNo"/>
pin	<input type="text" value="IVR Variable"/>	<input type="text" value="General_Data"/>

End Point / Call Response	IVR Variable
account_status	<input type="text" value="Collection_Agency"/>

Add to DNC

Add to DNC	
Put in DNC List(2)	
In	Out

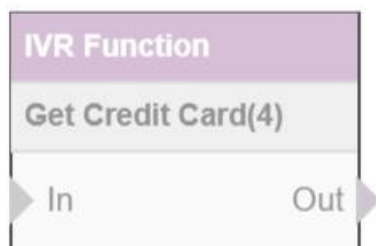
The Add to DNC command allows you to add phone a number to the local or global Do Not Call lists. The system does not call the numbers in the Do Not Call list.

The Add to DNC form below shows how to add a specific number (ExtensionNo) to the Global Do Not Call list.

Add to DNC (2)	Save Block Settings	Delete Block
Block Name	<input type="text" value="Put in DNC List"/>	
OnHangupGoto	<input type="text" value="RECORD_FILE(No Nan"/>	
Error Allowed	<input type="text" value="3"/>	
Timeout-Play/Next	<input type="text" value="Not Set"/>	<input type="text" value="RECORD_FILE(No Nan"/>
Default Play & Next Block	<input type="text" value="Not Set"/>	<input type="text" value="RECORD_FILE(No Nan"/>
Number to add to DNC	<input type="text" value="ExtensionNo"/>	
Choose DNC Type	<input type="text" value="Add to Global DNC"/>	

DRAFT

IVR Function



The IVR Function command allows you to activate a prebuilt call flow defined by a function name. This allows you to reuse frequently used IVR call flows throughout the IVR or across multiple IVRs without the need to create the same logic over and over.

The IVR Function form below defines a new function called “Get Credit Card” which asks the caller to enter his/her credit card number, expiration date, and validation code. Now, this function can be reused in other call flows to avoid rewriting the logic. (see example below)

IVR Function (4)	Save Block Settings	Delete Block
Block Name	<input type="text" value="Get Credit Card"/>	
OnHangupGoto	<input type="text" value="HANG_UP(Good Bye)"/>	
Next Block	<input type="text" value="PLAY(Account Number)"/>	
Error-Play/Next	<input type="text" value="Ask to Enter Credit Card"/>	<input type="text" value="PLAY(Account Number)"/>

IVR Function Variables to Pass and Return		
Please select an IVR Function		
<input type="text" value="Get the Credit Card Information(6)"/>		

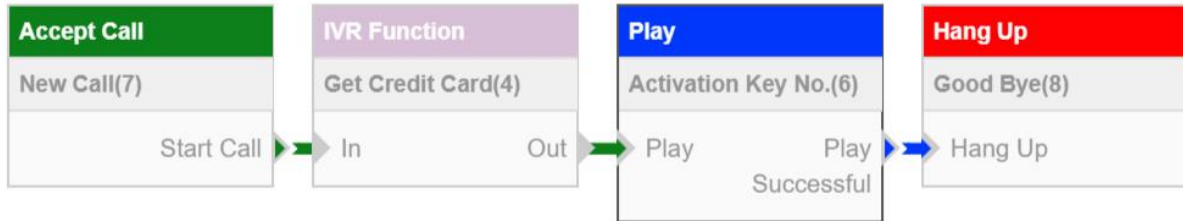
Variables to Pass to the IVR Function	Value Assignment	Value
There are no parameters to send.		

Variables to map back from the IVR Function	IVR Variable
There is no response to receive.	

	Order	Case Insensitive	Variable	Operand	Value	Play	Next
<input type="button" value="Add"/>							

IVR Function Example

In the call flow example below, the software receives the call and activates the Get Credit Card Number IVR function. The function performs all the tasks needed to collect the credit card number and then call a web service to validate the credit card. The function would then return back to the main IVR and continue with the information obtained from the “GetCreditCard()” function.



DRAFT March 20, 2017