

# Access to Voicemail from external DID

## REQUIREMENTS

Voicemail name must be the same of the sip name (e.g. john.doe).  
The voicemail password must be numeric (e.g. 0000).

## HOW IT WORKS

Access to voicemail can be handled by using ivr script and a custom application (VoiceMailMain).  
You need to set two different dial plans:

- An internal DID that need to execute the custom application (VoiceMailMain). The application VoiceMailMain allows you to manage the voicemail messages.
- An external DID to access to the ivr script. The ivr asks the voicemail extension you want to check and performs the query to xcall db to retrieve the voicemail name (sip name). Then the call goes to the internal dialplan to execute the VoiceMailMain application.

## IVR SCRIPT – ACCESS TO VOICEMAIL



## THE IVR SCRIPT

### Variables

Add two channel variables: VM\_EXT and VM\_USER

Name	Type
VM_EXT	Channel variable
VM_USER	Channel variable

### Database

Set the DB connection

Database *	<input type="text" value="xcally"/>
Host *	<input type="text" value="127.0.0.1"/>
Username *	<input type="text" value="root"/>
Password *	<input type="password" value="....."/>
Confirm Password *	<input type="password" value="....."/>
Type *	<input type="text" value="MySQL"/>

### Sounds

Upload the audio file

New Sound +

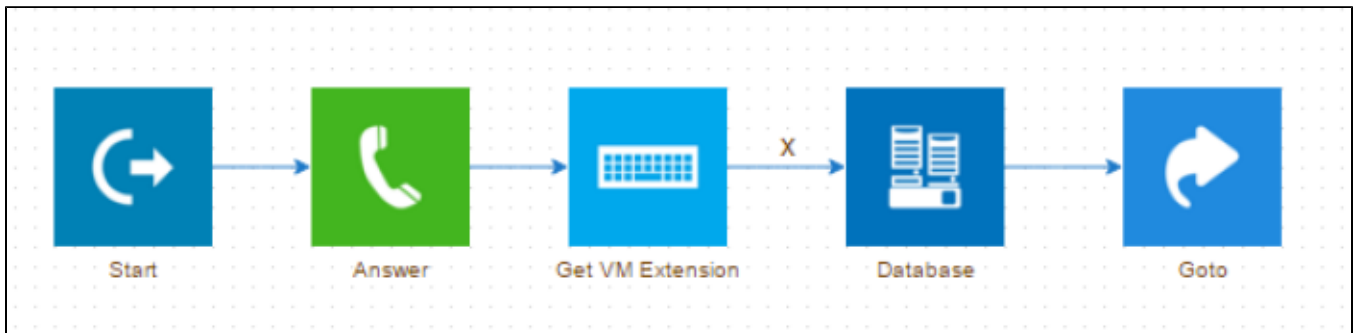
Name	Filename	Preview
Insert VM Extension	InsertVMext.wav	



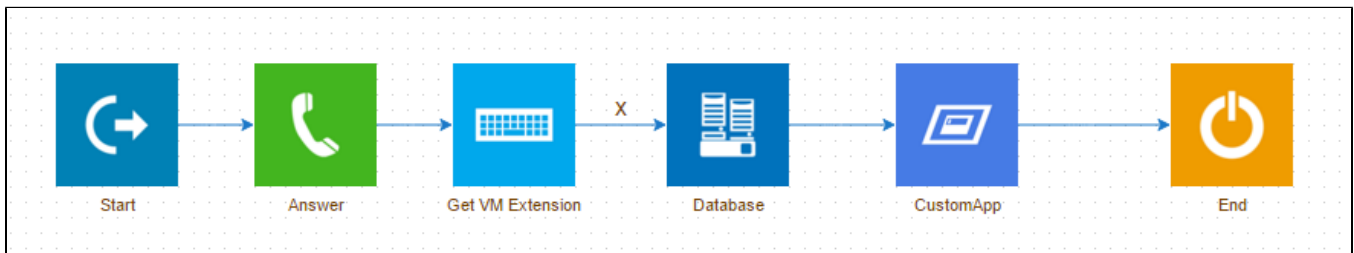
**Release note**

From xCally Shuttle release 2.4.21, you can use the new **CustomApp** IVR block instead the DialPlan.

**Design (Shuttle release up to 2.4.20)**



**Design (Shuttle release 2.4.21 or above)\*\***



**Block details**

GetDigit block settings

### Edit GetDigits

Label	<input type="text" value="Get VM Extension"/>
Filename	<input type="text" value="Insert VM Extension"/> ▾
Response Timeout	<input type="text" value="5"/>
Min Digits	<input type="text" value="4"/>
Max Digits	<input type="text" value="4"/>
Retry	<input type="text" value="3"/>
Variable	<input type="text" value="VM_EXT"/> ▾

Database block settings

### Edit Database

Label	<input type="text" value="Database"/>
Database	<input type="text" value="xcally"/> ▾
Query	<input type="text" value="SELECT name FROM sip WHERE reference={VM_EXT}"/>
Variable	<input type="text" value="VM_USER"/> ▾

GoTo block settings

Label	<input type="text" value="Goto"/>
Context	<input type="text" value="from-sip"/>
Extension	<input type="text" value="*1000"/>

Where \*1000 is the DID of internal dial plan we use to execute the VoiceMailMain application

**\*\*CustomApp block settings**

### Edit CustomApp

Label	CustomApp
Application	VoiceMailMain
Options	{VM_USER[0][name]}@from-voicemail

**DIAL-PLAN SETTINGS**

**The external inbound dialplan**

Add an external inbound dial-plan and set a DID you want to use to check the voicemail boxes

### VoiceMail Access

General    Advanced

#### General Settings

Field	Value	Description
Name	<a href="#">VoiceMail Access</a>	Name
DID	set your external DID <input type="text"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	DID
Description	<i>Empty</i>	Description

In the advanced settings add the cally square project

☒ VoiceMail Access

General **Advanced**

🔊 Application

+ New Application ▾

15 ▾ records

Priority ▲	Application ▾	Application Args
1	agi	agi://127.0.0.1/square,project=29 (Project Name: VoiceMail Access)

**The internal inbound dialplan**

Add an internal inbound dial plan and set the DID \*1000 (or the same you use in the GOTO block).

☒ CheckMyVoiceMail

General **Advanced**

🔊 General Settings

Field	Value
Name	<a href="#">CheckMyVoiceMail</a>
DID	<a href="#">*1000</a>
Description	<a href="#">Empty</a>

In the advanced settings add a **Custom Application**:

Application name: VoiceMailMain

Params: `_${VM_USER[0][name]}@from-voicemail`

Priority ▲	Application ▾	Application Args
1	VoiceMailMain	<code>_\${VM_USER[0][name]}@from-voicemail</code>
2	hangup	

Your remote access to check the voicemails is ready!

