



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Composer Help

Variables in Callflows

Variables in Callflows

Contents

- **1 Variables in Callflows**
 - 1.1 Types of Variables
 - 1.2 Variable Versus Static Data
 - 1.3 Entry Block Variables
 - 1.4 Attaching Results to User Data

You can define voice application (session) variables using the **Entry block Variables property**.

Types of Variables

Composer supports the following types of variables for callflow diagrams:

- **System--Pre-defined application variables** (System category above) hold Project and application-related values. While you cannot delete System variables, you can have your application modify the values.
- **User--User-defined custom variables** that you create by clicking the Add button in the Application Variables dialog box above and selecting User. Your application can delete and modify User variables.
- **Input**--These are variables supplied as input to the called diagram. Created by clicking the Add button in the Application Variables dialog box above and selecting User. During runtime, Input variables get auto-filled from the calling context. Typically Input variables are created on the SubCallflow side to notify the MainCallflow about the Parameter-passing details while designing the application flow. Composer does auto-synchronization of the Input variables in the **Subdialog block**. Input variables are also used on the MainCallflow while invoking the VoiceXML application from workflows in case of Voice Treatment execution - computer telephony integration (CTI) scenario (**Play Application**).
- **MainCallflow**--Automatically filled from either session.com.genesyslab.userdata or session.connection.protocol.sip.requesturi based on the Non-CTIC or CTIC flow.
- **SubCallflow**--Automatically filled from the VXML subdialog-invoking methodology.

Variable Versus Static Data

Many blocks enable the use of variables rather than static data. For example, the **Prompt block** can play the value of a variable as Text-to-Speech. Variables whose values are to be used in other blocks must be declared here so that they appear in the list of available variables in other blocks. The value collected by an **Input block** or a **Menu block** is saved as a session variable whose name is the same as the block Name. Also see information on the **AppState variable** used by the DB Data block.

Entry Block Variables

Entry block variables can access **User Data** (attached data from a routing workflow) from session.com.genesyslab.userdata and SIP Request-URI parameters from session.connection.protocol.sip.requesturi session variables.

Request URi parameters created in IVR Profiles during the VoiceXML application provisioning are passed to the Composer generated VoiceXML application as request-uri parameters in the session.connection.protocol.sip.requesturi session array. An Entry block variable can use these parameters by setting the following expressions to the variable values: typeof session.connection.protocol.sip.requesturi['var1'] == 'undefined' ? "LocalDefaultValue" : session.connection.protocol.sip.requesturi['var1'].

If parameters are set as part of IVR Profiles provisioning in the Genesys VoiceXML provisioning system, and if these parameters have the same names as variables set in the Entry block's **Variables** property with the above mentioned `sip.requesturi` expression, then the SIP-Request-URI parameters will take precedence over the user variable values set in the Entry block.

Important

For more information on valid values and syntax for the `gvp.services-parameter` section, refer to page number 121 in the [GVP 8.5 User's Guide](#).

IVR profiles for GVP can be created using the Genesys Administrator. For more information, refer to the [Voice Platform Solution Guide](#) and the `gvp.services-parameter` section in the [GVP 8.5 User's Guide](#).

Attaching Results to User Data

While you can assign Classify object results to a variable, Genesys does not recommend this. The recommended way of dealing with the classification results is to attach them to the interaction. Then User Data will have the keys listed in the table below with the corresponding values returned by Classification Server. As an example, User Data would have the following pairs after the attachment:

Parameter	Value
CtgId	00001a05F5U900QW
CtgRelevancy	95
CtgName	Cooking
CtgId_00001a05F5U900QW	95
CtgId_00001a05F5U900QX	85
CtgId_00001a05F5U900QY	75
CtgId_00001a05F5U900QZ	65