Genesys App Automation Platform Help

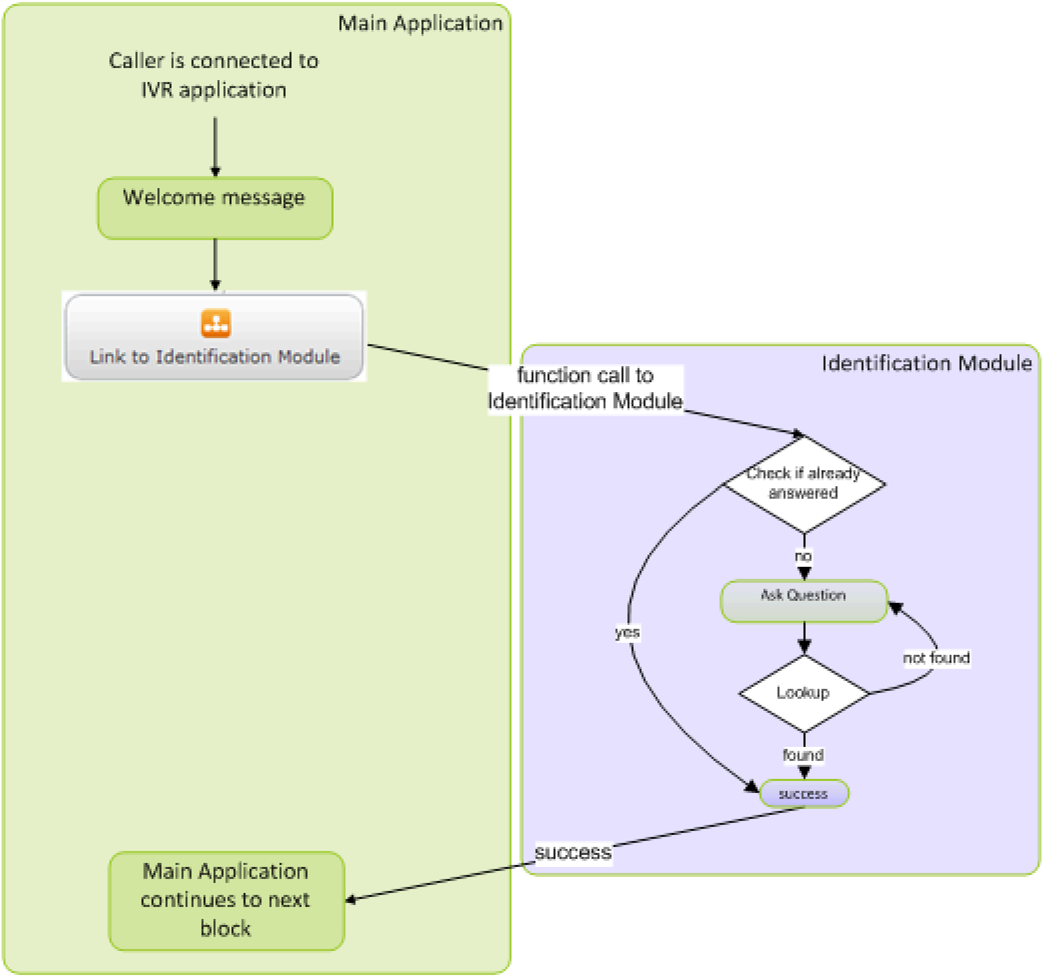
Link Block

12/14/2025

Link Block

You can use **Link** blocks to send a call within your callflow to another application, module, or an external VXML application. Once the external callflow finishes, the call returns to the point at which it left in the parent callflow.

For example, the main application might start with a Message block to welcome the caller and then use a **Link** block to call into an Identification module, which allows the caller to identify himself or herself. When the module finishes, the call routes back to the main application with a **success** result and the rest of the application callflow executes.


Main Application
Caller is connected to IVR application
Welcome message
Link to Identification Module
function call to Identification Module
Identification Module
Check if already answered
no
yes
Ask Question
Lookup
not found
found
success
success
Main Application continues to next block

Adding this block to the callflow

To add and configure **Link** blocks in a callflow:

1. Drag and drop a **Link** block onto the appropriate position in the callflow.
2. Click the **Link** block to view its properties.
3. In the **Link Details** tab, select one of the following options:
   - **Transfer to Another Genesys App Automation Platform Module** – Call a GAAP module within the same Company. This links to the Start block within the module.

     > ### Tip
     > You can click **Jump to Module** to open the linked module in the Callflow Editor. If you have unsaved changes in your current application, GAAP asks you to leave the page without saving.

   - **Transfer to an External VXML Application** – Use an HTTP URL to call into a VXML application. After this application finishes processing the call, it must return specific information back to the GAAP application (as defined in the third-party application contract), as well as the path that must be followed as a result of the call. You can use variables when specifying the URL, such as: `http://[var:server]:[var:port]/MyPage.vxml`.

     > ### Important
     > Contact your Genesys representative for more information on configuring the results from a VXML application.

     If the caller hangs up while in the VXML application portion of the call, or the VXML application wants to end the call, the VXML application must return a **hangup** or **system hangup** result, respectively.

Passing VXML Parameters to Other Applications

The following example describes how to pass Intelligent Automation variables to an external VXML application like Genesys Composer.

A sample GIA callflow can be as:

- a Start block
- a Script block
  - Create a variable and attach it to the interaction by selecting the **Attach to call** option.
- a Link block
  - Select **Transfer to an External VXML Application** and specify the VXML URL.
- an End block

In Genesys Composer:

- Add a text file, **my_script.js** to the directory referred by global variable SCRIPTSDIR inside the Entry block. This directory is **../Script**s by default.
- Add the following function to the script.

```
function getUrlVars( dataMessage )
{
    if(dataMessage == undefined)
        return undefined;
    var hashes = dataMessage.slice(dataMessage.indexOf('?') + 1).split('&');
    var hash;
    var resultArray = new Array();
    for(var i = 0; i < hashes.length; i++)
    {
        hash = hashes[i].split('=');
        resultArray.push(hash[0]);
        resultArray[hash[0]] = hash[1];
    }
    return resultArray;
}
```

- The URL variable can be used. For example, create an Assign block and define the value as `getUrlVars(_GetInfo('running_uri'))['test']`.
- Log the value being assigned and confirm that the value matches the value sent from the Intelligent Automation callflow.