



SmartApps Cloud Screen Pop Implementation Guide

Overview

The screen pop feature uses a custom URL protocol handler to launch applications running on the local agent's desktop running Genesys Cloud web application. Each screen pop to a local application is a bit different. Some create COM objects while others call out to an EXE files on the local machine or network drive, while some screen pops require interfacing with the browser using a browser extension. To execute the different screen pops, a Custom URL protocol is called the SmartApps Cloud Agent experience embedded in Genesys Cloud, issuing the screen pop type desired.

Custom URL Protocol Handler Overview

A custom URL protocol handler is a system feature that allows you to define your own URL scheme (such as smartappspop:) and associate it with a specific application on a user's device. When a user clicks a link with this custom protocol, the operating system recognizes the scheme and launches the designated application, optionally passing parameters from the URL to the app. This mechanism is commonly used to enable web pages or emails to trigger desktop or mobile apps directly for enhanced integration and automation.

SmartApps Screen Pop

The SmartApps Agent experience, embedded inside Genesys, leverages the custom URL protocol handler configured on each user's machine to initiate a screen pop to applications like Symitar Episys and Fiserv DNA Relationship Manager (among others). Specifically, when an event in Genesys requires a screen pop—such as an incoming interaction—the web page generates a link using the custom protocol (for example, smartappspop:), which encodes the necessary parameters for the screen pop action.

Upon clicking this pop button, the browser recognizes the custom protocol and passes the URL to the operating system, which then launches the registered executable. The executable extracts the parameters from the custom URL and performs the screen pop based on the configuration.

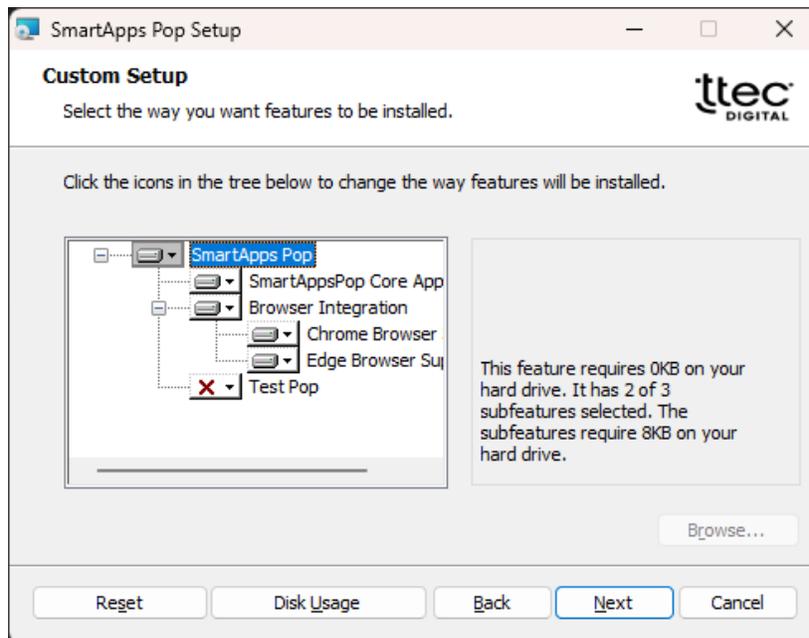


Installation Overview

The SmartAppsPop Installer will deploy the necessary screen pop definitions for all support screen pop types along with the ability to create extensible screen pops outside those supported. Given there are numerous screen pops supported, there are numerous components which can be selected with the installer which may only apply to specific implementations. This means the components may not be needed for all deployments

Installation Instructions

1. Download and start the installer: [SmartAppsPop.Installer.msi](#)
2. This application needs to be installed on each user's desktop. We suggest testing this on one desktop end to end prior to deploying to all desktops.
3. The installation deploys files to C:\Program Files (x86)\SmartAppsPop and registers the Custom URL Protocol handler to receive the screen pop request from interactions managed within Genesys + SmartApps Agent experience.



4. On the Feature selection page, there are numerous features which control what is installed. Below describes which applications are required for which screen pop type.
 - a. SmartAppsPop Core Application – Required for all screen pop types.



- b. Browser Integration – Required for any screen pop which uses a browser extension for the screen pop execution. Below is a list of screen pops which require the browser extension to be installed on each machine.
- c. Test Pop – This is an option feature which can be used to test any of the available screen pops with a test harness html page. By default, this is not installed as it is not needed for the application to work but only to troubleshoot the screen pop process with any specific application.

The table below breaks down the screen pop types and the required features from the install to be able to execute the desired screen pop type.

Screen Pop Type	Require SmartApps Pop	Requires Browser Integration
Symitar Quest (Episys)	Yes	No
Fiserv DNA Relationship Manager	Yes	No
Fiserv XP2	Yes*	Yes
Temenos	Yes*	Yes
Fiserv Spectrum Branchsuite	Yes*	Yes
URL Pop to new browser tab	No	No
URL Pop to specific browser type (Chrome or Edge)	Yes	No
URL Pop replacing existing tab (Chrome or Edge)	Yes	Yes

* For these, it is yes for requiring SmartApps Pop is only required if the Genesys Desktop Client is used. If the Genesys Desktop Client is not used, the Browser extension can be installed either through the installer or directly through the [Chrome Web Store](#).

Silent Install

The installer supports a silent installation along with allowing the browser components to be disabled. The installer needs to be run with elevated permissions command prompt so the installer can write HKLM registry entries.



Installer Flags

The install MSI supports numerous flags to control different behaviors of the install when running from a command line.

- "INSTALLBROWSERINTEGRATION = 0" Indicates that none of the browser components should be installed. This is for deployments where the screen pops required aren't any of the browser-based screen pops. For instance, Episys and Relationship Manager do not require any browser screen pop components
- "INSTALLEXTENSIONPOLICY=0" When installing the browser components, the install will register the browser extensions to be automatically deployed into Chrome and Edge when they are started next. If you prefer to deploy the browser extension manually through the Chrome store, use this flag to not add the extension for download within the browser automatically. The chrome extension can be manually deployed from the chrome store here: <https://chromewebstore.google.com/detail/ttec-digital-smartapps-sc/gjmecejdekmandfoohnhfdkmgpddmoomhh>

Standard silent install:

```
msiexec /i "Installer/bin/Release/SmartAppsPop.Installer.msi" /qn /norestart /!*"v  
"C:\temp\SmartAppsPop-install.log"
```

Silent install (disable browser integration):

```
msiexec /i "Installer/bin/Release/SmartAppsPop.Installer.msi" INSTALLBROWSERINTEGRATION=0  
/qn /norestart /!*"v "C:\temp\SmartAppsPop-install.log"
```

Symitar Episys

The Episys screen pop is executed by instantiated the Episys COM Object for Telephony Queue Access. Symitar's RemoteAdminServer.exe program installed with Episys is the local server for the object. The RemoteAdminServer is required to use the COM Object for screen pop.

In Site Manager, create a new Application Definition (Screen Pop -> Application Definitions). Using the table below as a reference for configuring the Application Definition. After saving the new screen pop definition, assign this screen pop definition to any screen pop profile which should pop this application and screen based on the call flow.



Field	Value
Client Addin Type	"Custom Application"
Description	"Episys" (or whatever description preferred)
Protocol Handler App	"smartappspop"
Parameter(s)	<p>episys,{{Account Number}},0,T,ChangeWorkAreaTo:TellerTransactions</p> <ul style="list-style-type: none"> • episys is the first parameter sent to the SmartAppsPop application so it knows which screen pop type is being executed. • {{Account Number}} is replaced during the interaction with the opened account within the SmartApps Agent experience. • "0" for the second parameter means the Account number is what is being searched. • "T" for the third parameter indicates if the RasQueue should be shown within Episys. Default is T. Any other value will not open the Episys RasQueue • "ChangeWorkAreaTo:TellerTransactions" for the fourth parameter denotes the screen to open. Other options include ChangeWorkAreaTo:TellerTransactions, ChangeWorkAreaTo:ApplicationProcessing, or ChangeWorkAreaTo:CollectionPackage

Message Flow

Actors

- SmartApps Client Addin (Running in browser or Genesys Desktop Client)
- Screen Pop App (SmartAppsPop)
- Symitar Episys (Quest)

Message Flow

1. **Interaction detected** - The SmartApps Client Addin detects a new Genesys interaction and determines a Symitar (Episys) screen pop is required.
2. **Screen pop request sent** - The Addin invokes the SmartApps custom URL protocol (smartappspop:) with Episys screen pop parameters.



- **Example:**
smartappspop:episys,{AccountNumber},0,T,ChangeWorkAreaTo:TellerTransactions
- 3. **Screen Pop App launched** - The operating system launches the Screen Pop App, which parses the request and identifies it as a Symitar screen pop.
- 4. **Connection to Episys** - The Screen Pop App connects to Episys using the Telephony Queue Access COM interface provided by RemoteAdminServer.exe.
- 5. **Episys navigation executed** - The Screen Pop App instructs Episys to:
 - Open the specified account
 - Bring the Episys client to the foreground
 - Navigate to the configured work area
 - Optionally display the RAS Queue

Notes

- Native execution only (no browser or extensions)
- Execution path: Client Addin → Protocol Handler → Screen Pop App → COM → Episys
- RemoteAdminServer.exe must be running
- Citrix desktop support
 - Citrix VDI - The Citrix desktop session must be run in Single User Mode. This is a requirement for both SmartApps Screen Pop and Episys.
 - One Episys version installed - Ensure that the Citrix Master VM only has one version of the Episys client installed. We have seen problems if there are multiple versions of the Episys Client installed even if the user is using the latest version.
 - Enable RAS for Citrix - Screen pop uses Episys Remote Admin Server to open a member profile in the Episys Client. To enable this in Citrix VDI workstation, the following registry item needs to be updated on the Citrix Master VM:
 - The RemoteAdminServer key must be set to "1". By default, it is set to "0".
 - [HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Symitar\SFW\2.0\RemoteAdminServer]
 - Set "Run RAS Under Citrix Client Workstation" = dword:00000001

Fiserv DNA Relationship Manager

Relationship Manager can be screen popped by calling to the executable ShellLauncher pop is executed by instantiated the Episys COM Object for Telephony Queue Access. Symitar's



RemoteAdminServer.exe program installed with Episys is the local server for the object. The RemoteAdminServer is required to use the COM Object for screen pop.

In Site Manager, create a new Application Definition (Screen Pop -> Application Definitions). Using the table below as a reference for configuring the Application Definition. After saving the new screen pop definition, assign this screen pop definition to any screen pop profile which should pop this application and screen based on the call flow.

Field	Value
Client Addin Type	"Custom Application"
Description	"Relationship Manager" (or whatever description preferred)
Protocol Handler App	"smartappspop"
Parameter(s)	<p>dna,l:\OSI\ShellLauncher\ShellLauncher.exe,/UEI=Y /CMD="CORE LAUNCH 22000 Relationship Module {{Core Attribute.IVR_CoreApiCallerPersOrgType}} {{Core Attribute.IVR_CoreApiCallerPersOrgNumber}}"</p> <ul style="list-style-type: none"> • dna is the first parameter sent to the SmartAppsPop application, so it knows which screen pop type is being executed. • The second parameter is the path to where ShellLauncher.exe is located. This ShellLauncher.exe needs to be located in the same location on all users' machines. • The third parameter includes all parameters to be sent to ShellLauncher.exe. <ul style="list-style-type: none"> ○ {{Core Attribute.IVR_CoreApiCallerPersOrgType}} is replaced with PERS or ORG depending on the Account Type to be opened. ○ {{Core Attribute.IVR_CoreApiCallerPersOrgNumber }} is replaced with PERS or ORG number to pop the correct account.

Message Flow

Actors



- SmartApps Client Addin (Running in browser or Genesys Desktop Client)
- Screen Pop App (SmartAppsPop)
- Fiserv DNA Relationship Manager

Flow

1. **Interaction detected** – The SmartApps Client Addin detects a new Genesys interaction and determines a DNA Relationship Manager screen pop is required.
2. **Screen pop request sent** – The Addin invokes the SmartApps custom URL protocol (smartappspop:) with DNA screen pop parameters.
 1. **Example:** smartappspop:dna,l:\OSI\ShellLauncher\ShellLauncher.exe,/UEI=Y /CMD="CORE\LAUNCH\22000\Relationship Module\PERS\123456"
3. **Screen Pop App launched** – The operating system launches the Screen Pop App, which parses the request and identifies it as a DNA screen pop.
4. **DNA launch initiated** – The Screen Pop App executes ShellLauncher.exe with the provided command-line parameters to launch DNA Relationship Manager.
5. **DNA navigation executed** – DNA Relationship Manager:
 - o Launches the Relationship Module
 1. Opens the specified person or organization record
 2. Brings the DNA client to the foreground

Notes

- Native execution only (no browser or extensions)
- Execution path: Client Addin → Protocol Handler → Screen Pop App → ShellLauncher → DNA
- ShellLauncher.exe must be accessible on the agent workstation

COMMAND Parameters

Below are details from Relationship Manager documentation regarding the ShellLauncher screen pop approach. SmartAppsPop takes care of executing the ShellLauncher properly to the documentation below is only for reference.

Product Code	Always 'CORE'
Action to Take	Always 'LAUNCH' to automatically launch the specified screen



Module Application Number	Always 22000 to specify the Relationship Module
Menu Item ID	Always "Relationship Module" which is the Menu Item ID of the Relationship Profile screen
Customer/Member Type	PERS specifies a Person; ORG specifies an Organization or Business
Customer/Member Number	DNA assigned number that represents the Customer/Member. If the Customer/Member Type is equal to 'PERS' than this needs to be the PERS.PERSNBR; if its equal to 'ORG' than this needs to be the ORG.ORGNBR.

Example of Command Line Execution (all one command):

ShellLauncher.exe /UEI=Y /CMD="CORE|LAUNCH|22000|Relationship Module|PERS|621203"

- **CORE**=Product
- **LAUNCH**=Action to take
- **22000**=Module Form to Launch in is
- **Relationship Module**=Menu Item ID
- **PERS**=Entity of the Number to follow
- **621203**=Person Number being used for this example (Will change depending on execution)

Fiserv XP2

XP2 is a web application which is screen popped using Browser Extension (Chrome or Edge). While the Browser Extension is required to execute the screen pop, the SmartAppsPop is only required to be installed if the Genesys Desktop client. If the Genesys agent experience is used within a chrome or edge browser, the SmartApps Agent experience will send the pop request directly to the browser extension without using the SmartAppsPop. When using the Genesys Desktop client, the SmartApps Agent experience executes the SmartAppsPop through the Custom URL Protocol and then the SmartAppsPop sends the request to the running browser extension to complete the screen pop



Genesys web app within Chrome/Edge browser

When using Genesys Agent experience within Chrome/Edge browser, set up the screen pop using the settings in the table below.

Field	Value
Client Addin Type	"XP2 Screen Pop"
Use Screen Pop Assistant App	Off
Description	"XP2" (or whatever description preferred)
Option 1	<p>https://xp2web1 https://xp2web2 https://xp2web3</p> <p>This is a list of Web URLs of the XP2 application. The URLs are pipe delimited and should include any of the URLs where the XP2 could be running. These URLs are the base URLs the browser extension can use to locate any running browser tabs which are running XP2. Once the extension locates a browser tab which includes one of these URLs, it will replace that URL with the new account number.</p>

Message Flow

Actors

- SmartApps Client Addin (Running in browser)
- Browser Extension (Chrome or Edge)
- Fiserv XP2 Web Application

Message Flow

1. **Interaction detected** – The SmartApps Client Addin determines an XP2 screen pop is required.
2. **Screen pop request sent (direct)** – The Addin sends the XP2 request directly to the browser extension (no Screen Pop App / no Native Host).
3. **Browser tab located/opened** – The extension’s background.js runs XP2Pop() to find an existing XP2 tab by base URL or open a new one.
4. **In-page action** – background.js posts to content.js in the XP2 tab; content.js performs the account navigation/search.

Notes



- Web flow bypasses the Screen Pop App and Native Messaging Host; the Addin talks straight to the extension. Web client execution path: Client Addin → Browser Extension → XP2.
- Same browser extension logic (tab search + content.js action) as the desktop flow.

Genesys Desktop Client (Installed Application)

When using Genesys Desktop Client installed application, set up the screen pop using the settings in the table below.

Field	Value
Client Addin Type	"Custom Application"
Description	"XP2" (or whatever description preferred)
Protocol Handler App	"smartappspop"
Parameter(s)	<p>webextension,screenpop,{"type":"XP","XP2_BaseURL":"https://xp2web1 https://xp2web2 https://xp2web3","XP2_ScreenPopOption3":"","IVR_Account":"{{Account Number}}"},chrome</p> <ul style="list-style-type: none"> • webextension,screenpop is the first and second parameter sent to the SmartAppsPop application, so it knows it is sending a screenpop request to the browser extension. • The third parameter is a json object which describes the screen pop. <ul style="list-style-type: none"> ○ type - XP (Type of screen pop to execute) ○ XP2_BaseURL – Pipe delimited list of XP2 base web URLs to locate when looking for the web tab running XP2. ○ XP2_URLPath – The rest of the web path to append to XP2_BaseURL to replace the tab with. ○ IVR_Account – Set to the Account Number to pop. {{Account Number}} is replaced during the interaction with the opened account within the SmartApps Agent experience.



	<ul style="list-style-type: none"> The last parameter is optional specifying either chrome or edge to ensure the screen pop is only sent to an extension running in the desired browser.
--	---

Message Flow

Actors

- SmartApps Client Addin (Running in Genesys Desktop Client)
- Screen Pop App (SmartAppsPop)
- Native Messaging Host (exe)
- Browser Extension (Chrome or Edge)
- Fiserv XP2 Web Application

Message Flow

- Interaction detected** – The SmartApps Client Addin determines an XP2 screen pop is required.
- Screen pop request sent** – The Addin invokes the SmartApps custom URL protocol (smartappspop:) with XP2 parameters (e.g., base URLs and account).

Example:

```
webextension,screenpop,{"type":"XP","XP2_BaseURL":"https://xp2web1|https://xp2web2","IVR_Account":"123456"},chrome
```

- Screen Pop App launched** – The OS launches the Screen Pop App, which parses the request and recognizes an XP2 browser-based pop.
- Routed to Native Messaging Host** – The Screen Pop App forwards the request to the **Native Messaging Host (exe)** to reach the browser extension.
- Extension routing** – The Native Messaging Host relays the message to the browser extension’s background.js. background.js calls XP2Pop().
- Browser tab located/opened** – The extension searches for an XP2 tab using the **pipe-delimited** base URLs; opens a new tab if none found.
- In-page action** – background.js sends data to content.js in the XP2 tab; content.js performs the navigation/search to the account.

Notes

- Desktop execution path: Client Addin → Protocol Handler → Screen Pop App → Native Messaging Host → Browser Extension → XP2.
- Extension handlers (XP2Pop) perform tab search and in-page actions via content.js.



Fiserv Spectrum Branchsuite

Spectrum Branchsuite is a web application which is screen popped using Browser Extension (Chrome or Edge). While the Browser Extension is required to execute the screen pop, the SmartAppsPop is only required to be installed if the Genesys Desktop client. If the Genesys agent experience is used within a chrome or edge browser, the SmartApps Agent experience will send the pop request directly to the browser extension without using the SmartAppsPop. When using the Genesys Desktop client, the SmartApps Agent experience executes the SmartAppsPop through the Custom URL Protocol and then the SmartAppsPop sends the request to the running browser extension to complete the screen pop

Genesys web app within Chrome/Edge browser

When using Genesys Agent experience within Chrome/Edge browser, set up the screen pop using the settings in the table below.

Field	Value
Client Addin Type	"Spectrum"
Use Screen Pop Assistant App	Off
Description	"Branchsuite" (or whatever description preferred)
Fiserv Spectrum Branch Suite Settings	<p>https://branchsuitelive.customercu.com/</p> <p>The base web URL for Branchsuite. The browser extension uses the URL to locate any running browser tabs which are running Branchsuite. Once the extension locates a browser tab, it will inject javascript on the page to execute a search using the Account Number.</p>

Message Flow

Actors

- SmartApps Client Addin (Running in browser)
- Browser Extension (Chrome or Edge)
- Branchsuite website

Message Flow



1. **Interaction detected** – The SmartApps Client Addin determines an Branchsuite screen pop is required.
2. **Screen pop request sent (direct)** – The Addin sends the Branchsuite request directly to the browser extension (no Screen Pop App / no Native Host).
3. **Browser tab located** – The browser extension searches for an existing BranchSuite tab using the configured base URL. If no matching tab is found, a new tab is opened.
4. **BranchSuite navigation executed** – The extension injects logic into the BranchSuite page to perform an account search using the provided account number.

Notes

- Web flow bypasses the Screen Pop App and Native Messaging Host; the Addin talks straight to the extension. Web client execution path: Client Addin → Browser Extension → BranchSuite.
- Browser extension handles tab discovery and in-page interaction with Branchsuite.

Genesys Desktop Client (Installed Application)

When using Genesys Desktop Client installed application, set up the screen pop using the settings in the table below.

Field	Value
Client Addin Type	"Custom Application"
Description	"Branchsuite" (or whatever description preferred)
Protocol Handler App	"smartappspop"
Parameter(s)	<p>webextension,screenpop,{"type":"SP","BranchSuiteScreenPopURL":"http://branchsuitelive.customercu.com/","IVR_Account":"12345","BranchSuiteBrowser":""},chrome</p> <ul style="list-style-type: none"> • webextension,screenpop is the first and second parameter sent to the SmartAppsPop application, so it knows it is sending a screenpop request to the browser extension. • The third parameter is a json object which describes the screen pop. <ul style="list-style-type: none"> ○ type - SP (Type of screen pop to execute)



	<ul style="list-style-type: none"> ○ BranchSuiteScreenPopURL – Branchsuite URL to locate when looking for the web tab running Branchsuite. ○ IVR_Account – Set to the Account Number to pop. {{Account Number}} is replaced during the interaction with the opened account within the SmartApps Agent experience. ● The last parameter is optional specifying either chrome or edge to ensure the screen pop is only sent to an extension running in the desired browser.
--	--

Message Flow

Actors

- SmartApps Client Addin (Running in Genesys Desktop Client)
- Screen Pop App (SmartAppsPop)
- Native Messaging Host (exe)
- Browser Extension (Chrome or Edge)
- Branchsuite website

Message Flow

1. **Interaction detected** – The SmartApps Client Addin detects a new Genesys interaction and determines a BranchSuite screen pop is required.
2. **Screen pop request sent** – The Addin invokes the SmartApps custom URL protocol (smartappspop:) with BranchSuite screen pop parameters.
 1. **Example:**
`webextension,screenpop,{"type":"SP","BranchSuiteScreenPopURL":"https://branchsuite.live.customer.com/","IVR_Account":"12345"},chrome`
3. **Screen Pop App launched** – The operating system launches the Screen Pop App, which parses the request and identifies it as a browser-based BranchSuite screen pop.
4. **Routed to Native Messaging Host** – The Screen Pop App forwards the screen pop request to the Native Messaging Host (exe) for delivery to the browser extension.
5. **Extension routing** – The Native Messaging Host relays the request to the browser extension. The extension’s background.js processes the request and invokes the BranchSuite handler.



- 6. **Browser tab located** – The browser extension searches for an existing BranchSuite tab using the configured base URL. If no matching tab is found, a new tab is opened.
- 7. **BranchSuite navigation executed** – The extension injects logic into the BranchSuite page to perform an account search using the provided account number.

Notes

- Desktop execution path: Client Addin → Protocol Handler → Screen Pop App → Native Messaging Host → Browser Extension → Branchsuite.
- Browser extension handles tab discovery and in-page interaction with Branchsuite

Temenos

Temenos is a web application which is screen popped using Browser Extension (Chrome or Edge). While the Browser Extension is required to execute the screen pop, the SmartAppsPop is only required to be installed if the Genesys Desktop client. If the Genesys agent experience is used within a chrome or edge browser, the SmartApps Agent experience will send the pop request directly to the browser extension without using the SmartAppsPop. When using the Genesys Desktop client, the SmartApps Agent experience executes the SmartAppsPop through the Custom URL Protocol and then the SmartAppsPop sends the request to the running browser extension to complete the screen pop

Genesys web app within Chrome/Edge browser

When using Genesys Agent experience within Chrome/Edge browser, set up the screen pop using the settings in the table below.

Field	Value
Client Addin Type	"Temenos"
Use Screen Pop Assistant App	Off
Description	"Temenos" (or whatever description preferred)
Temenos Settings	
Search Type	Defines which value to search by which only ACCTNUMBER is currently supported.
Suppress Real Time Refresh	Real-time refresh will automatically update the screen. By disabling this, the Temenos screen will not automatically refresh upon screen pop.



Workgroups	No longer used
------------	----------------

Message Flow

Actors

- SmartApps Client Addin (Running in browser)
- Browser Extension (Chrome or Edge)
- Branchsuite website

Message Flow

1. **Interaction detected** – The SmartApps Client Addin detects a new Genesys interaction and determines a Temenos screen pop is required.
2. **Screen pop request sent (direct)** – The Addin sends the Temenos screen pop request directly to the browser extension (no Screen Pop App or Native Messaging Host involved).
3. **Browser tab located** – The browser extension locates an existing Temenos tab or opens a new tab if none exists.
4. **Temenos navigation executed** – The extension injects logic into the Temenos page to search for and open the specified account.

Genesys Desktop Client (Installed Application)

When using Genesys Desktop Client installed application, set up the screen pop using Site Manager in Screen Pop -> Application Definition using the settings in the table below.

Field	Value
Client Addin Type	"Custom Application"
Description	"Temenos" (or whatever description preferred)
Protocol Handler App	"smartappspop"
Parameter(s)	<pre>webextension,screenpop,{"type":"AK", adapt_AkcSuppressRealtimeRefresh":0,"adapt_AkcSearchType":"ACCTID ","IVR_Account":"{{Account Number}}"},chrome</pre> <ul style="list-style-type: none"> • webextension,screenpop is the first and second parameter sent to the SmartAppsPop application, so it knows it is sending a screenpop request to the browser extension.



	<ul style="list-style-type: none">• The third parameter is a json object which describes the screen pop.<ul style="list-style-type: none">○ type - AK (Type of screen pop to execute)○ adapt_AkcSearchType – Search type within Temenos. Only value supported is ACCTID.○ IVR_Account – Set to the Account Number to pop. {{Account Number}} is replaced during the interaction with the opened account within the SmartApps Agent experience.• The last parameter is optional specifying either chrome or edge to ensure the screen pop is only sent to an extension running in the desired browser.
--	--

Message Flow

Actors

- SmartApps Client Addin (Running in Genesys Desktop Client)
- Screen Pop App (SmartAppsPop)
- Native Messaging Host (exe)
- Browser Extension (Chrome or Edge)
- Temenos website

Message Flow

1. **Interaction detected** – The SmartApps Client Addin detects a new Genesys interaction and determines a Temenos screen pop is required.
2. **Screen pop request sent** – The Addin invokes the SmartApps custom URL protocol (smartappspop:) with Temenos screen pop parameters.
 1. **Example:**

```
webextension,screenpop,{"type":"AK","adapt_AkcSearchType":"ACCTID","IVR_Account":"123456"},chrome
```
3. **Screen Pop App launched** – The operating system launches the Screen Pop App, which parses the request and identifies it as a browser-based Temenos screen pop.
4. **Routed to Native Messaging Host** – The Screen Pop App forwards the screen pop request to the Native Messaging Host (exe) for delivery to the browser extension.



5. **Extension routing** – The Native Messaging Host relays the request to the browser extension. The extension’s background.js processes the request and invokes the Temenos screen pop handler.
6. **Browser tab located** – The browser extension locates an existing Temenos tab or opens a new tab if one is not already running.
7. **Temenos navigation executed** – The extension injects logic into the Temenos page to search for and open the specified account.

Notes

- Desktop execution path: Client Addin → Protocol Handler → Screen Pop App → Native Messaging Host → Browser Extension → Temenos.
- Browser extension handles tab discovery and in-page interaction

Generic URL

Generic URL screen pop is the ability to pop to any URL. There are numerous approaches to screen popping to a URL application. SmartApps can open a new browser tab with the URL but this may not be ideal as every call will cause a new browser tab to open using the default browser. Alternatively, it can screen pop to a new tab to a specific browser (chrome or edge). Finally, it can screen pop to the desired browser while looking for a specific browser tab with a specific URL to match and then replace it with the new URL. The last approach is ideal for scenarios where separate browser tabs should not be opened for the screen popped application.

Screen Pop new browser tab

When using Genesys Agent experience within Chrome/Edge browser, the screen pop will be to a new tab within that browser. This can be used in both using the Genesys web client in the browser as well as within the Genesys Desktop Client. When using the Genesys Desktop Client, the new tab will be opened in the local machine’s registered browser.

Field	Value
Client Addin Type	“URL”
Description	Enter description for screen pop
URL	Desired URL to pop using replacement tags like {{Account Number}} and {{Participant Attribute.GenesysAttribute}} to replace from Genesys Participant attributes



Message Flow

Actors

- SmartApps Client Addin (Running in Genesys Desktop Client)
- Web Browser (Default browser or specified browser)

Message Flow

1. **Interaction detected** – The SmartApps Client Addin detects a new Genesys interaction and determines that a URL screen pop to a new browser tab is required.
2. **Screen pop request sent** – The Addin issues a URL screen pop request containing the destination URL, including any dynamic replacement values (for example, account number or participant attributes).
3. **Browser launch initiated** –
 - If running in the **Genesys Web client**, the browser opens a new tab directly.
 - If running in the **Genesys Desktop Client**, the operating system opens a new tab in the registered default browser.
4. **URL loaded** – The browser opens a new tab and navigates to the resolved destination URL.

Notes

- **Opens a new browser tab for each screen pop**
- No browser extension required
- Execution path: Client Addin → Browser
- Does not search for or reuse existing browser tabs

Screen Pop to specific browser type

When needing to browser to a specific browser type regardless of which browser is the default registered browser, use this screen pop definition. This approach does not require use of the web extension but does require the SmartAppsPop to be installed on each machine.

Field	Value
Client Addin Type	"Custom Application"
Description	Enter description for screen pop
Protocol Handler App	"smartappspop"



Parameter(s)	<p>browser,chrome,https://www.google.com/search?q={{Account Number}}</p> <ul style="list-style-type: none"> • browser is the first parameter sent to the SmartAppsPop application to indicate popping to a URL. • The second parameter is the browser type to open the screen. Valid values: chrome or edge. • The third parameter is the URL to pop where replacement parameters can be included. The example above is opening Google search and putting the Account Number in the search box.
--------------	--

Message Flow

Actors

- SmartApps Client Addin (Running in browser or Genesys Desktop Client)
- Screen Pop App (SmartAppsPop)
- Web Browser (Chrome or Edge)

Message Flow

1. **Interaction detected** – The SmartApps Client Addin detects a new Genesys interaction and determines a screen pop to a specific browser type is required.
2. **Screen pop request sent** – The Addin invokes the SmartApps custom URL protocol (smartappspop:) with parameters indicating a browser-based URL pop.
 1. **Example:**
`smartappspop:browser,chrome,https://www.google.com/search?q={AccountNumber}`
3. **Screen Pop App launched** – The operating system launches the Screen Pop App, which parses the request and identifies it as a URL screen pop targeting a specific browser.
4. **Browser launched** – The Screen Pop App launches the specified browser (Chrome or Edge), regardless of the system default browser.
5. **URL loaded** – The browser opens a new tab and navigates to the resolved destination URL.

Notes

- Browser type is explicitly controlled (Chrome or Edge)
- No browser extension required



- Execution path: Client Addin → Protocol Handler → Screen Pop App → Open Browser from command line.
- SmartAppsPop must be installed on the agent machine

Screen Pop replacing an existing browser tab

When needing to pop to a browser tab by finding an existing tab with a matching URL. The ScreenPopApp will send the request to any browser running with the browser extension. The browser extension then searches for any tabs which have a matching URL and then replaces the URL based on the configuration below.

Field	Value
Client Addin Type	"Custom Application"
Description	Enter description for screen pop
Protocol Handler App	"smartappspop"
Parameter(s)	<pre>webextension,screenpop,{"type":"URL","BaseURL":"google.com","ReplaceURL":"https://www.google.com/search?q={{Account Number}}"},chrome</pre> <ul style="list-style-type: none"> • webextension,screenpop is the first and second parameter sent to the SmartAppsPop application, so it knows it is sending a screenpop request to the browser extension. • The third parameter is a json object which describes the screen pop. <ul style="list-style-type: none"> ○ type - URL (Type of screen pop to execute) ○ BaseURL – Pipe delimited list of base web URLs to locate when looking for the web tab running the desire web application. ○ ReplaceURL – URL to replace found browser tab where any parameters are replaced with tags. For example, the above URL, {{Account Number}} is replaced during the interaction with the opened account within the SmartApps Agent experience.



	<ul style="list-style-type: none">• The last parameter is optional specifying either chrome or edge to ensure the screen pop is only sent to an extension running in the desired browser.
--	---

Message Flow

Actors

- SmartApps Client Addin (Running in Genesys Desktop Client)
- Screen Pop App (SmartAppsPop)
- Native Messaging Host (exe)
- Browser Extension (Chrome or Edge)
- Web Application (Target URL)

Message Flow

1. **Interaction detected** – The SmartApps Client Addin detects a new Genesys interaction and determines a URL screen pop that replaces an existing browser tab is required.
2. **Screen pop request sent** – The Addin invokes the SmartApps custom URL protocol (smartappspop:) with parameters indicating a browser-extension-based URL replacement screen pop.
 1. **Example:**
`webextension,screenpop,{"type":"URL","BaseURL":"google.com","ReplaceURL":"https://www.google.com/search?q={AccountNumber}"},chrome`
3. **Screen Pop App launched** – The operating system launches the Screen Pop App, which parses the request and identifies it as a browser-based URL replacement screen pop.
4. **Routed to Native Messaging Host** – The Screen Pop App sends the screen pop request to the Native Messaging Host (exe) to communicate with the browser extension.
5. **Request delivered to browser extension** – The Native Messaging Host relays the request to the browser extension’s background process.
6. **Existing tab located** – The browser extension searches all open browser tabs for a URL matching the configured BaseURL.
7. **URL replaced** – When a matching tab is found, the browser extension replaces the tab’s URL with the resolved ReplaceURL.

Notes

- Reuses an existing browser tab instead of opening a new one



- Browser extension is required
- Execution path: Client Addin → Protocol Handler → Screen Pop App → Native Messaging Host → Browser Extension
- Optional browser parameter (chrome or edge) limits delivery to a specific browser

Troubleshooting

If there are any issues in testing, a log file will be created located at %TEMP%\SmartAppsPop.log on the agent's desktop.