



Avaya Solution & Interoperability Test Lab

Application Notes for Cicero with Avaya Communication Manager and Avaya IP Agent – Issue 1.0

Abstract

These Application Notes describe the steps for configuring Avaya IP Agent to start a screen pop application developed and deployed with Cicero when an incoming call is received. Cicero is a framework for integrating an organization's enterprise software applications and sharing data between the applications. In this solution, Avaya IP Agent, the Cicero Call Pop application, and Internet Explorer were integrated in the Cicero Agent Desktop. When calls are received by Avaya IP Agent, the calling party number is passed to the Cicero Call Pop application using the Screen Pops feature. The Call Pop application can then pass the caller number to other 3rd party business applications. In this configuration, the caller number was passed to Internet Explorer with access to Salesforce.com, which then displayed caller information stored in its database. The Call Pop application is part of the Cicero Avaya IP Agent Productivity Pack.

Information in these Application Notes has been obtained through DevConnect compliance testing and additional technical discussions. Testing was conducted via the DevConnect Program at the Avaya Solution and Interoperability Test Lab.

1. Introduction

These Application Notes describe the steps for configuring Avaya IP Agent to start a screen pop application developed and deployed with the Cicero Platform when an incoming call is received. Cicero is a framework for integrating an organization's enterprise software applications and sharing data between the applications. Cicero includes a comprehensive toolset called Cicero Studio and a Runtime environment, which is installed where the applications are used. In this solution, Avaya IP Agent, the Cicero Call Pop application, and Internet Explorer were integrated in the Cicero Agent Desktop as shown in **Figure 1**. When calls are received by Avaya IP Agent, the calling party number is passed to the Cicero Call Pop application using the Screen Pops feature. The Call Pop application then passes the caller number to other 3rd party business applications. In this configuration, the caller number was passed to Internet Explorer with access to Salesforce.com, which then displayed caller information stored in its database for the agent. Integrating business applications with Cicero improves business processes and productivity because users no longer have to manually enter data in different business applications. The Call Pop application is part of the Cicero Avaya IP Agent Productivity Pack and is available upon request from a Cicero Support Representative.

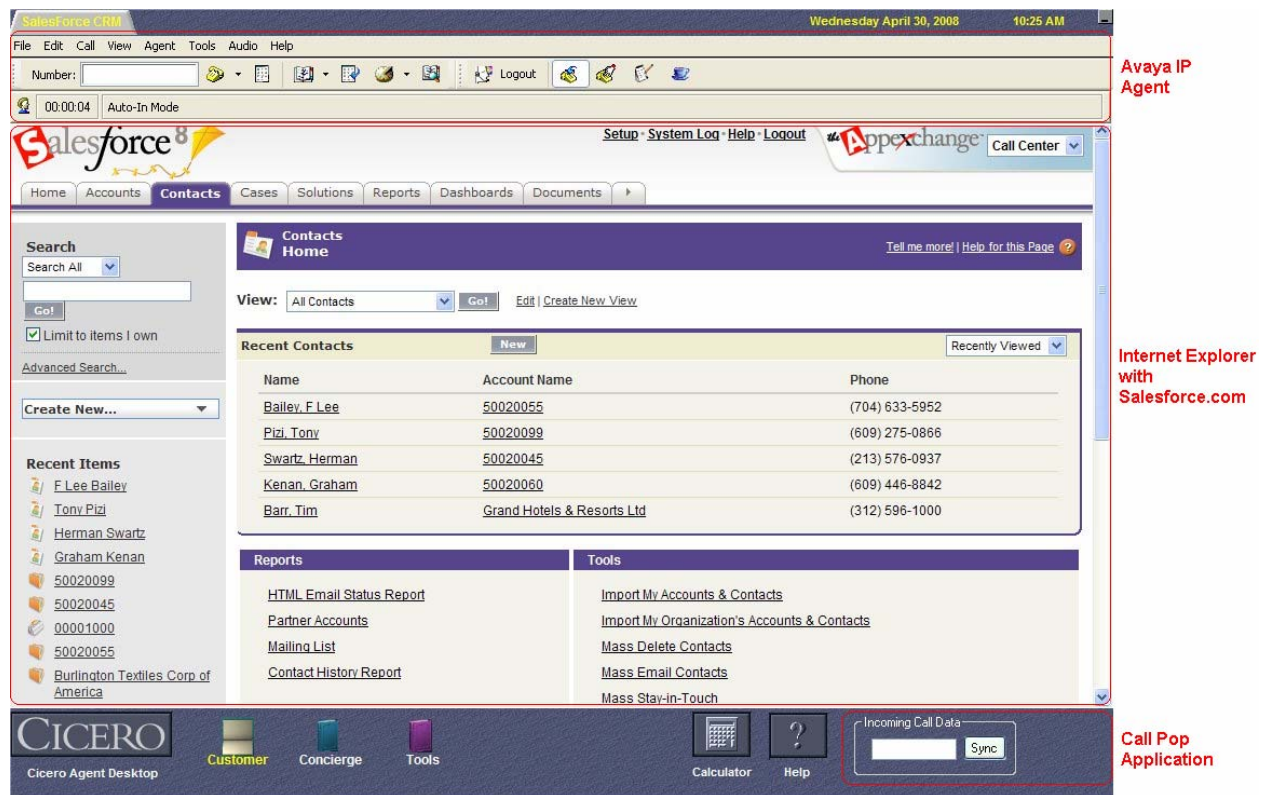


Figure 1: Cicero Agent Desktop

Figure 2 illustrates the configuration used for testing. In this configuration, Avaya Communication Manager received incoming calls from the simulated PSTN over an ISDN trunk

and routed the calls to an agent in the ACD call center. Avaya IP Agent, which was running on the Cicero Agent Desktop, was registered to Avaya Communication Manager and logged in as an agent in the ACD skill group. Avaya IP Agent and the Cicero Agent Desktop were running on the same Windows PC.

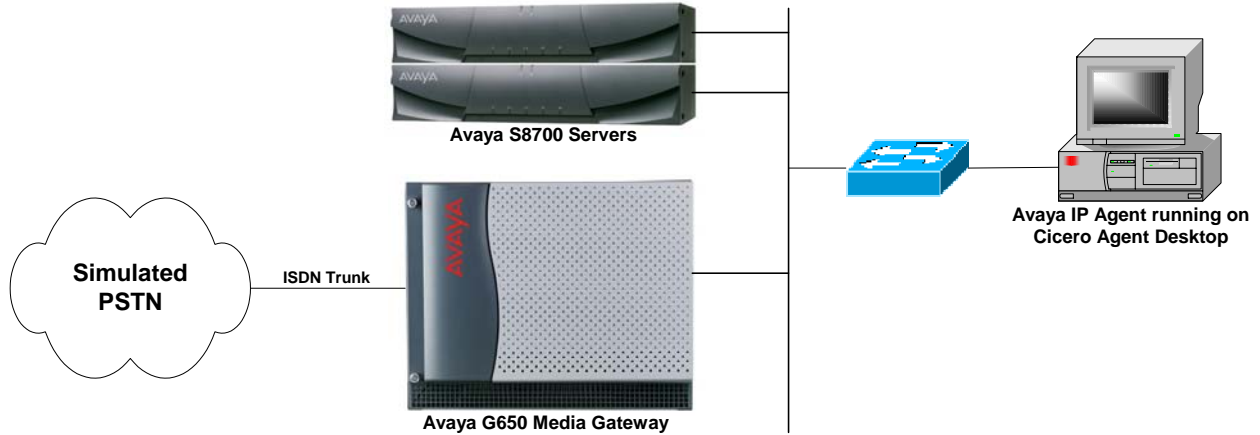


Figure 2: Configuration with Avaya Communication Manager, Avaya IP Agent and Cicero

1.1. Equipment and Software Validated

The following equipment and software were used for the sample configuration:

Equipment	Software
Avaya S8700 Servers with a G650 Media Gateway	Avaya Communication Manager 4.0 (R014x.00.1.731.2)
Avaya IP Agent	7.0.23.116
Cicero	6.3

2. Configure Avaya Communication Manager

Avaya Communication Manager provided PSTN connectivity and ACD call center functionality. For PSTN connectivity, an ISDN trunk was used. This required a trunk group and signaling group to be configured. The ISDN trunk must be configured on the PSTN side to send the calling party number. However, to successfully pass the calling party number from Avaya IP Agent to the screen pop application, the ISDN trunk on the PSTN side must be configured to *not* send the calling party name.

Avaya Communication Manager required configuration of a Vector Directory Number (VDN), Vector, Hunt Group/Skill, and Agent Login IDs for the ACD call center functionality. In addition, an IP station was required for Avaya IP Agent. The configuration of Avaya Communication Manager is outside the scope of these Application Notes, refer to [1] and [2] for information on configuring an ACD call center and the IP station for Avaya IP Agent. For this configuration, an IP station with type of '8434D' was used, but other options are also available.

3. Configure Avaya IP Agent

This section covers the configuration of the Screen Pops feature in Avaya IP Agent. When an incoming call is received by Avaya IP Agent, it would pass the calling party number to the Cicero Call Pop application. The calling party number is passed using the %m parameter as described below.

From the menu bar of the Avaya IP Agent window, select **Tools→Screen Pops** as shown in **Figure 3**. Avaya IP Agent displays the **Screen Pops** window shown in **Figure 4**.

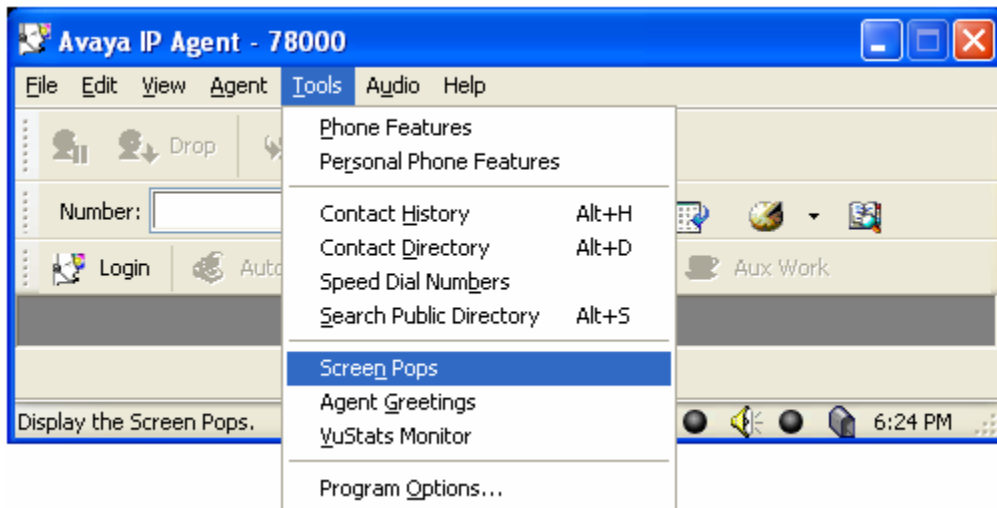


Figure 3: Avaya IP Agent Tools Menu

In the **Screen Pops** window below, select **File→New** to display the **Screen Pops Wizard** as shown in **Figure 5**.

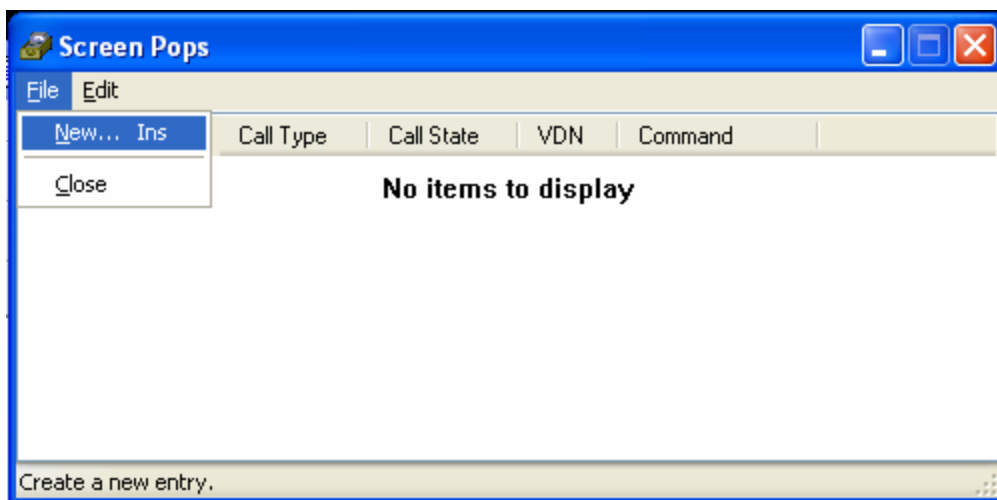


Figure 4: Screen Pops

In the **Screen Pops** window, enter a name for this screen pop in the text box. *Cicero_IPAgent* was used in this configuration. Select the **Windows Explorer** option button and click the **Next** button.

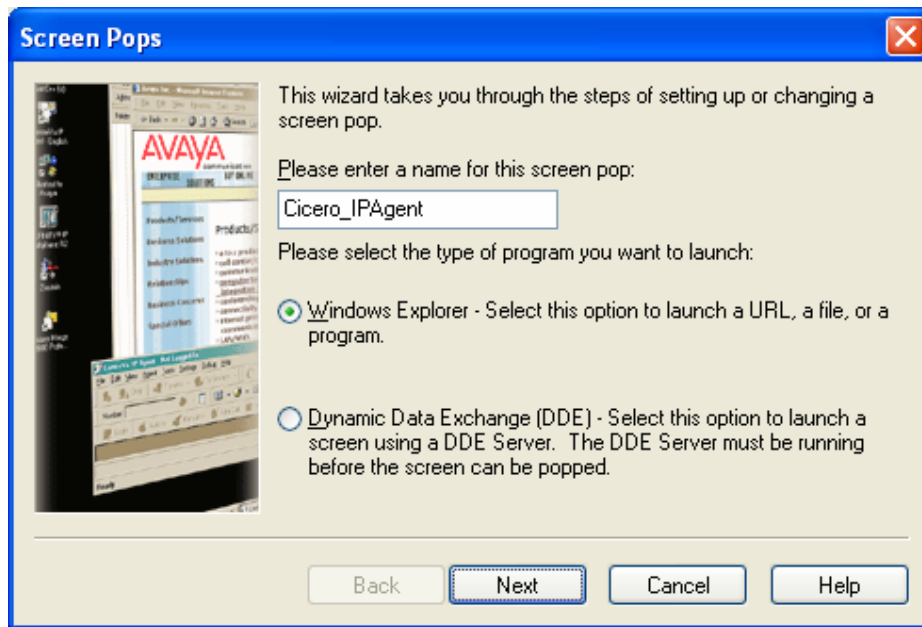


Figure 5: Screen Pops Wizard

In the **Screen Pops – Trigger** window, select *Ring* or *Answered* as the trigger condition. If *Ring* is selected, the screen pop application will start when Avaya IP Agent receives an incoming call. If *Answered* is selected, the screen pop application will start when an incoming call has been answered. Click the **Next** button.

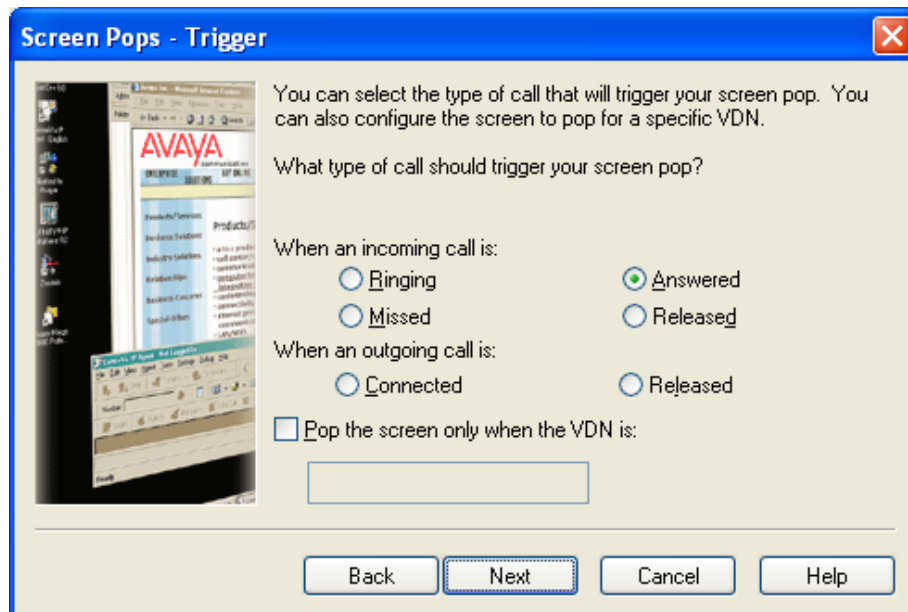


Figure 6: Screen Pops - Trigger

Next, Avaya IP Agent displays the **Screen Pop – Action** window. In this window, specify the program name of the screen pop application. The program name for the Cicero Call Pop Application is:

```
C:\CiceroData\Projects\CiceroCallCenterDemo\Applications\AvayaIPAgent_Callpops\bin\CallPopUp.exe %m
```

CallPopUp.exe is a Cicero application file that was stored in the directory specified above when the Cicero demo project was imported. See... The %m parameter included in the program name passes the telephone number of the calling party, if available. The caller number will be passed to the 3rd party business application to retrieve caller information from a database. In this example, the caller number is passed to SalesForce.com, which displayed additional caller information for the agent. Click the **Next** button.

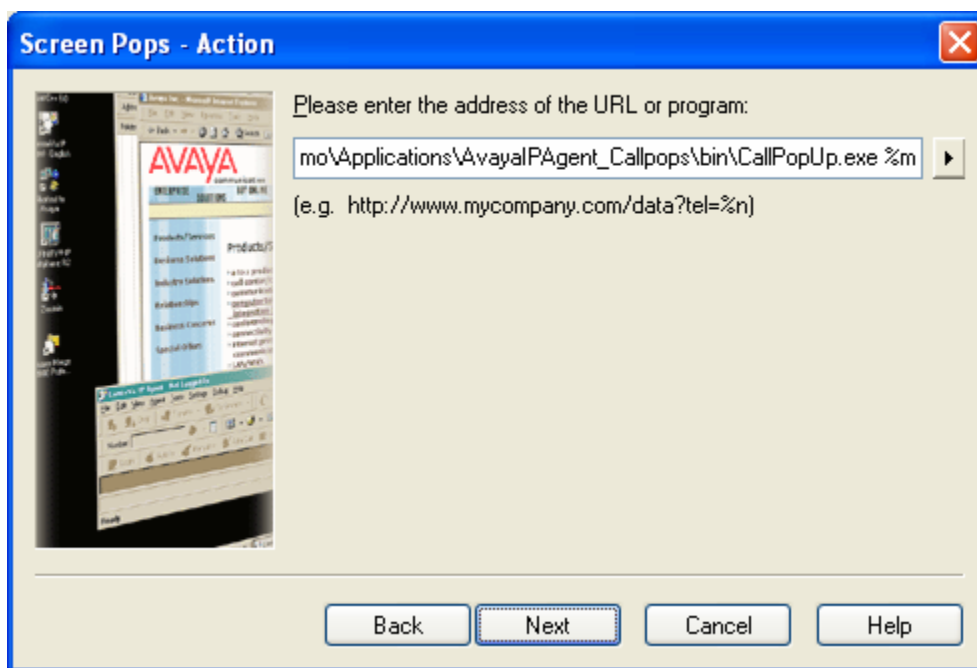


Figure 7: Screen Pops - Action

In the **Screen Pops – Format Call Information** window, specify the number of digits of the caller number to use for the screen pop. In this example, all the digits were passed to the screen pop. Click the **Next** button.

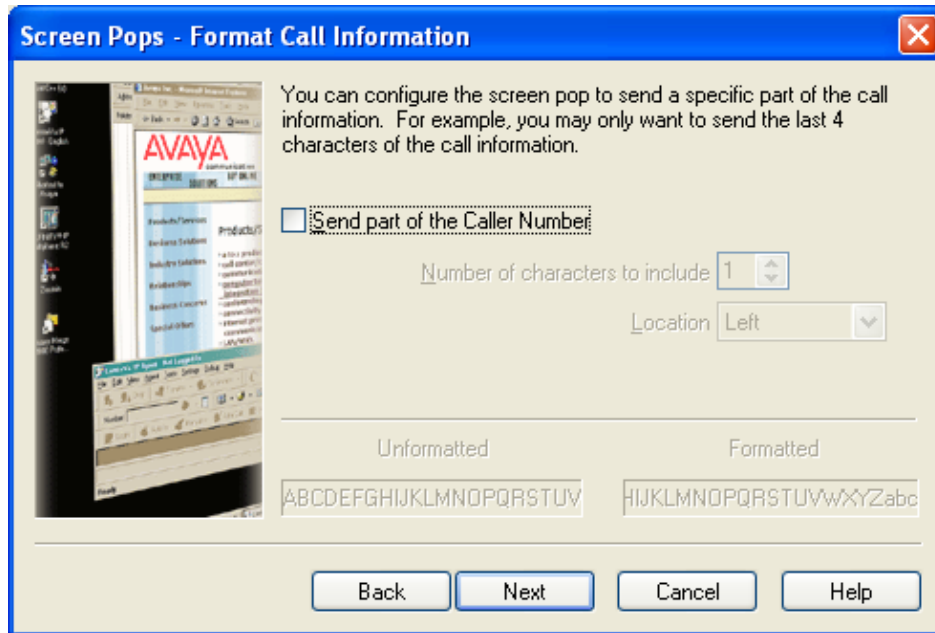


Figure 8: Screen Pops – Format Call Information

From the **Screen Pops – Testing** window, select the **Test** button to verify that the configuration of this screen pop works as intended, then click the **Next** button.

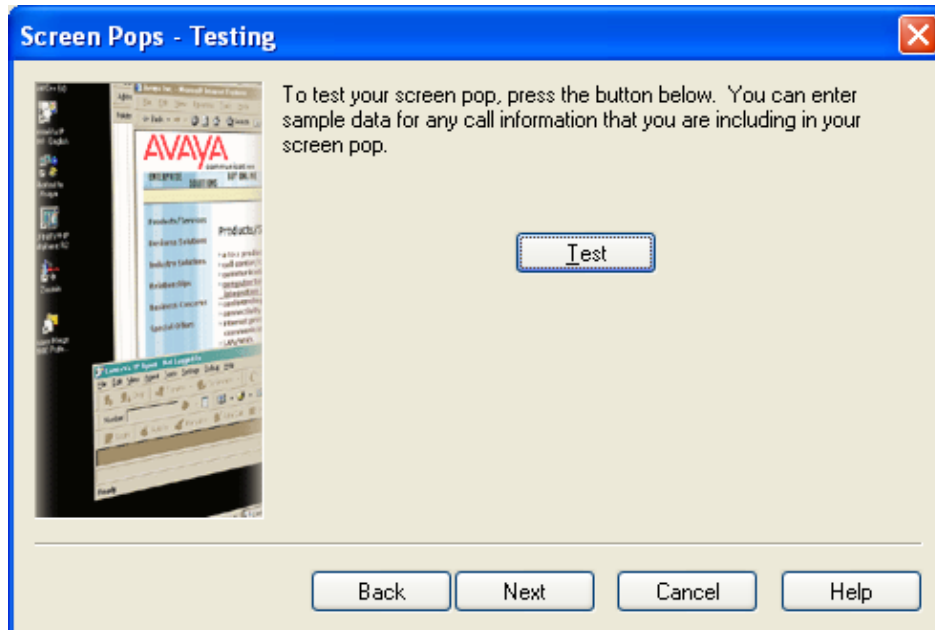


Figure 9: Screen Pops – Testing

In the **Screen Pops – Setup Completed** window, click the **Finish** button.

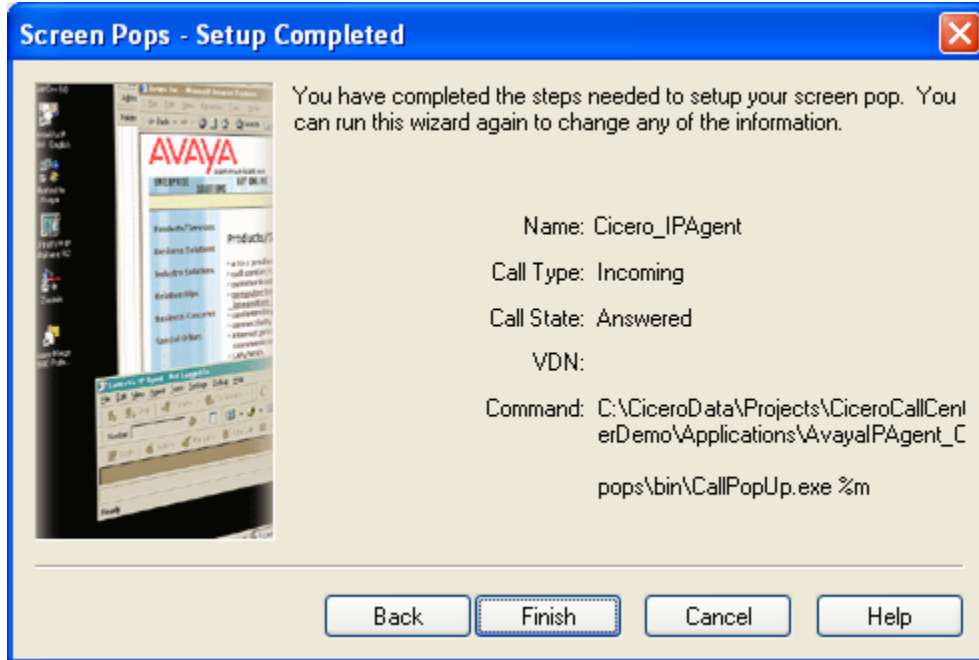


Figure 10: Screen Pops – Setup Completed

Avaya IP Agent saves this screen pop and displays it in the **Screen Pops** window shown in **Figure 11**. Select the *Cicero_IPAgent* application in the list to activate it.

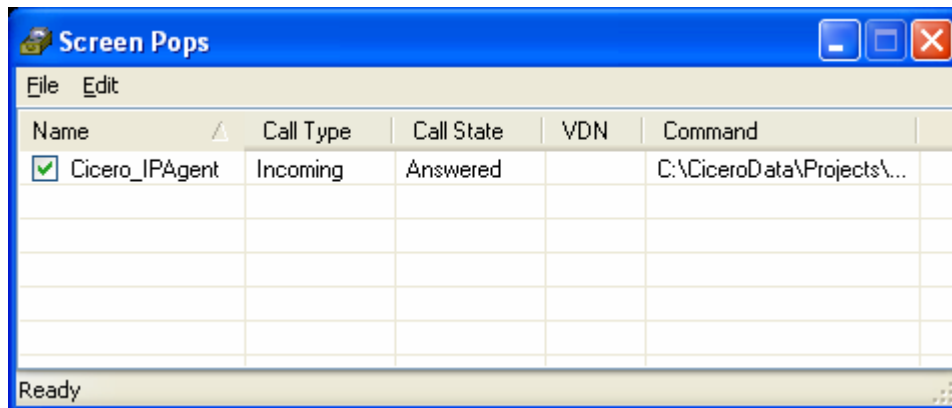


Figure 11: Screen Pops

4. Configure Cicero

The Cicero Agent Desktop, shown in **Figure 12**, is a “project” developed with Cicero Studio. A “project” is the container for a set of applications. The Cicero Agent Desktop is comprised of Avaya IP Agent, the Call Pop application, and Internet Explorer with Salesforce.com, a 3rd party application. The Cicero Runtime environment runs on the same Windows PC where the Cicero Agent Desktop is installed. Refer to [3] for information on developing and deploying applications with Cicero. It is assumed that the Cicero Agent Desktop has already been developed and deployed to the agent desktops. This section will provide a brief description on how to use the Cicero Agent Desktop for screen pop applications.

When the Cicero Agent Desktop is started, the user is prompted to log in Avaya IP Agent. Once logged in, the Cicero Agent Desktop appears as shown in **Figure 12**.

The screenshot displays the Cicero Agent Desktop interface. At the top, there is a menu bar with options: File, Edit, Call, View, Agent, Tools, Audio, Help. Below the menu bar is a search bar with a 'Number:' field and a 'Logout' button. The main content area is divided into several sections:

- Search:** A search bar with a 'Search All' dropdown, a 'Go!' button, and a checkbox for 'Limit to items I own'. There is also an 'Advanced Search...' link and a 'Create New...' dropdown.
- Recent Items:** A list of recent items including 'F Lee Bailey', 'Tony Pizi', 'Herman Swartz', 'Graham Kenan', '50020099', '50020045', '00001000', '50020055', and 'Burlington Textiles Corp of America'.
- Contacts Home:** A section with a 'View:' dropdown set to 'All Contacts', a 'Go!' button, and a 'Create New View' link.
- Recent Contacts:** A table with columns 'Name', 'Account Name', and 'Phone'. The data is as follows:

Name	Account Name	Phone
Bailey, F Lee	50020055	(704) 633-5952
Pizi, Tony	50020099	(609) 275-0866
Swartz, Herman	50020045	(213) 576-0937
Kenan, Graham	50020060	(609) 446-8842
Barr, Tim	Grand Hotels & Resorts Ltd	(312) 596-1000
- Reports:** A list of reports including 'HTML Email Status Report', 'Partner Accounts', 'Mailing List', and 'Contact History Report'.
- Tools:** A list of tools including 'Import My Accounts & Contacts', 'Import My Organization's Accounts & Contacts', 'Mass Delete Contacts', 'Mass Email Contacts', and 'Mass Stay-in-Touch'.

At the bottom of the interface, there is a 'CICERO' logo and the text 'Cicero Agent Desktop'. There are also icons for 'Customer', 'Concierge', and 'Tools'. On the right side, there is a 'Calculator' icon, a 'Help' icon, and an 'Incoming Call Data' field with a 'Sync' button.

Figure 12: Cicero Agent Desktop

When Avaya IP Agent receives an incoming call, it will pass the caller number to the Call Pop application. This event is triggered when the incoming call is in the ringing or answered state depending on the configuration. In this case, the caller number was passed to the Call Pop application after the incoming call was answered. Note that the text box of the Call Pop application was automatically populated with “21000”, the caller number. In addition, the **Sync** button starts to flash.

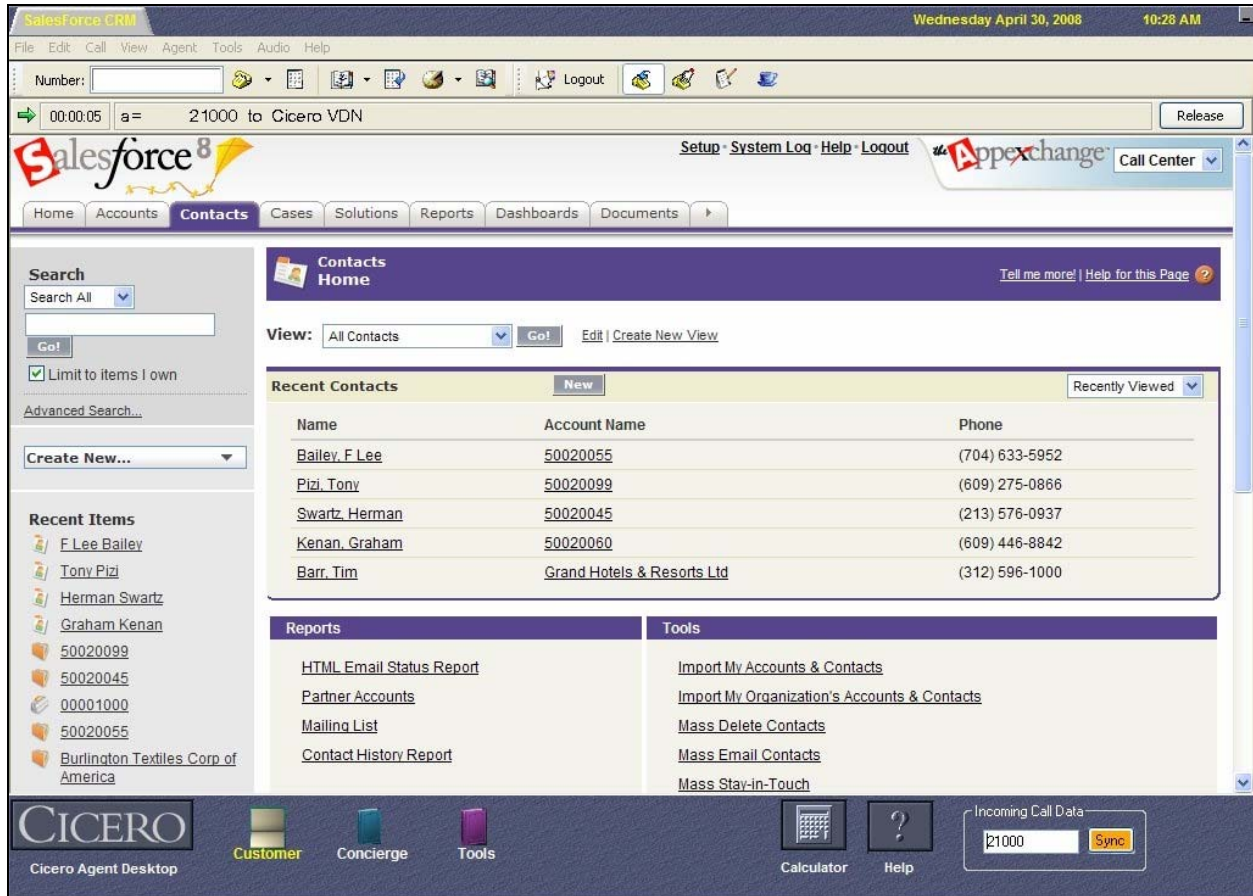


Figure 13: Cicero Agent Desktop with Caller Number in Call Pop Application

Once the Call Pop application receives the caller number and flashes the **Sync** button, the caller number can then be passed to a 3rd party business application, in this case Salesforce.com, by clicking on the **Sync** button. Salesforce.com then retrieves and displays caller information from its database as shown in **Figure 14**. Integrating business applications with Cicero allows agents to share data between applications easily, with little manual intervention.

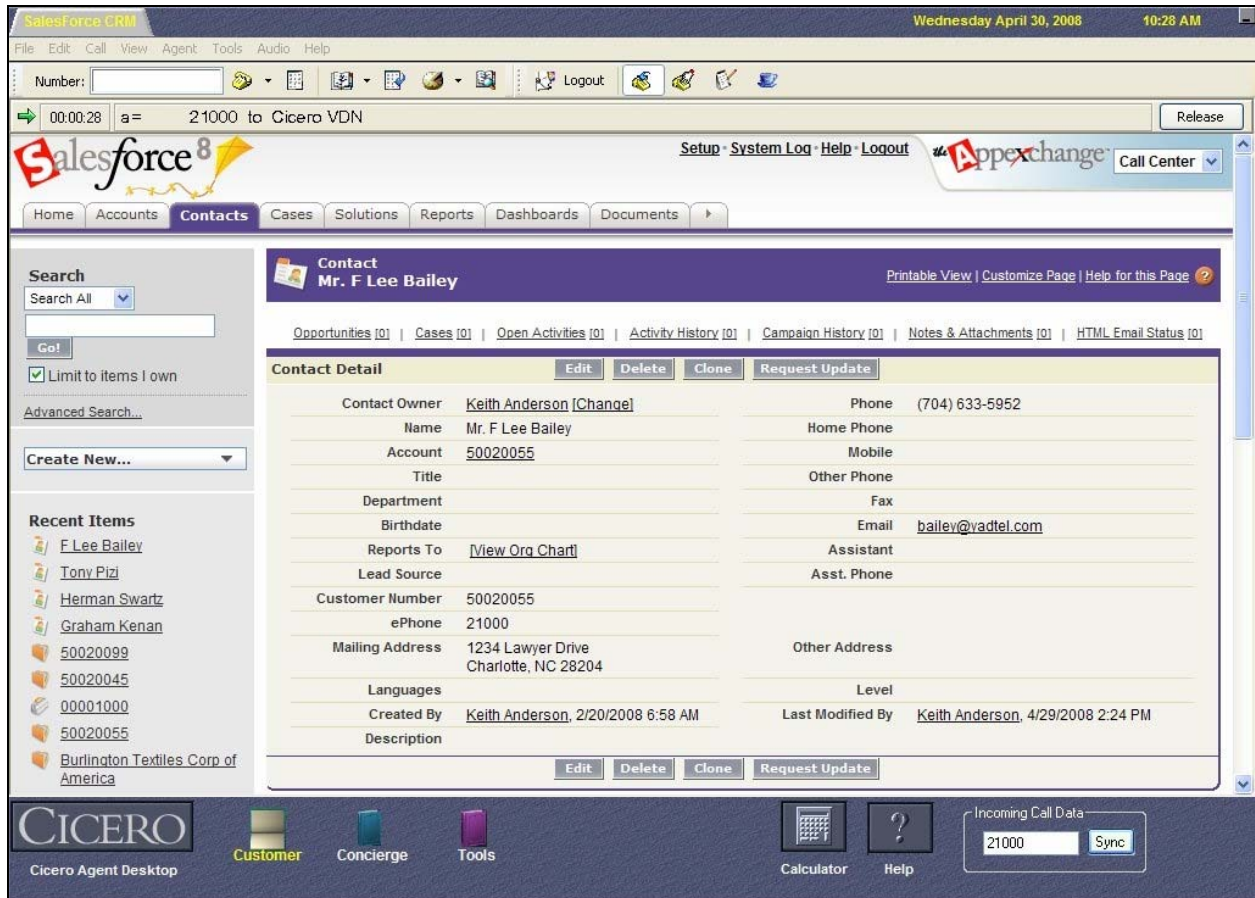


Figure 14: Cicero Agent Desktop with Salesforce.com Caller Lookup

5. Interoperability Compliance Testing

This section describes the interoperability compliance testing used to verify that Avaya IP Agent passes the calling party number to the Call Pop application on the Cicero Agent Desktop.

5.1. General Test Approach

The interoperability compliance test focused on feature testing and verified that Avaya IP Agent would successfully pass the calling party number to the Call Pop application on the Cicero Agent Desktop when an incoming call was in the ringing or answered state, depending on the configuration. Although not required for the compliance test, it was also verified that Cicero could pass the calling party number to a 3rd party business application, such as Salesforce.com.

5.2. Test Results

All test cases passed. Avaya IP Agent was successful in passing the calling party number to the Call Pop application on the Cicero Agent Desktop. However, it was required that the ISDN trunk configuration between the simulated PSTN and Avaya Communication Manager be configured to *not* send the calling party name.

6. Verification Steps

This section provides the verification steps that may be performed to verify that Avaya IP Agent and the screen pop application is properly configured.

1. From the System Access Terminal (SAT) of Avaya Communication Manger, use the **list registered-ip-stations** command to verify that Avaya IP Agent is successfully registered.
2. From the SAT, use the **monitor bcms skill <skill #>** to verify that the agent is logged into the skill, if supported for the skill.
3. Test the screen pop application from the **Screen Pops – Testing** window shown in **Figure 9**.

7. Support

For Cicero technical support, call 1-866-538-3588 or contact them through their website at <http://www.level8.com/contact.html>.

8. Conclusion

These Application Notes describe the steps for configuring Avaya IP Agent to start the Call Pop application that is part of the Cicero Avaya IP Agent Productivity Pack when an incoming call is received. All feature test cases were completed successfully.

9. Additional References

This section references the product documentation that is relevant to these Application Notes.

- [1] *Administrator Guide for Avaya Communication Manager*, Document 03-300509, Issue 3.1, February 2007, available at <http://support.avaya.com>.
- [2] *Feature Description and Implementation for Avaya Communication Manager*, Document 555-245-205, Issue 5, February 2007, available at <http://support.avaya.com>.
- [3] *Cicero 6.3 Integration Guide*, available from Cicero.

©2008 Avaya Inc. All Rights Reserved.

Avaya and the Avaya Logo are trademarks of Avaya Inc. All trademarks identified by ® and ™ are registered trademarks or trademarks, respectively, of Avaya Inc. All other trademarks are the property of their respective owners. The information provided in these Application Notes is subject to change without notice. The configurations, technical data, and recommendations provided in these Application Notes are believed to be accurate and dependable, but are presented without express or implied warranty. Users are responsible for their application of any products specified in these Application Notes.

Please e-mail any questions or comments pertaining to these Application Notes along with the full title name and filename, located in the lower right corner, directly to the Avaya DevConnect Program at devconnect@avaya.com.