5 Enmore Gardens
London
SW14 8RF

www.iteloffice.com
e: support@iteloffice.com

Tel: +44 (0) 20 8878 7367
Fax: +44 (0) 20 8876 7257

Cuba for Remedy – TAPI Manual Cuba Version 2.0

Document Number: 0403-00288
Revision: G
Date: 21st July 2013

0. PRELIMINARIES

0.1 PUBLISHED BY

Itel Office Software Limited, 5 Enmore Gardens, London, SW14 8RF Tel: +44(0)20 8878 7367, Fax: +44(0)20 8876 7257

0.2 COPYRIGHT

0.3 DOCUMENT HISTORY

Issued @ revision G, on 2013-07-21, by A. Comber

0.4 BIBLIOGRAPHY

0.4.1 Referenced Documents

[1] TAPI, http://msdn.microsoft.com/en-us/library/ms737218(VS.85).aspx

Table of contents

1. INTRODUCTION

This manual provides a complete guide to preparing, installing and configuring your telephone system, Cuba and BMC Remedy for computer telephony integration. For best results follow the manual in chronological order. No specific skills are required but some knowledge of IT can be useful. This manual attempts to make all the necessary steps as painless as possible.

2. OVERVIEW

Cuba for Remedy software enables Computer Telephony Integration (CTI) between most business grade telecom systems, PBX's, and BMC Remedy.

The Cuba components used depend on whether Remedy clients are thick client (using Remedy Windows client) or thick client (using a web browser).

For thick client environments, Cuba Server and Cuba Windows Client are required. For thin client, Cuba Server and Cuba Servlet (a third component, Cuba Thin Client, is actually embedded inside selected Remedy pages as javascript).

The Cuba Server communicates with your telephone system via a TAPI interface. The Cuba client, whether it be the Cuba Windows Client used in Remedy thick client environments, or the embedded Cuba javascript client used in Remedy thin client environments, communicates directly with Remedy. Cuba Windows Client connects directly to Cuba Server to exchange telephony messages while Cuba Thin Client connects to Cuba Servlet which acts as a proxy between Cuba client and Server.

The diagram below illustrates the two types of installation.

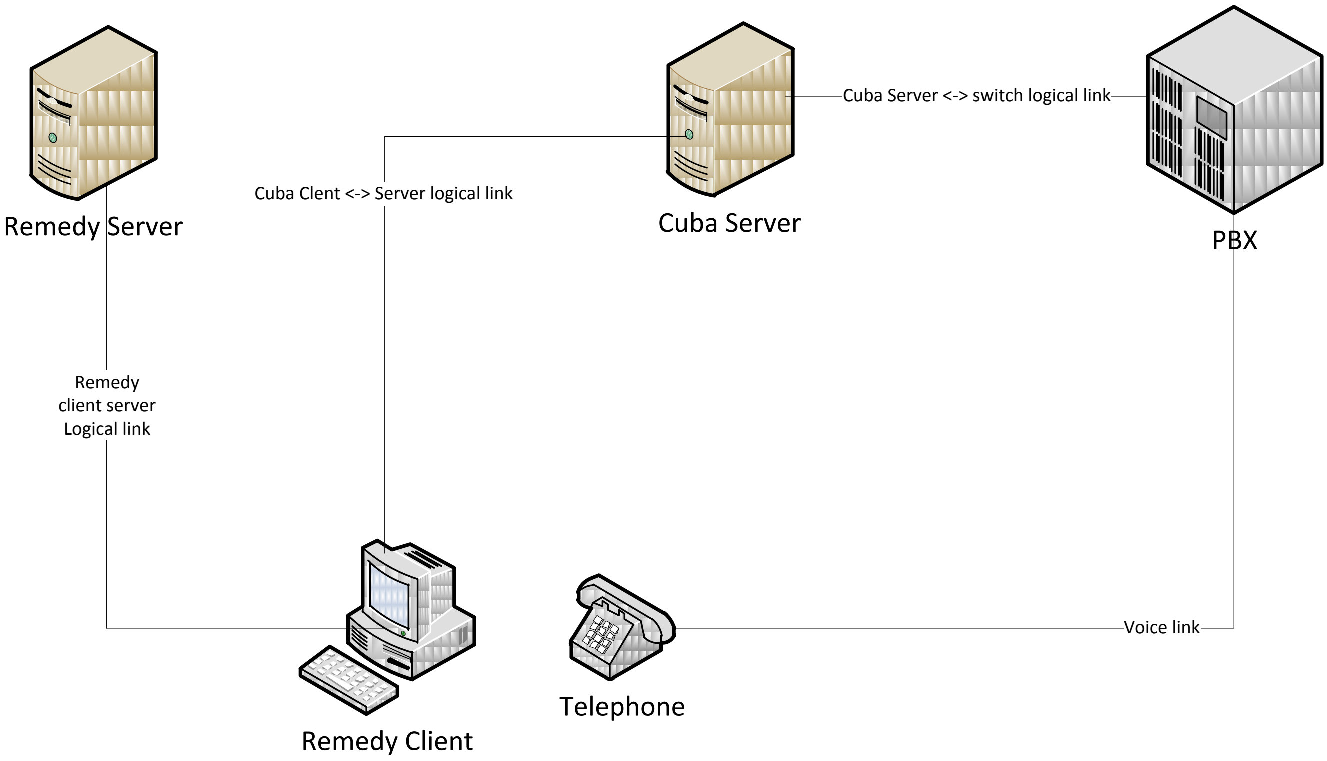


Figure 1: Remedy Thick Client Environment

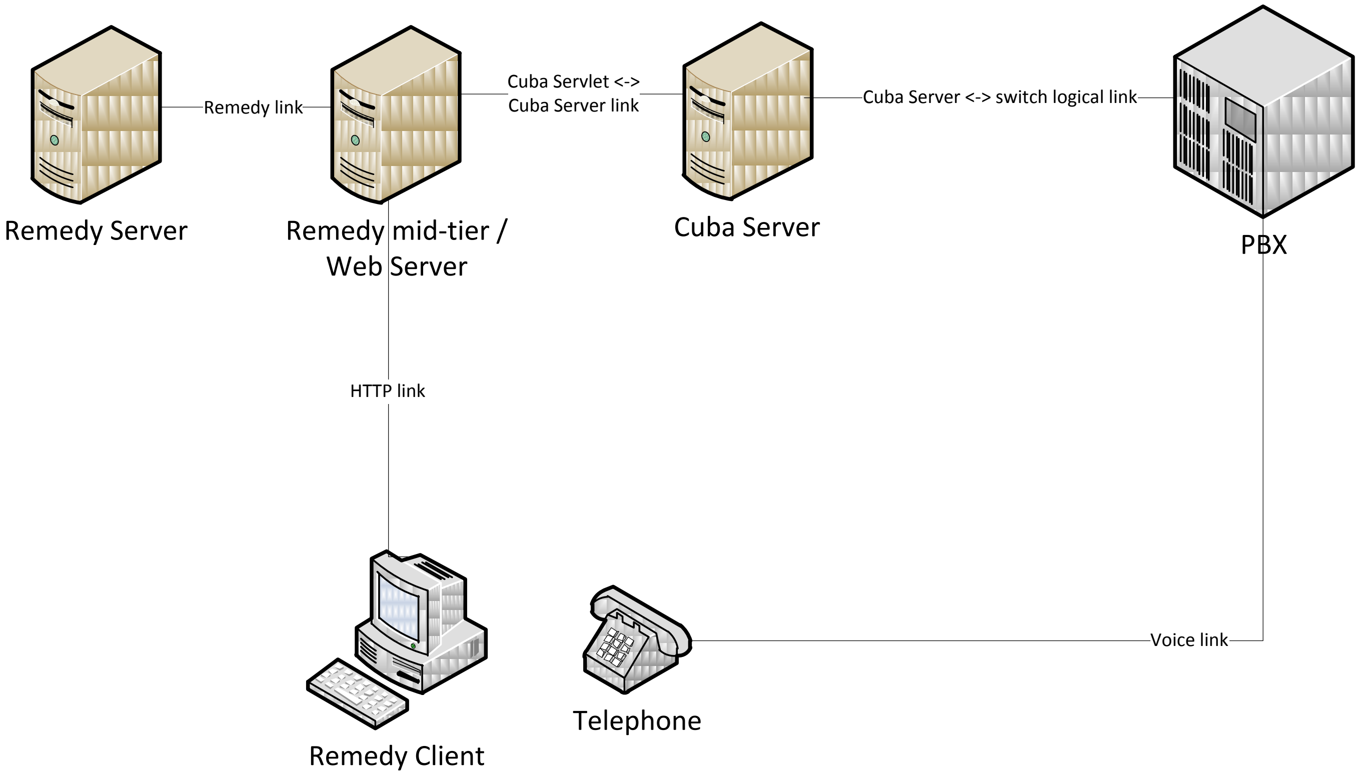

Remedy Server
Remedy link
Remedy mid-tier / Web Server
Cuba Servlet <-> Cuba Server link
Cuba Server
Cuba Server <-> switch logical link
PBX
HTTP link
Remedy Client
Telephone
Voice link

Figure 2: Remedy Thin Client Environment

TAPI provides an application programming interface which allows Cuba Server to control your telephone system by issuing third party requests such as makecall and receive telephony events such as call progress and device status. This interaction is two-way. For example, the Cuba client can make a call on a selected extension which is a message essentially from Cuba to the telephone switch. However, some telephony data is initiated on the telephone system. An example would be an incoming call. The TAPI interface presents this 'delivered' event to Cuba Server and Cuba client can screenpop this information at a BMC Remedy users form.

The PBX component above may be hardware or a software based server. The software server will ultimately communicate with telephone hardware. For example, Cuba Server connects directly via a network connection to an Avaya IP Office but via a CCT Server to a Nortel Meridian system. The common part is a 3rd party TAPI driver. This component is generally installed on the same machine which hosts Cuba Server.

The TAPI driver must be installed and configured before Cuba Server can communicate with your telephony hardware. Email us on support@iteloffice.com if you require assistance in selecting, installing or configuring the TAPI driver. Generally, the TAPI driver is provided by your telephone system maintainer.

The TAPI driver provides an application programming interface which allows Cuba Server to control your telephone system and receive events. This interaction is two-way.

A telephony request is initiated by Cuba and is sent to the telephone switch. A telephony event is received by Cuba from the telephone switch.

An example of a telephony request would be a dial request initiated by right clicking on the Cuba client systemtray icon, selecting the Softphone menu, entering a number and clicking on the dial button.

An example of a telephony event would be an alerting event on a phone causing the TAPI interface to send alerting call information to Cuba which could result in Cuba client screen popping your application.

2.1 USER DATA

Cuba version 1.3+ introduced the capability to associate custom data to a call. In this way custom data can be transported with a call. The custom data could for example be a database record ID so that a user answering a call can lookup or create a record, attach the record ID to the call and have the information transferred to an agent with the necessary skills to resolve the call. This application of attaching user data is sometimes referred to as 'screen passing'.

This ability means that if the first recipient of a customer call adds the caller details to Remedy but later realises that a different person should handle this call, then that user can attach the Remedy request ID (unique form reference) to the call and transfer to another agent. Processing on the end recipient can check for user data and if found use the data to screenpop the Remedy form instance previously worked on by the first call recipient. This means that the caller does not have to re-iterate everything just explained to the last agent they talked to and can dramatically improve customer service as a result.

3. PREPARATION

3.1 OPERATING SYSTEM REQUIREMENTS

The Cuba Server should be installed on an 'always on' PC on the network.

System Requirements are:

512MB RAM (Approx 52Kb per monitored call)

1.2GHz CPU

Ethernet Port

Operating System – Microsoft Windows: Vista, XP, 2000, 2003, 2008, NT, 7, 8

Cuba Server is currently a 32 bit application but will run on 64 bit Windows platforms.

Cuba Windows client can run on any Microsoft Windows platform from Windows 95+.

Administering Cuba Settings
Cuba's settings are configured in the Cuba Web Administrator - a web interface to Cuba Server. Your web browser must support Java Script.

3.2 CONFIGURING TELEPHONY INTERFACE

Configuring the telephony interface is very specific to the manufacturer of your telephone system. Essentially, you will require all the components necessary to install a third party TAPI driver for your equipment. On some systems, all that is required is installation of the TAPI driver software. On some systems extra hardware may be involved. The TAPI driver must be able to operate in third party mode. In other words, provide the capability to control multiple devices. First party TAPI drivers, used to control a single telephone are not suitable. Some TAPI drivers work in both modes and it is necessary to select a 'third party mode'.

Contact your manufacturer or your supplier for further details. Alternatively, feel free to email us on support@iteloffice.com with any questions.

3.3 CONFIGURING BMC REMEDY

Cuba for Remedy can screenpop Remedy Windows client or the Remedy thin client via a web browser. Cuba Windows Client requires Remedy Windows User Tool version 4.5 or higher to be installed on the user workstation. Cuba Thin Client is supported On Remedy thin client versions up to v8.

3.3.1 BMC Remedy Thick Client Configuration

Cuba for Remedy can pass a caller or called number to a Remedy search form and automatically perform the search. Or telephone data can be passed to Remedy client for Remedy workflow to process.

Softphone functionality can be provides within a Remedy form. Active links linked to Remedy controls such as buttons invoke telephony services such as makecall, transfer, hold and answer. In essence, Cuba provides the interface between Remedy and the telephone system so users need not learn new software.

For handling telephony events such as inbound call for screenpop, a mapping between the telephony 'data' and a Remedy field is configured in the Cuba web administrator. The mapping is straightforward. Simply select a telephony data item, for example a Caller ID, and map to a Remedy field.

For handling telephony softphone functions create an active link.

Performing these configurations is explained in detail in the following sections.

3.3.2 BMC Remedy Thin Client Configuration

The functionality provided is similar to thick client as described above. However, the Remedy 'client' in this case is a web browser. The Cuba client component is embedded within each Remedy form and so is available if required. To embed the Cuba client follow these steps:

1. Go to the Remedy mid-tier host.

2. Navigate to the location where Remedy mid-tier has been installed. For example, on the Windows platform this is likely to be C:\Program Files\BMC Software\ARSystem\midtier\resources\javascript and on UNIX /usr/ar/mid-tier/resources/javascript. The exact directory required is browser dependent. To support Internet Explorer it is ie folder, Firefox moz, Chrome or Safari use saf.

3. Append the contents of cubarem.js, which is copied into the Cuba Server install folder (default C:\Program Files\IOffice\CubaTAPI\), to the end of ClientCore.js file.

4. Deploy the Cuba Servlet. Launch the generate_war_file.exe program also copied to the Cuba server install folder (see step above). Enter the host and port of your Cuba Server and the program will generate a customised war file. a WAR, web archive, file is a package used to deploy a Java web application, in this case Cuba servlet. Copy the generated cuba4web.war file to your Tomcat root/webapps folder. The Tomcat folder is in a Tomcat folder, not a BMC folder. On Microsoft Windows platform for example, the folder to deploy too might be: C:\Program Files\Apache Software Foundation\Tomcat6.0\webapps

Tomcat will automatically unpack the war file into a webapps/cuba4web folder.

For these changes to take effect, clear the Remedy mid-tier cache.

3.4 NETWORK CONFIGURATION

Your Cuba Server communicates on ports 6090 and 8080. Port 6090 is used for communication of telephony data and port 8080 is used for web administration. So you should make sure that these ports are not used by other applications. Port 8080 is often used by Tomcat so if you are running Cuba Server on the same host as Tomcat you should stop Tomcat, (re)start cuba Server, reconfigure the web port to be port 8081 (for example) then restart Cuba Server.

Also, ensure that any firewalls configured between the Cuba Server and Cuba Windows Clients / Cuba servlet do not block port 6090.

3.5 CUBA WINDOWS CLIENT

The Cuba Client communicates with the Cuba Server on port 6090* so this (tcp) port must be open.

* This can be edited in the 'Telephony Server Port' field under 'Server Settings' in the Cuba Web Administrator.

3.6 CUBA SERVLET

Cuba servlet communicates with Cuba server on port 6090 so this (tcp) port must be open between Cuba servlet running on your web server (Tomcat) and Cuba server.

4. INSTALLATION

This section covers the basics of downloading and installing your Cuba software and introduces you to the Cuba Web Administrator - the web administration panel where you can modify your settings. The Web Administrator is covered in depth in Section 4.6.

Download and installation of Cuba is fast and intuitive but we've covered it here anyway in a step-by step guide.

4.1 TRIAL PERIOD & LICENCING

During the trial period your Cuba Server software restricts your operation of Cuba for Remedy to three hours of continuous use at a time. Restart the Cuba Server to resume operation.

You are free to install the Cuba Client on as many workstations as you have available.

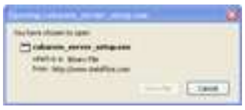
You can apply for a licence by filling in the application form under 'Unlock you Trial' in the Cuba **Web Administrator** – or contact us direct: support@iteloffice.com

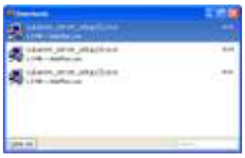
4.2 DOWNLOADING CUBA

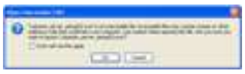
Download your Cuba Software at: **www.iteloffice.com**

To get going with Cuba you need to first download the software. The download operates intuitively and takes just a couple of minutes – but just for the record here's a simple visual overview – using Firefox as the web browser…

Log onto the designated Windows NT/2000/XP/Vista/2003/2008/7/8 PC as an administrator.



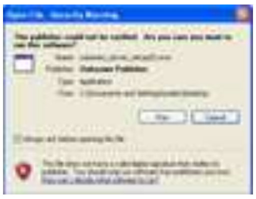


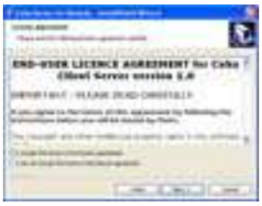


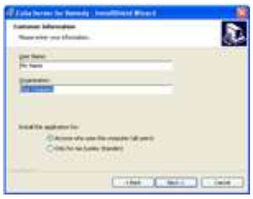
click 'Save File ' >> double click on details >> click 'OK' >>

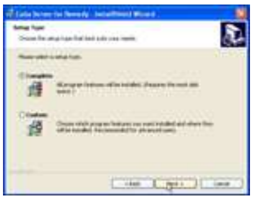
4.3 INSTALLING CUBA SERVER

Then the InstallShield Wizard will launch - this will first install the Cuba Server. In this section we will first install the Cuba Server then return to install the Cuba Client.

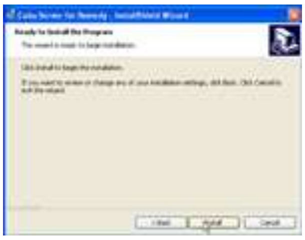


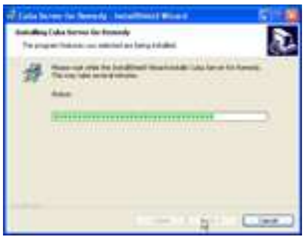


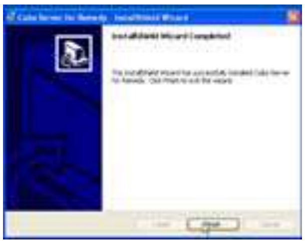




Click Next >> accept licence >> enter name >> click Next >>

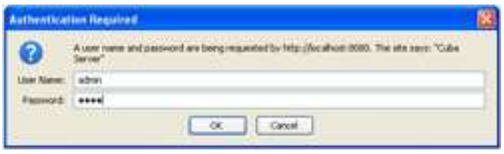






click finish >>

A dialog box will appear…



Authentication Required

A user name and password are being requested by http://localhost:8080. The site says: "Cuba Server"

User Name: admin

Password: ••••

Enter:

User Name: admin

Password: cuba

Click 'OK' and the Cuba Web Administrator launches automatically on the 'Settings' page. You can start entering your settings immediately but for the purpose of this tutorial we'll return to install the Cuba Client and deal with settings in the next Section.

The Cuba Server installation copies a file, TSLIB.INI to the install folder. The default install folder is **C:\Program Files\Ioffice\CubaTSAPI.** Edit this file by entering the IP address of the TSAPI Server. If Cuba server is installed on the same computer as the TSAPI server you can enter 127.0.0.1 as the IP address.

Example where IP address of TSAPI Server is 192.168.0.50:

```
[Telephony Servers]
192.168.0.50=450
```

After editing, copy this file to the computers Windows directory. This is usually C:\Windows. On some computers it may be C:\WINNT or a custom location. If in doubt type echo %WINDIR% <ENTER> at a command prompt.

4.4 INSTALLING THE CUBA WINDOWS CLIENT

Install the Cuba Windows Client to support Remedy thick Client environments.

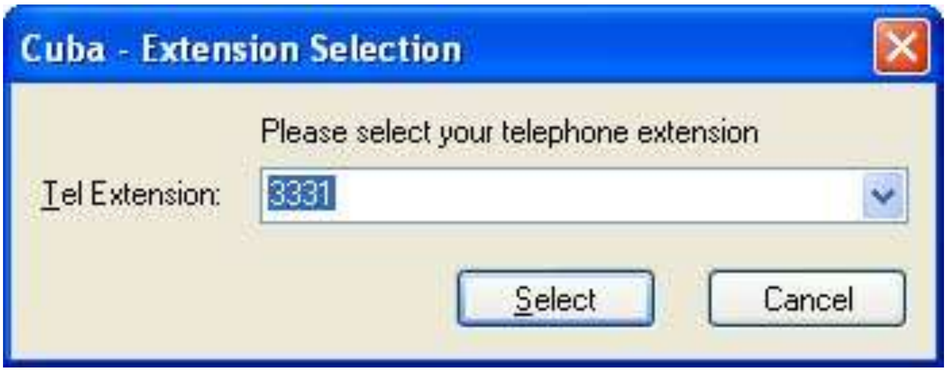
To install your Cuba Client,

Click: 'Start' > 'All Programs' > 'Cuba' > 'Install Cuba Client' >

As previously, the InstallShield Wizard will launch… again, work through the wizard until you reach 'Finish'.

Once you have completed the wizard you will notice that the Cuba Telephone Icon… ☎ appears in your System Tray.

Double Click on the Telephone Icon and a dialog box with a drop down menu will appear - click on the down arrow to reveal a drop down box that displays your extensions.



Cuba - Extension Selection

Please select your telephone extension

Tel Extension: 3331

Choose your extension and Click 'Select'. You'll see that the Cuba Telephone Icon has turned green to indicate you have successfully connected Cuba to your telephone extension.

☎

4.5 INSTALLING CUBA CLIENT ON A TARGET WORKSTATION

You'll find the Cuba Client installation program in your Cuba Server installation folder - which by default is: C:\Program Files\Ioffice\Cuba4Remedy.

The Cuba Client installation file is: cubarem_client_setup.exe.

If you want to install the Cuba client on other workstations, copy the cubarem_client_setup.exe file to a shared network location - accessible from other workstations - then run the Setup Program on as many workstations as you require.

Note: You can install the Cuba Clients on as many workstations as you like. Your licence will determine how many concurrent Cuba Clients can connect to a Cuba Server.

Our free trial allows unlimited concurrent Cuba Clients but the Cuba Server shuts down after 3 hours operation. Restart the Cuba Server to resume operation.

4.6 THE CUBA WEB ADMINISTRATOR

You can control Cuba's Application Settings in the Cuba Web Administrator.

This is a web front end that allows quick and easy configuration of your Cuba Server.

The Web Administrator launches automatically during initial installation and opens on the 'Settings' tab. This gives you a series of simple form fields where you can enter configuration details.

4.6.1 NAVIGATING TO THE WEB ADMINISTRATOR

You'll be able to return here to review and edit all of your settings whenever you need to. You'll find the Cuba Web Administrator alongside the Cuba Software Files in your Program File - simply click on:

Start > All Programs > Cuba > Cuba Web Administrator > or

If your Cuba Server is installed on your local machine you can use the address 127.0.0.1.

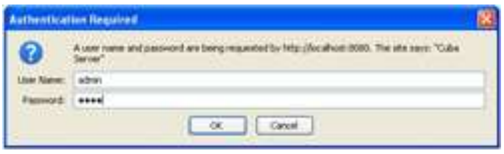
If your Cuba Server is installed elsewhere on your network, launch your web browser and navigate to the hostname or IP address of the computer where your Cuba Server is installed.

Append : 8080 to the hostname, for example: **http://MyComputerName:8080**

This is because the Cuba web server uses the non standard port 8080 so as not to conflict with alternative web servers which may be configured on the Cuba Server computer.

4.6.2 PASSWORD PROTECTION

Each time you open it a dialogue box will appear…



Authentication Required

A user name and password are being requested by http://localhost:8080. The site says: "Cuba Server"

User Name: admin

Password: ••••

As before enter:

User Name: admin

Password: cuba

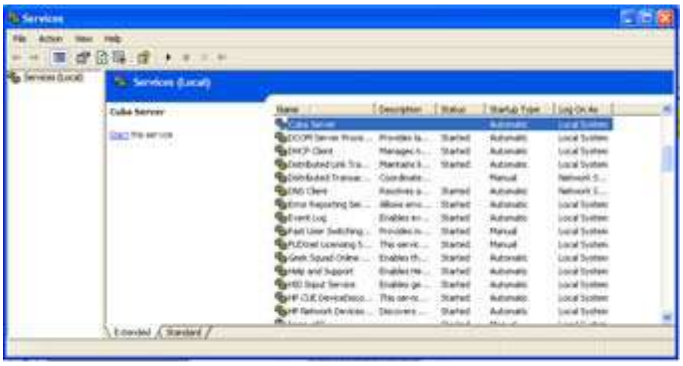
Click 'OK' and the Cuba Web Administrator launches automatically on the 'Settings' page.

If you get a Page Load Error then either the Cuba Server is not hosted on your machine or your trial version needs to be restarted.

Go to:

Start > Control Panel > Administrative Tools > Services >

Scroll Down to 'Cuba Server' > Click 'Start'



Services

Services (Local)

Cuba Server

Start this service

Name	Description	Status	Startup Type	Log On As
Cuba Server			Automatic	Local System
DCOM Server Proce...	Provides la...	Started	Automatic	Local System
DHCP Client	Manages n...	Started	Automatic	Local System
Distributed Link Tra...	Maintains li...	Started	Automatic	Local System
Distributed Transac...	Coordinate...		Manual	Network S...
DNS Client	Resolves a...	Started	Automatic	Network S...
Error Reporting Ser...	Allows erro...	Started	Automatic	Local System
Event Log	Enables ev...	Started	Automatic	Local System
Fast User Switching ...	Provides m...	Started	Manual	Local System
FLEXnet Licensing S...	This servic...	Started	Manual	Local System
Geek Squad Online ...	Enables th...	Started	Automatic	Local System
Help and Support	Enables He...	Started	Automatic	Local System
HID Input Service	Enables ge...	Started	Automatic	Local System
HP CUE DeviceDisco...	This servic...	Started	Automatic	Local System
HP Network Devices ...	Discovers ...	Started	Automatic	Local System

The Cuba Server will restart. You may need to restart your browser in order to access the Cuba Web Administrator.

5. HOW IT WORKS

This section explains how the Cuba solution interacts with Remedy to provide a bird's eye view. This should make configuration more natural with an appreciation of what is happening at a higher level. It will be useful to refer to Figure 1 and 2, dependent on Remedy environment, while reading.

A PBX, or private branch exchange, and its related hardware has historically been separated from premise data networks. Telephone handsets have typically been wired directly into PBX interfaces. A telephone on a call centre agent's desk can be used to send requests to the PBX via entering digits on the telephone. The PBX itself generates events, for example, when a customer calls, and can set a telephone ringing. So the telephone can be used to send requests and also can receive events. So we have a request - event based system.

By connecting the data network to the PBX via Cuba Server, this request-event facility can be extended to PC users.

Therefore, Cuba provides two sets of facilities:

A) Ability to send requests via a computer to the PBX.

B) Ability to be notified of events from the PBX.

This request-event facility is often referred to as computer telephony integration or CTI. CTI and the traditional telephone handset work in tandem.

Various editions of Cuba Server are available which communicate with a PBX via a switch protocol. Cuba server for TAPI uses the TAPI interface. So Cuba Server provides the connection to the switch. The other Cuba components ultimately communicate with Cuba Server to provide CTI features.

The Cuba client components are specialised to communicate seamlessly with Remedy. The Cuba windows Client communicates directly with Cuba Server using a proprietary Cuba protocol. It communicates with Remedy Windows client by the Remedy COM interface.

The Cuba Thin Client is Javascript code embedded within Remedy web pages. The Remedy client functionality is provided in the same way. In fact, Cuba javascript code invoked via Remedy or via a Cuba telephony communication calls Remedy javascript functions to provide functionality.

The Cuba Thin Client cannot communicate directly with Cuba Server because of a javascript security restriction that requests can only be sent or received via the same host that served the web page. Hence, a Cuba component must be installed on the web server to complete the communications link from Cuba Thin client to Cuba server. Hence the communication is as follows:

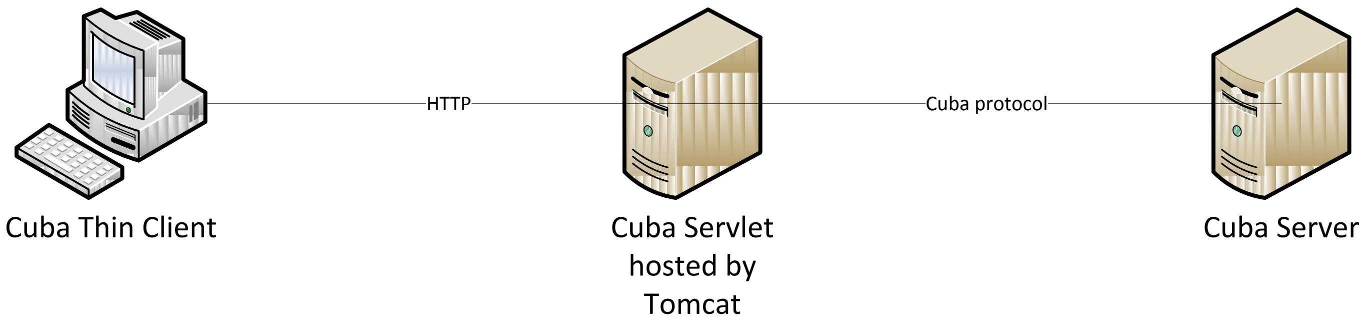


Figure 3: Cuba Thin Client architecture

The request part of CTI functionality can be provided by Cuba by invoking Cuba functions via Remedy active links. A Remedy active link can invoke Cuba makecall, answercall, clearcall, transfercall, and a whole range of other request types. Telephony events are data travelling in the reverse direction, from PBX to Remedy and these are initiated by Cuba calling Remedy active links.

Configuration of requests is performed in Remedy, usually using the BMC Remedy Developer Studio or for earlier versions of Remedy, Remedy Administrator. Configuration of event handling is dependent on whether the client is thick or thin client. Thick client handling must be configured in the Cuba Web Admin which sets rules for how telephony events are handled in the Cuba client. Event handling is rules based. For each event type a specific type of telephony data can be copied to a Remedy field. After posting the data an event can be triggered which can call a Remedy active link (trigger event option). The active link which is to be triggered has to be created in Remedy of course.

However, handling for Remedy thin clients is different. There is no configuration required in Cuba Web Admin because Cuba data is passed to Remedy by an event. And Remedy workflow responds to the event as required. This means that Remedy thin client developers use their familiar Remedy workflow concepts to create a telephony solution.

The details of configuring CTI behaviour follow in the next section.

6. CUBA CONFIGURATION

As described above, CTI functionality can be split into configuring requests and configuring telephony event handling. Configuring requests is generally adding softphone functionality to a Remedy form. Configuring event handling is configuring which remedy active links are triggered for a telephony event type.

6.1 CUBA REQUEST CONFIGURATION

6.1.1 Cuba Windows Client Request Configuration

To enable a telephony request from your Remedy Form – e.g. ‘Dial’, ‘Answer’, ‘Hold’ or ‘End Call’ - you’ll need to create a control - typically a button - that runs an active link sending a command to the Cuba (Client) to perform an action.

Outbound actions are not restricted to a particular Remedy Form. You are free to create any number of SoftPhone buttons on any - or all - of your Remedy Forms.

The following tutorials explain in detail how to create a form and add ‘Dial’, ‘Hold’, ‘End Call’ and ‘Answer’ buttons.

6.1.1.1 The commands

Optional parameters are in italics.

Command	Parameters	Notes
DIAL	<Number> <Userdata>	Make a call. <Number> is destination number to dial. <Userdata> is string data which can be attached to the call.
ANSR		Answer a call.
HGUP		Hangup a call.
HOLD		Hold current active call.
RETR		Retrieve held call.
ITFR	<Number> <Userdata>	Initiate transfer. Creates consult call so caller can consult with other called party. <Number> is destination number to dial. <Userdata> is string data which can be attached to the call.
CTFR		Complete transfer. Connects parties and disconnects transferor.
ICFR	<Number> <Userdata>	Initiate conference. Creates consult call so caller can consult with other called party. <Number> is destination number to dial. <Userdata> is string data which can be attached to the call.
CCFR		Complete conference. Connects new party to conference.
FWDA	<Number>	Forward all calls. Only forward ALL calls is supported. <Number> is destination number to forward to.

CFWD		Cancel forwarding.
UDAT	<UserData>	Attach user data. <Userdata> is string data which can be attached to the currently active call.
SSTF	<Number> <Userdata>	Single step transfer. Also known as blind transfer. Transfers call without consultation with destination party. <Number> is destination number to dial. <Userdata> is string data which can be attached to the call.
REDR	<Number>	Redirect call. Redirect an alerting call to another destination (without answering). <Number> is destination number to dial.
RCON		Reconnect Call. Drop active consult call and retrieve previously held (main) call. Used in conjunction with initiate transfer or initiate conference. If an initiate is not answered, reconnect reverts to previous state.
DTMF	<digits>	SendDTMF. The specified digits (or string) will be send as DTMF tones on the active call.

Many of the functions above require no parameters. For example the Answer function just answers an alerting call. Required parameters must be passed otherwise the function will fail. Optionally information such as user data to attach to a call can be passed for some functions.

No notification of function failure is provided to the user. However, the failure is logged to the Cuba client log file.

To use these functions Remedy fires an active link which creates a text file (Cuba command file) in a location on the users workstation that the Cuba client periodically polls. The command files are intercepted by Cuba client and actioned against the currently registered extension.

The command used should be of this format:

```
%ComSpec% /c Echo <Function> <Parameters> >
"%APPDATA%\Itel\cubacmd%date:/=% %time::=% %RANDOM%.cub"
```

Here is a DIAL example where $Work Phone$ is a Remedy character field:

```
%ComSpec% /c Echo DIAL $Work Phone$ >
"%APPDATA%\Itel\cubacmd%date:/=% %time::=% %RANDOM%.cub"
```

Some examples follow.

6.1.1.2 Creating a 'Dial' Button

The following gives an example of how to set up a new test form and add a dial button to dial a Caller ID field on a Remedy Form.

You can use this guide to create a dial button on an existing form - simply open the form you want to ScreenPop and substitute the details of your existing form and/or CallerID field where appropriate.

Create a Form
Launch your Remedy Administrator and locate – or create - the form that you want to ScreenPop. For the purpose of this exercise we'll call this form 'MyFormName'.

Go to 'File' > 'Save Form As' > then enter the Form Name of 'MyFormName'. Click 'OK'.

Double click on the Form and a 'Form Properties' dialog box will appear. Select the 'Permissions' tab and add 'Public'. This allows any user access to the form. Click 'OK'.

Creating a Caller ID/telephone no Field
Right click on the Form and create a Character Field. Double click on the Character Field and change the label to 'Work Phone'. Select the 'Permissions' tab and add 'Public'. Click 'OK'.

Building your 'Dial' Button
Right click on the Form next to the Telephone Number Field and create a Button.

Double click on the Button and in the 'Button Label' field that appears, change the label to Dial.

Select the 'Permissions' Tab and add 'Public'. Click 'OK'.

Select the 'Database' Tab and in the Name Field change the name to 'WorkDial'

Save and close the form.

Creating an Active Link
You now need to create a new Active Link.

Click on 'Active Links' in the left hand navigation in your Remedy Administrator.

Scroll down until you find the name of your button - in this case it would be 'WorkDial'

A 'Modify Active Link' dialogue box will open.

Name the Active Link 'WorkDialActiveLink'.

Under 'Form Name' tick 'MyFormName'.

Tick the 'Button/Menu Item' and in the Field Dropdown Menu select 'WorkDial'.

Under 'Run If" enter: **'Work Phone' != $NULL$**

Select the 'If Actions' tab.

In the 'New Action' select 'Run Process'.

In the 'Command Line' field enter the following text:

```
%ComSpec% /c Echo DIAL $Work Phone$ >
"%APPDATA%\Itel\cubacmd%date:/=% %time::=% %RANDOM%.cub"
```

[The data highlighted in pink must be changed to your Caller ID Field Name - you can access this by clicking the arrow button and selecting 'Fields']

Alternative to send user data with dial request:

In the 'Command Line' field enter the following text:

```
%ComSpec% /c Echo DIAL $Work Phone$ $Request ID$ >
"%APPDATA%\Itel\cubacmd%date:/=% %time::=% %RANDOM%.cub"
```

The above example sends the Request ID of the current form with the dial request. If the dial request is successful the Request ID will be attached as userdata to the call. The userdata will be present in call events for the call. For example, the ringing (alerting) and connected (call established) events. In this way the current Remedy record can be 'passed' with the call to the recipient of the call.

Select the 'Permissions' tab and add 'Public'.

Save the Active Link.

6.1.1.3 Creating a 'Hold' Button

Building your Hold Button

Right click on the Form and create a Button. Double click on the Button and in the 'Button Label' field that appears, change the label to 'Hold'.

Select the 'Permissions' Tab and add 'Public'. Click 'OK'.

Save and close the form.

Creating an Active Link

You now need to create a new Active Link.

Click on 'Active Links' in the left hand navigation in your Remedy Administrator.

Scroll down until you find the name of your button - in this case it would be 'Hold'

A 'Modify Active Link' dialogue box will open.

Name the Active Link 'HoldActiveLink'.

Under 'Form Name' tick 'MyFormName'.

Tick the 'Button/Menu Item' and in the Field Dropdown Menu select 'Hold'.

Select the 'If Actions' tab.

In the ‘New Action’ select ‘Run Process’.

In the ‘Command Line’ field enter the following text:

```
%ComSpec% /c Echo HOLD > "%APPDATA%\Itel\cubacmd%date:/=%
%time::=% %RANDOM%.cub"
```

Select the ‘Permissions’ tab and add ‘Public’.

Save the Active Link.

6.1.1.4 Creating a ‘Hang Up’ Button

Building your ‘End Call’ Button

Right click on the Form and create a Button. Double click on the Button and in the ‘Button Label’ field that appears, change the label to ‘Hang Up’.

Select the ‘Permissions’ Tab and add ‘Public’. Click ‘OK’.

Save and close the form.

Creating an Active Link

You now need to create a new Active Link.

Click on ‘Active Links’ in the left hand navigation in your Remedy Administrator.

Scroll down until you find the name of your button - in this case it would be ‘Call’

A ‘Modify Active Link’ dialogue box will open.

Name the Active Link ‘HangUpActiveLink’.

Under ‘Form Name’ tick ‘MyFormName’.

Tick the ‘Button/Menu Item’ and in the Field Dropdown Menu select ‘HangUp’.

Select the ‘If Actions’ tab.

In the ‘New Action’ select ‘Run Process’.

In the ‘Command Line’ field enter the following text:

```
%ComSpec% /c Echo HGUP > "%APPDATA%\Itel\cubacmd%date:/=%
%time::=% %RANDOM%.cub"
```

Select the ‘Permissions’ tab and add ‘Public’.

Save the Active Link.

6.1.1.5 Creating an ‘Answer’ Button

Building your ‘Answer’ Button

Right click on the Form and create a Button. Double click on the Button and in the ‘Button Label’ field that appears, change the label to ‘Answer’.

Select the ‘Permissions’ Tab and add ‘Public’. Click ‘OK’.

Save and close the form.

Creating an Active Link

You now need to create a new Active Link.

Click on ‘Active Links’ in the left hand navigation in your Remedy Administrator.

Scroll down until you find the name of your button - in this case it would be 'Answer'

A 'Modify Active Link' dialogue box will open.

Name the Active Link 'AnswerActiveLink'.

Under 'Form Name' tick 'MyFormName'.

Tick the 'Button/Menu Item' and in the Field Dropdown Menu select 'Answer'.

Select the 'If Actions' tab.

In the 'New Action' select 'Run Process'.

In the 'Command Line' field enter the following text:

```
%ComSpec% /c Echo ANSR > "%APPDATA%\Itel\cubacmd%date:/=%
%time::=% %RANDOM%.cub"
```

Select the 'Permissions' tab and add 'Public'.

Save the Active Link.

6.1.1.6 Creating an 'Initiate Transfer' Button

Initiate transfer is the first part of transferring the call. A consult call is made to a designated destination and the caller can talk to the destination. To actually transfer the call a complete transfer operation is required.

Building your 'Initiate Transfer' Button

Right click on the Form and create a Button. Double click on the Button and in the 'Button Label' field that appears, change the label to 'Initiate Transfer'.

Select the 'Permissions' Tab and add 'Public'. Click 'OK'.

Create a new character field called XferTo. Make Permissions public as above. Click OK. User enters the number to transfer to in this field.

Save and close the form.

Creating an Active Link

You now need to create a new Active Link.

Click on 'Active Links' in the left hand navigation in your Remedy Administrator.

Scroll down until you find the name of your button - in this case it would be 'Initiate Transfer'

A 'Modify Active Link' dialogue box will open.

Name the Active Link 'InitXferActiveLink'.

Under 'Form Name' tick 'MyFormName'.

Tick the 'Button/Menu Item' and in the Field Dropdown Menu select 'Initiate Transfer'.

Select the 'If Actions' tab.

In the 'New Action' select 'Run Process'.

In the 'Command Line' field enter the following text:

```
%ComSpec% /c Echo ITFR $XferTo$ > "%APPDATA%\Itel\cubacmd%date:/=%
%time::=% %RANDOM%.cub"
```

Alternative to send user data with dial request:

In the 'Command Line' field enter the following text:

%ComSpec% /c Echo ITRF $XferTo$ $Request ID$ >

"%APPDATA%\Itel\cubacmd%date:/=% %time::=% %RANDOM%.cub"

The above example sends the Request ID of the current form with the initiate transfer request. If the request is successful the Request ID will be attached as userdata to the call. The userdata will be present in call events for the call. For example, the ringing (alerting) and connected (call established) events. In this way the current Remedy record can be 'passed' with the call to the recipient of the call.

Select the 'Permissions' tab and add 'Public'.

Save the Active Link.

Technical Notes: It's not really necessary to understand what the 'Run Process' command is doing.

However some explanation might help troubleshooting if you experience problems.

When the active link is invoked, Remedy uses the Windows command interpreter to echo the text - DIAL <contents of Telephone Number Field> to a text file.

%COMSPEC% is the path variable for the Command Interpreter.

On Windows XP this path is usually C:\WINDOWS\system32\cmd.exe but will depend on the Windows operating system and how Windows has been installed.

APPDATA is the Windows Application Data path and on Windows XP for example might be: C:\Documents and Settings\John\Application Data if the currently logged on user logged into Windows as username John.

Itel is the hard-coded Itel subdirectory of this APPDATA path.

The date time and RANDOM processing are used to ensure that each command will create a unique file name. The Cuba (Client) will pick up these commands in the APPDATA file.

This configuration only works for the Remedy Thick Client because the Remedy Thin Client does not support active links which access the local file system. To reproduce this functionality in the Thin Client, a different type of active link must be created as described in the next section.

6.1.2 Cuba Thin Client Request Configuration

In principle, configuration for thin client is the same as for thick client. The type of active link to create to invoke a Cuba telephony function is Run Process and in the run process command line the javascript function is entered in the form:

javascript:cuba.<request_name>(<number_to_dial);

It really is as simple as that. You then bind a control to the active link and that is all the configuration you need.

6.1.2.1 Cuba thin client function list

The full list of available functions as of Cuba version 2.0 is as follows:

Function name	Parameters	Notes

startCuba		Used to start Cuba internal event handling. If startCuba is not called requests can be issued but no events will be received back from the switch.
getConfigItem	key	Each configuration item key in Cuba Web Admin can be looked up using this function. For the Cuba key names check the Cuba Server config file or you can check the Cuba Web admin html pages. For example, the Remedy form name to receive events is specified in The Remedy thin client configuration and the keyname is form.
getText	fieldid	This function retrieves the text in the current Remedy form in the fieldid specified. This can be used for example to grab some text in a character field. Generally however, the same text can just as easily be obtained using Remedy field variables.
postText	fieldid, text	This function can be used to post text to a specific field. The text parameter is the text to insert.
setSelection	fieldid, newselection	This function can change the selection in a Remedy Drop-Down List, a Radio button, or a Checkbox field. Enter the zero based index. Note the first selection is zero. As an example to change to selection in the default Status field with fieldid 7 to Fixed on a Remedy form the fieldid would be 7 and the newselection would be set to 2.
triggerEvent	data1, data2	This function calls the Remedy ARProcessEvent function which can be detected by Remedy to fire an active link. In an active link execution options section, there is an Event tickbox. Tick this to fire the active link on ARProcessEvent being generated. data1 and data are data items that can be passed to the trigger and intercepted in the active link in the $EVENTTYPE$ and $EVENTDATA$ field respectively. This function is used internally by Cuba for the trigger event event handling feature. The function is also made available to integrators.
getDevices		Function to retrieve the list of stations defined in the Device List section of Cuba Web Admin. This list is generally used to provide a list of extensions to select to register a station to receive telephony events. A station must be registered before Cuba server will accept requests on the stations behalf or to receive

		events.
registerDN	dn	registerDN must be called for each station (DN) which intends to use Cuba CTI features. The dn parameter is the station identifier. registerDN starts monitoring on the PBX for the specified device.
unregisterDN	dn	unregisterDN stops monitoring on the switch for the specified device.
getMonitoredDN		getMonitoredDN returns the monitored DN, if any, for the browser session.
getClientReport		This is an internal function used to display the Cuba connected clients report.
makeCall	dn	makeCall delivers a request to dial a destination dn.
answerCall		answerCall can be used to answer a ringing call via CTI.
clearCall		clearCall can be used to drop a call.
cubaMapGet	Searchstring, key	This function can be used to extract the data value from a key in $EVENTDATA$ data from a Cuba event. eg if $EVENTDATA$ = <string>dn=3331<morestring> cubaMapGet(<the_string>, 'dn') would retrieve 3331

These functions are at the bottom of the cubarem.js file appended to the bottom of the ClientCore.js file during installation so for full details you can examine the source code here.

Note that in the javascript an optional callback function can be provided as a parameter. This is used internally by Cuba and in general would not be used when called by integrators.

6.1.2.2 Remedy MakeCall Button Example

Given the list of functions above, to create a dial button on a Remedy form, follow these steps.

1. Add a Button control to a Remedy form, with name=btnDial, Button Label='Dial' and Permission='Public'. Save form.

2. Add a Character Field to same Remedy form, with name=txtNumber, Label='No. to Dial', Permissions='Public'.

2. Create an active link named CUBA:AL_makeCall of type Run Process.

3. Associate the form above with the active link.

4. In execution options set Button/Menu Field to btnDial.

5. In the Run Process Command Line field enter: javascript:cuba.makeCall($txtNumber$);

6. Save active link.

Note that you can also configure a response field in Cuba Web Admin which is described in the event handling part of this document. Then the user can see the response for each request or the responses could be logged for troubleshooting purposes.

6.2 CUBA EVENT HANDLING CONFIGURATION

How Cuba passes event data from the PBX to Remedy is dependent on whether client is using Windows User Tool or a web browser. Thick client is configured by setting rules in the Cuba Web Admin. Thin clients receive events.

6.2.1 Remedy Thick Client Configuration

This section is used to configure screenpop for Remedy Windows user tool aka thick client users.

Essentially, the way it works is that a target field is specified for a specific item of telephony data. The target is specified for each event type. For example, if the only item of telephony data that should be passed to Remedy is the CalledID (called number) for the ringing event on an Remedy users phone, then the target field for CalledID must be specified and CalledID selected for the ringing event (alerting event in Cuba speak).

The Cuba web admin is a web interface which can be accessed by loading any web browser and navigating to the host on which you have installed Cuba Server and appending :8080 to the hostname. If for example, you installed the Cuba Server on 192.168.0.45 or mycubahost or telserver.mydomain.com you would enter 192.168.0.45:8080 or mycubahost:8080 or telserver.mydomain.com:8080 respectively. Login using user: admin, password: cuba. Refer to section installation section for further details.

The Cuba web administrator uses a left hand navigation bar menu. Some of the menu items are not related directly to telephony event handling in Remedy and so are discussed in the next section.

6.2.1.1 Thick Client Options

Configuration Item	Description
Remedy Server name	This is the name of the Remedy Server your users connect to.
Remedy form	This is the name of the Remedy form to be screenpopped. Only one form can be selected. If screenpop of multiple forms is required, then an intermediate form should be designated as the form here and workflow in this form can then screenpop alternative forms based on some criterion set. For example, in a serviced environment where Remedy users service a number of different clients who dial different numbers, then workflow in the screenpopped form can evaluate the called number and present varying forms to the user based on number dialed.

AutoSearch	AutoSearch is a Cuba4Remedy feature to simplify screenpop. If AutoSearch is specified, a user can select this option and then specify a telephony data item to be used in the search and an event to screenpop and Cuba client will automatically search for a Remedy form instance based on the telephony data passed. The simplest example would be screenpop using Caller ID on a ringing call. If the callers number was present in the Remedy database then this caller can be automatically searched and presented to the user without any user intervention.
Save form on screenpop	Select this option to make Cuba client save the Remedy form after screenpop. If the form is saved then Cuba will save a copy of the Remedy form RequestID and so on subsequent screenpop events can ensure that the correct form instance is loaded. For example, if a ringing call saves the Caller ID and a timestamp, then the cleared call event can be used to set a cleared call timestamp on the same form. By saving the form on the ringing screenpop event, Cuba can ensure this same form record is used for stamping the cleared timestamp.
Close form on Cleared	If Remedy form processing is set on a cleared event (end of call) then the screenpopped form is closed.
Update Remedy status field on Cleared	If Remedy form processing is set on a cleared event (end of call) then the screenpopped form status is updated to the specified enumerated value. If for example, the Remedy form has status New, Progressing and Closed then selecting enumerated value 2 means that status is set to Closed.
On Alerting event close any open instances of form	If set, this option closes any open instances of the screenpop form specified above as the first action. This option is to prevent a multitude of Remedy screenpop forms being open on a users workstation.
Open form Mode	On a screenpop, a Remedy form can be opened in either Search or Submit mode. The default used by Cuba is Search so be sure to change here if Submit (or modify) mode is preferred.
Screenpop outbound calls	Typically, screenpop is only used for calls where the user is the destination of the call. However, it is also possible to perform a screenpop on an outbound call so select this option to enable this feature.
Telephony data mapping fields	
This Device	This device refers to the actual phone which receives the telephony event. The this device may or may not be the called number. If a call is originated from a caller who dials a distribution device then the called number will be

	the distribution device and the distribution device will divert the call to the this device. In telephony speak the called number would be the distribution device number and this device would be the user station (their telephone extension). Enter the name of the Remedy field which should receive the this device information.
Caller ID	Enter the name of the Remedy field which should receive the Caller ID information. This is the callers number also known as the ANI.
Called ID	Enter the name of the Remedy field which should receive the Called ID information. This is the called number also known as the DNIS.
Call ID	Enter the name of the Remedy field which should receive the Call ID information. The Call ID is the telephone switch identifier for the call in question. This identifier can be tracked for reporting purposes or cross reference to switch vendor supplied call logging information.
Connection ID	Enter the name of the Remedy field which should receive the Connection ID information. Connection ID is a Cuba concept and is a Cuba identifier for a call. A Connection ID is similar to a Call ID but remains the same across transfers. Most Call ID's change when a call is transferred. Cuba uses Connection ID to track a call lifecycle from start to end. Generally this information is not required for screenpop but can be passed to Remedy if required.
User data	Enter the name of the Remedy field which should receive the User data information. Refer to the Userdata section for further details on this feature.
Agent ID	Enter the name of the Remedy field which should receive the Agent ID information. If the Agent ID of the agent processing the call can be obtained, then this agent id can be passed to Remedy.
Collected Digits	Enter the name of the Remedy field which should receive the collected digits information. Some telephone switches provide a facility to prompt callers for a string of digits or at least some choice. This information, if available on the call, can be passed to Remedy. For example, an IVR system, Interactive voice response system, can prompt for a callers customer id.
Ringing Timestamp	Enter the name of the Remedy field which should receive the ringing timestamp information. Cuba provides the facility to add a timestamp for your Remedy records. Specify here the name of the Remedy field to post the ringing (alerting) timestamp. Note that the difference

	between ringing and answered timestamp could be used to calculate call answer time. The Ringing Timestamp is a date time stamp taken from the Cuba Server's time clock when the incoming call presents to your telephone switch. Enter a Remedy field name here if you want to post the alerting event timestamp data to your Remedy database. Cuba string date timestamps use the UK presentation style - DD/MM/YYYY HH:MM:SS. Cuba date-timestamps can also be passed to Remedy as 'true' date/time values. In this case locality is not an issue. So for a solution independent of locality, post Cuba timestamps to Remedy date/time fields. Note also that the timestamp is the local time on the cuba Server.
Answered Timestamp	Enter the name of the Remedy field which should receive the answered timestamp information. Answered timestamp is the time that the call was answered.
Dropped Timestamp	Enter the name of the Remedy field which should receive the dropped timestamp information. The dropped timestamp is the time that the call ended for the This Device party.
Diverted Timestamp	Enter the name of the Remedy field which should receive the diverted timestamp information. A divert event occurs if a call is deflected to another device. An example is if forwarding is set on the ringing station. Alternatively, if voicemail is set, then the call might be diverted to voicemail if not answered within a set time period.
Transferred Timestamp	Enter the name of the Remedy field which should receive the transferred timestamp information. The timestamp is the time that a call is transferred to another device.
Initiated timestamp	Enter the name of the Remedy field which should receive the initated timestamp information. The initiated event occurs when an originated call connects to its destination and hears ringback. The call has not yet been answered but the destination ringing tone has been detected by the switch.
Set Focus	Enter the name of the Remedy field which should receive the focus. Set focus can invoke Remedy workflow so this option can be used for this purpose or simply to set focus to a particular field for user convenience.
Call Direction	Enter the name of the Remedy field which should receive the call party direction information. Call direction can be origination, destination or unknown. A user making an call to a customer would have origination direction.
Network Call ID	Enter the name of the Remedy field which should receive the network call id information. The network call id

	allows tracking of a call transferred between switches. Note that the trunk link between switches must be configured appropriately to support this feature across switches.
Call logging information	Enter the name of the Remedy field which should receive the call logging information when the call ends on this party/this device. Call logging information provides: Description of the data: station - users extension. otherDN - other party on call. ANI – caller. DNIS - dialed number. CallID - call identifier. ConnectionID - Cuba specific call identifier - connection ID of main call remains throughout lifetime of call NetworkCallID - a call id which is unique across the network. party_role - whether party is an originator of a call or destination. agent_id - agentID if available. userdata - any userdata on call. alert/initiated_timestamp - timestamp when call starts ringing or outbound call initiated. answered_timestamp - timestamp when call answered. cleared_timestamp - timestamp when call cleared on this station. If two calls end together, then the second call is appended to the end of the existing call info with a ; character as delimitter between existing record and new. This can be the case with a transfer. The transfer scenario works as follows: Existing call between party A and party B - call with connection id 1. Party B makes consult call to party C - consult call has connection id 2. Party B transfers the main call (call with party A) to party C. This results in both the main call and the consult call ending on party B. So for party B, two calls will be posted. Cuba handles this situation by checking the contents of the call info field before posting. If the field contains some data, this data is copied to a buffer, a semi-colon (;) delimitter appended and then the new call info data.
Screenpop Rules (What happens, when)	
Ringing Call Actions	Set here the telephony information to be passed to the Remedy fields specified above on the ringing event. Minipop only means that there will be no screenpop of remedy initially on a ringing event but instead a small

	minipop window will appear. The user can click on the minipop window to perform the remedy screenpop.
Answered Call Actions	Set here the telephony information to be passed to the Remedy fields specified above on the call answered event.
Dropped Call Actions	Set here the telephony information to be passed to the Remedy fields specified above on the call dropped/cleared event.
Diverted Call Actions	Set here the telephony information to be passed to the Remedy fields specified above on the diverted event.
Transferred Call Actions	Set here the telephony information to be passed to the Remedy fields specified above on the transferred event.
Initiated Call Actions	Set here the telephony information to be passed to the Remedy fields specified above on the initiated event.
Other Options	
Revert focus	This option can be used to save the Remedy MDI client window handle of the active window before screenpop and after screenpop revent focus to this saved previous Window. This is a convenience for the user and prevents a screenpopped window overlaying itself on the Remedy form instance that the user might have been working on at the point of the screenpop.
Stealth mode for callinfo posting	Apply stealth mode for the posting of callinfo at the end of each call. See Steath mode feature below.

Stealth mode screenpop

Stealth mode is a form of screenpop which is as unobtrusive to the user as possible. It uses the Remedy global fields option to post data to the currently open form and therefore not switch forms which can be distracting and inconvenient to the user.

Stealth mode can be used in conjunction with the Remedy Global Fields feature so that data posted will always be saved to Remedy, no matter which form the data was entered.

Global fields allow data entry to a shared database table/column from multiple Remedy forms. Global fields must be defined with database IDs in the range 1000000 to 1999999. Refer to your Remedy documentation for further details.

For simple inbound call handling it’s very simple - configuration is achieved directly within the Cuba Web Administrator. Simply enter the data relating to the Form Fields – Caller ID or Called ID - that you want Cuba to search when the phone rings and Cuba will Screenpop your form.

There’s no need to configure your Remedy Form, unless you decide to create a new form - or add a new field to an existing form - to post your telephony data to.

Simply put, if you have an existing Remedy Form that you'd like Cuba to ScreenPop by recognising, say, the CallerID (the callers phone number), and the Caller ID field is already on the form then you get started straight away by entering your data in the Basic Settings section of the Cuba Web Administrator.

However, for onscreen telephony functionality - like a 'Dial', 'HangUp', 'Answer' or 'Transfer' button – there's a little bit more to it. You'll need to add a button and create an Active Link to create a Command File.

6.2.1.2 CONFIGURING FOR SCREENPOP

Cuba can only ScreenPop one form. So the first step is to select an existing Remedy form - or create a new one – that you want Cuba to display when it recognises your caller.

You can ScreenPop your Remedy Form by requesting Cuba to recognise either the Caller ID or the Called ID –i.e. the phone no the caller is calling from, or has called.

When you create a new Field in Remedy remember that all data is alphanumeric - so Remedy Character fields should be used.

You only need to create fields for the telephony information you wish to post to Remedy. For example, if only CallerID is required then only this one field will be required on the form.

The next step is to enter the designated form name and field names in the Cuba Web Administrator.

Creating a ScreenPop

This example will configure a Remedy form to ScreenPop the Caller ID of an incoming call.

Launch the Cuba Web Administrator by clicking:

'Start'> 'All Programs' > 'Cuba' > 'Cuba Web Administrator'

You will be prompted for a username and password. The defaults are:

user: admin

password: cuba

Note: these are all lowercase.

If the **Web Administrator** fails to launch - either the Cuba (Server) is not hosted on your machine or your trial version needs to be restarted. For more details see section 4.6.

In the 'Remedy Settings' Section enter the following information:

Remedy Server Name — Enter the Host Name - or IP address – of the computer/server that your Remedy (Server) is located on.

Remedy Form Name — Enter the name of the Remedy Form that you want to ScreenPop.

Caller ID — Enter the name of the Caller ID field that you want Cuba to ScreenPop.

AutoSearch — check this box then click the 'Save' button.

Testing

To test this your will need to have TSAPI configured and installed and the Cuba (Client) will need to be running on the extension that you are phoning.

Double click on the grey telephone icon in the Windows system tray area and select an extension. Call this extension and your selected Remedy form will appear with the selected Caller ID field populated with the extension (or number) you dialled from.

6.2.2 Remedy Thin client Event Handling

Event handling for thin client works differently to thick client. Every telephony event and responses for requests for a monitored telephone device are presented to the particular Remedy thin client interface as an event. Remedy has an 'event' feature which can be used to send a message. In typical Remedy workflow this might be used to send a message between Remedy forms. Cuba uses this same feature to send a message from outside remedy (ie from Cuba) into Remedy. This action is equivalent to Cuba creating a run process active link using the PERFORM-ACTION-SEND-EVENT feature. Cuba populates the $EVENTTYPE$ and $EVENTDATA$ fields as follows:

Cuba 'event'	**$EVENTTYPE$**	**$EVENTDATA$**
Response from telephony request	response	dn=<dn>&rqid=<request_identifier>&data=<userdata>&error=<errorcode>
Error occurred - usually from a request	EventError	
Link between Cuba Server and switch down	EventCTILink Connected	
Link between Cuba Server and switch up	EventCTILink Disconnected	
Dn out of service	EventDevice OOS	EventDeviceOOS&thisdn=<dn>
Dn back in service	EventDeviceIS	EventDeviceIS&thisdn=<dn>
Device monitor on dn successfully started	EventRegistered	EventRegistered&thisdn=<dn>
Monitor stopped	EventUnregistered	EventUnregistered&thisdn=<dn>
Ringing on dn	EventAlerting	eventname=EventAlerting&thisdn=<dn>&otherdn=<otherdn>&callerid=<callerid>&calledid=<calledid>&timestamp=<integer_timestamp>&callid=<callid>&udata=<userdata>&agentid=<agentid>&digits=<collected_digits>
Call established	EventConnected	eventname=EventConnected&thisdn=<dn>&otherdn=<otherdn>&callerid=<callerid>&calledi

		d=<calledid>&timestamp=<integer_timestamp>&callid=<callid>&udata=<userdata>&agentid=<agentid>&digits=<collected_digits>
Call created from dn	EventInitiated	eventname=EventInitiated&thisdn=<dn>&otherdn=<otherdn>&callerid=<callerid>&calledid=<calledid>&timestamp=<integer_timestamp>&callid=<callid>&udata=<userdata>&agentid=<agentid>&digits=<collected_digits>
Call dropped on dn	EventCleared	eventname=EventCleared&thisdn=<dn>&otherdn=<otherdn>&callerid=<callerid>&calledid=<calledid>&timestamp=<integer_timestamp>&callid=<callid>&udata=<userdata>&agentid=<agentid>&digits=<collected_digits>
Active call held	EventHeld	eventname=EventHeld&thisdn=<dn>&otherdn=<otherdn>&callerid=<callerid>&calledid=<calledid>&timestamp=<integer_timestamp>&callid=<callid>&udata=<userdata>&agentid=<agentid>&digits=<collected_digits>
Held call retrieved	EventRetrieved	eventname=EventRetrieved&thisdn=<dn>&otherdn=<otherdn>&callerid=<callerid>&calledid=<calledid>&timestamp=<integer_timestamp>&callid=<callid>&udata=<userdata>&agentid=<agentid>&digits=<collected_digits>
Call queued on dn	EventQueued	eventname=EventQueued&thisdn=<dn>&otherdn=<otherdn>&callerid=<callerid>&calledid=<calledid>&timestamp=<integer_timestamp>&callid=<callid>&udata=<userdata>&agentid=<agentid>&digits=<collected_digits>
Dn on hook	EventOnHook	eventname=EventOnHook&thisdn=<dn>
Dn off hook	EventOffHook	eventname=EventOffHook&thisdn=<dn>
Destination is busy - busy tone on phone	EventDestinationBusy	eventname=EventDestinationBusy&thisdn=<dn>&otherdn=<otherdn>&callerid=<callerid>&calledid=<calledid>&timestamp=<integer_timestamp>&callid=<callid>&udata=<userdata>&agentid=<agentid>&digits=<collected_digits>
Forwarding set	EventForwardSet	eventname=EventForwardSet&thisdn=<dn>&otherdn=<otherdn>&fwdtype=<forwarding_type>
Forwarding cancelled	EventForwardCancel	eventname=EventForwardCancel&thisdn=<dn>&otherdn=<otherdn>&fwdtype=<forwarding_type>
DND set on dn	EventDNDOn	eventname=EventDNDOn&thisdn=<dn>

DND off on dn	EventDNDOff	eventname=EventDNDOff&thisdn=<dn>
A party in call changed	EventParty Changed	Eventname=EventPartyChanged&thisdn=<dn>&otherdn=<otherdn>&callerid=<callerid>&calledid=<calledid>&timestamp=<integer_timestamp>&callid=<callid>&udata=<userdata>&agentid=<agentid>&digits=<collected_digits>
A party added to call (used in a conference)	EventPartyAdded	eventname=EventPartyAdded&thisdn=<dn>&otherdn=<otherdn>&callerid=<callerid>&calledid=<calledid>&timestamp=<integer_timestamp>&callid=<callid>&udata=<userdata>&agentid=<agentid>&digits=<collected_digits>
Agent logged onto dn	EventAgentLogon	eventname=EventAgentLogon&thisdn=<dn>&agentid=<agentid>
Agent logged off this dn	EventAgentLogoff	eventname=EventAgentLogoff&thisdn=<dn>&agentid=<agentid>
Agent now ready to receive calls	EventAgentReady	eventname=EventAgentReady&thisdn=<dn>&agentid=<agentid>
Agent not ready to receive calls from distribution devices	EventAgent NotReady	eventname=EventAgentNotReady&thisdn=<dn>&agentid=<agentid>&mode=<not_ready_code>
Client (or Cuba Servlet) (re)connected to Cuba Server	EventServer Connected	
Cannot connect to Cuba Server	EventServer Disconnected	
Some data has been attached to an event	EventPrivateData	
Call on dn transferred	EventTransferred	eventname=EventTransferred&thisdn=<dn>&otherdn=<otherdn>&callerid=<callerid>&calledid=<calledid>&timestamp=<integer_timestamp>&callid=<callid>&udata=<userdata>&agentid=<agentid>&digits=<collected_digits>
Call on dn diverted	EventDiverted	eventname=EventDiverted&thisdn=<dn>&otherdn=<otherdn>&callerid=<callerid>&calledid=<calledid>&timestamp=<integer_timestamp>&callid=<callid>&udata=<userdata>&agentid=<agentid>&digits=<collected_digits>

The task of creating a creating a telephony solution then is to respond to the relevant events with appropriate workflow. See the sample def file for a simple demonstration. (cuba.def copied to Cuba Server install folder).

In summary form, responding to the Cuba events described would be configured as follows:

1. In BMC Developer create an active link.

2. In Associated forms section list all forms which should respond to the specific event type.

3. In execution options select 'Event'.

4. In Run If Qualification enter a filtering qualification. For example $EVENTTYPE$ = "EventAlerting"

5. Then configure the desired active link workflow action. All the data from the event is available through the $EVENTDATA$ variable. The javascript function cubaMapGet can be used to extract fields in $EVENTDATA$. Useful Cuba javascript functions are listed in the Cuba Thin client function list.

For Cuba thin client screenpop to work it is necessary to enable Javascript in Internet Explorer. Follow these steps: Tools... Internet Options, Security tab, click Custom Level button, Scripting section, enable Active scripting.

7. CUBA SERVER GENERAL CONFIGURATION

This section describes the general Cuba Server options some of which are essential for the cuba solution to function. Others are purely for maintenance and troubleshooting.

7.1 CUBA LOGS

Select the Logs section to view or delete Cuba Server logs. The log file provides a history of what is happening on the Cuba Server.

7.2 DEVICES

In this section you need to enter all the call centre agents extension numbers. Not the agent ID, the actual telephone hardware/handset extension number. The Cuba client extension logon dialog is populated with the contents of the list defined here.

7.3 QUEUES

In a call centre environment agents sometimes login to a queue. The switch vendor may call this something else. Pilot is a common name. But these devices allow calls to be queued and distributed to agents. For some CTI links Cuba can monitor these queuing devices and report useful information or possibly obtain useful information that might not otherwise be available.

If the TAPI driver for your PBX supports queue monitoring enter the queue devices in this section.

7.4 SERVER SETTINGS

Web Server Port	This is the port that the Cuba web server listens on and is 8080 by default. If you wish to communicate on

	another port enter the port no. here. Remember to restart the Cuba (Server) after editing this value.
Telephone Server Port	This is the port that the Cuba telephony server listens on and is 6090 by default. If you wish to communicate on another port enter the port no. here. Remember to restart Cuba (Server) after editing this value.

7.5 HELP

Help on configuring Cuba Server.

7.6 UNLOCK TRIAL

To get a full user licence you will need to purchase an Activation Key. Enter your contact details here and click 'Send Registration Request'. We will contact you to discuss payment options and the registration process.

7.7 REGISTER

Once you have received your Activation Key copy and paste it here and click 'Save'. You will then get unrestricted use of Cuba for the number of seats that you have purchased.

8. TROUBLESHOOTING

8.1 LOGGING

Cuba Server writes logging information to a log file, cubasvr.txt, in the Itel subdirectory of the common application data folder. The location of this folder varies dependent on Windows version. On Windows XP this folder is likely to be C:\Documents and Settings\All Users\Application Data\Itel. Type ECHO %ALLUSERSPROFILE% <ENTER> at a command prompt to find this folder on your system. Prepend \Application Data\Itel to the returned string.

To enable full logging select Full in the Logging level dropdown in the server section of Cuba web administrator.

Now debug level errors and informational messages will be logged to cubasvr.txt which can provide very useful troubleshooting information.

9. SIMULATING CALLS

Currently the simulator only works in a Remedy thick client environment because simulation is via the Cuba Windows client. Version 2.1 Cuba for Remedy will provide a PBX simulator to simulate a complete end-to-end telephony environment.

You can use the Cuba simulator to test a Remedy thick client configuration without requiring a connection to a real telephone system. Therefore, CTI development and basic testing can be performed on a laptop. Customer testing can then be performed at a later stage when a CTI link is available.

To use the simulator, install the Cuba client program and follow these steps:

1. Right click on Cuba client systemtray icon and select Open menu.

2. Click Settings button.

3. Click Simulator button.

4. The Cuba Simulator screen allows you to enter a caller ID, called number, call ID, user data, agent id or collected digits which can be used to screenpop a Remedy form.

Note that the behaviour will be dependent on the rules configured in the Cuba web administrator. Complete this configuration first.

10. SOLUTION EXAMPLE

10.1 REMEDY THICK CLIENT SOLUTION EXAMPLE

To demonstrate the power and flexibilty available to Remedy developers, an advanced configuration is demonstrated providing sufficient information for implementers to deploy a similar system.

The example requirement is to screenpop different Remedy forms based on Caller ID, called number, collected digits and user data.

The combination of Cuba and Remedy's advanced workflow capabilities can enable sophisticated call processing capabilities. This example, demonstrates how the contents of the telephony data can be used to screenpop different forms.

Scenario: Ajax Systems is a manufacturer of components for the airline industry. Ajax provides a single helpdesk for internal users and customers. The helpdesk manager would like to use Cuba for screenpop to improve customer service. Before Cuba was introduced, considerable time was taken to identify the caller. In addition, due to the complexity of the products, customers very often had to communicate their problem to a number of helpdesk agents before reaching the most appropriate person to handle the call.

Proposed Solution: Cuba was proposed to provide a solution for the following two problems:

Faster identification and processing of caller.

Reduction in time taken for customer to explain problem when passed between agents.

Low level detailed Solution: The two high level problems above were solved using the following Cuba, telephone switch and Remedy features.

Use an IVR, interactive voice response, system to prompt internal users to enter employee ID to identify themselves.

Major account customers were provided with dedicated helpdesk numbers to assist in identifying these callers and routing their call to the right agents.

In situations where agents must transfer a caller onto another agent, the agents were trained to enter customer details directly into Remedy and where a transfer was required, the agent attached the Remedy record ID to the call when transferring. The destination agent would then be provided with the information collected by the other agent therefore saving the time required to obtain the information and dramatically improving customer satisfaction. The Cuba userdata feature was used to facilitate this.

All other callers could be identified by Caller ID.

Remedy workflow was used to analyse the telephony information provided by Cuba and intelligently screenpop the most appropriate helpdesk forms.

This five point solution will be demonstrated for both Remedy thick and thin client.

It is assumed that all your agent telephone extensions are already added to the devices section and the CTI link section is already completed.

10.1.1 CTI Form

The first step is to configure a form to receive all the telephony data from Cuba on specified telephone system events. A telephone system will generate a number of events. For the purposes of this example, the call related events are only considered. The call events are events which relate to a party to a call. For example an alerting event is the event which occurs when a phone starts ringing on a telephone extension. If the call is answered, then Cuba will generate a connected event.

Cuba supports a range of events including alerting, connected, cleared, offhook, onhook, initiated, diverted, transferred. To simplify implementation, offhook and onhook are not passed to Remedy. So if a user picks up the telephone and decides not to make a call and puts immediately goes onhook again, no events are passed to Remedy.

For the purposes of this example, the only event which will be used is the connected event. So a screenpop will only occur if a call is answered where the answering party is the destination for the call. In other words, no screenpop will occur for originated calls.

Follow these steps for Cuba configuration for Remedy thick client.

Create a new Remedy form called cuba:events. Create the following character fields: this dn, caller id, called id, userdata, agent id, digits, setfocus. Create a checkbox called thininvoke (checkbox required for Remedy thin client operation).

Using any web browser navigate to the host running Cuba Server. If, for example, the Cuba host was remcuba then navigate to http://remcuba:8080. Login with username admin and password: cuba.

Click on the remedy settings menu in the left hand menu bar. Enter the Remedy server name.

In the Remedy form name field enter cubaevents.

Tick the Screenpop internal calls checkbox.

In the Cuba telephony data section enter the following: This Device=this dn, Caller ID=caller id, Called ID=called id, Userdata=userdata, AgentID=agent id, SetFocus=setfocus.

In the screenpop rules section, Answered call actions section, tick the following: Target dn, Set focus, CallerID, CalledID, Userdata, Digits, AgentID.

Click the Save button to commit these changes.

Follow these steps for Cuba configuration for Remedy thin client:

Click on the remedythin menu in the left hand menu bar.

Enter the Remedy server name.

In the Remedy form name field enter cubaevents.

Enter the web server. In the case of Ajax, the Remedy mid-tier runs on the same host as the Remedy server so this entry is the same as for Remedy server name.

Tick the screenpop internal calls checkbox.

In the Cuba telephony data section enter the Field ID’s of the equivalent Remedy field names for Caller ID, Called ID, Userdata, AgentID, Digits.

Enter the Field ID of the thininvoke checkbox in the Tick/check checkbox field.

In the screenpop rules section, Answered call actions section, tick the following: Target dn, Set focus, CallerID, CalledID, Userdata, Digits, AgentID.

Click the save button to commit these changes.

10.1.2 Dedicated Helpdesk Numbers for Major Accounts

Ajax systems have two major accounts, AA and BA. Configure these numbers on the PBX. Cuba will extract the called number and present it as CalledID. Two numbers ending in 2001 and 2002 were configured on the switch.

To process calls to these two numbers, a Remedy active link will be configured to screenpop an AA form for calls to 2001 and a BA form for calls to 2002.

These are the steps:

Create two Remedy forms called cuba:aa and cuba:ba. These form should contain the following character fields: this dn, caller id, called id, userdata, agent id.

Create an active link called cuba:loadaa. Link to the cubaevents form. In the Run If section enter the following condition: ‘called id’ = “2001”.

Tick the Gain focus checkbox and in the field dropdown select the field named setfocus.

Now click on the If action tab and for New Action select Open Window. For Window type select Search. In the Form details section select form name: cuba:aa.

In the Fields in section map create the following mappings:

Fields in Search Form	Value
Agentid	$agentid$
Calledid	$calledid$
Callerid	$callerid$
Userdata	$userdata$
This dn	$this dn$

Perform steps 2-5 again but this time naming the active link cuba:loadba and the condition should be ‘called id’ = “2002”

The steps above are required to process Remedy thick client processing. For Remedy thin client operation, the active links are slightly different.

Perform these steps to configure for Remedy thin client:

Create an active link called cuba:thinloadaa. Link to the cubaevents form. In the Run If section enter the following condition: ‘called id’ = “2001”.

Tick the Return/Table or Level Dbl-Clk checkbox and in the field dropdown select the field named thininvoke.

Now click on the If action tab and for New Action select Open Window. For Window type select Search. In the Form details section select form name: cuba:aa.

In the Fields in section map create the following mappings:

Fields in Search Form	Value
agentid	$agentid$
Calledid	$calledid$
Callerid	$callerid$
Userdata	$userdata$
This dn	$this dn$

10.1.3 Avoiding caller having to explain problem each time they are transferred to a new agent

This problem is solved by using the Cuba user data feature. Implementation on Remedy thick client only is demonstrated.

For screenpop, Cuba is configured to pass telephony data to one Remedy form. However to implement softphone functionality, any number of Remedy forms can be used. Ajax decided to add dial, transfer and attach user data functionality to all forms containing contact information. But for this example, we will create just one form to demonstrate softphone features. Implementing a softphone for thin client is more complex so only thick client will be considered here.

Create a new Remedy form called cuba:ontacts. On this form create a button called Transfer and attach. Create a character field called destinationagent.

Also create the following character fields: firstname, lastname, telephone, notes.

Create an active link as follows:

Name the active link cuba:xferandattach, link to cuba:events form.

Tick the Button/Menu item check box and in the field dropdown select Transfer and attach.

In the Run If section enter the condition: ‘destinationagent’ != “”

Click the If Action tab and in Current Actions select Run Process. In the command line field enter:

```
%ComSpec% /c Echo ITFR $destinationagent$ $Request ID$ >
"%APPDATA%\Itel\cubacmd%date:/=% %time::=% %RANDOM%.cub"
```

Save the active link.

Note that this command writes a file to the local machine. This operation is not possible from within a web browser so can only be supported for Remedy thick clients.

The next stage in configuring userdata processing is to create Remedy workflow to handle the userdata on the destination agent.

Follw these steps:

Perform these steps to configure for Remedy thick client:

Create an active link called cuba:processudata. Link to the cubaevents form. In the Run If section enter the following condition: ‘userdata’ != “”.

Tick the Gain focus checkbox and in the field dropdown select the field named setfocus.

Now click on the If action tab and for New Action select Open Window. For Window type select Search. In the Form details section select form name: cuba:contacts.

In the Fields in section map create the following mappings:

Fields in Search Form	Value
Userdata	$Request ID$

The user will need to press enter on the form for the search to be performed.

Perform these steps to configure for Remedy thin client:

Create an active link called cuba:thinloadaa. Link to the cubaevents form. In the Run If section enter the following condition: ‘userdata’ != “”.

Tick the Return/Table or Level Dbl-Clk checkbox and in the field dropdown select the field named thininvoke.

Now click on the If action tab and for New Action select Open Window. For Window type select Search. In the Form details section select form name: cuba:contacts.

In the Fields in section map create the following mappings:

Fields in Search Form	Value
Userdata	$Request ID$

The user will need to press enter on the form for the search to be performed.

10.1.4 Screenpop on CallerID for everything else

This is now the last step, the catch-all processing where none of the above conditions are met. To prevent this processing to happen it is not required it is necessqary to set a condition to not run if any of the conditions for the above outcomes are valid.

Create a Remedy forms called cuba:popcallerid. This form should contain the following character fields: this dn, caller id, called id, agent id. Note that userdata or digits are not added because this form will only be screenpopped if no digits or userdata is available.

Create an active link called cuba:loadpopcallerid. Link to the cubaevents form. In the Run If section enter the following condition:

‘called id’ != “2001” AND ‘called id’ != “2002” AND ‘digits’ = “” AND ‘userdata’ = “”

Tick the Gain focus checkbox and in the field dropdown select the field named setfocus.

Now click on the If action tab and for New Action select Open Window. For Window type select Search. In the Form details section select form name: cuba:popcallerid.

In the Fields in section map create the following mappings:

Fields in Search Form	Value
Callerid	$callerid$

The user will need to press enter to invoke the search.

The steps above are required to process Remedy thick client processing. For Remedy thin client operation, the active links are slightly different.

Perform these steps to configure for Remedy thin client:

Create an active link called cuba:thinloadpopcallerid. Link to the cubaevents form. In the Run If section enter the following condition: ‘called id’ != “2001” AND ‘called id’ != “2002” AND ‘digits’ = “” AND ‘userdata’ = “”

Tick the Return/Table or Level Dbl-Clk checkbox and in the field dropdown select the field named thininvoke.

Now click on the If action tab and for New Action select Open Window. For Window type select Search. In the Form details section select form name: cuba:popcallerid.

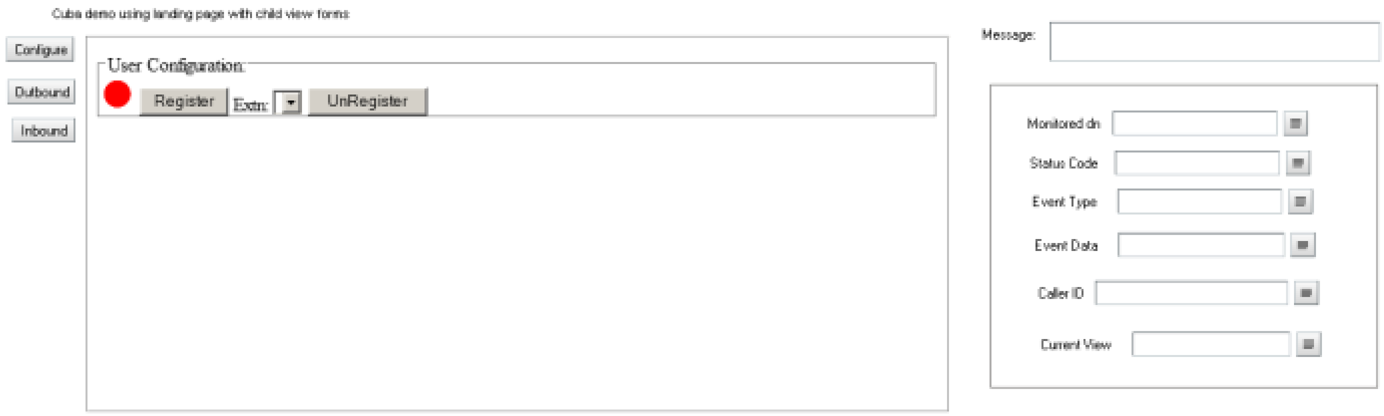
In the Fields in section map create the following mappings:

Fields in Search Form	Value
Callerid	$callerid$

10.2 REMEDY THIN CLIENT SOLUTION EXAMPLE

The sample Cuba def file, cuba.def uses a simulation of SHR:LandingConsole, called CUBA:LandingTest to demonstrate a sample of how a solution could be put together in a real Remedy application. cuba.def is copied to the Cuba server installation folder on installation.

Here is a how CUBA:LandingTest looks like in BMC Developer Studio.



The web page within a view field in CUBA:LandingTest is web page, configure.html, provided as part of the Cuba Servlet package. It is inside cuba4web.war and is unpacked if the war file is deployed to Tomcat. If deploying to Tomcat is inconvenient, the war file can be opened by:

1. unzipping using 7-Zip.
2. Running this command at the command line: jar xvf cuba4web.war

The def file view field links to the URL: http://remedywebserver:8080/cuba4web/configure.html

Probably your mid-tier URL will be different from this so either edit the text in the def file manually or via BMC Developer studio. Usually you will only need to edit the string remedyserver to your mid-tier hostname or domain name.

To use the telephony features, a CTI link between Cuba Server and a PBX will be required. however, by populating the list of devices in Cuba Web Admin, this list should be visible in the Extensions dropdown list which shows that the communication link from the web client via Cuba Servlet to Cuba Server is working.

The important steps in creating a working Remedy CTI solution like CUBA:LandingTest are as follows:

1. Follow installation steps described above for Cuba in a Remedy thin client environment.

3. On the designated Remedy form create a Run Process Active Link to run on loading the form to call startCuba function. In the demo the active link is called CUBA:AL_StartCuba and the run process command is: javascript:cuba.startCuba();

4. To create softphone functionality you create an active link with a run process command such as:

```
 javascript:cuba.makeCall($txtNumber$);
```

The demo uses javascript:window.top.cuba.makeCall(536870915);

This is because the Remedy form with the button is a child form of CUBA:LandingTest and the Cuba object resides in that form. window.top means refer to the top - which is the parent. Every web page instance has a window object. window.top refers to the top level parent of this window.

Then create a button on a form and link it to the active link to invoke the makecall request.

For further information contact support@iteloffice.com