Installing and configuring **Avaya Callback Assist**

Release 4.4.1.0

August 2016

Notice

While reasonable efforts have been made to ensure that the information in this document is complete and accurate at the time of printing, Avaya assumes no liability for any errors. Avaya reserves the right to make changes and corrections to the information in this document without the obligation to notify any person or organization of such changes.

Documentation disclaimer

"Documentation" means information published by Avaya in varying mediums which may include product information, operating instructions and performance specifications that Avaya may generally make available to users of its products and Hosted Services. Documentation does not include marketing materials. Avaya shall not be responsible for any modifications, additions, or deletions to the original published version of documentation unless such modifications, additions, or deletions were performed by Avaya. End User agrees to indemnify and hold harmless Avaya, Avaya's agents, servants and employees against all claims, lawsuits, demands and judgments arising out of, or in connection with, subsequent modifications, additions or deletions to this documentation, to the extent made by End User.

Link disclaimer

Avaya is not responsible for the contents or reliability of any linked websites referenced within this site or documentation provided by Avaya. Avaya is not responsible for the accuracy of any information, statement or content provided on these sites and does not necessarily endorse the products, services, or information described or offered within them. Avaya does not guarantee that these links will work all the time and has no control over the availability of the linked pages.

Warranty

Avaya provides a limited warranty on Avaya hardware and software. Refer to your sales agreement to establish the terms of the limited warranty. In addition, Avaya's standard warranty language, as well as information regarding support for this product while under warranty is available to Avaya customers and other parties through the Avaya Support website: http://support.avaya.com or such successor site as designated by Avaya. Please note that if You acquired the product(s) from an authorized Avaya Channel Partner outside of the United States and Canada, the warranty is provided to You by said Avaya Channel Partner and not by Avaya.

Licenses

THE SOFTWARE LICENSE TERMS AVAILABLE ON THE AVAYA WEBSITE, HTTP://SUPPORT.AVAYA.COM/LICENSEINFO OR SUCH SUCCESSOR SITE AS DESIGNATED BY AVAYA, ARE APPLICABLE TO ANYONE WHO DOWNLOADS, USES AND/OR INSTALLS AVAYA SOFTWARE, PURCHASED FROM AVAYA INC., ANY AVAYA AFFILIATE, OR AN AVAYA CHANNEL PARTNER (AS APPLICABLE) UNDER A COMMERCIAL AGREEMENT WITH AVAYA OR AN AVAYA CHANNEL PARTNER. UNLESS OTHERWISE AGREED TO BY AVAYA IN WRITING, AVAYA DOES NOT EXTEND THIS LICENSE IF THE SOFTWARE WAS OBTAINED FROM ANYONE OTHER THAN AVAYA, AN AVAYA AFFILIATE OR AN AVAYA CHANNEL PARTNER; AVAYA RESERVES THE RIGHT TO TAKE LEGAL ACTION AGAINST YOU AND ANYONE ELSE USING OR SELLING THE SOFTWARE WITHOUT A LICENSE. BY INSTALLING, DOWNLOADING OR USING THE SOFTWARE, OR AUTHORIZING OTHERS TO DO SO, YOU, ON BEHALF OF YOURSELF AND THE ENTITY FOR WHOM YOU ARE INSTALLING, DOWNLOADING OR USING THE SOFTWARE (HEREINAFTER REFERRED TO INTERCHANGEABLY AS "YOU" AND "END USER"), AGREE TO THESE TERMS AND CONDITIONS AND CREATE A BINDING CONTRACT BETWEEN YOU AND AVAYA INC. OR THE APPLICABLE AVAYA AFFILIATE ("AVAYA").

Avaya grants You a license within the scope of the license types described below, with the exception of Heritage Nortel Software, for which the scope of the license is detailed below. Where the order documentation does not expressly identify a license type, the applicable license will be a Designated System License. The applicable number of licenses and units of capacity for which the license is granted will be one (1), unless a different number of licenses or units of capacity is specified in the documentation or other materials available to You. "Software" means computer programs in object code, provided by Avaya or an Avaya Channel Partner, whether as stand-alone products, pre-installed on hardware products, and any upgrades, updates, patches, bug fixes, or modified versions thereto. "Designated Processor" means a single stand-alone computing device. "Server" means a Designated Processor that hosts a software application to be accessed by multiple users. "Instance" means a single copy of the Software executing at a particular time: (i) on one physical machine; or (ii) on one deployed software virtual machine ("VM") or similar deployment.

License type(s)

Designated System(s) License (DS). End User may install and use each copy or an Instance of the Software only on a number of Designated Processors up to the number indicated in the order. Avaya may require the Designated Processor(s) to be identified in the order by type, serial number, feature key, Instance, location or other specific designation, or to be provided by End User to Avaya through electronic means established by Avaya specifically for this purpose.

Concurrent User License (CU). End User may install and use the Software on multiple Designated Processors or one or more Servers, so long as only the licensed number of Units are accessing and using the Software at any given time. A "Unit" means the unit on which Avaya, at its sole discretion, bases the pricing of its licenses and can be, without limitation, an agent, port or user, an e-mail or voice mail account in the name of a person or corporate function (e.g., webmaster or helpdesk), or a directory entry in the administrative database utilized by the Software that permits one user to interface with the Software. Units may be linked to a specific, identified Server or an Instance of the Software.

Database License (DL). End User may install and use each copy or an Instance of the Software on one Server or on multiple Servers provided that each of the Servers on which the Software is installed communicates with no more than one Instance of the same database.

CPU License (CP). End User may install and use each copy or Instance of the Software on a number of Servers up to the number indicated in the order provided that the performance capacity of the Server(s) does not exceed the performance capacity specified for the Software. End User may not re-install or operate the Software on Server(s) with

a larger performance capacity without Avaya's prior consent and payment of an upgrade fee.
Named User License (NU). You may: (i) install and use the Software on a single Designated Processor or Server per authorized Named User (defined below); or (ii) install and use the Software on a Server so long as only authorized Named Users access and use the Software. "Named User", means a user or device that has been expressly authorized by Avaya to access and use the Software. At Avaya's sole discretion, a "Named User" may be, without limitation, designated by name, corporate function (e.g., webmaster or helpdesk), an e-mail or voice mail account in the name of a person or corporate function, or a directory entry in the administrative database utilized by the Software that permits one user to interface with the Software.
Shrinkwrap License (SR). You may install and use the Software in accordance with the terms and conditions of the applicable license agreements, such as "shrinkwrap" or "clickthrough" license accompanying or applicable to the Software ("Shrinkwrap License").

Copyright

Except where expressly stated otherwise, no use should be made of materials on this site, the Documentation, Software, Hosted Service, or hardware provided by Avaya. All content on this site, the documentation, Hosted Service, and the product provided by Avaya including the selection, arrangement and design of the content is owned either by Avaya or its licensors and is protected by copyright and other intellectual property laws including the sui generis rights relating to the protection of databases. You may not modify, copy, reproduce, republish, upload, post, transmit or distribute in any way any content, in whole or in part, including any code and software unless expressly authorized by Avaya. Unauthorized reproduction, transmission, dissemination, storage, and or use without the express written consent of Avaya can be a criminal, as well as a civil offense under the applicable law.

Third Party Components

"Third Party Components" mean certain software programs or portions thereof included in the Software or Hosted Service may contain software (including open source software) distributed under third party agreements ("Third Party Components"), which contain terms regarding the rights to use certain portions of the Software ("Third Party Terms"). As required, information regarding distributed Linux OS source code (for those products that have distributed Linux OS source code) and identifying the copyright holders of the Third Party Components and the Third Party Terms that apply is available in the products, Documentation or on Avaya's website at: http://support.avaya.com/Copyright or such successor site as designated by Avaya. You agree to the Third Party Terms for any such Third Party Components.
THIS PRODUCT IS LICENSED UNDER THE AVC PATENT PORTFOLIO LICENSE FOR THE PERSONAL USE OF A CONSUMER OR OTHER USES IN WHICH IT DOES NOT RECEIVE REMUNERATION TO (i) ENCODE VIDEO IN COMPLIANCE WITH THE AVC STANDARD ("AVC VIDEO") AND/OR (ii) DECODE AVC VIDEO THAT WAS ENCODED BY A CONSUMER ENGAGED IN A PERSONAL ACTIVITY AND/OR WAS OBTAINED FROM A VIDEO PROVIDER LICENSED TO PROVIDE AVC VIDEO. NO LICENSE IS GRANTED OR SHALL BE IMPLIED FOR ANY OTHER USE. ADDITIONAL INFORMATION MAY BE OBTAINED FROM MPEG LA, L.L.C. SEE HTTP://WWW.MPEGLA.COM.

Note to Service Provider

The product or Hosted Service may use Third Party Components subject to Third Party Terms that do not allow hosting and require a Service Provider to be independently licensed for such purpose. It is your responsibility to obtain such licensing.

Preventing Toll Fraud

"Toll Fraud" is the unauthorized use of your telecommunications system by an unauthorized party (for example, a person who is not a corporate employee, agent, subcontractor, or is not working on your company's behalf). Be aware that there can be a risk of Toll Fraud associated with your system and that, if Toll Fraud occurs, it can result in substantial additional charges for your telecommunications services.

Avaya Toll Fraud intervention

If You suspect that You are being victimized by Toll Fraud and You need technical assistance or support, call Technical Service Center Toll Fraud Intervention Hotline at +1-800-643-2353 for the United States and Canada. For additional support telephone numbers, see the Avaya Support website: http://support.avaya.com or such successor site as designated by Avaya. Suspected security vulnerabilities with Avaya products should be reported to Avaya by sending mail to: securityalerts@avaya.com.

Downloading Documentation

For the most current versions of Documentation, see the Avaya Support website: http://support.avaya.com, or such successor site as designated by Avaya.

Contact Avaya Support

See the Avaya Support website: http://support.avaya.com for product or Hosted Service notices and articles, or to report a problem with your Avaya product or Hosted Service. For a list of support telephone numbers and contact addresses, go to the Avaya Support website: http://support.avaya.com (or such successor site as designated by Avaya), scroll to the bottom of the page, and select Contact Avaya Support.

Trademarks

The trademarks, logos and service marks ("Marks") displayed in this site, the Documentation, Hosted Service(s), and product(s) provided by Avaya are the registered or unregistered Marks of Avaya, its affiliates, or other third parties. Users are not permitted to use such Marks without prior written consent from Avaya or such third party which may own the Mark. Nothing contained in this site, the Documentation, Hosted Service(s) and product(s) should be construed as granting, by implication, estoppel, or otherwise, any license or right in and to the Marks without the express written permission of Avaya or the applicable third party.
Avaya is a registered trademark of Avaya Inc.
All non-Avaya trademarks are the property of their respective owners. Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

Table of Contents

About this document 10

Intended audience 10

Related documents 10

Introduction to Callback Assist installation 11

Installation roadmap 11

Installation prerequisites 12

Referring to CBA Overview and planning guide 12

Acquiring a license file 12

Logging in as an Administrative User 12

Environment configuration 13

Browser requirements 13

Hardware requirements 13

Virtualization 14

High Availability 14

Network considerations 14

Firewall 14

Hosts files 17

Network Latency 17

Software requirements 18

CTI environment 18

SIP environment 19

AACC Environment 20

Linux Shared Memory 20

Setting up Google Profile Authentication System 21

Installing Callback Assist in a CTI environment 25

Checklist for setting up the system for Callback Assist 25

Configuring Avaya Aura® Communication Manager 27

Beginning the configuration 27
Global configuration 27
Understanding specific configurations for a callback 37
Setting up VDN Callback for ISDN PRI or IP trunks 40

Configuring AES 40
Pre-requisites 41
Global configuration 41

Installing Callback Assist Software 42
Software Installation Steps 43
Software Post installation steps 48

Configuring Avaya Aura® Experience Portal 57
Setting up Callback Customer Voice Application 57
Setting up Callback Agent Voice Application 58

Configuring Avaya Orchestration Designer 59

Installing Callback Assist in a SIP environment 61

Checklist for setting up the system for Callback Assist 61

Installing Callback Assist Software 62
Software Installation Steps 62

Configuring Avaya Aura® Communication Manager 70
Basic configuration 70
Creating Polling Vectors and VDNs for Best Service Routing (BSR) 73
Creating Queuing Vectors and VDNs for Best Service Routing (BSR) 75

Configuring Avaya Aura® Session Manager (ASM) 79
Logging on to System Manager Console 79
Adding a SIP domain 79
Adding SIP Entities for Communication Manager 79
Adding SIP Entities for Experience Portal 80
Adding SIP Entities for Callback Assist BSR Service 80
Adding SIP Entity Link for Communication Manager 81
Adding SIP Entity Link for Experience Portal 82
Adding SIP Entity Link for Callback Assist BSR Service 82
Adding Routing Policy for Communication Manager 83
Adding Routing Policy for Experience Portal 84
Dial Patterns 84

Configuring Avaya Aura® Experience Portal 86
Configuring VoIP Connections 86

Configuring Inbound and Outbound Applications 87

Post Installation Steps 93
Callback Assist Web Administration 93
Configuring Avaya Orchestration Designer 106

Installing Callback Assist in an AACC environment 110

Checklist for setting up the system for Callback Assist 111

Setting up a Dial Pattern (route) to AAEP in System Manager 112

Setting up the SIP trunk in Avaya Aura® Experience Portal 112

Setting up inbound application in Avaya Aura® Experience Portal 114

Setting up Outbound Application 115

Sample SCE Scripts 117

Sample SCE script to queue a call to a skillset 118

Installing Callback Assist Software 119
Software Installation Steps 119

Managing status of Callback services 123

Post-installation steps 126
Callback Assist Web Administration 126
Configuring Avaya Orchestration Designer 141

Post Installation 143

License Installation 143

Installing Callback Assist in a High Availability mode 147

Installing PostgreSQL Database Server for Callback Assist 147

Installing Callback Assist Core Components with external Database 149

Installing Webcallback and DD Apps Components 158
Checklist for setting up the system for Callback Assist 158
Installing Webcallback and DD Apps 158
Post Installation Steps 163
Uninstalling Webcallback and DD Apps 165

Reinstalling Webcallback and DD Apps 166
Upgrading Webcallback and DD Apps 169

Managing status of Callback services 174

Post-installation steps 176
Configuring license 178
Adding a custom certificate to connect to the WebLM Server 179
Configuring Global Settings 181
Adding Site Definitions 182
Adding Zone Definitions 186
Defining the Global Time Zone 190

Configuring Load Balancer 191
Configuring a Load Balancer for the Callback Assist Web Administration application 191
Configuring a Load Balancer for the IVR applications 192

Changing Linux box Time Zone 192

Installing Callback Assist in Dual Server mode 193

Installing PostgreSQL Database Server for Callback Assist 193

Installing Callback Assist Core Components with external Database 193

Post-installation steps 201

Changing the existing installation platform 202

Uninstalling Callback Assist 204

Reinstalling Callback Assist 206

Stop and Start Callback Assist in HA mode 207

Stop Callback Assist in HA mode 207

Start Callback Assist in HA mode 207

Backup and Rollback Callback Assist 207

Backup 207

Rollback 208

Persona Administration 209

Default Persona 209

Persona Package file 209

Add Persona 210

Update Persona 212

Remove Persona 214

Assigning Persona to Callback Configurations 215

Zoning Administration 216

Sites to Zones Migration 216

Zones to Sites Migration 218

Upgrading Callback Assist 219

Upgrading Single Server Deployment 220
Performing the upgrade 220

Upgrading High Availability Deployment 227
Checklist before upgrading 227
Upgrade Database Server(s) 228
Upgrade CBA Server(s) 230
Restore Configurations 230
Verify Audio Files in Riak File Server 231

Upgrading Dual Server Deployment 232
Backup 232
Upgrade Database Server(s) 232
Upgrade CBA Application Server 232
Restore Configurations 240

Post-upgrade steps 240
Restore Custom Audio Files 240
Restore Riak File Server 241
Customer First Callback Configuration changes for 4.1.6 or higher 242
Additional step when upgrading to 4.4.0.0 or higher in SIP/AACC Platform 242
Additional Steps when upgrading to 4.1.4 or higher 242

Examples of High Availability Installation and configuration 247

Environment 247

Installing Callback Assist 248
Installing Callback Assist Database on a separate server 248

Installing Callback Assist server 248
Installing the first Callback Assist server 248
Installing the second Callback Assist server 249

Configuring the Load Balancer 251

Configuring Callback Assist 251
Configuring the Audio Storage Server 251
Configuring Sites 252
Configuring Zones 253
Adding Callback Configurations 255
Configuring BSR 255
Configuring Experience Portal 256

Miscellaneous 258

Security-Enhanced Linux (SELinux) 258

Invoking CBASIPOffer as VXML sub dialog Sample 258

Database Tuning 260

Increase JVMs heap size for CBA components 260

Database Local Cache Setup 261

Connection Pool and Caching Settings for CBA Components 262
Engine 262
DD Apps 264
Admin 268
ICR Admin 271
Maintenance 274

CBA Alarms List 278

Result Size of getWebCallbackConfigurations WS API 280

About this document

Installing and configuring Avaya Callback Assist provides information on environment configuration for installing Callback Assist in CTI, SIP and AACC environments and also provides detailed installation procedures.

Intended audience

The document is addressed to users who want to install Callback Assist.

Related documents

Installing and configuring Avaya *Callback Assist* is part of the Callback Assist documentation set. The other documents in the documentation set are as follows:

- *Avaya Callback Assist Overview and Planning guide*
  Provides an overview and functional description of the CBA application. The document also lists the features of the CBA application and the environment configuration requirements to install the application.
- *Administering Avaya Callback Assist guide*
  Provides information on administering and using CBA.
- *Avaya Callback Assist reporting guide*
  Provides information on the CBA Reporting feature.
- *Avaya Callback Assist migration Guide*
  Provides information on preparing the user for migrating to the latest version of Callback Assist and procedure to migrate.
- *Avaya Callback Assist Web services API guide*
  Provides information on CBA Web services API.
- *Avaya Callback Assist Release Notes*
  Provides information on the latest version of CBA.

Introduction to Callback Assist installation

Starting from release 3.1, Avaya Callback Assist (CBA) runs only on Linux operating system. The installation is based on Linux shell scripts, also known as Installer Scripts. Using these scripts, you can install all the CBA modules with ease and flexibility.

The CBA application consists of the following modules:

- Callback Assist Web Administration application
- Callback Engine Service
- Callback DBMS and database (PostgreSQL)
- Callback Maintenance and Reporting Service
- Callback VXML applications (also known as Dialogs)
- Callback Call Control Application (CCA), available on SIP environment
- Callback File Server (also known as Audio Storage)
- Web Callback feature
- WebLM Server (4.7.1)

Installation roadmap

An overview of the tasks involved in installing and configuring CBA are as follows:

- Check the hardware and software requirements.
- Check the browser requirements.
- Configure the required applications, for example, Avaya Aura® Communication Manager, Avaya Application Enablement Services (AES) and Avaya Aura® Experience Portal (AAEP).
- Secure valid license files for CBA and other required applications.
- Chose the required CBA installation mode: single server, or high availability.
- Install CBA on a CTI environment or a SIP environment or an AACC environment and pick Delivery Strategy.
- Chose the authentication type: Internal, or External (OpenID ex. Google)
- Check the Administration Web interface.
- Check the configurations of applications through Global Settings.
- Place a call to test the CBA system.

Installation prerequisites

This section describes the procedures that you must follow to prepare yourself for the installation, configuration, and administration of the CBA software.

Referring to CBA Overview and planning guide

Read the *Avaya Callback Assist Overview and planning guide* and follow the instructions in the document before using this guide. If you do not have access to this document, or if you have any issues, contact your Avaya Professional Services representative or Business Partner for details.

Acquiring a license file

For more information, see the *Licensing* section in the *Avaya Callback Assist Overview and planning guide*.

Logging in as an Administrative User

To successfully install CBA, you must be logged in Linux as root, as all the services run as daemons under control of the init process.

Environment configuration

Callback Assist works in a *CTI environment or SIP environment or AACC environment*.

The *Installing and configuring Avaya Callback Assist* refers to the *CTI environment* for CBA for a call center in which the call control functionality of the application is implemented through Avaya AES; and Avaya Aura® Experience Portal, is deployed behind Communication Manager and connected to the Communication Manager using H.323 ports.

This document refers to a *SIP environment* for CBA for a call center in which the call control functionality of the application run on Avaya Aura® Experience Portal, and Avaya Aura® Experience Portal and Communication Manager are both SIP Entities of either Avaya Aura® Session Manager or Avaya Aura® Avaya SIP Enablement Services, connected through SIP trunks.

This document refers to an AACC environment for CBA for a call center in which the call control functionality of the application runs on Avaya Aura® Experience Portal, and Avaya Aura® Experience Portal, AACC and CM are SIP Entities of Avaya Aura® Session Manager, while all the call center administration is performed using Avaya Aura® Contact Center connected to CM.

Browser requirements

- Firefox 15.0 or higher
- Internet Explorer 7.0

Hardware requirements

Server hardware requirements for CBA depend on two factors, namely, number of Experience Portal ports and number of agent skills with callback functionality. Contact your Avaya Professional Services representative to know more about the hardware requirements based on your organizational needs.

The minimum hardware requirements for the server configuration are as follows:

Hardware requirement	Description
Processor	Intel Xeon E5520 Quad-Core processor, 8MB Cache, 2.26 GHz or higher.
RAM	16 GB ECC DDR3
Hard disk drive	120 GB Disk, 7200 RPM
Network	Gigabit Ethernet (GbE or 1 GigE) NIC
Screen Resolution	1280×800 or higher with 16-bit color

Virtualization

Callback Assist supports virtualization if the Virtual Machine in which you are installing Callback Assist meets the minimum hardware requirements. For more information, see the *Hardware requirements* section.

High Availability

Callback Assist can be deployed in a high availability format. A fully HA deployment requires at least four application servers. Out of these four servers, two servers will be used to host the Callback Assist Core Components (primary/slave) and two servers will be used to host the Callback Assist database and the database replication. All the servers must meet the minimum hardware requirements.
For more information, see the *Hardware requirements* section.

Network considerations

Firewall

If you enable a firewall on the Callback Assist application server, for example, Linux IP Tables or between the major supporting components of the solution, that is, AES, Experience Portal, and Session Manager, and the Callback Assist server on the LAN, you might require to open the following TCP ports on the firewall to enable proper operation of Callback Assist:

Source	Port	Destination	Port	Traffic Type	Notes
Web browser	Dynamic	Callback Assist Web Administration application	80	Network and Local	

Source	Port	Destination	Port	Traffic Type	Notes
Web browser	Dynamic	Callback Assist Web Administration application	8093	Network and Local	
Experience Portal	Dynamic	Callback Assist Customer and Agent applications	8080	Network	
Any client application	Dynamic	Callback Assist Web Services server	8081	Network	
All Callback Assist services	Dynamic	BSR Application Server (SIP only)	8086	Network	
All Callback Assist services	Dynamic	BSR Application Server (SIP only)	8087	Network	
All Callback Assist services	Dynamic	BSR Application Server (SIP only)	8089	Network	
Callback Assist WebLM	Dynamic	Callback Assist WebLM server (http)	9090	Network	
All Callback Assist services	Dynamic	Callback Assist WebLM server (https)	8443	Network	
Callback Assist WebLM	Dynamic	Callback Assist WebLM tomcat (shutdown)	8445	Local	
Callback Assist WebLM	Dynamic	Callback Assist WebLM tomcat (AJP)	8449	Local	
Callback Assist Engine	Dynamic	Experience Portal Outcall Web Service (SIP only)	80	Network	
Callback Assist Engine	Dynamic	AES (CTI only)	450	Network	
Callback Assist Management Scripts	Dynamic	BSR Application Server (shutdown) (SIP only)	8006	Network	
All Callback	Dynamic	Callback Assist Engine	6177	Local	

Source	Port	Destination	Port	Traffic Type	Notes
Assist services					
All Callback Assist services	Dynamic	Callback Assist Engine	6178	Local	
All Callback Assist services	Dynamic	Callback Assist Engine	6179	Local	
All Callback Assist services	Dynamic	Callback Assist Postgres Database	6198	Local	
Callback Assist Management Scripts		Callback Assist VXML applications (shutdown)	8008	Local	
Callback Assist Management Scripts		Callback Assist Web Administration application (shutdown)	8011	Local	
Callback Assist Management Scripts		Callback Assist Web Callback (shutdown)	8009	Local	
Callback Assist Storage Backend	Dynamic	Callback Assist Storage Backend (cluster/single)	TCP:4369	Network and Local	Epmd's listener (Erlang port manager deamon)
Callback Assist Storage Backend	Dynamic	Callback Assist Storage Backend (cluster/single)	TCP:8099	Network and Local	Handoff Port Listener
Callback Assist Storage Backend	Dynamic	Callback Assist Storage Backend (cluster/single)	TCP:[6200-7999]	Network and Local	inter-node communication port range
Callback Assist Storage Backend	Dynamic	Callback Assist Storage Backend (cluster/single)	TCP:8098	Network and Local	Web Port
Callback Assist Storage Backend	Dynamic	Callback Assist Storage Backend (cluster/single)	TCP:8070	Network and Local	Protocol Buffers

Hosts files

You must have the */etc/hosts* file configured with the current server's IP Address for Callback Assist to work properly. If this file is not configured with the current server's IP Address, then Callback Assist will not be able to determine the current server's IP Address causing the main components to malfunction. You must configure this setting irrespective of whatever platform, Delivery Strategy, or deployment type you choose for installing Callback Assist.

Moreover, if you plan to install a High Available environment, you must configure all the Callback Assist Servers' IP addresses involved in HA operation in the */etc/hosts* file. If you are planning to use hostname in the input fields (installer options), make sure the hostname is a valid FQDN (Fully Qualified Domain Name) and properly configured in /etc/hosts file. Before running script makes sure the remote server(s) are able to reach using ping command [ping hostname].

The following is an example of /etc/hosts file:

```
127.0.0.1 localhost.localdomain localhost
::1 localhost6.localdomain6 localhost6
10.10.10.1 gateway.domain.com
10.10.10.2 cbaappserver1.domain.com
10.10.10.3 cbaappserver2.domain.com
10.10.10.4 cbamasterdb.domain.com
10.10.10.5 cbastandbydb.domain.com
```

Network Latency

The following network latency requirements should be met for the correct functioning of Callback Assist.

- Latency between EPMS/MPP and CBA server: < 100 ms
- Latency between CBA components (for a HA deployment, including DB): < 30 ms

Software requirements

This section provides a list of software requirements for CBA.

CTI environment

- Callback Application Server Operating Systems (OS):
    - Red Hat Enterprise Linux (RHEL) or CentOS Linux, versions 5.3, 5.4, 6.0, 6.2 or 6.5 either 32 bits or 64 bits and 6.6, 6.7, 6.8 64 bits.
- Avaya Application Enablement Services (AES): 5.2.x, 6.1.x, 6.2.x, 6.3.0 to 6.3.3 or 7.0
- Avaya Aura® Experience Portal: 6.0 Service Pack 2, 6.0 Service Pack 3 (6.0.3), 7.0, 7.0 Service Pack 1 (7.0.1), 7.0 Service Pack 2 (7.0.2)
- Avaya Aura® Communication Manager with Avaya Call Center Elite Software package: 5.x, 6.2, 6.3, 7.0

  Note: Note: Make sure that you have Special Application (Green Feature) SA8874 with your Communication Manager. You must have a call status messages for 7434ND IP phones. SA8874 provides detailed call-progress indications to Avaya Aura® Experience Portal when the incoming or outgoing call is either a station-to-station call or is over a PRI trunk.
  See Avaya Aura® Communication Manager Documentation and Configuration Note 3910 for more information.
- CBA Local WebLM version 4.7.1 installed during the CBA installation. Also it supports the external WebLM server versions 4.6, 4.7.1, 6.3 or 6.3.10.

  Note:
  WebLM Servers that CBA supports out of the box:
  CBA supports the WebLM Servers shipped with Avaya Aura® Experience Portal 6.0 or 7.0, AES 5.2, and Avaya Aura® System Manager 6.1. If you plan to use a different WebLM Server, you must add the WebLM certificate to the CBA keystore files manually.
  For more information, see the *Adding a custom certificate to connect to the WebLM Server* section in this document.

IMPORTANT: CBA does not support Windows Server, Avaya Interactive Response (IR) and Media Processing Platform (MPS).

SIP environment

- Callback Application Server Operating Systems (OS):
    - Red Hat Enterprise Linux (RHEL) or CentOS Linux, versions 5.3, 5.4, 6.0, 6.2 or 6.5 either 32 bits or 64 bits and 6.6, 6.7, 6.8 64 bits.
- Avaya Aura® Session Manager: 6.1, 6.2, 6.3 or 7.0
- Avaya Aura® Experience Portal: 6.0 Service Pack 2, 6.0 Service Pack 3 (6.0.3), 7.0, 7.0 Service Pack 1 (7.0.1), 7.0 Service Pack 2 (7.0.2)
    - Experience Portal 7.0.1 Zoning feature is supported
    - Enhanced Call Classification license is required for using any call classification services
    - If using AAEP Signaling Ports and shuffling media on SIP calls, Avaya Aura Experience Portal 6 requires Patch 6.0.2.0.0517
- Avaya Aura® Communication Manager with Avaya Call Center Elite Software package: 5.2.1, 6.0.1, 6.2, 6.3 or 7.0
    - For using the AAEP Signaling ports feature, versions 5.2.1 and 6.0.1 require the following patch:
        - For Communication Manager Version 6.0.1 install Service Pack 4.
        - For Communication Manager Version 5.2.1 install Service Pack 12.
- Session Border Controller requirement: If outbound calls are routed directly to a SBC from Session Manager the SBC must support SIP RFC5589 (http://tools.ietf.org/html/rfc5589) to perform SIP REFER w/Replaces.
- On HA deployments, a Load Balancer tool with support of session affinity strategies.
- CBA Local WebLM version 4.7.1 installed during the CBA installation. Also it supports the external WebLM server versions 4.6, 4.7.1, 6.3 or 6.3.10.

Note:
WebLM Servers that CBA supports out of the box:
CBA supports the WebLM Servers shipped with AVP 5.1, Avaya Aura® Experience Portal 6.0 or 7.0, AES 5.2 and Avaya Aura® System Manager 6.1. If you plan to use a different WebLM Server, you must add the WebLM certificate to the CBA keystore files manually.
For more information, see the *Adding a custom certificate to connect to the WebLM Server* section in this document.

IMPORTANT: CBA does not support Windows Server, Avaya Interactive Response (IR) and Media Processing Platform (MPS).

AACC Environment

- Callback Application Server Operating System (OS):
    - Red Hat Enterprise Linux (RHEL) or CentOS Linux, versions 5.3, 5.4, 6.0, 6.2 or 6.5 either 32 bits or 64 bits and 6.6, 6.7, 6.8 64 bits.
- Avaya Aura® Contact Center 7.
  Note: CBA supports only deployments with Avaya Aura® Communication Manager.
- Avaya Aura® Experience Portal 7.0 Feature Pack 1 (7.0.1).
  Note: In addition, you must have an Enhanced Call Classification license for using any call classification services.
- Session Border Controller requirement: If outbound calls are routed directly to a SBC from Session Manager the SBC must support SIP RFC5589 (http://tools.ietf.org/html/rfc5589) to perform SIP REFER w/Replaces.

Linux Shared Memory

The table below shows the minimum shared memory settings required for installing Callback Assist software.

64 Bit Architecture		32 Bit Architecture	
parameter	**Value**	**parameter**	**Value**
kernel.shmmax	17179869184	kernel.shmmax	2147483648
kernel.shmall	4194304	kernel.shmall	4194304

On the Linux shell logged on as root run the following commands to verify the current values.

cat /proc/sys/kernel/shmmax
cat /proc/sys/kernel/shmall

Changing the Shared Memory parameters

Ignore this section if the values are already above the required minimum values.

1. Open sysctl.conf file
   vi /etc/sysctl.conf
2. Add or edit the following values in the file depending on the system architecture
   kernel.shmmax = 2147483648
   kernel.shmall = 4194304
3. Save the changes by entering
   Escape + wq!
4. Run the below command to make the changes take effect
   sysctl –p

Setting up Google Profile Authentication System

Basics

During the installation process (After selecting the Platform & Strategy, it will ask to select the authentication type (INTERNAL or EXTERNAL).

Internal authentication type is nothing but the users will be authenticated via CBA's Postgres database. External authentication type is CBA users will be authenticated via Google Profile (Oauth 2.0). Initially the username will be validated against CBA's internal database, if the user is present, then it will redirect to Google Authentication server to authenticate and redirect to CBA Administration page.

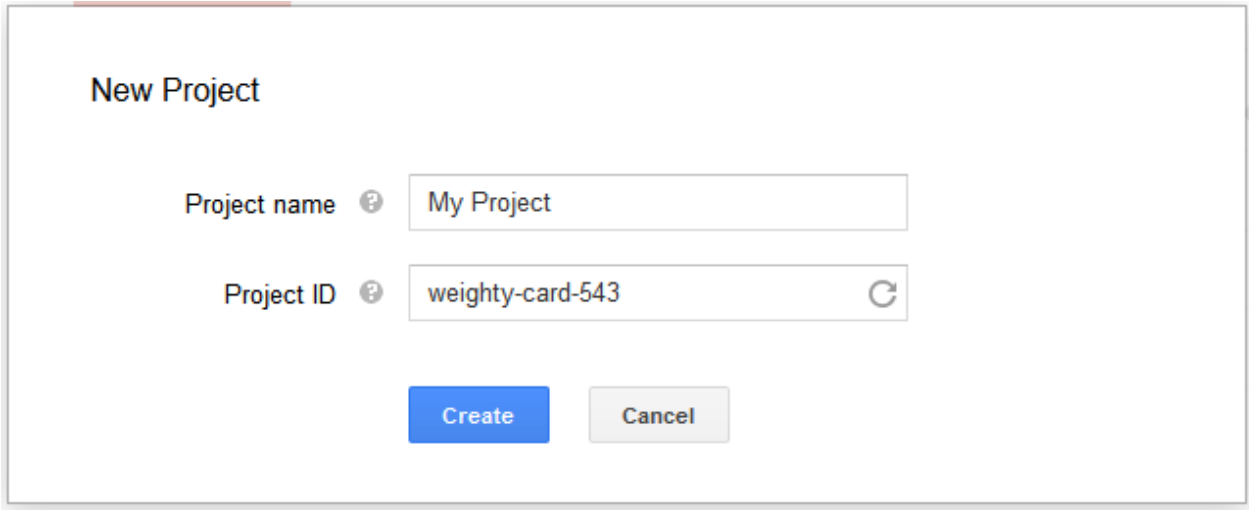
Before CBA application can use Google's Oauth 2.0 authentication system for user login, we must set up a project in the Google Developers Console to obtain Oauth 2.0 credentials (client_secret and client_id), set a redirect URI, and (optionally) customize the branding information that CBA users see on the user-consent screen.

Obtain Oauth 2.0 credentials

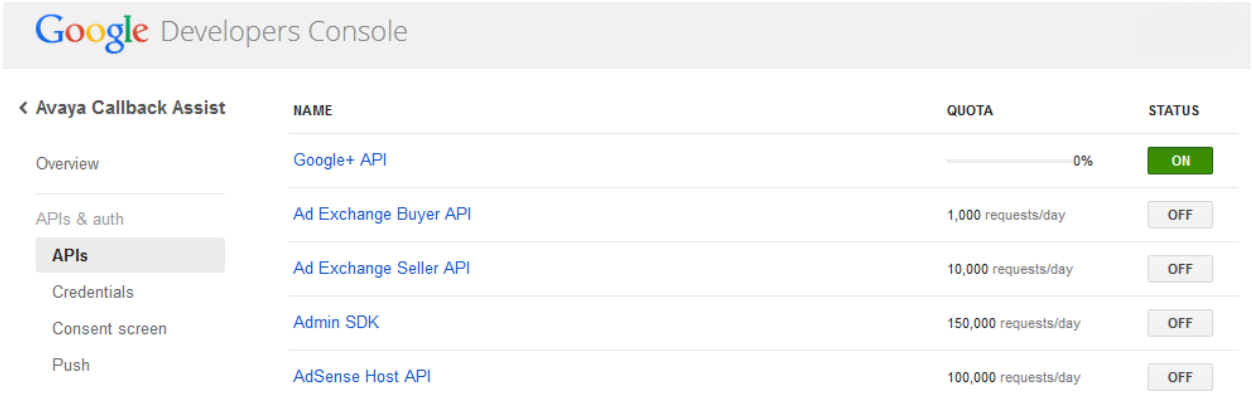
We need Oauth 2.0 credentials, including a client ID and client secret, to authenticate users and gain access to Google's APIs.

To find our project's client ID and client secret, do the following:

1. Go to the Google Developers Console.
2. Create a project by clicking "**CREATE PROJECT**" button. Provide the project name as "Avaya Callback Assist" and leave the project ID as default.

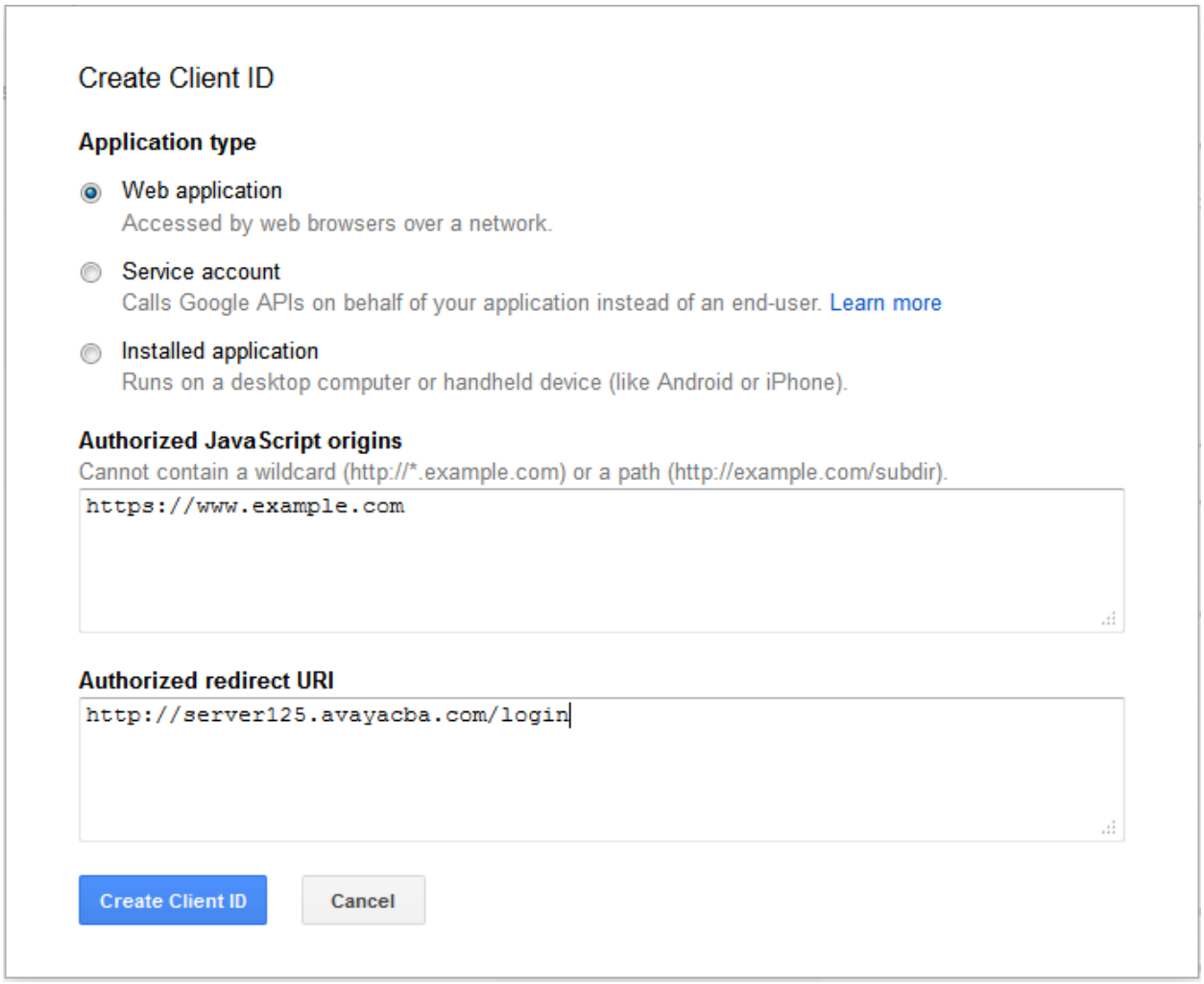


3. In the sidebar on the left, select **APIs & auth**. In the displayed list of APIs, make sure Google+ API should be selected (status ON) to use google profile as authentication type.



4. In the sidebar on the left, select **Credentials** and click **CREATE NEW CLIENT ID** button.
   i) Choose the "Application Type" as Web application
   ii) Leave the default value in "Authorized JavaScript origins".
   iii) Change the "**Authorized Redirect URI**" text box to Callback Assist Administration Login URL http://[hostname]/login.

   Note: The redirect URI must be valid hostname. This field will not accept IP Address or invalid hostname. If this is HA deployment then provide the load balancer machine's hostname.


Create Client ID
Application type
Web application
Accessed by web browsers over a network.
Service account
Calls Google APIs on behalf of your application instead of an end-user. Learn more
Installed application
Runs on a desktop computer or handheld device (like Android or iPhone).
Authorized JavaScript origins
Cannot contain a wildcard (http://*.example.com) or a path (http://example.com/subdir).
https://www.example.com
Authorized redirect URI
http://server125.avayacba.com/login
Create Client ID
Cancel

5. Click the “Create Client Id” button to submit the details.

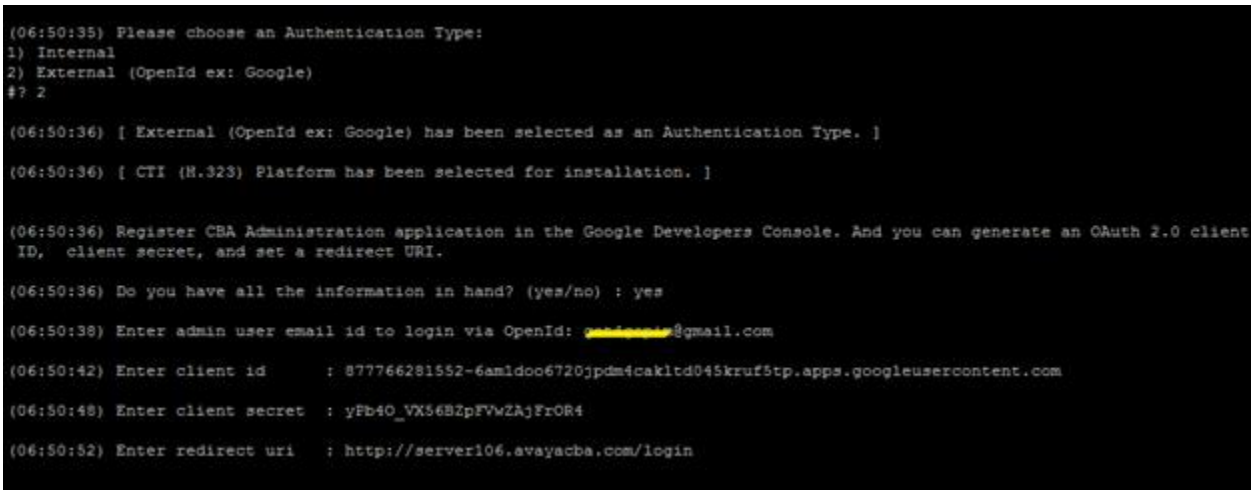
6. The Client ID, Client secret and Redirect URIs information will be required during the installation process.

CBA Installation as External Authentication

During the installation process (After selecting the Platform & Strategy) user, asked to select the authentication type (INTERNAL or EXTERNAL).

1. If the authentication type selected as External then the installation will confirm whether all the information available to proceed the installation. If not then the installation will be terminated.
2. Information required are:
   - i) Admin user’s email id (This is required to login into Callback Assist Administration application for the first time using this username. If this is an upgrade installation then the existing default username “admin” will be changed to this username.)
   - ii) Client Id
   - iii) Client Secret
   - iv) Redirect URI

   Note: Client Id, Client Secret and Redirect URI are obtained during the Oauth2.0 Credentials Setup process.



3. Once all the information is provided then the installation will be continued as per the strategy selected.

Installing Callback Assist in a CTI environment

This section guides you through the steps to configure different applications to install Callback Assist in a CTI environment, including the following:

- Configuration of Avaya Aura® Communication Manager
- Configuration of Avaya Application Enablement Services (AES)
- Installation and Configuration of Callback Assist
- Configuration of Avaya Aura® Experience Portal (AAEP)
- Configuration of Orchestration Designer runtime (AAOD)

Checklist for setting up the system for Callback Assist

You must be careful in configuring Communication Manager to ensure a flawless Avaya Callback Assist installation. Use the following checklist to plan the activities to set up Communication Manager to work with Avaya Callback Assist, and also set up the other required components.

#	Activity	Description/Reference	Completed?
1	Check the licensing requirements.	For more information, see the *Licensing* section in the *Avaya Callback Assist overview and planning*	

		guide.	
2	Check the hardware requirements.	For more information, see the Hardware requirements section in this document.	
3	Check the software requirements.	For more information, see the Software requirements for CTI environment section in this document.	
4	Install the required Linux OS in the Callback Assist server.		
5	Verify the Linux Shared Memory	For more information, see the Linux Shared Memory section in this document.	
6	Check the network considerations (firewall).	For more information, see the Network considerations section in this document.	
7	Map the information of the AAEP/AVP extensions (ports) numbers to the AAEP/VP Ports.		
8	Check the ASAI Enhanced Features settings, Converse-on Feature Access Code, and Data Delay settings.		
9	Create the Phantom Extensions.		
10	Create the Skills to transfer calls to AAEP/VP Ports.		
11	Create Agent Login IDs for AAEP/VP Ports.		
12	Create one call vector and VDN to transfer calls to AAEP/VP, known as “Phantom Port To IVR Transfer VDN”.		
13	Create the Incoming Calls Vector and Initial VDNs for each of the callback configurations.		
14	Create the Outgoing Calls Vector and Outgoing VDNs for each of the callback configurations.		
15	Install and configure Callback Assist as shown on Installing Callback Assist Software		
16	Configure the user credentials for Application		

	Enablement Services (AES).		
17	Configure the Avaya Aura® Experience Portal (Callback Customer Voice Application and Callback Agent Voice Application).		
18	Configure Avaya Orchestration Designer for the licensing server.		
19	Configure Google OAuth2.0 Authentication System. (Only for the Google Profile Authentication Type)	For more information, see the Setting up Google Profile Authentication System section in this document.	

Configuring Avaya Aura® Communication Manager

The following procedures are applicable only for Avaya Aura® Communication Manager. For additional help or support on Avaya Aura® Communication Manager, see the respective product documentation.

Beginning the configuration

Make sure you have access to Avaya Aura® Communication Manager and your user name has administrative privileges. You can use either *Avaya Site Administration* or other remote access methods. Use the *Graphical Enhanced Interface (GEDI).*

Also, make sure the Class of Restriction (COR) for all elements (stations and hunt groups) allows making outbound calls. Otherwise the agents will not be able to connect to the customers when the system executes the callback requests.

Global configuration

1. *Checking the Optional Features settings*

- Go to the **Customer System Parameter** screen by running the command: **`display system-parameters customer-options`**. Verify that the **Computer Telephony Adjunct Links** field parameter is set to ***y***:

```
display system-parameters customer-options                    Page   3 of  11
                              OPTIONAL FEATURES

    Abbreviated Dialing Enhanced List? y          Audible Message Waiting? y
        Access Security Gateway (ASG)? n            Authorization Codes? y
        Analog Trunk Incoming Call ID? y                    CAS Branch? n
 A/D Grp/Sys List Dialing Start at 01? n                      CAS Main? n
Answer Supervision by Call Classifier? y                Change COR by FAC? n
                                ARS? y  Computer Telephony Adjunct Links? y
                 ARS/AAR Partitioning? y  Cvg Of Calls Redirected Off-net? y
          ARS/AAR Dialing without FAC? y                      DCS (Basic)? y
          ASAI Link Core Capabilities? y                DCS Call Coverage? y
          ASAI Link Plus Capabilities? y                DCS with Rerouting? y
       Async. Transfer Mode (ATM) PNC? n
  Async. Transfer Mode (ATM) Trunking? n    Digital Loss Plan Modification? n
              ATM WAN Spare Processor? n                        DS1 MSP? y
                                ATMS? y            DS1 Echo Cancellation? y
                  Attendant Vectoring? y

       (NOTE: You must logoff & login to effect the permission changes.)
```

Figure 1 - Checking CM Optional Features

- If this field is not enabled, please contact your Avaya Professional Services representative or Business Partner to enable this field for you.

2. *Setting Converse-on Feature Data Return Code*

- Go to the **Feature Access Code** screen by running the following command: **`change feature-access-code`**. Navigate to page 6 or search for **Converse Data Return Code** and set a number for this field. For example: *116* or **65*. Do not change the value if the field is already configured. Make a note of the access code number, as you will require this number to configure the CBA Web Administration application:

```
display feature-access-codes                                  Page   6 of   9
                            FEATURE ACCESS CODE (FAC)

                        Call Vectoring/Prompting Features

   Converse Data Return Code: *65

Vector Variable 1 (VV1) Code:
Vector Variable 2 (VV2) Code:
Vector Variable 3 (VV3) Code:
Vector Variable 4 (VV4) Code:
Vector Variable 5 (VV5) Code:
Vector Variable 6 (VV6) Code:
Vector Variable 7 (VV7) Code:
Vector Variable 8 (VV8) Code:
Vector Variable 9 (VV9) Code:
```

Figure 2 - Setting CM Converse Data Return Code

3. *Setting Converse-on Data Delay*

- Go to the **System Parameters Features** screen by running the following command: **`change system-parameters feature`**. Navigate to page 11 or search for *Vectoring* and set one of the

following values for **First Data Delay** and **Second Data Delay** fields. The following table provides an example to show that another set of values can also work.

```
display system-parameters features                              Page  11 of  18
                     FEATURE-RELATED SYSTEM PARAMETERS
CALL CENTER SYSTEM PARAMETERS
  EAS
        Expert Agent Selection (EAS) Enabled? y
       Minimum Agent-LoginID Password Length:
         Direct Agent Announcement Extension:                    Delay:
   Message Waiting Lamp Indicates Status For: station

  VECTORING
                    Converse First Data Delay: 1      Second Data Delay: 0
              Converse Signaling Tone (msec): 100         Pause (msec): 70
                    Prompting Timeout (secs): 10
                Interflow-qpos EWT Threshold: 2
    Reverse Star/Pound Digit For Collect Step? n
          Available Agent Adjustments for BSR? n
                           BSR Tie Strategy: 1st-found
   Store VDN Name in Station's Local Call Log? n
  SERVICE OBSERVING
             Service Observing: Warning Tone? y     or Conference Tone? n
     Service Observing Allowed with Exclusion? y
             Allow Two Observers in Same Call? y
```

Figure 3 - Setting CM Converse Data Delay

Platform	First Data Delay	Second Data Delay
AAEP/AVP	1	0

Table 1 - Data Delay parameters

4. *UCID Settings*

Make sure that both the UCID and Send UCID to ASAI are enabled in CM > system-parameters features

- On page 5 of the system-parameters features form, set ***Create Universal Call ID (UCID)?*** to y and ***UCID Network Node ID*** to an unassigned node ID.

```
display system-parameters features                          Page   5 of  18
                        FEATURE-RELATED SYSTEM PARAMETERS

SYSTEM PRINTER PARAMETERS
  Endpoint:               Lines Per Page: 60

SYSTEM-WIDE PARAMETERS
                                    Switch Name:
           Emergency Extension Forwarding (min): 10
         Enable Inter-Gateway Alternate Routing? n
Enable Dial Plan Transparency in Survivable Mode? n
                             COR to Use for DPT: station

MALICIOUS CALL TRACE PARAMETERS
               Apply MCT Warning Tone? n   MCT Voice Recorder Trunk Group:
      Delay Sending RELease (seconds)? 0
SEND ALL CALLS OPTIONS
     Send All Calls Applies to: station    Auto Inspect on Send All Calls? n

UNIVERSAL CALL ID
     Create Universal Call ID (UCID)? y    UCID Network Node ID: 2
```

Figure 4 - Example of UCID Settings Page 5

- On page 13, set ***Send UCID to ASAI?*** to y.

```
display system-parameters features                          Page  13 of  18
                        FEATURE-RELATED SYSTEM PARAMETERS

 CALL CENTER MISCELLANEOUS
                        Clear Callr-info: next-call
       Allow Ringer-off with Auto-Answer? n

   Reporting for PC Non-Predictive Calls? n


         Interruptible Aux Notification Timer (sec): 3
       Interruptible Aux Deactivation Threshold (%): 95

  ASAI
           Copy ASAI UUI During Conference/Transfer? y
       Call Classification After Answer Supervision? y
                                 Send UCID to ASAI? y
```

Figure 5 - Example of UCID Settings Page 13

5. *Creating phantom extensions*

- Create as many phantom stations as at least the number of Experience Portal channels to be used for CBA. Create one phantom station for each IVR channel. To create a new station, use the command: ***add station [number]*** or ***add station next.*** For more information, see

Avaya Aura® Communication Manager Documentation.
Examples:

```
display station 51215                                        Page   1 of   5
                                    STATION

Extension: 51215                       Lock Messages? n              BCC: 0
     Type: 6408D+                      Security Code:                 TN: 1
     Port: X                         Coverage Path 1:                COR: 1
     Name: CBA_Phantom_1             Coverage Path 2:                COS: 1
                                     Hunt-to Station:
STATION OPTIONS
                                          Time of Day Lock Table:
             Loss Group: 2          Personalized Ringing Pattern: 1
           Data Module? n                     Message Lamp Ext: 51215
        Speakerphone: 2-way                Mute Button Enabled? y
     Display Language: english

         Survivable COR: internal               Media Complex Ext:
   Survivable Trunk Dest? y                       IP SoftPhone? n

                                                  IP Video? n
```

Figure 6 - Example of phantom station creation – Page 1

```
display station 51215                                        Page   2 of   5
                                    STATION
FEATURE OPTIONS
          LWC Reception: spe                Auto Select Any Idle Appearance? n
         LWC Activation? y                          Coverage Msg Retrieval? y
  LWC Log External Calls? n                               Auto Answer: none
            CDR Privacy? n                          Data Restriction? n
   Redirect Notification? y                   Idle Appearance Preference? n
 Per Button Ring Control? n                 Bridged Idle Line Preference? n
   Bridged Call Alerting? n                     Restrict Last Appearance? y
  Active Station Ringing: single

       H.320 Conversion? n          Per Station CPN - Send Calling Number?
      Service Link Mode: as-needed                    EC500 State: enabled
        Multimedia Mode: basic                 Audible Message Waiting? n
    MWI Served User Type:                    Display Client Redirection? n
            AUDIX Name:                      Select Last Used Appearance? n
                                               Coverage After Forwarding? s
                                                 Multimedia Early Answer? n
                                           Direct IP-IP Audio Connections? y
  Emergency Location Ext: 51215                      IP Audio Hairpinning? n
```

Figure 7 – Example of phantom station creation – Page 2

- Make a note of the numbers of the phantom stations created as you will require the number during CBA configuration.

6. *Creating a hunt group/skill to transfer calls to AAEP/VP Ports*

- Create an ACD skill for AAEP/AVP agent's login ID to send data to AAEP/AVP as required by the converse-on step feature. By using this feature, the Expected Wait Time (EWT) and Queue Position (QPOS) of the caller can be sent to the CBA application running on AAEP/AVP.

  To create a new hunt group (skill), use the command ***add hunt-group [number]*** or ***add hunt-group next***.
  For more information, see *Avaya Aura® Communication* Manager Documentation.
  Examples:

```
display hunt-group 62                                          Page   1 of   3
                                  HUNT GROUP

           Group Number: 62                                 ACD? y
             Group Name: CBA_VP_Port_SKill                Queue? n
        Group Extension: 51101                           Vector? y
             Group Type: ead-mia
                     TN: 1
                    COR: 1                        MM Early Answer? n
               Security Code:                  Local Agent Preference? n
 ISDN/SIP Caller Display:
```

Figure 8 - ACD Skill creation

```
display hunt-group 62                                          Page   2 of   3
                                  HUNT GROUP

                    Skill? y      Expected Call Handling Time (sec): 180
                      AAS? y        Service Level Target (% in sec): 80 in 20
                 Measured: internal
     Supervisor Extension:


      Controlling Adjunct: none


        VuStats Objective:

   Multiple Call Handling: none


Interruptible Aux Threshold: none
                              Redirect on No Answer (rings):
                                          Redirect to VDN:
                  Forced Entry of Stroke Counts or Call Work Codes? n
```

Figure 9 - Page 2 for ACD Skill creation Screen

7. *Disabling CBA skills to support queue to callers or agents*

- When you configure AAEP/VP skill set the **Queue** to ***n*** (No), as shown on Figure 8.

8. *Creating an Agent-login ID for each AAEP/VP Port*

- Each AAEP/VP port, which is a Communication Manager station, must have an *Agent-loginID* associated to the AAEP/VP Skill created in the previous step. This is required for *converse-on* function. To add a new agent, use the command ***add agent-loginID [number]*** or ***add agent-loginID next.***
  Example:

```
display agent-loginID 51310                                    Page   1 of   2
                                  AGENT LOGINID

               Login ID: 51310                                    AAS? y
                   Name: CBA_VP_Agent_1                         AUDIX? n
                     TN: 1                                LWC Reception: spe
                    COR: 1                        LWC Log External Calls? n
          Coverage Path:                       AUDIX Name for Messaging:
          Security Code:
        Port Extension: 51210          LoginID for ISDN/SIP Display? n


                                                      Auto Answer: none
                                                  MIA Across Skills: system
                                           ACW Agent Considered Idle: system
                                           Aux Work Reason Code Type: system
                                             Logout Reason Code Type: system
                                 Maximum time agent in ACW before logout (sec): system
                                                  Forced Agent Logout Time:   :

    WARNING:  Agent must log in again before changes take effect
```

Figure 10 - Agent Login ID creation

- Set the **AAS** field value to ***y*** to allow the association of the *Login ID* to a *Port Extension*. In the **Port Extension** field, associate the Agent Login ID to an IVR extension number. The **COR** value must allow outbound dialing to PSTN as the AAEP/VP Agent/Port will perform the outbound call to the customer.
- In page 2, associate the **Agent Login ID** to the **AAEP/VP Skill** previously created:

```
display agent-loginID 51310                                    Page   2 of   2
                                  AGENT LOGINID
      Direct Agent Skill:                                Service Objective? n
Call Handling Preference: skill-level                  Local Call Preference? n

    SN   RL SL                SN   RL SL
 1: 62      1          16:
 2:                    17:
 3:                    18:
 4:                    19:
 5:                    20:
 6:
 7:
 8:
 9:
10:
11:
12:
13:
14:
15:
```

Figure 11 - Page 2 for Agent Login creation

9. *Creating a call vector and a VDN to transfer calls to AAEP/VP*

- Create a call vector to send calls to AAEP/VP. To create or modify a call vector, use the command **`chance vector [number]`**.
  Perform the following tasks to complete this operation:

- Queue the call to the *AAEP/VP Skill*.

```
display vector 57                                          Page   1 of   6
                                CALL VECTOR

    Number: 57                Name: CBA_3_IVR
Multimedia? n       Attendant Vectoring? n    Meet-me Conf? n           Lock? n
     Basic? y    EAS? y    G3V4 Enhanced? y   ANI/II-Digits? y   ASAI Routing? y
 Prompting? y    LAI? y   G3V4 Adv Route? y   CINFO? y    BSR? y    Holidays? y
 Variables? y    3.0 Enhanced? y
01 queue-to      skill 62    pri m
02 stop
03
04
05
06
07
08
09
10
11
12

                        Press 'Esc f 6' for Vector Editing
```

Figure 12 - Call Vector creation

- Create a VDN number (called *Phantom Port to IVR Transfer VDN*) associated to previous vector. To create a new VDN, use the command ***add vdn [number]*** or ***add vdn next***. Example:

```
display vdn 51012                                          Page   1 of   3
                          VECTOR DIRECTORY NUMBER

                            Extension: 51012
                                 Name*: CBA CTI IVR
                          Destination: Vector Number         57
                Attendant Vectoring? n
              Meet-me Conferencing? n
                Allow VDN Override? n
                               COR: 1
                               TN*: 1
                          Measured: none


       VDN of Origin Annc. Extension*:
                           1st Skill*:
                           2nd Skill*:
                           3rd Skill*:



* Follows VDN Override Rules
```

Figure 13 - VDN Creation

10. Creating IVR station in Communication Manager 7.0.1

To create a new IVR station, use the command: ***add station [number]*** or ***add station next***.
For more information, see *Avaya Aura® Communication Manager* Documentation.
Example:

```
display station 64211                                        Page   1 of   6
                                    STATION

Extension: 64211                      Lock Messages? n                BCC: 0
     Type: 7434ND                     Security Code: 1234              TN: 1
     Port: S00267                   Coverage Path 1:                  COR: 1
     Name: CBAIVRST                 Coverage Path 2:                  COS: 1
                                    Hunt-to Station:
STATION OPTIONS
                                         Time of Day Lock Table:
             Loss Group: 2          Personalized Ringing Pattern: 1
           Data Module? n                    Message Lamp Ext: 64211
        Display Module? y
      Display Language: english              Coverage Module? n

         Survivable COR: internal             Media Complex Ext:
   Survivable Trunk Dest? y                        IP SoftPhone? y
                                              Remote Office Phone? n
                                              IP Video Softphone? n
                                Short/Prefixed Registration Allowed: default
```

Figure 14 - Example of IVR station creation - Page 1

```
display station 64211                                        Page   2 of   6
                                    STATION
FEATURE OPTIONS
          LWC Reception: spe              Auto Select Any Idle Appearance? n
         LWC Activation? y                      Coverage Msg Retrieval? y
  LWC Log External Calls? n                               Auto Answer: none
            CDR Privacy? n                          Data Restriction? n
   Redirect Notification? y                 Idle Appearance Preference? n
Per Button Ring Control? n                Bridged Idle Line Preference? n
  Bridged Call Alerting? n                     Restrict Last Appearance? y
 Active Station Ringing: single

       H.320 Conversion? n          Per Station CPN - Send Calling Number?
     Service Link Mode: as-needed                    EC500 State: enabled
      Multimedia Mode: enhanced                Audible Message Waiting? n
   MWI Served User Type:                     Display Client Redirection? n
           AUDIX Name:                      Select Last Used Appearance? n
                                             Coverage After Forwarding? s
                                              Multimedia Early Answer? n
Remote Softphone Emergency Calls: as-on-local Direct IP-IP Audio Connections? y
 Emergency Location Ext: 64211        Always Use? n IP Audio Hairpinning? n
```

Figure 15 - Example of IVR station creation - Page 2

```
display station 64211                                    Page   4 of   6
                                    STATION
 SITE DATA
      Room:                                          Headset? n
      Jack:                                          Speaker? n
     Cable:                                        Mounting: d
     Floor:                                    Cord Length: 0
  Building:                                      Set Color:



ABBREVIATED DIALING
     List1:                    List2:                    List3:

BUTTON ASSIGNMENTS
 1: call-appr                          6:
 2: call-appr                          7:
 3:                                    8:
 4:                                    9:
 5:                                   10: ucid-info
```

Figure 16 - Example of IVR station creation – Page 4

```
display station 64211                                    Page   6 of   6
                                    STATION

DISPLAY BUTTON ASSIGNMENTS

 1: normal
 2:
 3:
 4:
 5:
 6:
 7:
```

Figure 17 - Example of IVR station creation – Page 6

Understanding specific configurations for a callback

This section describes minimal requirements for a callback configuration.

1. *Creating Incoming Calls Vector and Initial VDN*

a) Create a call vector to handle the incoming calls and associate the call vector to a VDN for each callback configuration you want to add to CBA.

b) Create one vector that will queue the incoming calls to the agents' skill and then send the agent's skills to the CBA AAEP/VP Application, passing both queue position (*QPOS*) and estimated wait time (*EWT*) values using *converse-on function*. To create or modify a call vector, use the command **`chance vector [number]`**.

Perform the following tasks to complete this operation:

```
display vector 55                                          Page   1 of   6
                              CALL VECTOR

    Number: 55               Name: CBA_1_VP
Multimedia? n      Attendant Vectoring? n    Meet-me Conf? n           Lock? n
     Basic? y   EAS? y   G3V4 Enhanced? y   ANI/II-Digits? y   ASAI Routing? y
 Prompting? y   LAI? y  G3V4 Adv Route? y   CINFO? y   BSR? y   Holidays? y
 Variables? y   3.0 Enhanced? y
01 wait-time    1   secs hearing silence
02 queue-to     skill 60   pri m
03 converse-on  skill 62   pri m passing qpos   and wait
04 collect      1    digits after announcement none     for none
05 goto step    8             if digits           =      1
06 wait-time    20  secs hearing music
07 goto step    6             if unconditionally
08 disconnect   after announcement none
09 stop
10
11
12
```

Figure 18 - Call Vector configuration

- Use Step 01 as shown in the *Call Vector configuration* example to avoid timing synchronization issues.
- Use Step 02 as shown in the *Call Vector configuration* to send the calls to the agents' queue.
- If an agent does not immediately attend the call and if the system puts the call in queue, the processing goes to Step 03 and then is sent to the AAEP/VP though the system executes the *converse-on* command and the CBA AAEP/VP Application. The system sends the call to the AAEP/VP skill (in this example, skill 62), passing the **Queue Position** first and then **EWT** to the AAEP/VP application. The system then halts the vector processing until the AAEP/VP application is available.
- The CBA *AAEP/VP Application* sends one DTMF digit to the vector, which is collected in Step 04. The IVR application can return either ***zero***, indicating that the caller did not request a callback, or ***one***, indicating that the caller was able to request a callback.
- Step 5 compares the returned digit from *AAEP/VP Application* and, in case of success for the callback request, it send the processing to step 9 to disconnect the call. Otherwise, it will continue to next step;

- On step 6 and 7, it will keep the caller hearing music while waiting in queue, as the callback request wasn't successfully requested;
- The vector above is presented as a template, and you can customize accordingly your needs. Below is another example of a vector programming, which will offer again the possibility for a callback if one attempt fails after 10 seconds:

```
display vector 55                                              Page   1 of   6
                                CALL VECTOR

    Number: 55              Name: CBA_1_VP
Multimedia? n      Attendant Vectoring? n    Meet-me Conf? n           Lock? n
     Basic? y   EAS? y   G3V4 Enhanced? y   ANI/II-Digits? y   ASAI Routing? y
 Prompting? y   LAI? y  G3V4 Adv Route? y   CINFO? y   BSR? y   Holidays? y
 Variables? y   3.0 Enhanced? y
01 wait-time    1   secs hearing silence
02 queue-to     skill 60   pri m
03 converse-on  skill 62   pri m passing qpos   and wait
04 collect      1    digits after announcement none     for none
05 goto step    8              if digits           =      1
06 wait-time    20  secs hearing music
07 goto step    3              if unconditionally
08 disconnect   after announcement none
09 stop
10
11
12
```

Figure 19 - Alternate Call Vector configuration

- (Optional) Check the status of the AAEP/ AVP agent skill first to ensure that ports are available. If not, the call should not be sent to AAEP/VP with the converse-on step.

c) Create one VDN associated with previous vector and take note of the VDN extension number created. To create a new VDN, use the command ***add vdn [number]*** or ***add vdn next***. Example:

```
display vdn 51010                                              Page   1 of   3
                        VECTOR DIRECTORY NUMBER

                          Extension: 51010
                               Name*: CBA_CTI_Converse
                        Destination: Vector Number         55
                Attendant Vectoring? n
               Meet-me Conferencing? n
                 Allow VDN Override? y
                                COR: 1
                                TN*: 1
                           Measured: both
       Acceptable Service Level (sec): 20

       VDN of Origin Annc. Extension*:
                           1st Skill*:
                           2nd Skill*:
                           3rd Skill*:


* Follows VDN Override Rules
```

Figure 20 - Associated VDN creation

2. *Creating the Outgoing Calls Vector and Outgoing VDN*

- For each Callback configuration you want to add to Avaya Callback Assist, you must create a call vector to handle the outgoing calls to the agent's queue and associate it to a VDN;
- Create one vector that will simply queue the outgoing callbacks ("*phantom calls*") that the **Avaya Callback Assist** Engine will create to agents' skill. To create or modify a call vector, use the command ***chance vector [number]***. The steps that must be created in order to perform the operation are listed below:

```
display vector 56                                          Page   1 of   6
                                CALL VECTOR

    Number: 56                Name: CBA_2_QUEUE
Multimedia? n      Attendant Vectoring? n    Meet-me Conf? n           Lock? n
     Basic? y   EAS? y   G3V4 Enhanced? y   ANI/II-Digits? y   ASAI Routing? y
 Prompting? y   LAI? y  G3V4 Adv Route? y   CINFO? y   BSR? y   Holidays? y
 Variables? y   3.0 Enhanced? y
01 queue-to     skill 60   pri m
02 stop
03
04
05
06
07
08
09
10
11
12
```

Figure 21 - Call Vector for agent's queue

- This call vector simply queues the phantom call to the original agents' skill;
- ***Important:*** On the chosen skill, check that the ***queue*** option is set to "***yes***" to allow queuing phantom calls. Use command ***change hunt [skill number]*** to verify the value***.***
- Create one VDN associated with previous vector and take note of the VDN extension number created. To create a new VDN, use the command ***add vdn [number]*** or ***add vdn next***. Example:

```
display vdn 51011                                          Page   1 of   3
                        VECTOR DIRECTORY NUMBER

                           Extension: 51011
                                Name*: CBA_CTI_QUEUE
                          Destination: Vector Number         56
                 Attendant Vectoring? n
               Meet-me Conferencing? n
                 Allow VDN Override? n
                                COR: 1
                                 TN*: 1
                           Measured: both
       Acceptable Service Level (sec): 20

       VDN of Origin Annc. Extension*:
                           1st Skill*:
                           2nd Skill*:
                           3rd Skill*:


* Follows VDN Override Rules
```

Figure 22 - VDN associated with agent's vector

Setting up VDN Callback for ISDN PRI or IP trunks

When calls arrive on an **ISDN PRI or IP trunk**, they have the DNIS sent from the network presented to **Avaya Callback Assist**. Refer to the section **DNIS or Called Numbers** on ***Avaya Callback Assist*** **Overview and Planning Guide**.

To prevent this behavior, you must confirm that:

1) VDN Override is set on all VDNs that were touched by the call prior to delivery to **Avaya Callback Assist**;

2) The feature "*VDN Override for ISDN Trunk ASAI Messages*" is set to "***y***" for the VDN which routes to the vector that converses to Callback Assist. Below there is an example screenshot from the CM administration. To access that screen, use the command ***change VDN [number]*** and go to page 2.

```
display vdn 51010                                              Page   2 of   3
                          VECTOR DIRECTORY NUMBER

                                AUDIX Name:

                          Return Destination*:

                             BSR Application*:
                 BSR Available Agent Strategy*: 1st-found
                           BSR Tie Strategy*: system

                      Observe on Agent Answer? n


                  Display VDN for Route-To DAC*? n
VDN Override for ISDN Trunk ASAI Messages*? y

                          BSR Local Treatment*? n

              Reporting for PC Predictive Calls? n
               Pass Prefixed CPN to VDN/Vector*? system
* Follows VDN Override Rules
```

Figure 23 - VDN setting for ISDN PRI or IP trunk

Configuring AES

These steps apply to configuring **Avaya Application Enablement Services (AES)** for Callback Assist, it does not provide AES installation steps. For additional help or support on **Avaya Application Enablement Services**, see AES documentation.

Pre-requisites

It is required that Avaya Application Enablement Services (AES) is already installed and configured with Communication Manager (CM).

Global configuration

In order to use AES TSAPI services, it is needed to create a user for Callback Assist on AES with proper permissions.

1. Login to AES OAM Administration.
2. Create a user with CT rights under "***User Management -> Add User***". For example:

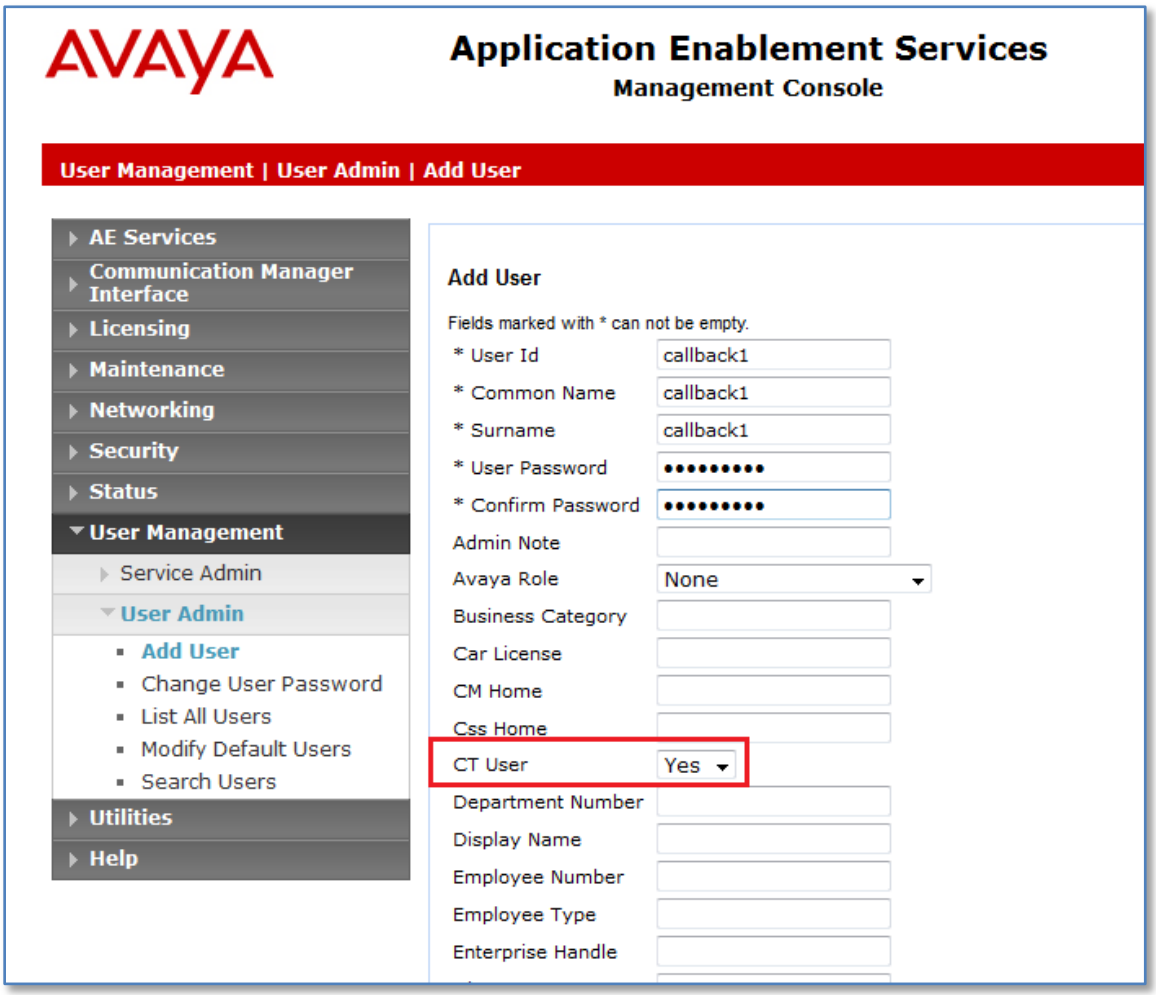


Figure 24 - AES user creation

3. Click Save.

4. Under menu "***Security -> Security Database -> CTI Users***", select the newly created user and click on "Unrestricted Access" rights.

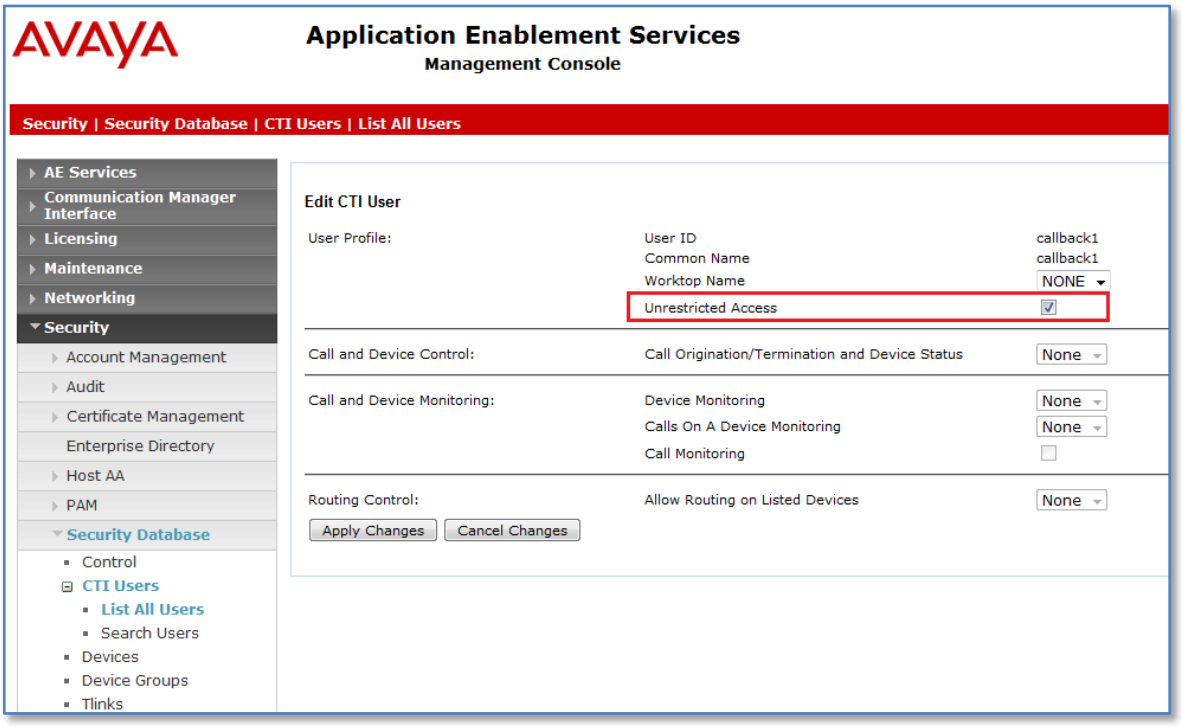


Figure 25 -AES Edit CTI User- Enabled Unrestricted Access

5. Make a note of the user and password created. These will be required to complete the installation steps.

Installing Callback Assist Software

The Avaya Callback Assist Installation is based on Linux shell scripts. The installer of Avaya Callback Assist is a single tar file whose name has the form callbackassist-<version>.tar (for example: callbackassist-4.3.1.0-GA.tar, where 4.3.1.0-GA is the version label). All Avaya Callback Assist components run as Linux daemons registered under /etc/init.d.

The callback-install.sh script installs all the available components in the server where the script is run (including the PostgreSQL DBMS, if single server option was chosen), and prepares most of the configuration files automatically.

The installation script prompts you to specify the Deployment Platform in the call center (for example, CTI, SIP or AACC) to enable the corresponding functionality.

Software Installation Steps

1. On a Linux shell logged on as root, go to the directory where the CBA installer file is located.
2. Run the following command to extract the files to a temporary folder in your system:
   `tar -xvf callbackassist_<version>.tar`
   One of the extracted file is callback-install.sh which is the main installation script.
3. Run the following command to start the installation process:
   `./callback-install.sh`
   The system writes the command output of the scripts to the standard output and to the *callback-install.log* file in your system.
4. Select [**3**) ***Callback Assist Single Server Deployment (Core Components & DB)*** as the installation mode.
5. Enter the installation directory where to install Callback Assist or just press ***Enter*** to use default location (default directory "/opt").
6. Select [**1) *CTI (H.323)*** as the platform type and press Enter.
7. Choose the authentication type
   1. Internal
   2. External (Open ID ex: Google)

   If the authentication type is selected as External then follow the CBA Installation as External Authentication section for more details.
8. Choose whether you need Local Web LM server or not. If you decide to use the Local Web LM server as license server then select "yes", otherwise "no".

   Note: If you select "no" at the time of installation and later if you decide to use Local Web LM server as a license server; then the WebLM (tomcat-weblm) service can be restored by running the script reinstallservices.sh from <CBA_INSTALLATION_LOCATION>/support folder.

The system starts installing Callback Assist in a CTI environment. After completing the installation, the system displays an installation successful message.

Step by step output of the *callback-install.sh* script:

```
[root@server206 4.3.1.0]# ./callback-install.sh


************************************************************
Avaya Callback Assist Installer
************************************************************


(06:23:08) [ No Callback Assist Components were found. ]
```

```
(06:23:08) Please choose the Components to be Installed on this server:


1) Callback Assist Core Components with external Database

2) PostgreSQL Database Server for Callback Assist

3) Callback Assist Single Server deployment (Core Components & DB)

#? 3


(06:23:16) [ About to Install Callback Assist Single Server deployment
(Core Components & DB) ]

Please, enter the base directory where Avaya Callback Assist will be
installed (default directory /opt):

Directory where Avaya Callback Assist is going to be installed:
/opt/Avaya/callbackassist

(06:23:18) Checking if there is enough disk space...

(06:23:18) Available disk space is enough.

(06:23:18) Please choose a Platform type for Callback Assist
Application to be Installed from the options below:

1) CTI (H.323)

2) SIP

3) AACC/CM (SIP Based)

#? 1


(06:23:21) Please choose an Authentication Type:

1) Internal

2) External (OpenId ex: Google)

#? 1


(06:23:24) Are you going to use local WebLM service? (yes/no) :  yes


(06:23:29) [ Local WebLM server will be added as a daemon service. ]


(06:23:29) [ Internal has been selected as an Authentication Type. ]


(06:23:29) [ CTI (H.323) Platform has been selected for installation. ]
```

```
(06:23:29) Unpackaging distribution file callbackassist.package...


(06:23:51) Creating callback Group...
(06:23:51) Creating callback User...
(06:23:51) 32 bit Architecture detected ...
(06:23:51) Using 32 bit PostgreSQL Installer...
(06:23:51) Installing PostgreSQL Server, this step may take several
minutes...
(06:24:20) Creating 'callback' PostgreSQL user...
(06:24:20) Creating 'callback' database...
(06:24:20) 'callback' database created.
server signaled
(06:24:20) Restarting database to reset max_connections
Restarting PostgreSQL 9.4:

waiting for server to shut down.... done
server stopped
waiting for server to start.... done
server started
PostgreSQL 9.4 restarted successfully
(06:24:22) Signaling Postgresql postmaster...
(06:24:22) Done.
(06:24:22) Setting ownership to callback user.
(06:24:22) Performing JDK silent install...
(06:24:26) JDK installed.
java version "1.7.0_75"
Java(TM) SE Runtime Environment (build 1.7.0_75-b13)
Java HotSpot(TM) Server VM (build 24.75-b04, mixed mode)
(06:24:26) Installation of Callback Assist Maintenance done.
(06:24:26) Check
/opt/Avaya/callbackassist/maintenance/logs/execution.log file for
Callback Assist Maintenance service startup details.
(06:24:26) Installation of Callback Assist Engine done.
```

```
(06:24:26) Check /opt/Avaya/callbackassist/engine/logs/execution.log
file for Callback Assist Engine service startup details.

(06:24:26) 32 bit Architecture detected ...

(06:24:26) Installing file server for Red Hat 6.x 32 bits

(06:24:26) Local Ip Address: 135.122.99.206

(06:24:26) INFO: Using hostname/FQDN: server206.avayacba.com to run the
file-server.

(06:24:26) Configuring File Server IP Address...

(06:24:26) Done.

(06:24:26) Changing SELinux context to some CBA files...

(06:24:27) Waiting for the Database Schema to be created or updated...

Migration successful

(06:24:32) Database Schema was successfully created or updated.

(06:24:33) Platform successfully set.

(06:24:33) Storing Release version and Build Number...

(06:24:33) Release version and Build number successfully stored.

(06:24:33) Running Platform dependant changes...

Migration successful

(06:24:34) Database Schema was successfully updated.

(06:24:34) Updating authentication type into database ...

(06:24:34) Installing weblm Tomcat service (tomcat-weblm)...

(06:24:34) Installation of tomcat-weblm done.

(06:24:34) Installing adminapp Tomcat service (tomcat-adminapp)...

(06:24:34) Installation of tomcat-adminapp done.

(06:24:34) Installing ddapps Tomcat service (tomcat-ddapps)...

(06:24:34) Installation of tomcat-ddapps done.

(06:24:34) Installing webcallback Tomcat service (tomcat-
webcallback)...

(06:24:34) Installation of tomcat-webcallback done.

(06:24:34) Deploying Applications...

(06:24:34) Moving tomcat realm related jar files

(06:24:34) Installation of Tomcat instances done.

(06:24:34) Setting ownership to callback user.

Starting Callback Assist Engine...
```

```
Callback Assist Engine Started.                                [  OK  ]
Callback Assist Engine ( pid 11070 ) is running...
Starting Callback Assist Maintenance...
Callback Assist Maintenance Started.                           [  OK  ]
Callback Assist Maintenance ( pid 11145 ) is running...
Starting tomcat-weblm...                                       [  OK  ]
tomcat-weblm ( pid 11233 ) is running...
Starting tomcat-adminapp...                                    [  OK  ]
tomcat-adminapp ( pid 11329 ) is running...
Starting tomcat-webcallback...                                 [  OK  ]
tomcat-webcallback ( pid 11451 ) is running...
Starting tomcat-ddapps...                                      [  OK  ]
tomcat-ddapps ( pid 11550 ) is running...
Starting Callback Assist File Server...
Callback Assist File Server Started.                           [  OK  ]
Callback Assist File Server ( 11715 ) is running...            [  OK  ]
Callback Assist File Server Ping test                          [  OK  ]
Callback Assist File Server Read/Write cycle Test              [  OK  ]
Callback Assist File Server Ring Status is Up                  [  OK  ]
Callback Assist File Server is joined to a cluster             [  NO  ]
If this is not a HA deployment, then disregard this warning.

(06:24:49) [ Installation of Avaya Callback Assist (CTI (H.323))
completed. ]



**********************************************************
ACTION REQUIRED on Platform Configuration
**********************************************************
You have to MANUALLY Configure the following files:


        1) tsapi.pro: Here you will have to configure the AES IP
Address and Port Number.
```

```
        2) callbackEngineConfiguration.properties: In this file you
will have to configure

        the Phantom Station Extensions, and the Voice Portal Ports.


These files are both located in /opt/Avaya/callbackassist/engine/lib.
**********************************************************


**********************************************************
ACTION REQUIRED on Time Zone Configuration
**********************************************************
The default Time Zone of CBA is UTC. If your system requires a
different Time Zone
you must manually configure it in the Global Settings of the Admin
Portal.
**********************************************************
```

After the execution of the installer scripts for a CTI Environment, you must manually configure the Experience Portal ports and Phantom Stations used by Avaya Callback Assist in the ***callbackEngineConfiguration.properties*** file, and the AES server location parameters in the ***tsapi.pro*** file. See Software Post installation steps for detailed explanation.

Software Post installation steps

This section describes the minimal configuration steps to be performed on Callback Assist application to get started.

Callback Engine Configuration files

It is required to configure the AES TSAPI interface and CM artifacts to be monitored and/or controlled by Callback Assist Engine component. Edit the file ***callbackEngineConfiguration.properties*** located under <CBA_HOME>/engine/lib directory and enter the CM phantom stations and the AAEP/VP ports that are going to be used by Callback Assist. See the following example:

```
# Voice Portal ports extension mapping
#ivr.channel.31000 = 31000
#ivr.channel.31001 = 31001

# phantom station extensions
#phantom.extension.1 = 2001
#phantom.extension.2 = 2002

ivr.channel.51210  =  51210
ivr.channel.51211  =  51211
ivr.channel.51212  =  51212
ivr.channel.51213  =  51213
ivr.channel.51214  =  51214

phantom.extension.1 = 51215
phantom.extension.2 = 51216
phantom.extension.3 = 51217
phantom.extension.4 = 51218
phantom.extension.5 = 51219
```

Figure 26 - Callback Engine Configuration Properties – After Editing

Where “ivr.channel.XXXXX = XXXXX” format is used for the AAEP/VP ports and “phantom.extension.X = XXXXX” format is used for CM phantom stations.

Important: Make a note of the minimum number of phantom stations extensions to be configured. The minimum required number is 4.

The file ***tsapi.pro*** located under <CBA_HOME>/engine/lib is used to specify the AES TSAPI interface. This file will

The following image shows an example of the tsapi.pro file being edited using Linux vim editor:

```
# tsapi.pro
#
# This file must be located in one of the directories found in CLASSPATH
#
# This is a list of the servers offering Telephony Services via TCP/IP.
# Either domain name or IP address may be used; default port number is 450
# The form is: host_name=port_number   For example:
#
# tserver.mydomain.com=450
# 127.0.0.1=450
#debugLevel=6
#altTraceFile=/opt/Avaya/callbackassist/engine/logs/tsapi_trace.txt
#
# (Remove the '#' when creating actual server entries.)
135.122.6.155=450
```

Figure 27 - An Example of a File being edited in Linux Server

In the above example 135.122.6.155 is the AES Server IP address and 450 is the TCP port of TSAPI Service on AES. Please assure that this port number is properly configured to be accessible in firewall or network security access software.

For the above changes to take effect, then you must restart the Callback Assist Engine service by using the Operating System service command, as follows:

```
[root@denpsqacba64 4.1.4]# /sbin/service cbaengine stop
Stopping Callback Assist Engine...
Callback Assist Engine Stopped.                            [  OK  ]
[root@denpsqacba64 4.1.4]# /sbin/service cbaengine start
Starting Callback Assist Engine...
Callback Assist Engine Started.                            [  OK  ]
[root@denpsqacba64 4.1.4]#
```

License Installation

For more information, see the License Installation section in this document.

Login to Administration Interface

The Administration interface is accessible from the URL **http://<server-hostname>/admin** (uses port tcp/80).

Default log in credential is user ***admin*** and password ***123456*** as shown below:

Figure 28 - Avaya Callback Administration Login Screen

After login into the application for the first time the user will be prompted to change the admin user password, as follows:

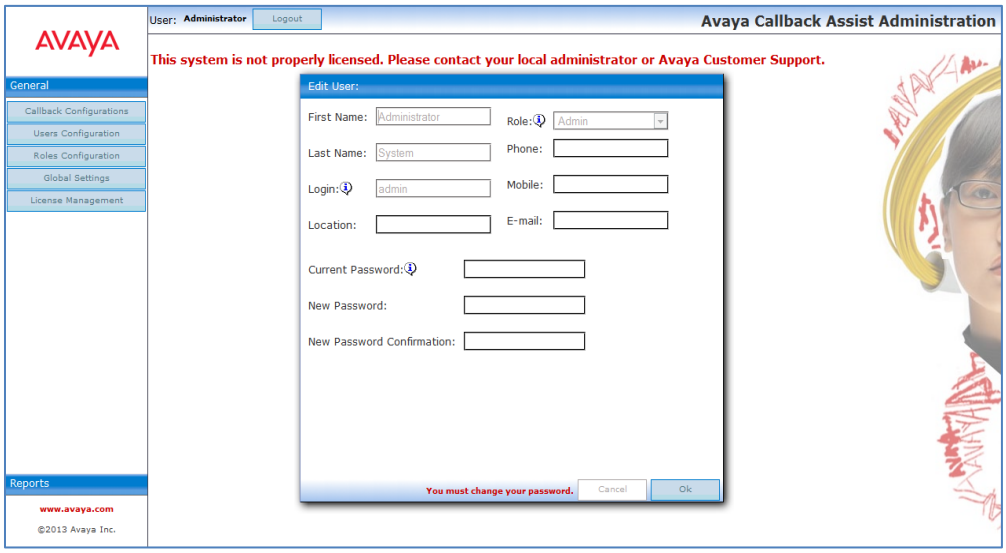


Figure 29 - Avaya Callback Assist – Change Password Screen

Enter new password and retype it on the confirmation field in order to complete the password change procedure.

Configuring license

Configure the WebLM Server URL where the Callback Assist license file is installed:

1. Go to **License Management** on the left navigation pane.
2. Enter the WebLM License Server URL replacing [Host] and [Port] with WebLM Server hostname or IP address and port number.
3. Enter the Dialog Designer WebLM License Server URL in the following format: https://\[Host\]:\[Port\]/WebLM/LicenseServer. The purpose of the Dialog Designer URL here is to display the Dialog Designer license status only and it will not affect CBA operations if not updated
4. Specify the number of CBA ports to be used based on your license. On a CTI installation, the number of CBA ports, as minimum, must match the number of phantom ports configured on Engine component.
5. Click the **Update** button to update the **License Management** page with WebLM URL and port usage information.

Note: If you have installed Callback Assist Application freshly then **Administration** → **Licensing Management** will be automatically pointed to Integrated WebLM URL. If you want to use external WebLM, modify the URL in **License Management** menu. If the Callback Assist Application is upgraded to 4.1.6 then the License URL will still point to the existing URL.

Example 1: The **License Management** page looks as follows when the currently installed Callback Assist version does not have a valid license file.

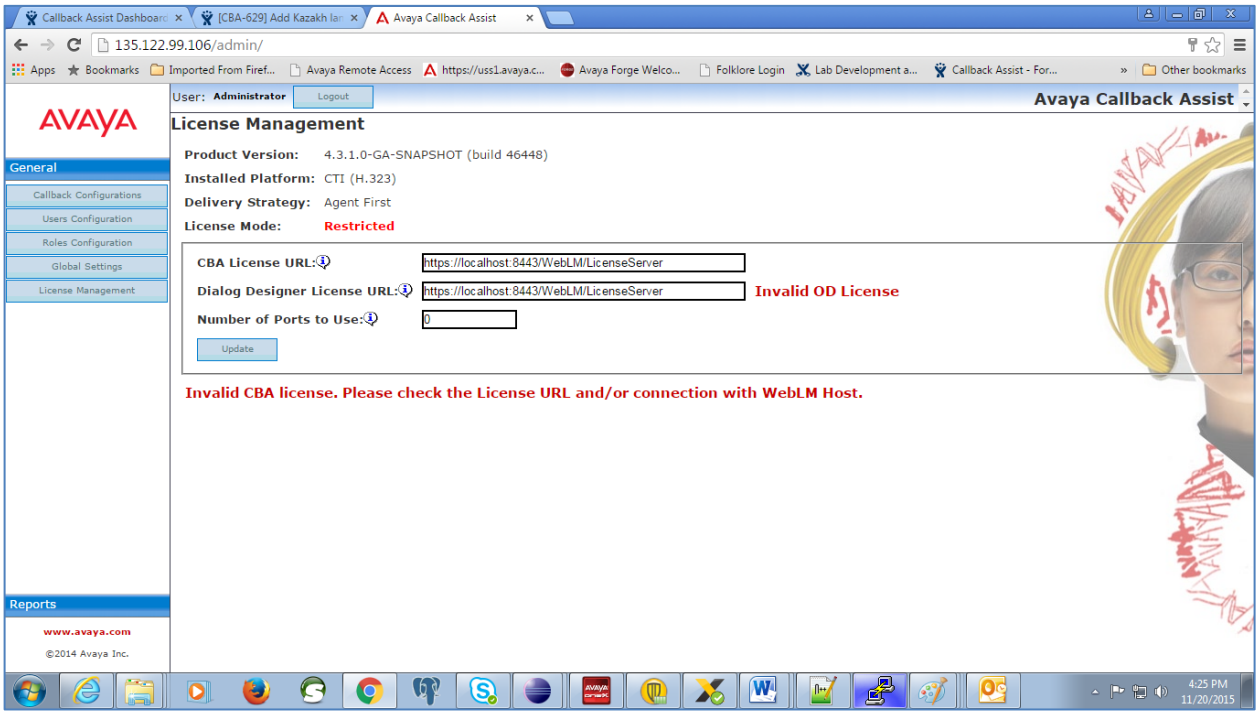


Figure 30 - License Management Screen - Invalid License

Example 2: The **License Management** page looks as follows when the currently installed Callback Assist version has a valid license file.

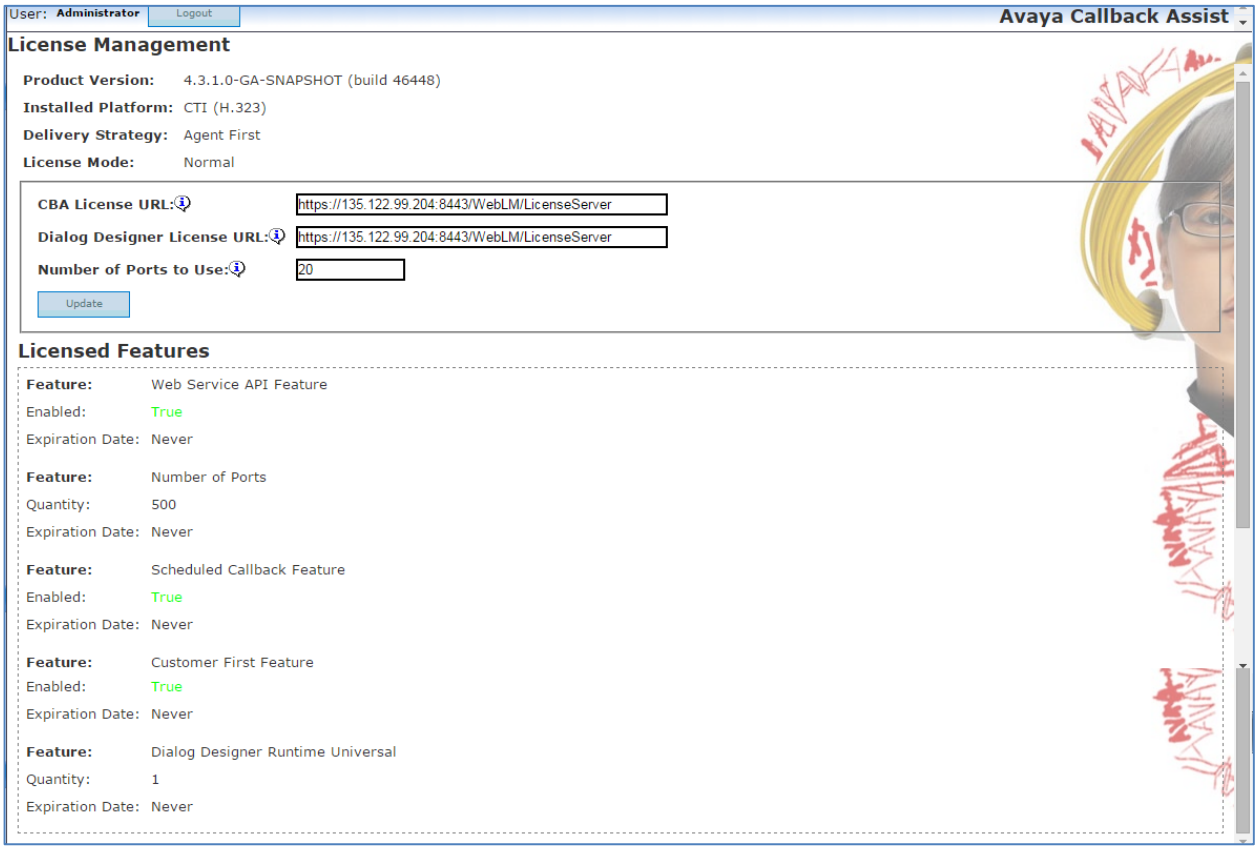


Figure 31 - License Management Screen - Valid License

Configuring Global Settings

- ### *Configuring the AES Tab*

Go to **Global Settings > AES** tab.
The system displays the **AES** page with the current AES configurations as follows:

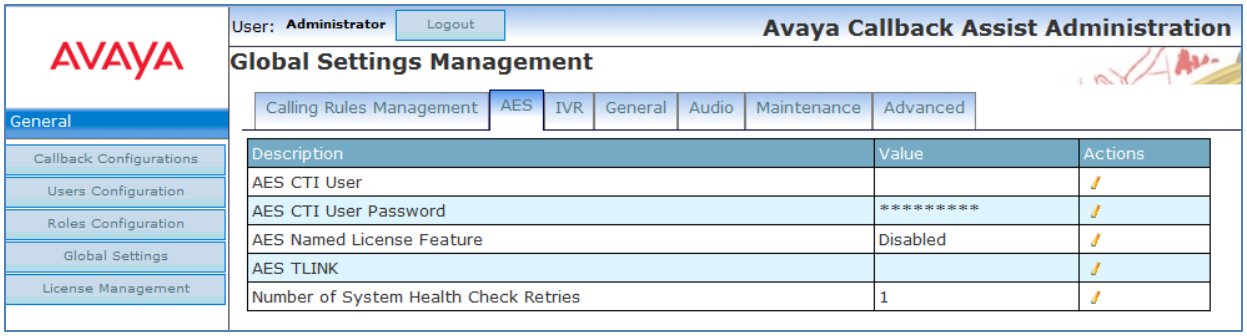


Figure 32 - AES tab configuration screen

- ***AES CTI User**:* User created previously on AES OAM.
- ***AES CTI User Password**:* password set previously on AES OAM.
- ***AES Named License Feature***: Allows usage of AES Application Specific Licensing.
- ***AES TLINK***: TLINK configured on AES. TLINK must be a secured one if AES Named License Feature is enabled.

Use default values for the rest of the parameters. For more information on this operation, see the *Administering Avaya Callback Assist* guide.

- ***Configuring the IVR Tab***

Go to **Global Settings > IVR** tab.
The system displays the **IVR** page with the default IVR parameters as follows:

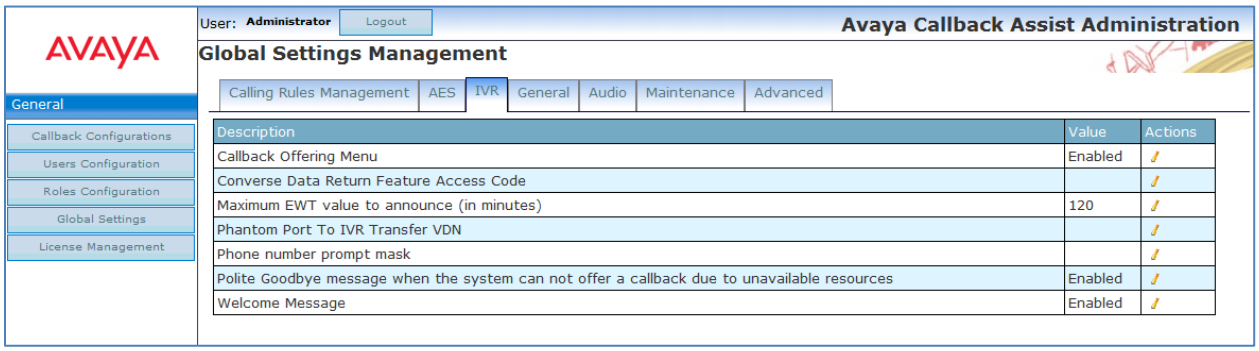


Figure 33 - IVR tab configuration screen

- ***Converse Data Return Feature Access Code***: Enter the return access code configured on Communication Manager Feature Access code as shown on section *Setting Converse-on Feature Data Return Code*.
- ***Phantom Port to IVR Transfer VDN:*** Enter VDN number created on section *Creating a call vector and a VDN to transfer calls to AAEP/VP*.

Use default values for the rest of the parameters. For more information on this operation, see the *Administering Avaya Callback Assist* guide.

- ***Configuring the Audio Tab***

Go to **Global Settings > Audio** tab.
The system displays the **Audio** page with the audio parameters configuration options as follows:

Figure 34 - Audio tab configuration screen

- ***Storage URL***: Change [Host] mask with Callback Assist Server IP Address or hostname.
  For more information on this operation, see the *Administering Avaya Callback Assist* guide.

Creating a new Callback Configuration

The following steps shows minimum required fields to create a Voice Configuration. Additional fields may be required, if that's the case see *Administering Callback Assist Guide* for further details.

Go to Callback Configurations and follow these steps:

- Click on ***Add New***.
- Select ***Voice*** configuration type and click on ***Next…***. Other types depend on the license features available.
- The ***Add Voice Callback Configuration*** window is displayed.
- Click on ***General Tab*** and, as minimum, configure required fields shown below:
  - ***i. Name:*** a simple name for this configuration
  - *ii.* ***DNIS:*** Initial VDN created on section *Creating Incoming Calls Vector and Initial VDN*
  - iii. ***Agent Queue:*** Outgoing VDN created on section *Creating the Outgoing Calls Vector and Outgoing VDN.*
  - iv. ***Minimum EWT Threshold***
  - v. ***Maximum EWT Threshold***
- Click on Availability Tab and configure required fields:
  - i. ***Time Cushion:*** This is a time space between the call and the first schedule slot to be offered
  - ii. ***Number of Offered Slots:*** Number of callback timeslots to be offered to the callers.

iii. ***Time Zone Message:*** specify the time zone message to be played while offering schedule callback slots.
iv. Select current day of week and configure, as minimum, current time of day as an available slot.

- Click on ***Customer Tab*** and, as minimum, configure required fields shown below:
  i. ***Welcome Message:*** specify the first message to be played to the caller.
  ii. ***Goodbye Message:*** specify the message when caller leaves gracefully.

  Note: *There are sample messages located on Callback Assist installation directory under /opt/Avaya/callbackassist/apache-tomcat-ddapps/webapps/CBAPhrases/samples/*

- Click on ***Agent Tab***
- Click on ***Voice Questions Tab*** and configure the following Phone Number question fields:
  i. ***Question Message:*** message to request phone number to caller.
  ii. ***Agent Message:*** message to playback phone number to Agent.
  iii. ***No Input Message:*** message to play when caller does not enter any value.
  iv. ***No Match Message:*** message to play when caller enters invalid value.
  v. Click ***OK*** on Question window.

  Note: There are sample messages located on Callback Assist installation directory under /opt/Avaya/callbackassist/apache-tomcat-ddapps/webapps/CBAPhrases/samples/

- Click OK button on ***Add Voice Callback Configuration*** window.

Configuring Avaya Aura® Experience Portal

These steps apply to *Avaya Aura® Experience Portal*. For additional help or support, see the latest *Avaya Aura® Experience Portal* documentation.

Setting up Callback Customer Voice Application

1. Enter EPM web page.
2. Browse menu to "Applications" option.
3. Add a new application for Customer interaction with Avaya Callback Assist offering prompts.
    a. Add URL: **http://<callback-application-server-hostname>:8080/CBACTICustomer/Start**
    b. Enable **Converse-on feature** (set it to "**Yes**"). All other parameters leave the default configuration.
    c. Add all **Initial VDN numbers** created previously as inbound numbers. For Example:

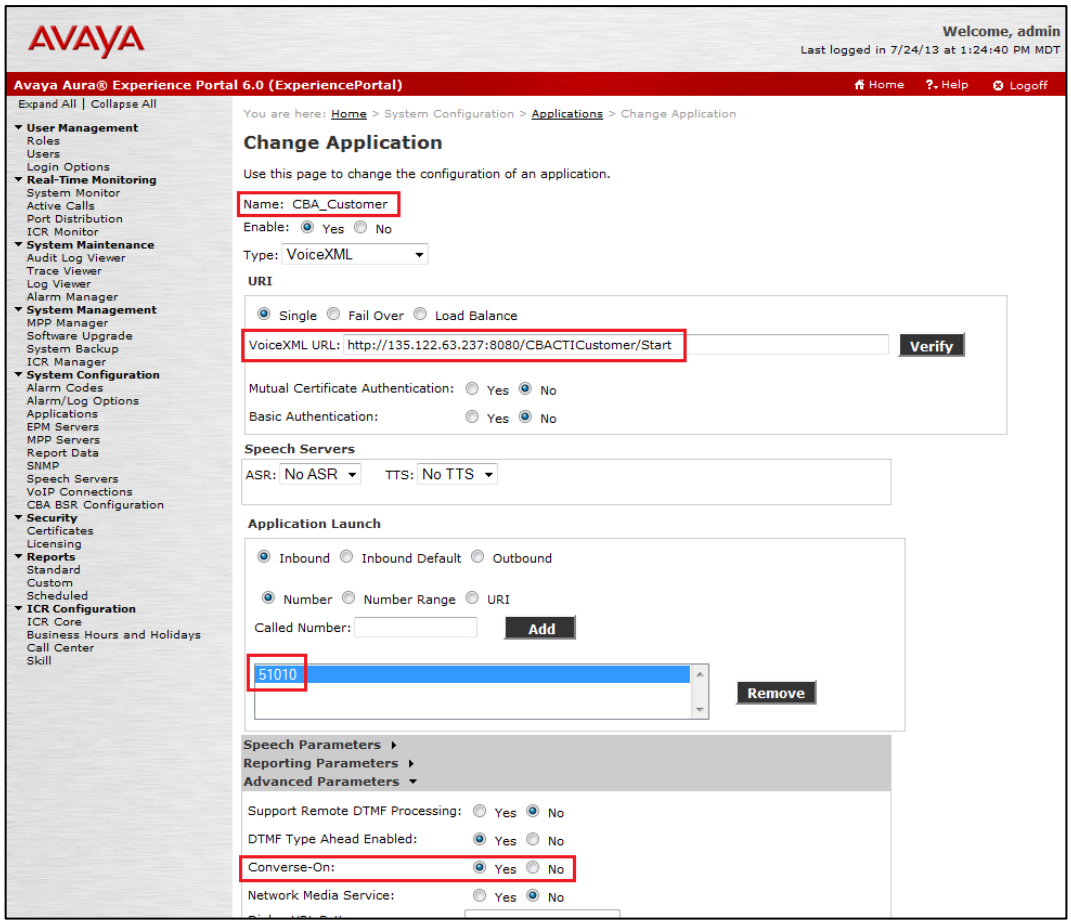


Figure 35 - Experience Portal Web page configuration and detail of Converse-On setting

Setting up Callback Agent Voice Application

1. Enter EPM web page.
2. Browse menu to "Applications" option.
3. Add another application for Agent interaction during processing of callbacks.
4. Add CBACTIAgent URL: **http://<callback-application-server-hostname>:8080/CBACTIAgent/Start**
5. Assign the ***Phantom Port to IVR Transfer VDN*** number as inbound number.
6. All other parameters leave the default configuration (including the **Converse-On** switch, which must be set to "**No**"). See example below:

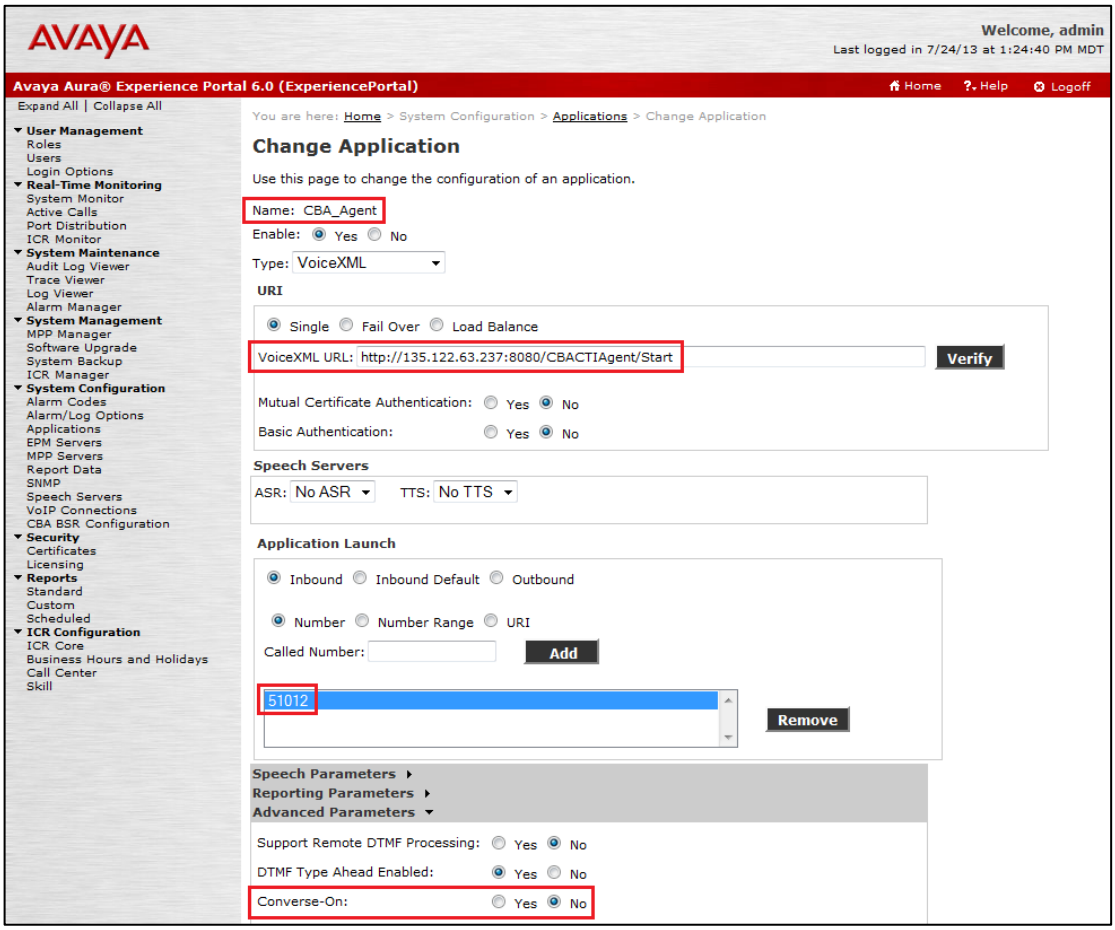


Figure 36 - Voice App Agent Setting

Configuring Avaya Orchestration Designer

Access the URL **http://<callback-application-server-hostname>:8080/runtimeconfig/** and logon with user/password = ddadmin/ddadmin, as shown below:

Figure 37 - Avaya Orchestration Designer Login screen

After logon, the system displays the following page:

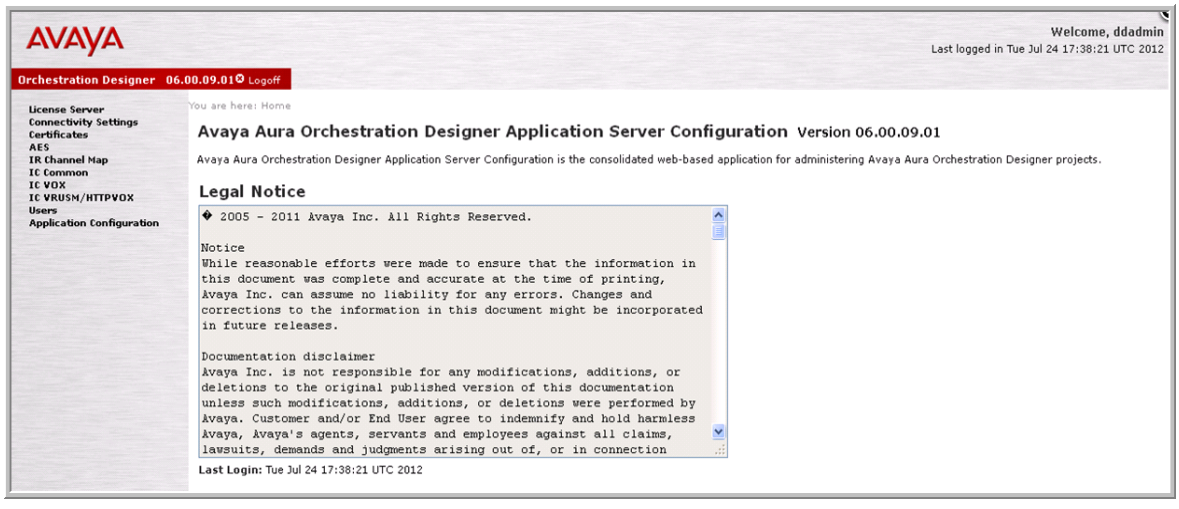


Figure 38 - Avaya Orchestration Designer page after Login

Click on the **License Server** link on the left navigation pane, and enter the URL for license server to configure the Licensing Server URL as follows:

Here you can use the Local WebLM Server (i.e., shipped with the CBA) or external WebLM Server.

Figure 39 - Configuration of Licensing Server

If the URL is correct the system displays a page similar to the following:

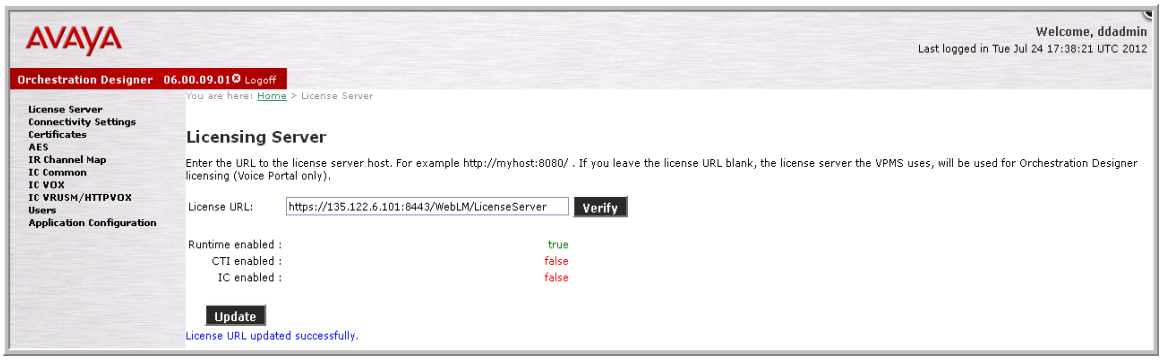


Figure 40 - Details of Configured Licensing Server

Installing Callback Assist in a SIP environment

This section guides you through the steps to configure different applications to install Callback Assist in a SIP environment, including the following:

- Configuration of Avaya Aura® Communication Manager
- Install and configure Avaya Callback Assist
- Configuration of Avaya Aura® Session Manager (ASM)
- Configuration of Avaya Aura® Experience Portal (AAEP)
- Configuration of Avaya Aura® Orchestration Designer (AAOD)
- Post Installation Steps

Checklist for setting up the system for Callback Assist

Prerequisites:

Before you set up the system to install Callback Assist in a SIP environment, make sure that you meet the following requirements:

- Session Manager Environment is installed and functional.
- Avaya Aura® Experience Portal Environment is installed and functional.
- Avaya Aura® Communication Manager Environment is installed and functional.
- Basic routing between AAEP/AVP and CM is working through ASM.
- Verify the Linux Shared Memory

Note:

- For configuring AAEP/AVP trunk to ASM, see the respective AAEP/AVP documentation available at Avaya Support site.
- For configuring Communication Manager routing to ASM, see the respective Communication Manager documentation available at Avaya Support site.

You can test the configuration by configuring a test application on AAEP/AVP and making a call from Communication Manager before installing Avaya Callback Assist.

Installing Callback Assist Software

The Avaya Callback Assist Installation is based on Linux shell scripts. The installer of Avaya Callback Assist is a single tar file whose name has the form callbackassist-<version>.tar (for example: callbackassist-4.3.1.0-GA.tar, where 4.3.1.0-GA is the version label). All Avaya Callback Assist components run as Linux daemons registered under /etc/init.d.

The callback-install.sh script installs all the available components in the server where the script is run (including the PostgreSQL DBMS, if single server option was chosen), and prepares most of the configuration files automatically.

You can install CBA on one of the three strategies for SIP platforms as follows:

- Agent First
- Customer First Phantom Pool
- Customer First Priority Queuing

You can also choose to use one of the two AAEP types for SIP platforms as follows:

- Site
- Zone

The installation steps are the same for all strategies. If you want to change the strategy, you must run the installation script again and select the desired strategy after specifying platform type.

If you want to change the AAEP type after installation, you must run the migration script provided in support directory and follow the instructions. For more information, refer Zoning Administration

For more information on Agent First and Customer First strategies, see *Avaya Callback Assist Overview and Planning Guide*.

Software Installation Steps

1. On a Linux shell logged on as root, go to the directory where the CBA installer file is located.
2. Run the following command to extract the files to a temporary folder in your system:
   `tar -xvf callbackassist-<version>.tar`
   One of the extracted file is callback-install.sh which is the main installation script.
3. Run the following command to start the installation process:
   `./callback-install.sh`
   The system writes the command output of the scripts to the standard output and to the callback-install.log file in your system. Then the system prompts you to choose the required platform for CBA installation.

4. Select [**3**) ***Callback Assist Single Server Deployment (Core Components & DB)*** as the installation mode.
5. Enter the installation directory where to install Callback Assist or just press ***Enter*** to use default location (default directory "/opt").
6. Select [**2)**] SIP as the platform type from the following options and press Enter.
    1. CTI (H.323)
    2. SIP
    3. AACC/CM (SIP Based)
7. Choose desired AAEP system type from the following options and press Enter. (Default is zone)

    Note: For site, no additional inputs are required. For zone, user will be asked to provide *Default Zone Name, Primary EPM Information (EPM Address, Username, Password)* as inputs.

    1. Site
    2. Zone
8. Choose desired Strategy from the following options and press Enter.
    1. Agent First
    2. Customer First Phantom Pool Strategy
    3. Customer First Priority Queuing Strategy

The system starts installing SIP platform with the desired strategy. After completing the installation, the system displays an installation successful message.

9. Choose the authentication type
    1. Internal
    2. External (Open ID ex: Google)

    If the authentication type is selected as External then follow the CBA Installation as External Authentication section for more details.
10. Choose whether you need Local Web LM server or not. If you decide to use the Local Web LM server as license server then select "yes", otherwise "no".

    Note: If you select "no" at the time of installation and later if you decide to use Local Web LM server as a license server; then the WebLM (tomcat-weblm) service can be restored by running the script reinstallservices.sh from <CBA_INSTALLATION_LOCATION>/support folder.

Step by step output of the *callback-install.sh* script:

```
[root@server103 4.4.0.0]# ./callback-install.sh

************************************************************
Avaya Callback Assist Installer
************************************************************



(09:22:35) [ No Callback Assist Components were found. ]



(09:22:35) Please choose the Components to be Installed on this server:

1) Callback Assist Core Components with external Database
2) PostgreSQL Database Server for Callback Assist
3) Callback Assist Single Server deployment (Core Components & DB)
4) Webcallback & DDApps components for Callback Assist
#? 3

(09:22:38) [ About to Install Callback Assist Single Server deployment
(Core Components & DB) ]



Please, enter the base directory where Avaya Callback Assist will be
installed (default directory /opt):

Directory where Avaya Callback Assist is going to be installed:
/opt/Avaya/callbackassist

(09:22:39) Checking if there is enough disk space...
(09:22:39) Available disk space is enough.

(09:22:39) Please choose a Platform type for Callback Assist Application
to be Installed from the options below:
1) CTI (H.323)
```

```
2) SIP
3) AACC/CM (SIP Based)
#? 2

(09:22:41) Please choose AAEP System type (Default: Zone):
1) Site
2) Zone
#? 2

(09:22:43) Please provide following information for Default Zone

Default Zone Name: default zone

Primary EPM IPAddress/HostName: 135.122.99.98

Primary EPM User Name: admin

Primary EPM Password: *******

(09:22:59) Please choose the Delivery Strategy:
1) Agent First Strategy
2) Customer First Phantom Pool Strategy
3) Customer First Priority Queuing Strategy
#? 1

(09:23:00) Please choose an Authentication Type:
1) Internal
2) External (OpenId ex: Google)
#? 1

(09:23:02) Are you going to use local WebLM service? (yes/no) : n

(09:23:03) [ Local WebLM server will not be added as a daemon service. ]
```

```
(09:23:03) [ Internal has been selected as an Authentication Type. ]

(09:23:03) [ Agent First Strategy has been selected as the Delivery
Strategy. ]

(09:23:03) [ Zone based AAEP system has been selected for installation. ]

(09:23:03) [ SIP Platform has been selected for installation. ]



(09:23:03) Unpackaging distribution file callbackassist.package...

(09:23:10) Creating callback Group...
(09:23:10) Creating callback User...
(09:23:11) 32 bit Architecture detected ...
(09:23:11) Using 32 bit PostgreSQL Installer...
(09:23:11) Installing PostgreSQL Server, this step may take several
minutes...
(09:23:35) Creating 'callback' PostgreSQL user...
(09:23:35) Creating 'callback' database...
(09:23:35) 'callback' database created.
server signaled
(09:23:35) Taking backup of DB configuration files for setting up the
replication later
(09:23:35) Restarting database to reset max_connections
Restarting PostgreSQL 9.4:



waiting for server to shut down.... done
server stopped
waiting for server to start.... done
server started
PostgreSQL 9.4 restarted successfully
```

```
(09:23:37) Signaling Postgresql postmaster...
(09:23:37) Done.
(09:23:37) Setting ownership to callback user.
(09:23:37) Performing JDK silent install...
(09:23:40) JDK installed.
java version "1.7.0_75"
Java(TM) SE Runtime Environment (build 1.7.0_75-b13)
Java HotSpot(TM) Client VM (build 24.75-b04, mixed mode)
(09:23:40) Installation of Callback Assist Maintenance done.
(09:23:40) Check /opt/Avaya/callbackassist/maintenance/logs/execution.log
file for Callback Assist Maintenance service startup details.
(09:23:40) Installation of Callback Assist Engine done.
(09:23:40) Check /opt/Avaya/callbackassist/engine/logs/execution.log file
for Callback Assist Engine service startup details.
(09:23:40) 32 bit Architecture detected ...
(09:23:40) Installing file server for Red Hat 6.x 32 bits
(09:23:41) Local Ip Address: 135.122.99.103
(09:23:41) INFO: Using hostname/FQDN: server103.avayacba.com to run the
file-server.
(09:23:41) Configuring File Server IP Address...
(09:23:41) Done.
(09:23:41) Changing SELinux context to some CBA files...
(09:23:41) Waiting for the Database Schema to be created or updated...
Migration successful
(09:24:06) Database Schema was successfully created or updated.
(09:24:06) Platform successfully set.
(09:24:06) AAEP System type successfully set.
(09:24:06) Primary EPM is successfully added for Default Zone.
(09:24:07) Deployment type successfully set.
(09:24:07) Storing Release version and Build Number...
(09:24:07) Release version and Build number successfully stored.
(09:24:07) Running Platform dependant changes...
Migration successful
(09:24:27) Database Schema was successfully updated.
```

```
(09:24:27) Running Deployment Type changes...
Migration successful
(09:24:48) Database Schema was successfully updated.
(09:24:48) Updating authentication type into database ...
(09:24:48) Installing weblm Tomcat service (tomcat-weblm)...
(09:24:48) Installation of tomcat-weblm done.
(09:24:48) Installing adminapp Tomcat service (tomcat-adminapp)...
(09:24:48) Installation of tomcat-adminapp done.
(09:24:48) Installing ddapps Tomcat service (tomcat-ddapps)...
(09:24:48) Installation of tomcat-ddapps done.
(09:24:48) Installing webcallback Tomcat service (tomcat-webcallback)...
(09:24:49) Installation of tomcat-webcallback done.
(09:24:49) Deploying Applications...
(09:24:49) Moving tomcat realm related jar files
(09:24:49) Installation of Tomcat instances done.
(09:24:49) Installing BSR Server.  This operation may take several
minutes...
(09:24:49) Local Ip Address: 135.122.99.103
(09:24:49) Setting ownership to callback user.
Starting Callback Assist Engine...
Callback Assist Engine Started.                            [  OK  ]
Callback Assist Engine ( pid 13353 ) is running...
Starting Callback Assist Maintenance...
Callback Assist Maintenance Started.                       [  OK  ]
Callback Assist Maintenance ( pid 13419 ) is running...
Service tomcat-weblm is not present. Nothing to do
Starting tomcat-adminapp...                                [  OK  ]
tomcat-adminapp ( pid 13504 ) is running...
Starting tomcat-webcallback...                             [  OK  ]
tomcat-webcallback ( pid 13593 ) is running...
Starting tomcat-ddapps...                                  [  OK  ]
tomcat-ddapps ( pid 13687 ) is running...
Starting CBA BSR Server service...                         [  OK  ]
CBA BSR Server ( pid 13766 ) is running...
```

```
Starting Callback Assist File Server...
Callback Assist File Server Started.                        [  OK  ]
Callback Assist File Server ( 13896 ) is running...         [  OK  ]
Callback Assist File Server Ping test                       [  OK  ]
Callback Assist File Server Read/Write cycle Test           [  OK  ]
Callback Assist File Server Ring Status is Up               [  OK  ]
Callback Assist File Server is joined to a cluster          [  NO  ]
If this is not a HA deployment, then disregard this warning.

(09:25:24) [ Installation of Avaya Callback Assist (SIP - Agent First
Strategy) completed. ]



**********************************************************
ACTION REQUIRED on Time Zone Configuration
**********************************************************
The default Time Zone of CBA is UTC. If your system requires a different
Time Zone
you must manually configure it in the Global Settings of the Admin Portal.
**********************************************************


[root@server103 4.4.0.0]#
```

Configuring Avaya Aura® Communication Manager

Basic configuration

Creating a new Node Name for Session Manager Asset

Add a new Node Name for the Session Manager Asset IP Address to Communication Manager node-names list by entering the command: *change node-names ip*. Enter a descriptive name and Session Manager Asset IP Address.

```
change node-names ip                                          Page   1 of   2
                                IP NODE NAMES
    Name                  IP Address
1a08_61_217           135.122.61.190
ANS-181.228           148.147.181.228
ANS-181.229           148.147.181.229
ANS-DEV-TEST          135.27.175.44
DenverCM              135.9.160.21
H323.Abacus           135.122.61.85
Horaci                135.64.120.207
Jerry-ANS-DEMO        135.122.61.6
QA8700-CL01A08        135.122.61.142
RHEL53RC2             135.122.6.155
Vz-PuneSM             148.147.181.128
araes01               135.122.60.117
cba_asm_56.135        135.122.56.135
cba_asm_mes_122       135.122.60.122
ccr2                  135.122.61.239
cs1k                  135.122.63.44
( 16 of 26   administered node-names were displayed )
Use 'list node-names' command to see all the administered node-names
Use 'change node-names ip xxx' to change a node-name 'xxx' or add a node-name
```

Figure 41 - Details of created node names

Creating and configuring a new Signaling Group for Session Manager Asset

Create a new Signaling Group by entering the command: *add signaling-group xxxx* where xxxx is the Signaling Group number. Consider listing the available Signaling Groups first to pick the number.

The required fields to configure are as follows:

Field	Value
Group Type	sip
Transport Method	tls or tcp
Near-end Node Name	procr
Near-end Listen Port	5060 or 5061
Far-end Node Name	node-name configured above

Field	Value
Far-end Domain	SIP Domain
Session Establishment timer	**120** Note: Always set to 120, as the default value for this field on a new Signaling Group form is 3 minutes which causes the queued calls to be disconnected after three minutes.

```
display signaling-group 9
                                SIGNALING GROUP

 Group Number: 9                Group Type: sip
                              Transport Method: tcp
  IMS Enabled? n               Co-Resident SES? n
    IP Video? n



   Near-end Node Name: procr                 Far-end Node Name: cba_asm_mes_122
 Near-end Listen Port: 5060                Far-end Listen Port: 5060
                                         Far-end Network Region: 1
Far-end Domain: csilab.avaya.com

                                              Bypass If IP Threshold Exceeded? n

         DTMF over IP: rtp-payload            Direct IP-IP Audio Connections? n
Session Establishment Timer(min): 120                   IP Audio Hairpinning? n
         Enable Layer 3 Test? n

                                                 Alternate Route Timer(sec): 6
```

Figure 42 - Details of Configured Signaling Port

Creating and configuring a new SIP Trunk

Create a new Trunk Group by entering the command: *add trunk-group xxxx* where xxxx is the Trunk Group number. Consider listing the available Trunk Group first to pick the number.

The required fields to configure are as follows:

Field	Value
Page 1 - Group Type	sip
Page 1 - Signaling Group	Signaling Group number created above
Page 1 - Number of Members	the number of ports for this SIP connection

Page 3 - UUI Treatment	shared
Page 4 – ASAI	1

Sample Trunk configuration:

```
display trunk-group 9                                          Page   1 of  22
                                TRUNK GROUP

Group Number: 9                     Group Type: sip           CDR Reports: n
  Group Name: cba_asm_mes_60.122          COR: 1        TN: 1        TAC: 409
   Direction: two-way        Outgoing Display? y
 Dial Access? n                                  Night Service:
Queue Length: 0
Service Type: tie                    Auth Code? n

                                                      Signaling Group: 9
                                                    Number of Members: 255
```

Figure 43 - Details of Configured SIP Trunk

```
display trunk-group 9                                          Page   3 of  22
TRUNK FEATURES
          ACA Assignment? n            Measured: internal
                                                    Maintenance Tests? y


                   Numbering Format: public
                                          UUI Treatment: shared
                                       Maximum Size of UUI Contents: 128
                                          Replace Restricted Numbers? n
                                          Replace Unavailable Numbers? n


              Send UCID? y


 Show ANSWERED BY on Display? y
```

Figure 44 - Details of Configured SIP Trunk

```
display trunk-group 9                                          Page   4 of  22
                          SHARED UUI FEATURE PRIORITIES

                               ASAI: 1

            Universal Call ID (UCID): 2

MULTI SITE ROUTING (MSR)

                          In-VDN Time: 3
                              VDN Name: 4
                     Collected Digits: 5
               Other LAI Information: 6
                      Held Call UCID: 7
```

Figure 45 - Details of Configured SIP Trunk

Creating Polling Vectors and VDNs for Best Service Routing (BSR)

Minimal configuration for a single Callback Configuration requires configuring one Polling VDN and vector to obtain expected wait time (EWT) for the desired Agent skill.

Creating and configuring a new vector for Polling VDN

Create a new vector by entering the command: *check vector xx* where xx is the vector number.

The required fields to configure are as follows:

Field	Value
Name	Enter desired name
Multimedia?	N
Attendant Vectoring?	N
Meet-me Conf?	N
Lock?	N

Add the following vector steps:

- consider skill xx pri m adjust-by x
- reply-best
- stop

An example is shown below:

```
display vector 59                                          Page   1 of   6
                                CALL VECTOR

    Number: 59              Name: SIPCBA Poll
Multimedia? n      Attendant Vectoring? n    Meet-me Conf? n           Lock? n
     Basic? y   EAS? y   G3V4 Enhanced? y   ANI/II-Digits? y   ASAI Routing? y
 Prompting? y   LAI? y  G3V4 Adv Route? y   CINFO? y   BSR? y   Holidays? y
 Variables? y   3.0 Enhanced? y
01 consider     skill    60   pri m adjust-by 0
02 reply-best
03 stop
04
05
06
07
08
09
10
11
12

                      Press 'Esc f 6' for Vector Editing
```

Figure 46 - Polling vector example

Creating and configuring a Polling VDN

Create a new VDN by entering the command: *add vdn xxxxx* where xxxxx is the VDN number.

The required fields to configure are as follows:

Field	Value
Page 1 - Name	Enter desired name
Page 1 - Vector number	Created above
Page 2 - BSR Available Agent Strategy	1st-found (or desired if using multiple skills)

An example is shown below:

```
change vdn 51009                                                Page   1 of   3
                          VECTOR DIRECTORY NUMBER

                            Extension: 51009
                                 Name*: SIPCBA Poll
                          Destination: Vector Number        59
                    Attendant Vectoring? n
                  Meet-me Conferencing? n
                    Allow VDN Override? n
                                  COR: 1
                                  TN*: 1
                             Measured: none


          VDN of Origin Annc. Extension*:
                              1st Skill*:
                              2nd Skill*:
                              3rd Skill*:



* Follows VDN Override Rules
```

Figure 47 - Add Polling VDN example

```
change vdn 51009                                                Page   2 of   3
                          VECTOR DIRECTORY NUMBER

                                  AUDIX Name:

                           Return Destination*:

                              BSR Application*:
                  BSR Available Agent Strategy*: 1st-found
                             BSR Tie Strategy*: system

                        Observe on Agent Answer? n


                  Display VDN for Route-To DAC*? n
 VDN Override for ISDN Trunk ASAI Messages*? n

                           BSR Local Treatment*? n

             Reporting for PC Predictive Calls? n
              Pass Prefixed CPN to VDN/Vector*? system
* Follows VDN Override Rules
```

Figure 48 -Add Polling VDN example

Creating Queuing Vectors and VDNs for Best Service Routing (BSR)

Minimal configuration for a single Callback Configuration requires configuring one Queuing VDN and vector to place and transfer calls on the desired Agent skill. Notice that the queuing vector steps differ based on the delivery strategy selected.

Creating and configuring a new vector for Queuing VDN

Create a new vector by entering the command: *check vector xx* where xx is the vector number.

The required fields to configure are as follows:

Field	Value
Name	Enter desired name
Multimedia?	N
Attendant Vectoring?	N
Meet-me Conf?	N
Lock?	N

The vector steps must be the following ones:

- consider skill xx pri m adjust-by 0
- queue-to skill xx pri m
- stop

IMPORTANT:

- If configuring a SIP Customer First Priority Queuing installation*, queue-to skill* step must queue the call with **higher priority** than regular callers, so use high or top priority instead.
- *Do not add any wait treatment steps, like playing announcements, as there will be a phantom call queued and not a regular caller. Adding these additional steps invalidates the call flow.*

Agent First and ***Customer First Phantom Pool*** example:

```
change vector 51                                              Page   1 of   6
                                CALL VECTOR

    Number: 51                Name: SIPCBA Queue
Multimedia? n      Attendant Vectoring? n    Meet-me Conf? n           Lock? n
     Basic? y   EAS? y   G3V4 Enhanced? y   ANI/II-Digits? y   ASAI Routing? y
 Prompting? y   LAI? y  G3V4 Adv Route? y   CINFO? y   BSR? y   Holidays? y
 Variables? y   3.0 Enhanced? y
01 consider      skill    60   pri m adjust-by 0
02 queue-to      skill 60   pri m
03 stop
04
05
06
07
08
09
10
11
12

                        Press 'Esc f 6' for Vector Editing
```

Figure 49 - Queuing vector example

Customer First Priority Queuing example:

```
change vector 51                                              Page   1 of   6
                                CALL VECTOR

    Number: 51                Name: SIPCBA Queue
Multimedia? n      Attendant Vectoring? n    Meet-me Conf? n           Lock? n
     Basic? y   EAS? y   G3V4 Enhanced? y   ANI/II-Digits? y   ASAI Routing? y
 Prompting? y   LAI? y  G3V4 Adv Route? y   CINFO? y   BSR? y   Holidays? y
 Variables? y   3.0 Enhanced? y
01 consider      skill    60   pri h adjust-by 0
02 queue-to      skill 60   pri h
03 stop
04
05
06
07
08
09
10
11
12

                        Press 'Esc f 6' for Vector Editing
```

Figure 50 - Queuing vector example

Creating and configuring a Queuing VDN

Create a new VDN by entering the command: *add vdn xxxxx* where xxxxx is the VDN number.

The required fields to configure are as follows:

Field	Value
Page 1 – Name	Enter desired name
Page 1 - Vector number	Created above

Page 2 - BSR Available Agent Strategy	1st-found (or desired if using multiple skills)

An example is shown below:

```
change vdn 51009                                                Page   1 of   3
                          VECTOR DIRECTORY NUMBER

                            Extension: 51009
                                 Name*: SIPCBA Poll
                          Destination: Vector Number        59
                    Attendant Vectoring? n
                   Meet-me Conferencing? n
                     Allow VDN Override? n
                                  COR: 1
                                  TN*: 1
                             Measured: none


          VDN of Origin Annc. Extension*:
                              1st Skill*:
                              2nd Skill*:
                              3rd Skill*:



* Follows VDN Override Rules
```

Figure 51 - Add Polling VDN example

```
change vdn 51009                                                Page   2 of   3
                          VECTOR DIRECTORY NUMBER

                                  AUDIX Name:

                           Return Destination*:

                              BSR Application*:
                  BSR Available Agent Strategy*: 1st-found
                             BSR Tie Strategy*: system

                        Observe on Agent Answer? n


                  Display VDN for Route-To DAC*? n
 VDN Override for ISDN Trunk ASAI Messages*? n

                           BSR Local Treatment*? n

             Reporting for PC Predictive Calls? n
              Pass Prefixed CPN to VDN/Vector*? system
* Follows VDN Override Rules
```

Figure 52 - Add Polling VDN example

Configuring Avaya Aura® Session Manager (ASM)

This section describes basic System Manager Configuration in order to have Communication Manager, Experience Portal and Callback Assist Best Service Routing (BSR) communicating using SIP protocol. Additional steps may be required if customer has already an Avaya Session Manager platform in place. For more information, see Administering Avaya Aura ® Session Manager document from the support site at http://support.avaya.com.

Logging on to System Manager Console

1. On the Web browser, enter the System Manager URL **https://<FQDN>/SMGR**
2. In the ***User ID*** field, enter the user name.
3. In the ***Password*** field, enter the password.
4. Click ***Log On***.

Adding a SIP domain

1. From the System Manager main menu, select ***Routing > Domains***
2. On the Domain Management page, click ***New***.
3. Enter the ***domain name*** and notes for the new domain or sub-domain.
4. Select **sip** as the domain type from the drop-down list.
5. Click ***Commit***.

Adding SIP Entities for Communication Manager

1. From the System Manager main menu, select ***Routing > SIP Entities***
2. On the SIP Entities Details page, click ***New***.
3. Add the SIP Entities information:

Field	Value
Name	Name of the CM server. This name must be unique and can have between 3 and 64 characters.
FQDN or IP Address	IP address of the CLAN/Proc configured in the signaling-group on CM

Type	CM
Location	SIP Entity location for this CM
Time Zone	Default time zone to be used for this entity

4. Click ***Commit*** to save the configuration.

Adding SIP Entities for Experience Portal

1. From the System Manager main menu, select ***Routing > SIP Entities***
2. On the SIP Entities Details page, click ***New***.
3. Add the SIP Entities information:

Field	Value
Name	Name of the AAEP/AVP server. This name must be unique and can have between 3 and 64 characters.
FQDN or IP Address	IP address of Media Processing Platform (MPP). If you have multiple MPPs then add all Media Processing Platform with a single FQDN under *Session Manager -> Network Configuration -> Local Host Name Resolution*
Type	Experience Portal Note: Type depends on SMGR release.
Location	SIP Entity location
Time Zone	Default time zone to be used for this entity

4. Click ***Commit*** to save the configuration.

Adding SIP Entities for Callback Assist BSR Service

1. From the System Manager main menu, select ***Routing > SIP Entities***

2. On the SIP Entities Details page, click ***New***.
3. Add the SIP Entities information:

Field	Value
Name	Name of the CBA BSR server. This name must be unique and can have between 3 and 64 characters.
FQDN or IP Address	IP address of CBA server.
Type	Other
Location	SIP Entity location
Time Zone	Default time zone to be used for this entity

4. Click ***Commit*** to save the configuration.

Adding SIP Entity Link for Communication Manager

1. From the System Manager main menu, select ***Routing > Entity Links***
2. On the Entity Links page, click ***New***
3. Add the SIP Entity Links details:

Field	Value
Name	Name of SIP Entity link to Communication Manager.
SIP Entity 1	Select a Session Manager instance from the drop-down list.
Protocol	TCP or TLS
Port	Port to be used by SIP Entity 1.
SIP Entity 2	Select from the drop-down list the Entity link created for CM on previous step.
Port	Port to be used by SIP Entity 2.

Trusted	Select this option.

4. Click ***Commit*** to save configuration.

Adding SIP Entity Link for Experience Portal

1. From the System Manager main menu, select ***Routing > Entity Links***
2. On the Entity Links page, click ***New***
3. Add the SIP Entity Links details:

Field	Value
Name	Name of SIP Entity link to AAEP/AVP.
SIP Entity 1	Select a Session Manager instance from the drop-down list.
Protocol	TCP or TLS
Port	Port to be used by SIP Entity 1.
SIP Entity 2	Select from the drop-down list the Entity link created for MPPs on previous step.
Port	Port to be used by SIP Entity 2.
Trusted	Select this option.

4. Click ***Commit*** to save configuration.

Adding SIP Entity Link for Callback Assist BSR Service

1. From the System Manager main menu, select ***Routing > Entity Links***
2. On the Entity Links page, click ***New***
3. Add the SIP Entity Links details:

Field	Value

Name	Name of SIP Entity link to Callback Assist BSR box.
SIP Entity 1	Select a Session Manager instance from the drop-down list.
Protocol	TCP
Port	Port to be used by SIP Entity 1.
SIP Entity 2	Select from the drop-down list the Entity link created for CBA BSR Service on previous step.
Port	Port to be used by SIP Entity 2. By convention port 33337 is used.
Trusted	Select this option.

4. Click ***Commit*** to save configuration.

Adding Routing Policy for Communication Manager

1. From the System Manager main menu, select ***Routing > Policies***
2. On the Routing Policies page, click ***New*** to add the Routing Policy information for Communication Manager.
3. On the Routing Policy Details page, under General, provide the value for ***Name*** field.
4. Under SIP Entity as Destination, click ***Select***.
5. From the list of SIP Entities, select the Communication Manager SIP Entity.
6. Under Time of Day, click ***Add***.
7. From the Time Ranges List page, select the required ***Time Range***.
8. Click ***Select***.
9. Click ***Commit*** to save the configuration.

Adding Routing Policy for Experience Portal

1. From the System Manager main menu, select ***Routing > Policies***
2. On the Routing Policies page, click ***New*** to add the Routing Policy information for AAEP/AVP.
3. On the Routing Policy Details page, under General, provide the value for ***Name*** field.
4. Under SIP Entity as Destination, click ***Select***.
5. From the list of SIP Entities, select the AAEP/AVP SIP Entity.
6. Under Time of Day, click ***Add***.
7. From the Time Ranges List page, select the required ***Time Range***.
8. Click ***Select***.
9. Click ***Commit*** to save the configuration.

Dial Patterns

You must configure a minimum of three dial patterns.

- One pattern to route calls to Experience Portal SIP Entity for customer's incoming calls.
- One pattern to route calls to agents for polling/queuing VDN's to Communication Manager.
- One pattern to route outbound calls from Experience Portal to PSTN (through CM or SIP Gateway).

Pattern for incoming calls to Experience Portal

1. From the System Manager main menu, select ***Routing > Dial Patterns***
2. On the Dial Patterns page, click ***New*** to add a new Dial Pattern for Experience Portal.
3. Under General, provide the value for the fields provided in the table.

Field	Value
Pattern	Dial pattern to match.
Min	Minimum number of digits to be matched.
Max	Maximum number of digits to be matched.
SIP Domain	Domain for which to restrict the dial pattern.

	Must match the created domain.

4. Under Originating Locations and Routing Policies, click ***Add***.
5. Under Originating Location, select the ***originating location***.
6. Under Routing Policies, select the ***routing policy*** of Experience Portal.
7. Click **Select**.
8. Click **Commit** to save the configuration.

Pattern for polling and queuing VDN's call to Communication Manager

1. From the System Manager main menu, select ***Routing > Dial Patterns***
2. On the Dial Patterns page, click ***New*** to add a new Dial Pattern for Communication Manager.
3. Under General, provide the value for the fields provided in the table.

Field	Value
Pattern	Dial pattern to match VDN's. Additional Dial patterns may be needed if pattern cannot route all VDN's.
Min	Minimum number of digits to be matched.
Max	Maximum number of digits to be matched.
SIP Domain	Domain for which to restrict the dial pattern. Must match the created domain.

4. Under Originating Locations and Routing Policies, click ***Add***.
5. Under Originating Location, select the ***originating location***.
6. Under Routing Policies, select the ***routing policy*** of Communication Manager.
7. Click **Select**.
8. Click **Commit** to save the configuration.

Pattern to route outbound calls to PSTN

1. From the System Manager main menu, select ***Routing > Dial Patterns***
2. On the Dial Patterns page, click ***New*** to add a new Dial Pattern for Communication Manager.
3. Under General, provide the value for the fields provided in the table.

Field	Value
Pattern	Dial pattern of customer contact numbers. Most of the time will be already present.
Min	Minimum number of digits to be matched.
Max	Maximum number of digits to be matched.
SIP Domain	Domain for which to restrict the dial pattern. Must match the created domain.

4. Under Originating Locations and Routing Policies, click ***Add***.
5. Under Originating Location, select the ***originating location***.
6. Under Routing Policies, select the ***routing policy*** of:
    a. If routing PSTN calls to Communication Manger, select Communication Manager routing policy.
    b. If routing PSTN calls to a SIP Gateway or Session Border Controller, select SIP Gateway or Session Border Controller routing policy.
7. Click **Select**.
8. Click **Commit** to save the configuration.

Configuring Avaya Aura® Experience Portal

This section describes basic VoIP Connections and applications required on Experience Portal in order to communicate with Avaya Aura Session Manager using SIP protocol. Additional steps may be required if customer has an Experience Portal platform in place. For more information, see Implementing Avaya Aura Experience Portal document Support Avaya site at http://support.avaya.com.

Configuring VoIP Connections

A VoIP SIP connection is required in order to communicate with Avaya Aura Session Manager with SIP protocol and route calls to Avaya Aura Communication Manager and PSTN.

1. On the Web browser, enter the Experience Portal EPM Administration URL https://<FQDN>/VoicePortal.
2. In the ***User Name*** field, enter the user name.
3. Click **Submit**.
4. In the ***Password*** field, enter the password.
5. Click ***Logon***.

6. Click on ***System Configuration -> VoIP Connections***
7. Click on ***SIP*** tab.
8. Click on ***Add***.
9. On the Add application window, configure the following details:

Field	Value
Name	Enter a name for this connection
Enable	Yes
Proxy Transport	TCP or TLS
Address	Enter Avaya Aura Session Manager Asset IP Address
Port	Enter Avaya Aura Session Manager Asset Port
Listening Port	Enter port configured on Entity Link created for Experience Portal
SIP Domain	Enter SIP Domain configured on Avaya Aura Session Manager SIP Domain configuration section
Maximum Simultaneous Calls	Enter the amount of concurrent calls to be supported (See Experience Portal license). Note: There must be inbound and outbound allowed but not necessarily all must be inbound and outbound.

10. Click on ***Save***.

Configuring Inbound and Outbound Applications

It is required to configure two different applications on Experience Portal:

- Inbound Application: To handle inbound calls and offer call back option based on a DNIS list.
- Outbound Application: To complete registered callbacks to customers.

Inbound Application

The inbound application or the inbound call handling can be configured in two different ways.

- As a CCXML application (default) or
- Invoke Offer application as a VXML sub dialog from another IVR application.

Only one of the above approaches should be followed and not both. **In the VXML as sub dialog approach Wait Treatment on CBA is not supported**.

Configuring Inbound as a CCXML application

1. On the Web browser, enter the Experience Portal EPM Administration URL https://<FQDN>/VoicePortal.
2. In the ***User Name*** field, enter the user name.
3. Click **Submit**.
4. In the ***Password*** field, enter the password.
5. Click ***Logon***.
6. From the menu select ***System Configuration -> Applications***
7. Click on ***Add.***
8. On the Add application window, configure the following details:

Field	Value
Name	Callback_Assist_Inbound
Enable	Yes
Type	CCXML
CCXML URL	http://CBA_FQDN:8080/CBAScripts/cbaCallControl or http://<CBA_FQDN>:8080/CBAScripts/cbaCallControl?EmergencyDestination=XXXXXX if overriding Emergency Destination route. See *Administering Avaya Callback Assist Guide* for further details.
Speech Servers - ASR	No ASR
Speech Servers - TTS	If TTS is available, it can be used with CBA, otherwise just select No TTS.
Application Launch	Inbound and Number or Inbound and Number Range

9. Click on ***Save.*** See configured application sample below:

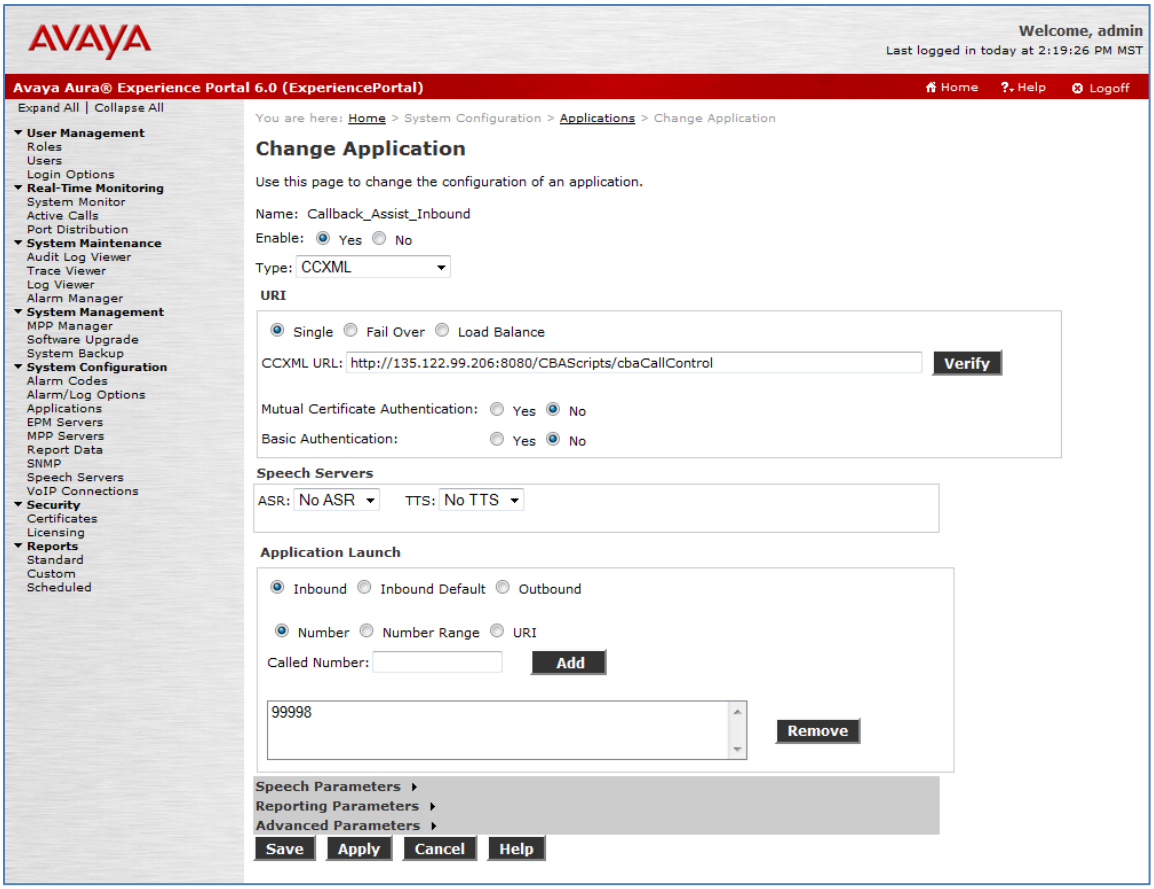


Figure 53 - Experience Portal Page for Change Application

Inbound as a VXML sub dialog

Callback Assist offer application or the application which handles inbound calls of CBA can also be invoked as a VXML sub dialog. This approach eliminates the need to do a SIP transfer to Callback Assist from an existing IVR application to offer callback to callers.

To use this approach **Global Settings → Advanced → Callback Offer Application as an Orchestration Designer Module** parameter should be enabled in the Administration UI.

The Orchestration Designer application of Callback Assist that needs to be invoked is CBASIPOffer. The URL of the CBASIPOffer OD module is **http://<CBA_FQDN>:8080/CBASIPOffer/Start.** The input and output parameters of CBASIPOffer application are listed in the tables below.

Input Parameters

dnis	The DNIS value of the *Callback Configuration* to use. Using this value CBA uniquely identifies the Callback Configuration to be used for the incoming call. If the input value does not match with any of the available Callback Configurations, callback won't be offered and

	an error result is returned.
vpmsAddress	The IP address/hostname of the AAEP through which the outbound calls will be delivered. This should be matching with one of the EPMS configured in the *Site Definition* page of CBA Administration UI. The application returns with an error if the values do not match. This parameter is mandatory if Site based CBA deployment is used and not required for zone based deployments.
callCenterId	This is the Id value of the *CBA BSR Call Center Application* configured in the *BSR Configuration* in Admin UI. This is an optional parameter. When passed as an input, CBA will use the Call Center application corresponding to the Id value to process the inbound and outbound calls instead of the default Call center application assigned to the Callback Configuration. When a corresponding match is not found the application returns with an error.
vpmsZoneId	Required if zone based deployment of AAEP is used. It is the Id of the zone in which the call needs to be delivered. The Id of a zone can be obtained using the Management Interface Webservice of Voiceportal. The zone should also be configured in the zone definitions page of CBA Administration UI.

CBA retrieves the UUI information from the session. The UUI information should be set to the session variable before invoking the offer application. Sample code shown below.

```
mySession.getVariableField(IProjectVariables.SESSION,
IProjectVariables.SESSION_FIELD_UUI).setValue(
                          "PD,04;C8,7465737464617461;FA,000202CC560B9377");
```

It is mandatory that the UUI information should be encoded based on the UUI Operation mode (Service Provider or Shared UUI) and set to session.

Output Parameters

result	The possible output values returned are ***callback_accepted*** Call back request is successfully registered.

	send_call_to_queue Callback not accepted by caller or callback denied to caller based on configuration. On this result the calling application should redirect the caller to the agent queue. ***exit*** Either caller dropped the call while callback was being offered or the caller has pending callback request and more callback requests are not allowed by configuration. On this result the calling application can end the session. ***error*** An error has occurred while processing the callback requests. On this result the caller needs to be redirected to the agent queue.
requestId	The callback request GUID. The unique Id value of the callback request.
queue_vdn	The Agent VDN to which the caller should be redirected when the result value is **send_call_to_queue** or **error**.
reason	Returns the reason when the returned result is **error** or **exit**

Refer Invoking CBASIPOffer as VXML sub dialog sample.

Outbound Application

1. On the Web browser, enter the Experience Portal EPM Administration URL https://<FQDN>/VoicePortal
2. In the ***User Name*** field, enter the user name.
3. Click **Submit**.
4. In the ***Password*** field, enter the password.
5. Click ***Logon***.
6. From the menu select ***System Configuration -> Applications***
7. Click on ***Add.***
8. On the Add application window, configure the following details:

Field	Value
Name	Callback_Assist_Outbound **Note:** The name of the CCXML application must exactly match the name configured in the Callback Assist Web administration in the Gobal Settings *EPM Callback Outbound application name*
Enable	Yes
Type	CCXML
CCXML URL	http://CBA_FQDN:8080/CBAScripts/cbaCallControl
Speech Servers - ASR	No ASR
Speech Servers - TTS	If TTS is available, it can be used with CBA, otherwise just select No TTS.
Application Launch	Outbound
Advanced -> Operation Mode	MUST match the same one used for Inbound application configured above.

9. Click on ***Save.*** See configured application sample below:

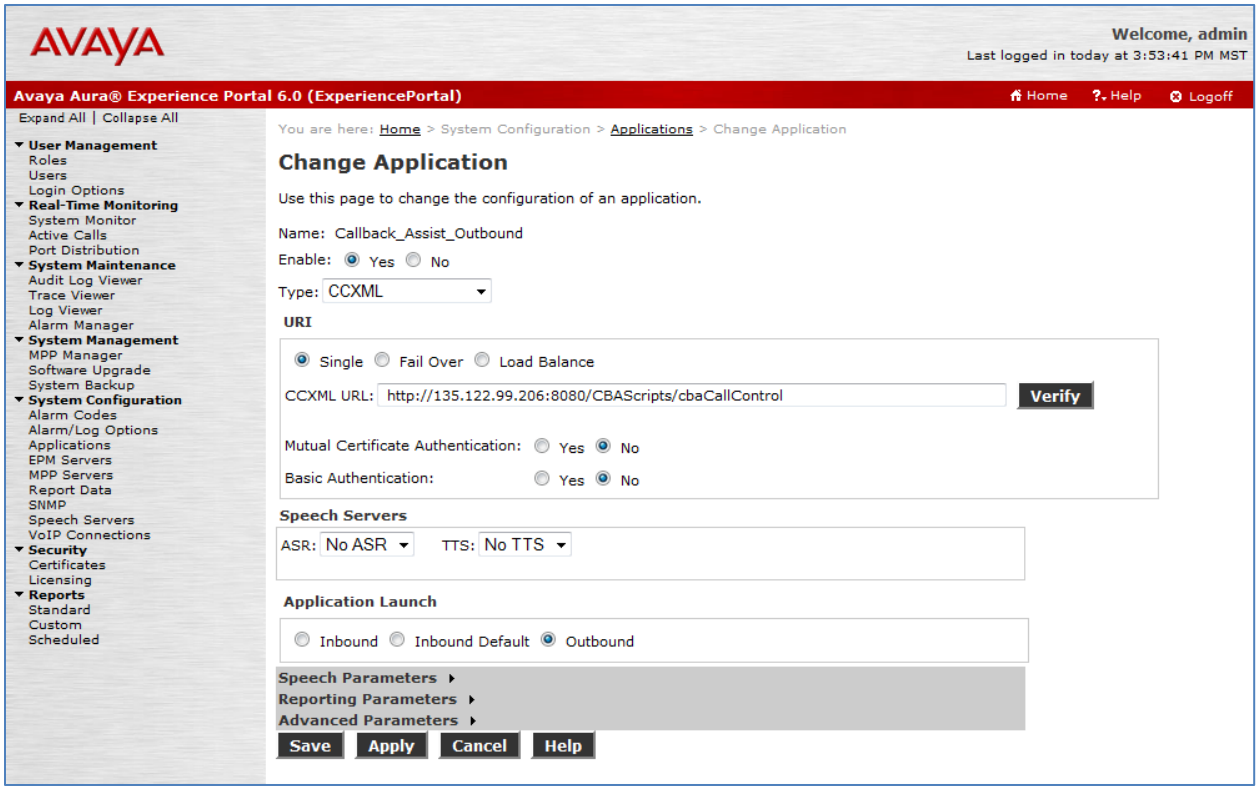


Figure 54 - Experience Portal Page- Add Application

Post Installation Steps

This section describes the minimum configuration required on Callback Assist software to start placing calls.

Callback Assist Web Administration

License Installation

For more information, see the License Installation section in this document.

Login to Administration Interface

The Administration interface is accessible from the URL http://<server-hostname>/admin (uses port tcp/80).

Default log in credential is user ***admin*** and password ***123456*** as shown below:

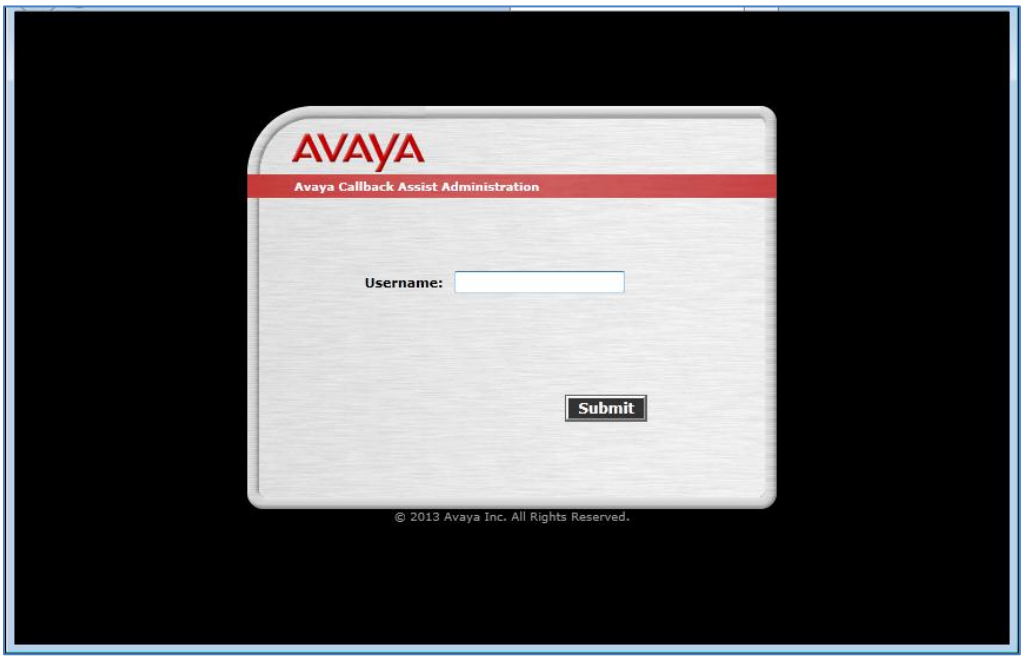


Figure 55 - Avaya Callback Administration Login Screen

After login into the application for the first time the user will be prompted to change the admin user password, as follows:

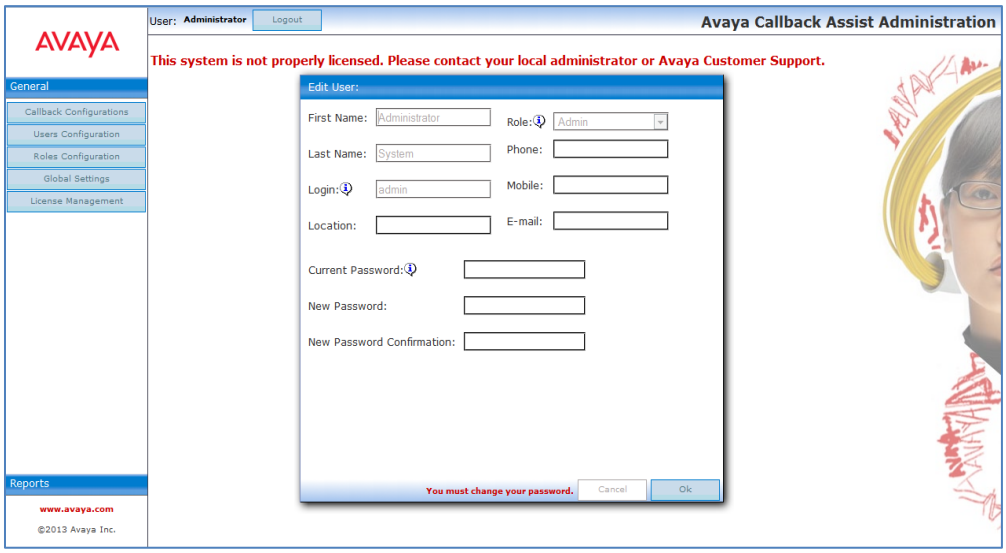


Figure 56 - Avaya Callback Assist – Change Password Screen

Enter new password and retype it on the confirmation field in order to complete the password change procedure.

Configuring license

Configure the WebLM Server URL where the Callback Assist license file is installed:

1. Go to **License Management** on the left navigation pane.
2. Enter the WebLM License Server URL replacing [Host] and [Port] with WebLM Server hostname or IP address and port number.
3. Enter the Dialog Designer WebLM License Server URL in the following format: https://\[Host\]:\[Port\]/WebLM/LicenseServer. The purpose of the Dialog Designer URL here is to display the Dialog Designer license status only and it will not affect CBA operations if not updated
4. Specify the number of CBA ports to be used based on your license. On a CTI installation, the number of CBA ports, as minimum, must match the number of phantom ports configured on Engine component.
5. Click the **Update** button to update the **License Management** page with WebLM URL and port usage information.

Note: If you have installed Callback Assist Application freshly then **Administration** → **Licensing Management** will be automatically pointed to Integrated WebLM URL. If you want to use external WebLM, modify the URL in **License Management** menu. If the Callback Assist Application is upgraded to 4.1.6 then the License URL will still point to the existing URL.

Example 1: The **License Management** page looks as follows when the currently installed Callback Assist version does not have a valid license file.

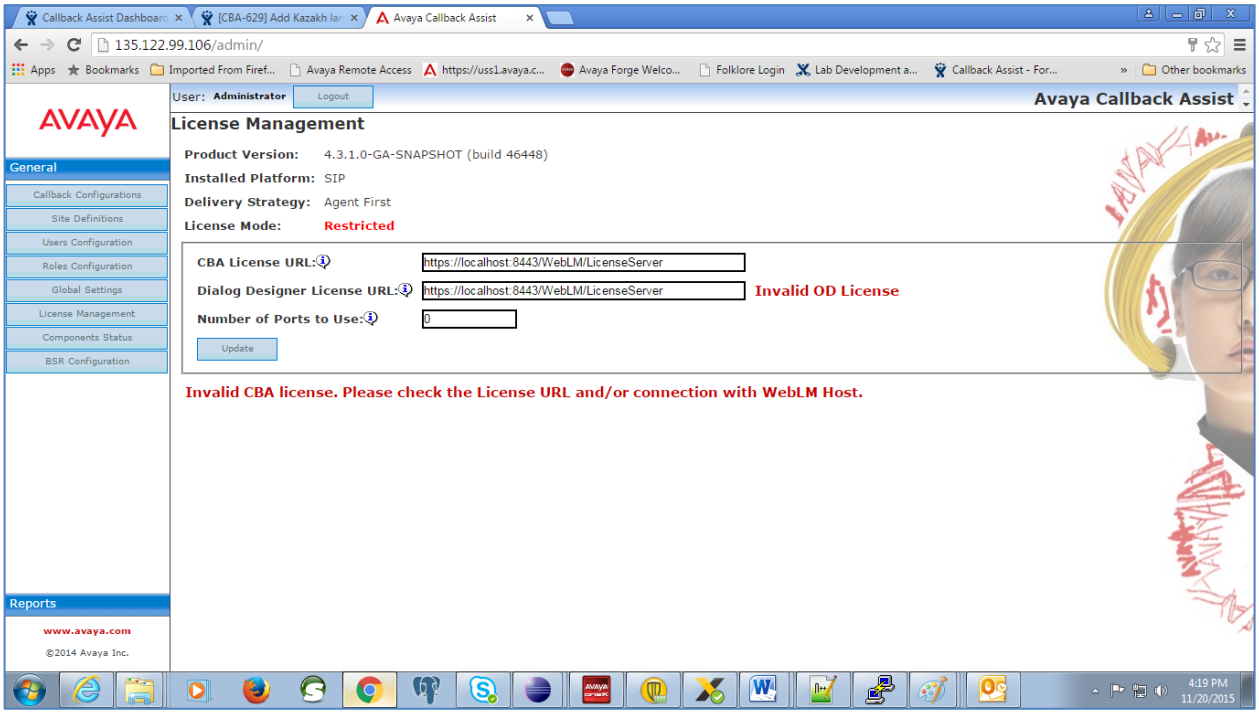


Figure 57 - License Management Screen - Invalid License

Example 2: The **License Management** page looks as follows when the currently installed Callback Assist version has a valid license file.

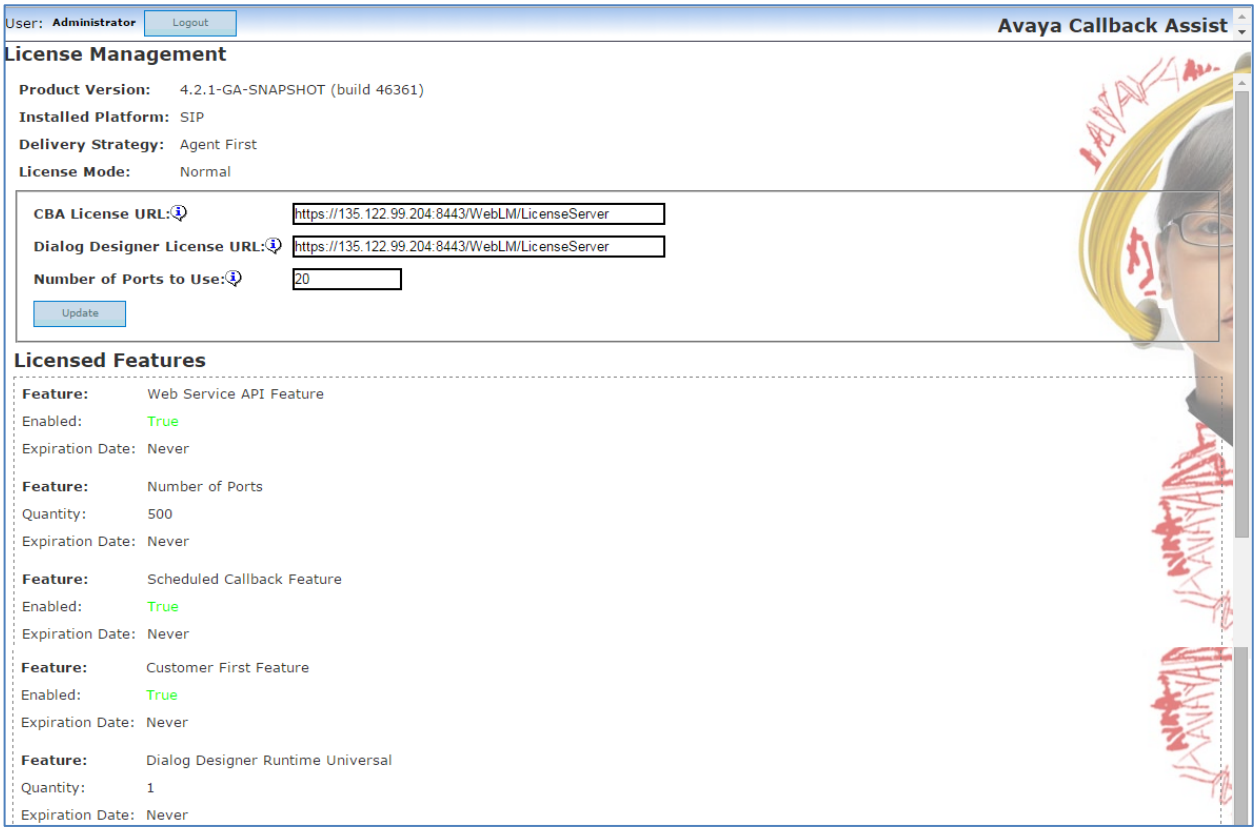


Figure 58 - License Management Screen - Valid License

Configuring Global Settings

- ### *Configuring the IVR Tab*

Go to **Global Settings > IVR** tab.
The system displays the **IVR** page with the default IVR parameters as follows:

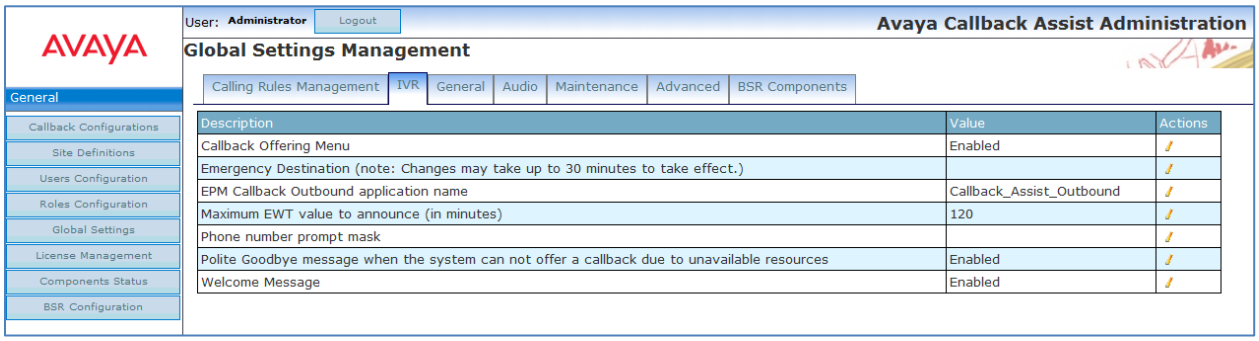


Figure 59 - Audio tab configuration screen

- ***Emergency Destination***: Optionally set a VDN or destination number to route calls on emergency scenarios.
- ***EPM Callback Outbound application name:*** Double check that the name entered matches the one defined on Experience Portal EPM.

Use default values for the rest of the parameters. For more information on this operation, see the *Administering Avaya Callback Assist* guide.

- ### *Configuring the General Tab*

Go to **Global Settings > General** tab.
The system displays the **General** page with the default General parameters as follows:

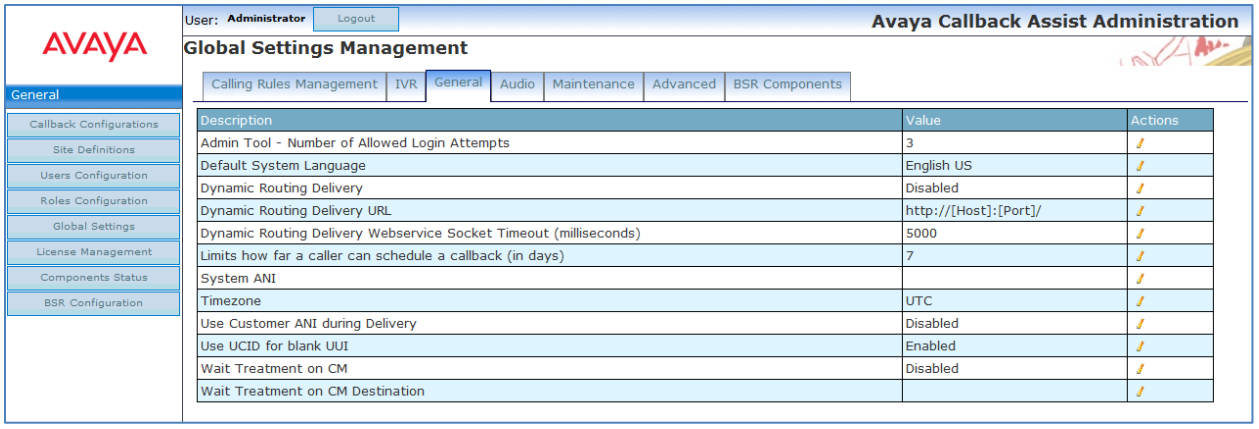


Figure 60 - Audio tab configuration screen

- ***System ANI***: Number used as ANI value for outbound calls to customers. This global parameter can be overridden per configuration basis, for more details see the *Administering Avaya Callback Assist* guide.

Use default values for the rest of the parameters. For more information on this operation, see the *Administering Avaya Callback Assist* guide.

- ### *Configuring the Audio Tab*

Go to **Global Settings > Audio** tab.
The system displays the **Audio** page with the audio parameters configuration options as follows:

Figure 61 - Audio tab configuration screen

- ***Storage URL***: Change [Host] mask with Callback Assist Server IP Address or hostname.
  For more information on this operation, see the *Administering Avaya Callback Assist* guide.

Configuring BSR Configuration

CM Configuration

Go to **CBA BSR Configuration > CM Configuration** option.

1. Click on ***Add*** from CM List.
2. Enter ***Name*** field for Communication Manager to be used.
3. Enter ***SIP Domain*** to be used on this Communication Manager.
4. Enter ***Timezone*** value from the drop-down list.
5. Click on ***Save***.

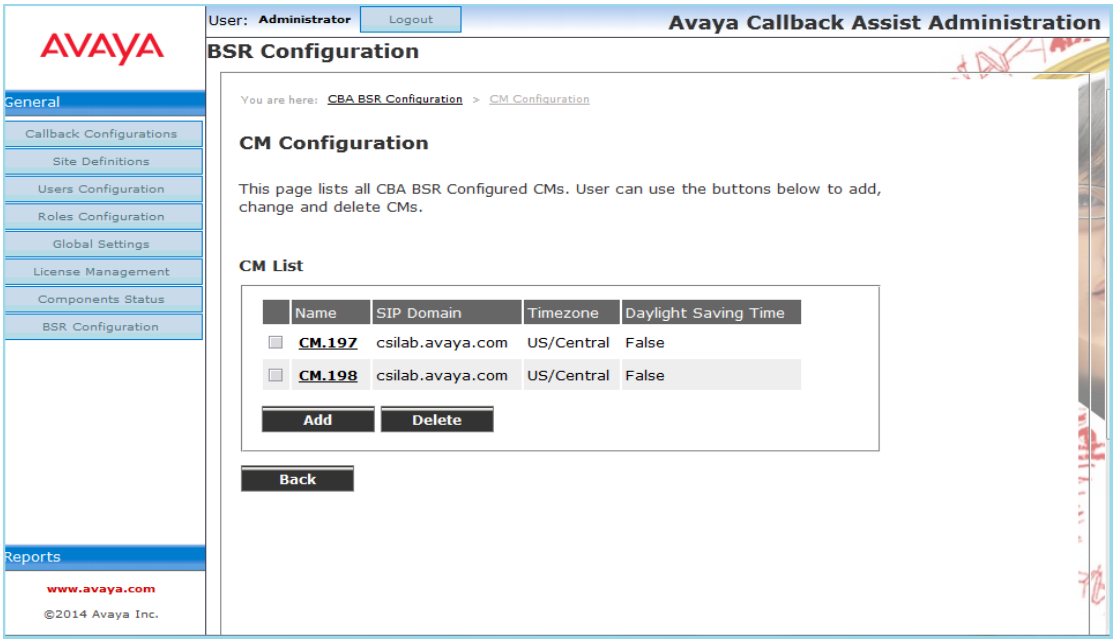


Figure 62 - BSR Configuration - Communication Manager

- ***Service Controller and SIP Proxy Configuration***

Go to **CBA BSR Configuration > Service Controller** option.

6. Click on ***Configuration*** button that belongs to *Polling Service*.
7. Click on ***Configure*** button.
8. Enter ***SIP Domain*** to be used.
9. Click on ***Configure*** button from *SIP Proxy* option.
10. Enter ***BSR Server Local Port*** value. From convention, port is 33337 but it should be the same used on *Adding SIP Entity Link for Callback Assist BSR Service* section.
11. On ***SIP Proxy IP Address***, enter Session Manager Asset IP Address.
12. On ***SIP Proxy Port***, enter Session Manager Asset SIP traffic listening port.
13. On ***Transport Protocol***, set tcp.
14. Set "bsr" as **User** and "bsr" as **Password**. Not used with TCP transport, only required if using TLS and must be a System Manager User Account.
15. Click on ***Save*** to get back to previous window and save configuration.
16. Click on ***Save*** on Direct Polling window.
17. Click on ***Save*** on Polling Service window.
18. Click on ***Start*** button of Polling Service.

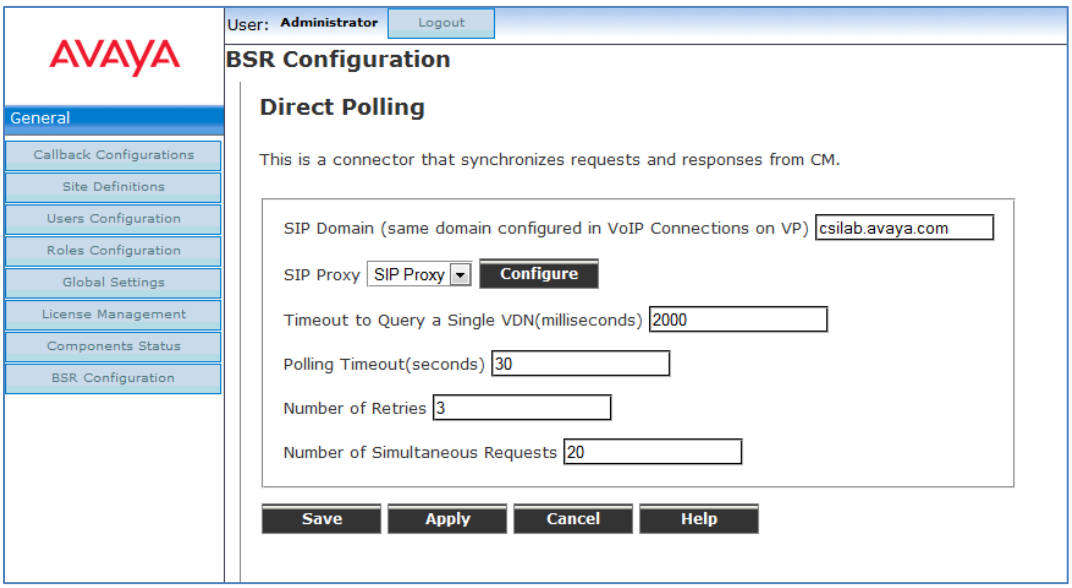


Figure 63 - BSR Configuration -> Direct Polling

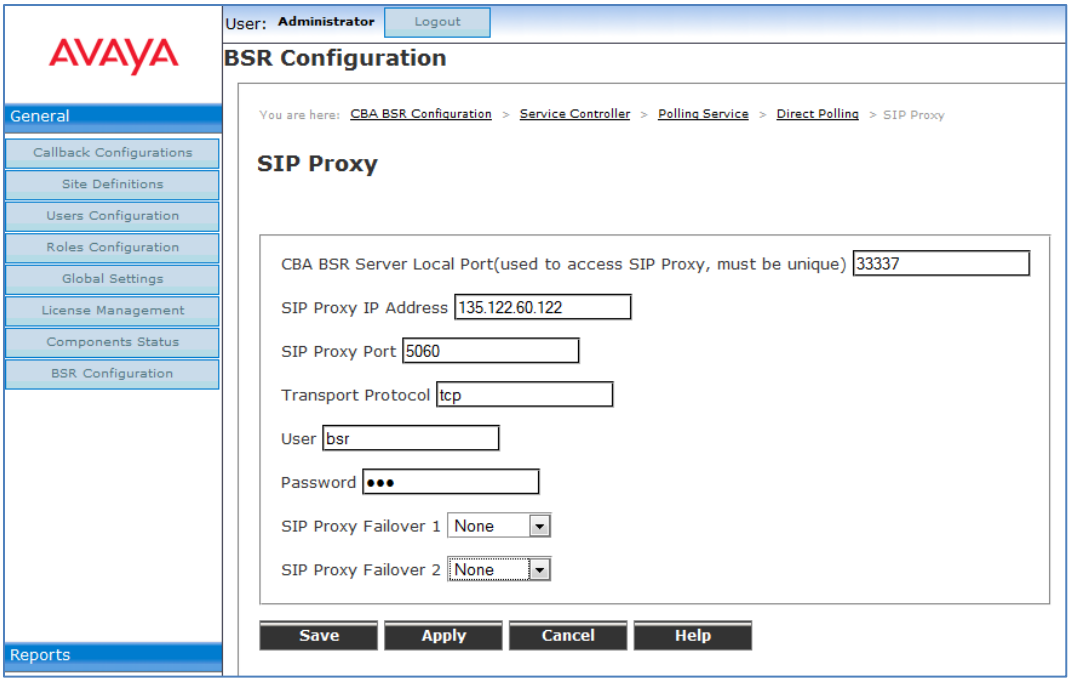


Figure 64 - BSR Configuration -> SIP Proxy

- ***CBA BSR Call Center Applications Configuration***

Go to **CBA BSR Configuration > CBA BSR Call Center Applications Configuration** option.

1. Click on ***Add***.
2. Enter ***Call Center Application ID***.
3. Enter ***Name*** field.
4. Enter brief ***Description***.
5. Enter ***default VDN*** to queue calls when the queuing destination cannot be evaluated.
6. Enter ***CM of default VDN*** from drop-down list.
7. Click on ***Add*** button from VDN List.
8. Enter ***Polling VDN*** previously created on *Creating Polling Vectors and VDNs for Best Service Routing (BSR)* section.
9. Enter ***Queuing VDN*** previously created on *Creating Polling Vectors and VDNs for Best Service Routing (BSR)* section.
10. Enter ***Adjust By*** value in seconds.
11. Enter a brief ***Description***.
12. Select ***CM*** from drop-down list.
13. Select ***Business Hours Group*** from drop-down list. By default, only "Full 24 hour Week" is available. Create others if needed on *Business Hours and Holidays Configuration* menu.
14. Select ***Holiday Group*** from drop-down list. By default, only "No Holidays" is available. Create others if needed on *Business Hours and Holidays Configuration* menu.
15. Click on ***Save***.
16. Repeat steps 7 to 15 for additional polling/queuing VDNs to be used.
17. Click on ***Save***.
18. Repeat same operation for additional Call Center Applications needed.

Configuring Site Definitions

The option "Site Definitions" would be enabled if the CBA installation has been done with site being enabled. The Site Definitions are required to be configured if using just one Experience Portal platform or if also having several ones. Below steps show minimal configuration for one Site only.

Go to Site Definitions and follow these steps:

1. Edit existing default site definition by clicking on pencil icon.
2. A window with site name text box and a button “Add New” for adding primary/auxiliary EPM will appear.
3. Optionally change ***Name*** field and it should be unique.
4. Click on “Add New” button to add primary EPM.
5. Enter ***Outbound Web Service IP Address*** of Primary EPM field.
6. Enter ***Outbound Web Service User*** defined on EPM. The User parameter must match the Outcall credentials administered on EPM Settings on Experience Portal Web Portal.
7. Enter ***Outbound Web Service Password*** defined on EPM. The User parameter must match the Outcall credentials administered on EPM Settings on Experience Portal Web Portal.
8. Click on OK button to save primary EPM detail
9. Optionally, if auxiliary EPMs are present, add the multiple auxiliary EPMs by clicking on “Add New” button and enter ***outbound web service IP Address, user name*** and ***password***.
10. Click on ***OK*** button to save the auxiliary EPM detail.

Configuring Zone Definitions

The option “Zone Definitions” would be enabled if the CBA installation has been done with zone being enabled. The Zone Definitions are required to be configured if using just one Experience Portal platform or if also having several ones. The primary EPM detail of the default zone would have been given during the CBA installation itself. Below steps show minimal configuration to add auxiliary EPMS for a Zone.

Go to Zone Definitions and follow these steps:

1. Edit existing default zone definition by clicking on pencil icon.
2. A window with zone name text box and a button “Add New” for adding auxiliary EPM will appear.
3. Select zone name from zone name drop down.
4. Click on “Add New” button to add auxiliary EPM.
5. Enter ***Outbound Web Service IP Address*** of auxiliary EPM field.
6. Enter ***Outbound Web Service User*** defined on EPM. The User parameter must match the Outcall credentials administered on EPM Settings on Experience Portal Web Portal.
7. Enter ***Outbound Web Service Password*** defined on EPM. The User parameter must match the Outcall credentials administered on EPM Settings on Experience Portal Web Portal.
8. Click on OK button to save auxiliary EPM detail
9. Optionally, if multiple auxiliary EPMs are present, add those by clicking on “Add New” button and enter the ***outbound web service IP Address, user name*** and ***password*** and click on OK button to save it.

Creating a new Callback Configuration

The following steps shows minimum required fields to create a Voice Configuration for Agent First Strategy. Additional fields may be needed based on implementation requirements; if that’s the case see *Administering Callback Assist Guide* for further details.

Go to Callback Configurations and follow these steps:

- Click on ***Add New***.
- Select ***Voice*** configuration type and click on ***Next…***. Other types depend on the license features available.
- The ***Add Voice Callback Configuration*** window is displayed.
- Click on ***General Tab*** and, as minimum, configure required fields shown below:
  - ***i. Name:*** a simple name for this configuration
  - *ii.* ***DNIS:*** Initial VDN (Initial Vector Directory Number), Service Number, or other route for receiving calls.
  - iii. ***Agent Queue:*** Select Call Center Application from drop-down list.
  - iv. ***Minimum EWT Threshold***
  - v. ***Maximum EWT Threshold***
- Click on Availability Tab and configure required fields:
  - i. ***Time Cushion:*** This is a time space between the call and the first schedule slot to be offered
  - ii. ***Number of Offered Slots:*** Number of callback timeslots to be offered to the callers.

iii. ***Time Zone Message:*** specify the time zone message to be played while offering schedule callback slots.
iv. Select current day of week and configure, as minimum, current time of day as an available slot.

- Click on ***Customer Tab*** and, as minimum, configure required fields shown below:
  i. ***Welcome Message:*** specify the first message to be played to the caller.
  ii. ***Goodbye Message:*** specify the message when caller leaves gracefully.
  iii. ***Customer WTA Message:*** specify the message to repeat when caller is waiting on queue.

  Note: *There are sample messages located on Callback Assist installation directory under /opt/Avaya/callbackassist/apache-tomcat-ddapps/webapps/CBAPhrases/samples/*

- Click on ***Agent Tab***
  i. ***Agent WTA Message:*** specify the message to play to agent when outbound call is in progress.

  Note: *There are sample messages located on Callback Assist installation directory under /opt/Avaya/callbackassist/apache-tomcat-ddapps/webapps/CBAPhrases/samples/*

- Click on ***Voice Questions Tab*** and configure the following Phone Number question fields:
  i. ***Question Message:*** message to request phone number to caller.
  ii. ***Agent Message:*** message to playback phone number to Agent.
  iii. ***No Input Message:*** message to play when caller does not enter any value.
  iv. ***No Match Message:*** message to play when caller enters invalid value.
  v. Click ***OK*** on Question window.

  Note: There are sample messages located on Callback Assist installation directory under /opt/Avaya/callbackassist/apache-tomcat-ddapps/webapps/CBAPhrases/samples/

- Click OK button on ***Add Voice Callback Configuration*** window.

Configuring Avaya Orchestration Designer

Access the URL **http://<callback-application-server-hostname>:8080/runtimeconfig/** and logon with user/password = ddadmin/ddadmin, as shown below:

Figure 65 - Avaya Orchestration Designer Login screen

After logon, the system displays the following page:

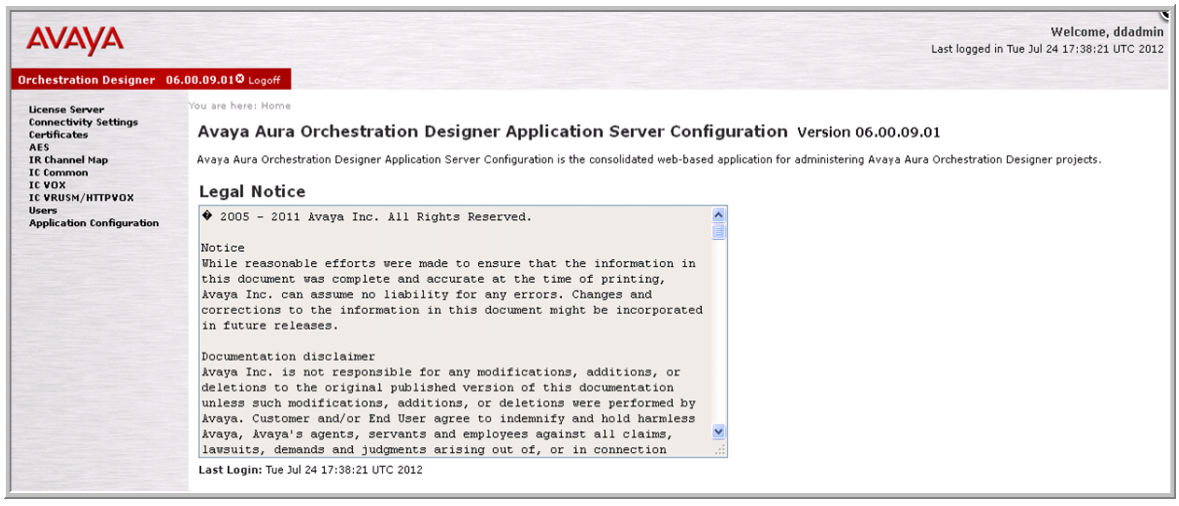


Figure 66 - Avaya Orchestration Designer page after Login

Click on the **License Server** link on the left navigation pane, and enter the URL for license server to configure the Licensing Server URL as follows:

Here you can use the Local WebLM Server (i.e., shipped with the CBA) or external WebLM Server.

Figure 67 - Configuration of Licensing Server

If the URL is correct the system displays a page similar to the following:

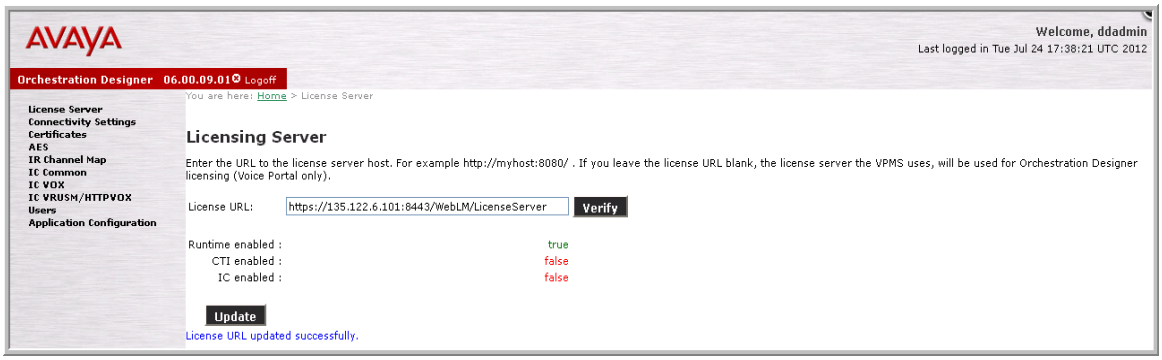
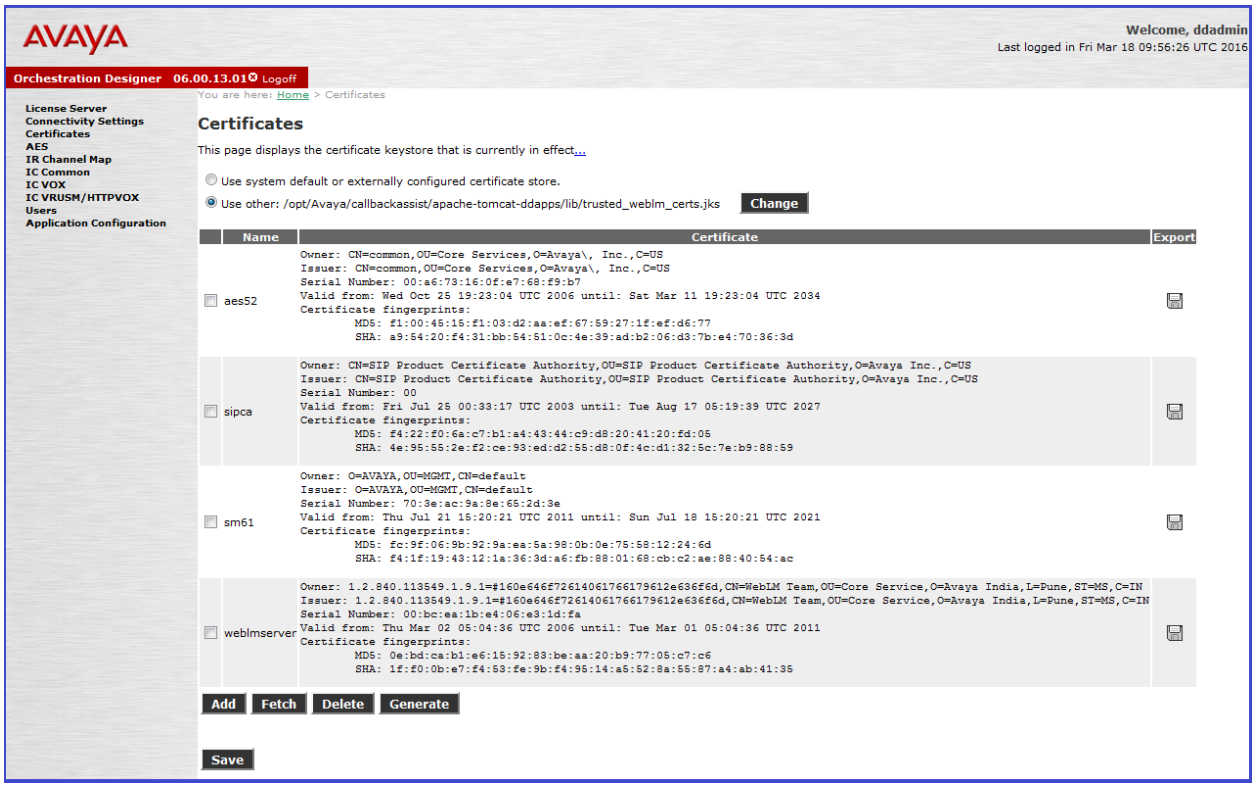


Figure 68 - Details of Configured Licensing Server

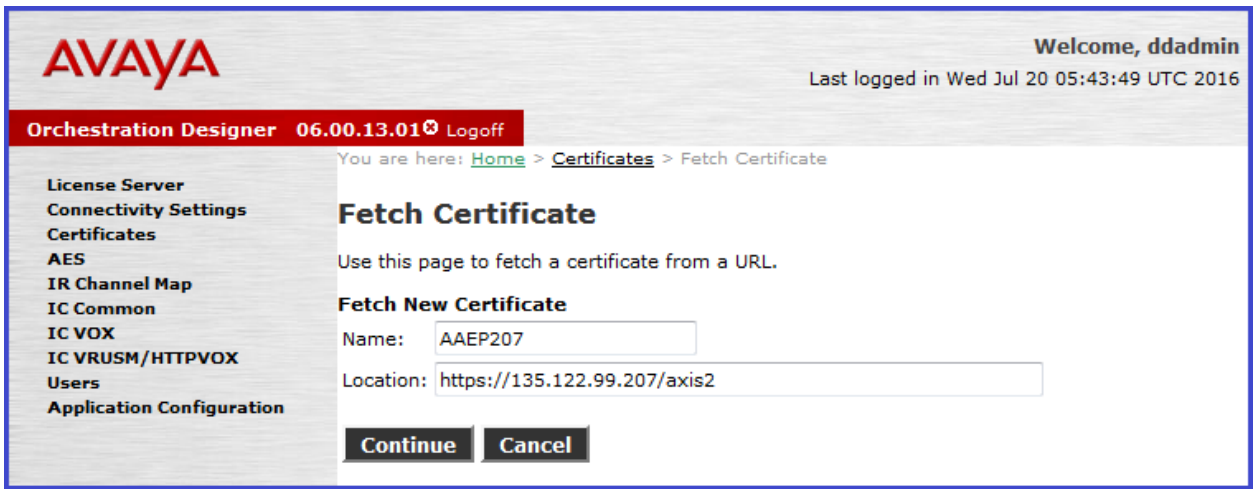
Add AAEP Security Certificate to CBA

Callback Assist in SIP/AACC platform uses axis2 based Application Interface web service of AAEP to place the outbound calls. The communication between CBA components and the Application Interface web service of AAEP is SSL encrypted. This requires the EPM axis2 security certificate of all the EPMs added to CBA. CBA uses the OD runtimeconfig application to manage these certificates. The default keystore file path in the runtimeconfig web application should not be changed. Follow the steps below to add the certificates.

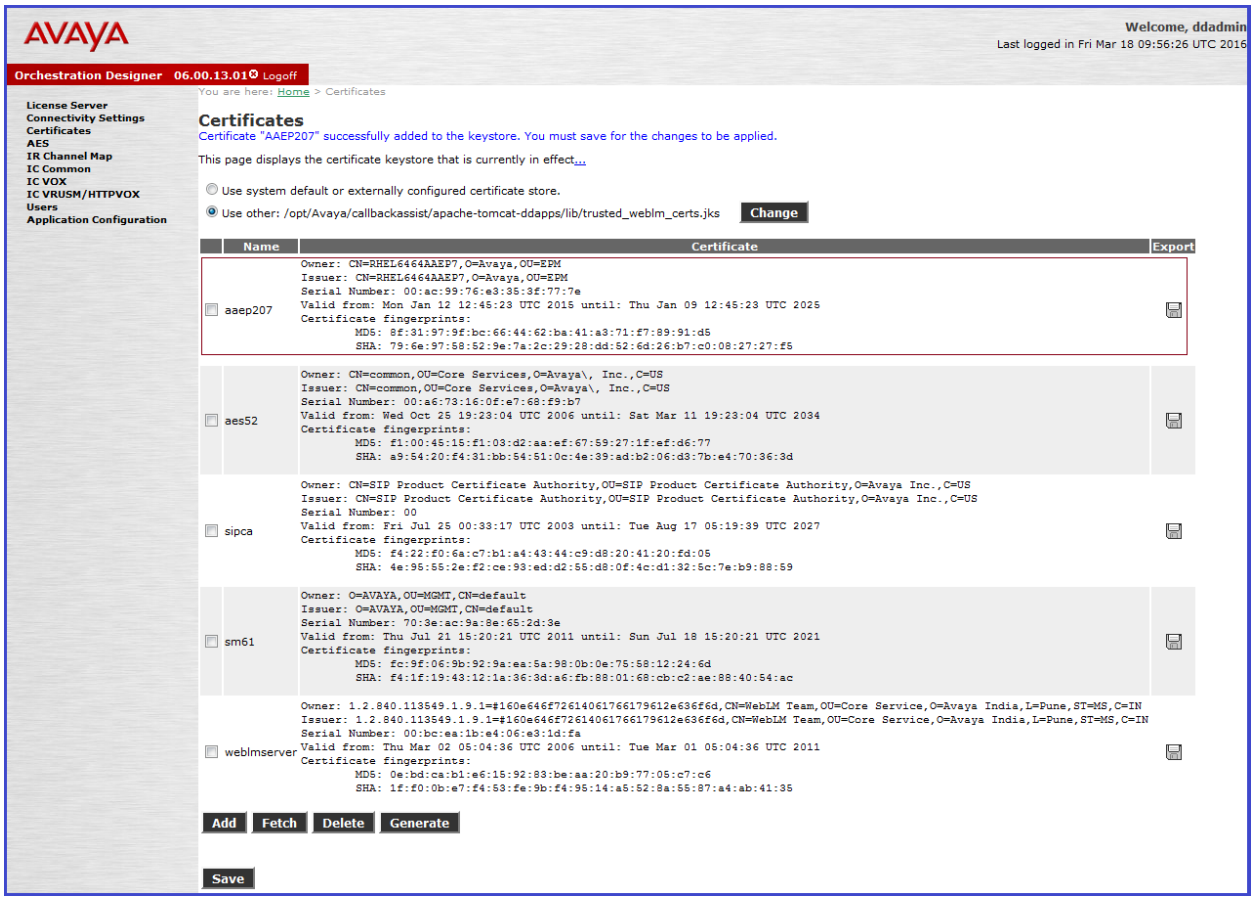
1. In the Avaya Orchestration Designer runtimeconfig application, click on the Certificates link on the left navigation pane.



2. Click on the **Fetch** button and enter the URL https://<EPM IP>/axis2 in the **Location** field with a valid name and click continue.



3. The new certificate gets displayed in the Certificates page. Click the Save button at the bottom of the page to permanently add the certificate to the keystore file



4. Repeat the above steps for all the EPMs configured in the Site Definitions/ Zone Definitions page of Callback Assist Administration UI.

5. Restart Callback Assist for the new certificate(s) to take effect.

Note: On HA deployments certificates should be administered in all the CBA Server Nodes.

Installing Callback Assist in an AACC environment

In an Avaya Aura® Contact Center (AACC) with Avaya Communication Manager (CM) environment, Avaya Callback Assist interacts with Avaya Aura ® Experience Portal (AAEP) placed behind an AACC with CM. That is, the customer call will arrive and based on the routing configuration in AACC, the system offers a callback option to the customer. If the customer accepts the callback option, then AACC will route the call to CBA.

There is a shared responsibility in CBA management in an AACC environment as follows:

- All estimated waiting time management and initial callback offering will be performed by AACC.
- Callback type selection, information collection, and delivery will be handled by CBA.

A basic call flow is as follows:

1. A call arrives at Avaya Aura Session Manager and it is routed to AACC.
2. AACC detects that it has to employ a CBA-enabled script to handle the call.
3. The given script, based on the EWT of the skillset queue, decides to offer the customer the option to be called back at a later time.
4. If the customer accepts the callback option, the AACC script routes the call to CBA.
5. CBA receives the customer call and starts interacting with the customer. CBA will perform the following based on the proper configuration:
    1. Offer an immediate or a scheduled callback.
    2. Ask for the customer information.
6. After the customer completes the callback scheduling, the system will end the call.
7. Based on the scheduled date and time to call the customer, CBA will queue a call in the skillset through the configured CDN.
8. When the agent answers the call, and if the CBA configuration allows it, the customer information will be played to the agent.
9. After the agent decides to accept the callback request, CBA will call the customer.
10. If the customer answers the call, the agent and customer calls will be merged and CBA will release control of the calls.
11. If the customer call was not successful, the agent will be informed of such event and the callback request will be scheduled for a later time, until the maximum number of attempts is reached.

This section guides you through the steps to configure different applications to install Callback Assist, including the following:

- Setting up an entity in Avaya Aura® System Manager
- Setting up a Dial Pattern (route) to AAEP in System Manager
- Setting up the SIP trunk in Avaya Aura® Experience Portal

- Setting up inbound application in Avaya Aura® Experience Portal
- Configuring AACC

Checklist for setting up the system for Callback Assist

Use the following checklist to plan the activities to set up AACC/CM to work with Avaya Callback Assist installation.

#	Activity	Description/Reference	Completed?
1	Check the licensing requirements. Note: If you are using the Call Classification feature, then make sure that that AAEP license includes the Call Classification feature.	For more information, see the *Licensing* section in the *Avaya Callback Assist overview and planning guide*.	
2	Check the hardware requirements.	For more information, see the Hardware requirements section in this document.	
3	Check the software requirements.	For more information, see the Software requirements for SIP environment section in this document.	
4	Install the required Linux OS in the Callback Assist server.		
5	Check the network considerations (firewall).	For more information, see the Network considerations section in this document.	
6	Configure the Avaya Aura® Experience Portal applications or Voice Portal applications (Callback Customer Voice Application and Callback Agent Voice Application).		
7	Configure Avaya Orchestration Designer for the licensing server.		

Setting up a Dial Pattern (route) to AAEP in System Manager

To configure a route to direct calls to the AAEP/AVP, perform the following tasks:

- On the System Manager, go to **Routing**> **Dial Patterns**.
- Create a new dial pattern by configuring the values for **Pattern**, **Min**, and **Max** fields based on the specific deployment requirements.
  The following image shows an example of routing, though routing always depends on the local Dial plan.

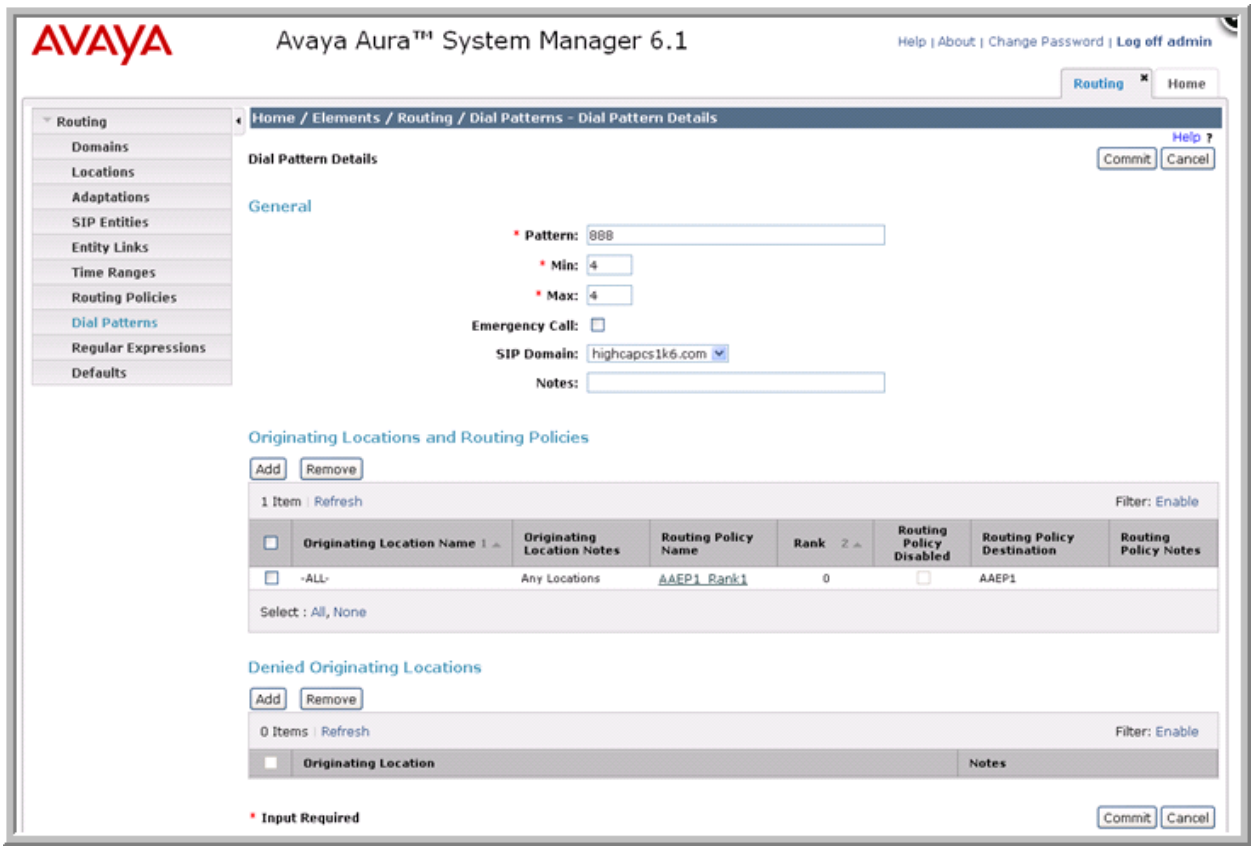


Figure 69 - Setting up a route in System Manager

Setting up the SIP trunk in Avaya Aura® Experience Portal

In the EPM web application:

1. Go to **System Configuration** > **VoIP Connections**.
2. Click on the **SIP** tab.
3. Click **Add** to add the new SIP trunk.

4. Enter a name for the new trunk.
5. Set “Enable” to "Yes".
6. Enter the IP Address and SIP port of the NRS.
7. Enter the port where the AVP will listen for SIP connections.
8. Enter the SIP Domain to use.
9. Configure the rest of the fields as needed by the specific deployment.
10. Save the configuration.

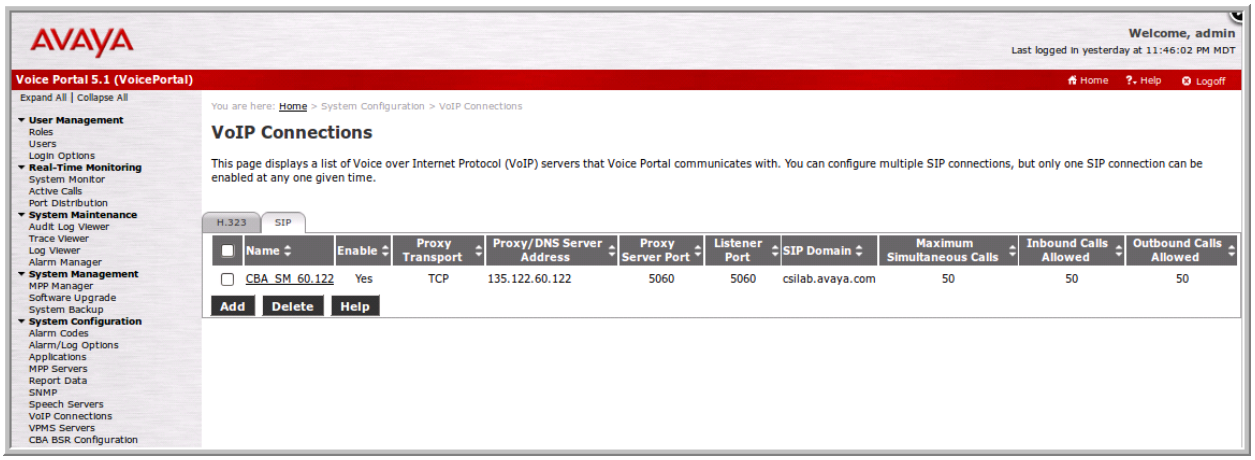


Figure 70 - Setting up SIP trunk in Experience Portal

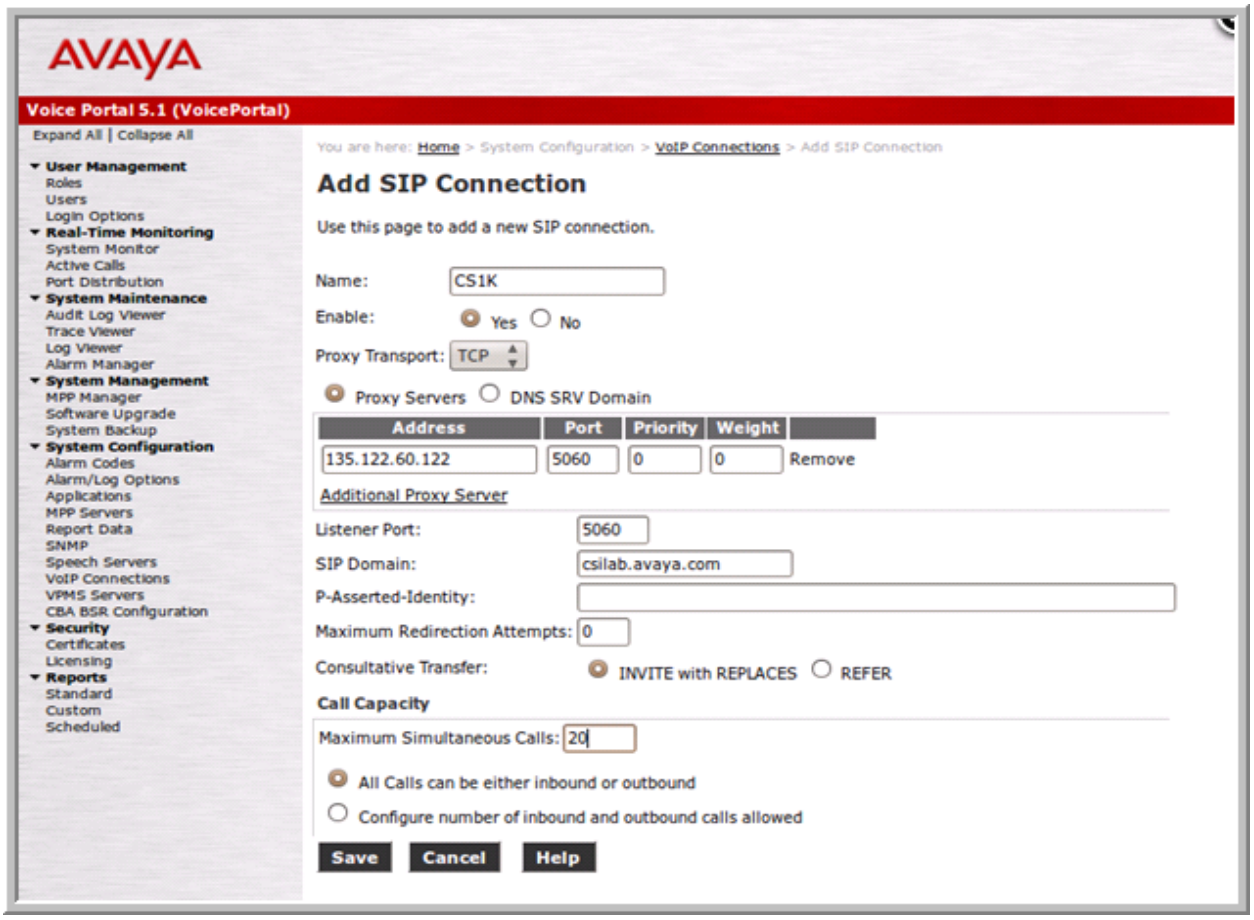


Figure 71 - Setting up SIP trunk in Experience Portal

Setting up inbound application in Avaya Aura® Experience Portal

In the EPM/VPMS web application:

1. Go to **System Configuration** > **Applications**.
2. On the **Add Application** page, specifying the required name to add a new application for the CBA AACC inbound connection.
3. Set "**Enable"** to "**Yes".**
4. Under **URL**, select the **Single** as the required option.
5. Add the following as CCXML URL:
   http://<CBA server IP>:8080/CBAScripts/cbaCallControl
6. Click **Verify**.
7. Under **Application Launch**, select **Inbound** as the required option.
8. Select the **Number** option and add the required DNIS, then click **Add**.
   Note: You must add at least one number.

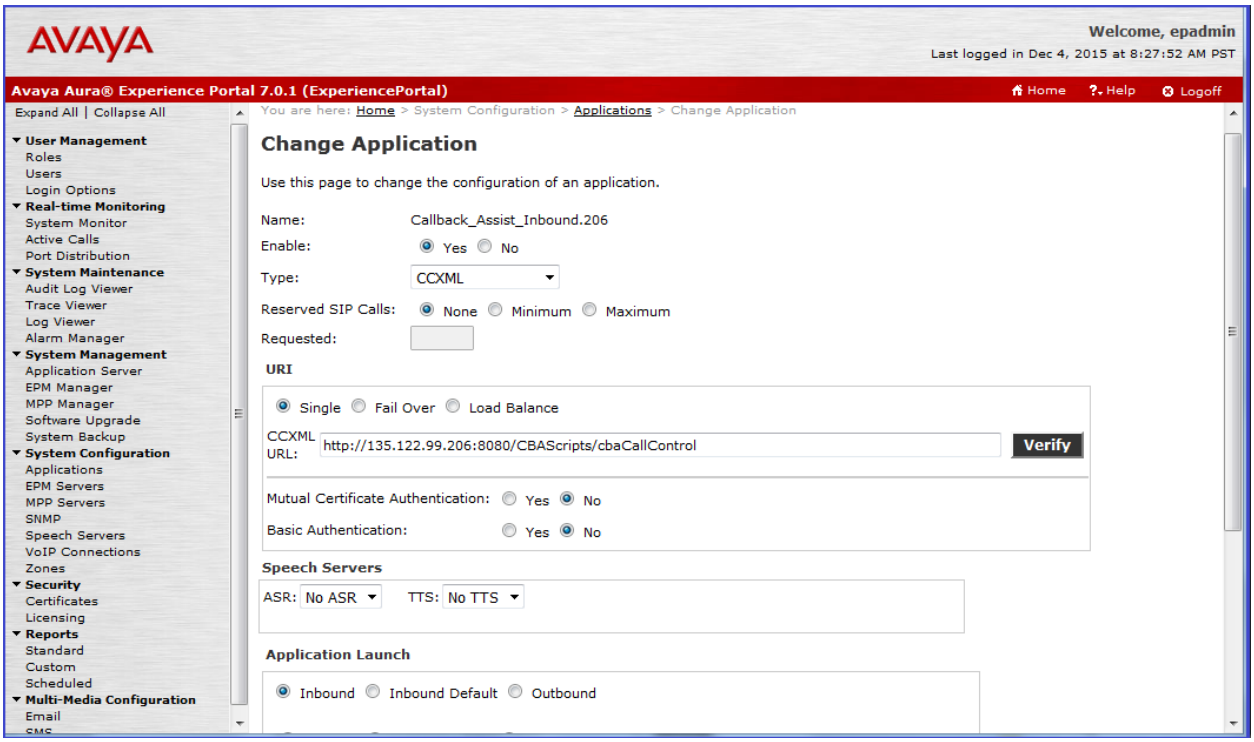


Figure 72 - Adding a new Inbound Application

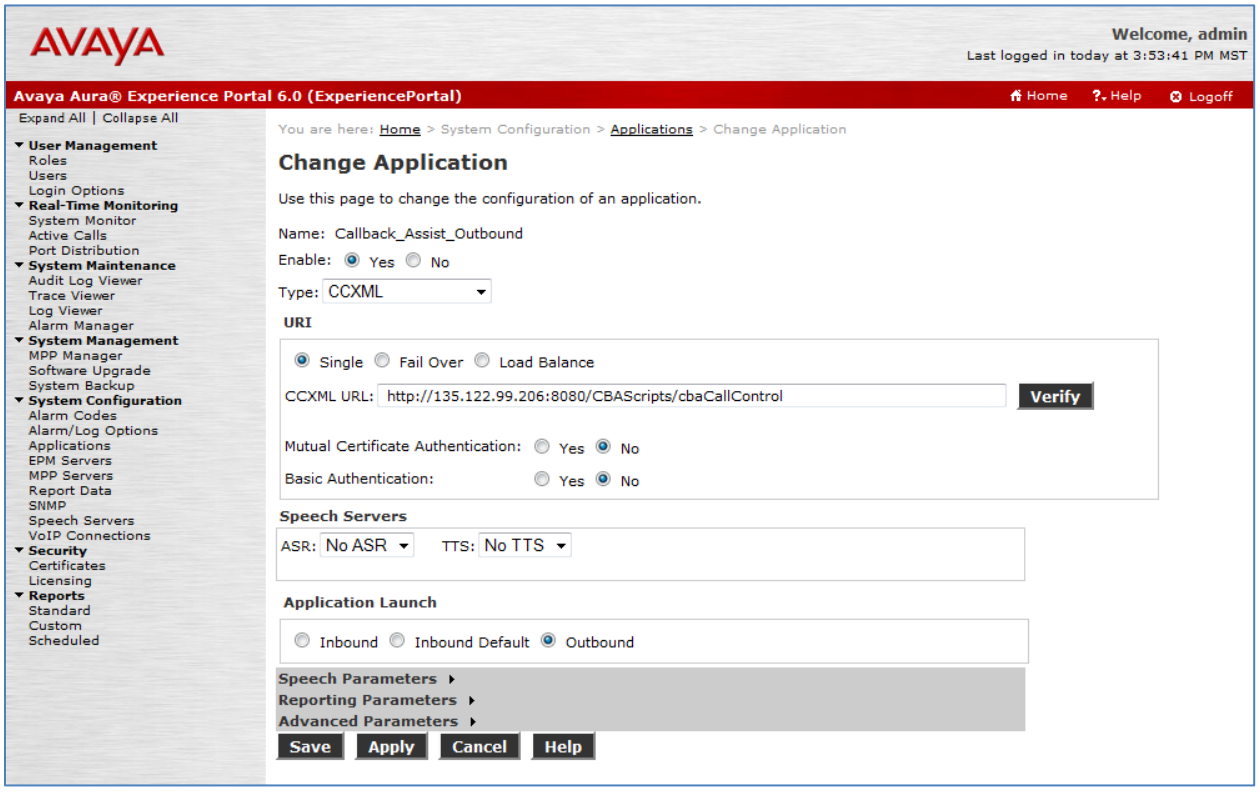
Setting up Outbound Application

1. On the Web browser, enter the Experience Portal EPM Administration URL https://<FQDN>/VoicePortal
2. In the ***User Name*** field, enter the user name.
3. Click **Submit**.
4. In the ***Password*** field, enter the password.
5. Click ***Logon***.
6. From the menu select ***System Configuration -> Applications***
7. Click on ***Add.***
8. On the Add application window, configure the following details:

Field	Value
Name	Callback_Assist_Outbound **Note:** The name of the CCXML application must exactly match the name configured in the Callback Assist Web administration in the Gobal Settings *EPM Callback Outbound application name*

Enable	Yes
Type	CCXML
CCXML URL	http://CBA_FQDN:8080/CBAScripts/cbaCallControl
Speech Servers - ASR	No ASR
Speech Servers - TTS	If TTS is available, it can be used with CBA, otherwise just select No TTS.
Application Launch	Outbound
Advanced -> Operation Mode	MUST match the same one used for Inbound application configured above.

9. Click on *Save.* See configured application sample below:



Configuring AACC

Create SCE scripts in the AACC that would interact with the caller, route the call to Callback Assist and in a subsequent phase to queue a call to the appropriate skillset on behalf of Callback Assist.

Sample SCE Scripts

A sample SCE script to receive a caller. This script will perform the following tasks:

- Welcome a caller.
- Inform the caller the estimated waiting time of the agent queue.
- Allow the caller to choose to wait in queue or arrange a callback.
- If the caller selects to keep waiting, queue the call and just play a music on hold until an agent is available.
- If the caller selects to arrange a callback, then route the call to CBA.

```
ASSIGN 0 TO Ewait_time_cv

QUEUE TO SKILLSET Prov_Rel_sk
WAIT 2

GIVE RAN 10  /* Route number of initial_greeting */

/* Check Estimated Wait Time */
ASSIGN EXPECTED WAIT TIME Prov_Rel_sk TO Ewait_time_cv /* Estimate wait
time in seconds */
ASSIGN (EWait_time_cv / 60) TO EWait_time_minutes_cv /* Convert
estimate wait time into minutes */

/* If EWT is over 10mins (or some threshold) then present the options
*/
IF EWait_time_minutes_cv > 10 THEN
      /* Play current estimate wait time */
      ASSIGN " ewt_is+%i0" TO prompttoplay
      ASSIGN EWait_time_minutes_cv TO vars
      GIVE IVR WITH VXML TREATMENT voicexml PARAMETERS prompttoplay,
vars

      /* Play & Collect IVR with menu options:
                                    1 - for callback
                                    2 - to continue waiting in queue */
      ASSIGN "cba_offer_menu " TO prompttoplay
      ASSIGN "PlayAndCollect.vxml" TO voicexml
      ASSIGN "1" TO numberofdigits
      ASSIGN -1 TO CC_ConfirmDigits
```

```
      GIVE IVR WITH VXML TREATMENT voicexml PARAMETERS prompttoplay,
numberofdigits RETURNS CC_ConfirmDigits

      IF CC_ConfirmDigits = 1 THEN
            ROUTE CALL 123456 /* Route the call to CBA */
      ELSE
            EXECUTE wait_loop /* Wait in the queue for next available
agent */
      END IF
ELSE
      /* EWT is less than the threshold, wait in the queue for next
available agent */
      EXECUTE wait_loop
END IF

SECTION wait_loop
      GIVE RAN 11 /* Route number of comfort message */
      WAIT 2
      GIVE MUSIC 111 /* Route number for music */
      WAIT 10
      EXECUTE wait_loop
```

Sample SCE script to queue a call to a skillset

The following script will place a call to a skillset in queue, without offering any wait treatment:

```
/* callback_assist_queue */
IF NOT OUT OF SERVICE CallBack_Skillset THEN
QUEUE TO SKILLSET CallBack_Skillset WITH PRIORITY 4
WAIT 2
ELSE
DISCONNECT
END IF

/* Loop to get SIP 183 Session Progress messages on AAEP */
SECTION LOOP_SILENCE
    GIVE SILENCE
    wait 60
    GIVE RINGBACK
    IF NOT QUEUED THEN
        DISCONNECT
    ELSE
        EXECUTE LOOP_SILENCE
    END IF
```

Installing Callback Assist Software

You can choose to install Callback Assist in one of the following modes based on your requirements:

- Callback Assist Single Server deployment (Core Components & DB)
  Installs CBA in a single box, that is, both Callback Assist Core Components and PostgreSQL Database Server will be installed in the same server.

The Avaya Callback Assist Installation is based on Linux shell scripts. The installer of Avaya Callback Assist is a single tar file whose name has the form callbackassist-<version>.tar. All Avaya Callback Assist components run as Linux daemons registered under /etc/init.d.

The callback-install.sh script installs all the available components in the server where the script is run (including the PostgreSQL DBMS), and prepares most of the configuration files automatically.

Software Installation Steps

1. On a Linux shell logged on as root, go to the directory where the CBA installer file is located.
2. Run the following command to extract the files to a temporary folder in your system:
   `tar -xvf callbackassist-<version>.tar`
   One of the extracted file is callback-install.sh which is the main installation script.
3. Run the following command to start the installation process:
   `./callback-install.sh`
   The system writes the command output of the scripts to the standard output and to the callback-installation.log file in your system. Then the system prompts you to choose the required platform for CBA installation.
4. Select **[3] Callback Assist Single Server Deployment (Core Components & DB)** as the installation mode.
5. Enter the installation directory where to install Callback Assist or just press **Enter** to use default location (default directory "/opt").
6. Select **[3] AACC/CM (SIP Based)** as the platform type from the following options and press Enter.
   1. CTI (H.323)
   2. SIP
   3. AACC/CM (SIP Based)

The system starts installing SIP platform with the desired strategy. After completing the installation, the system displays an installation successful message.

The following image shows the output of the *callback-install.sh* script for a Callback Assist installation in AACC environment:

```
[root@RHEL60EPAEP r44368]# ./callback-install.sh


*******************************************************
Avaya Callback Assist Installer
*******************************************************


(09:06:36) [ No Callback Assist Components were found. ]


(09:06:36) Please choose the Components to be Installed on this server:
1) Callback Assist Core Components with external Database
2) PostgreSQL Database Server for Callback Assist
3) Callback Assist Single Server deployment (Core Components & DB)
#? 3


(09:06:50) [ About to Install Callback Assist Single Server deployment (Core Components & DB) ]
Please, enter the base directory where Avaya Callback Assist will be installed (default directory /opt):
Directory where Avaya Callback Assist is going to be installed: /opt/Avaya/callbackassist
(09:06:55) Checking if there is enough disk space...
(09:06:55) Available disk space is enough.


(09:06:55) Please choose a Platform type for Callback Assist Application to be Installed from the options below:
1) CTI (H.323)
2) SIP
3) AACC/CM (SIP Based)
#? 3


(09:06:57) [ AACC/CM (SIP Based) Platform has been selected for installation. ]
(09:06:58) Unpackaging distribution file callbackassist.package...
(09:07:05) Creating callback Group...
```

```
(09:07:05) Creating callback User...
(09:07:06) 32 bit Architecture detected ...
(09:07:06) Using 32 bit PostgreSQL Installer...
(09:07:06) Installing PostgreSQL Server, this step may take several minutes...
(09:07:32) Creating 'callback' PostgreSQL user...
(09:07:32) Creating 'callback' database...
(09:07:32) 'callback' database created.
server signaled
(09:07:32) Restarting database to reset max_connections
Restarting PostgreSQL 9.4:
waiting for server to shut down.... done
server stopped
waiting for server to start.... done
server started
PostgreSQL 9.4 restarted successfully
(09:07:34) Signaling Postgresql postmaster...
(09:07:34) Done.
(09:07:34) Setting ownership to callback user.
(09:07:34) Performing JDK silent install...
(09:07:37) JDK installed.
java version "1.7.0_75"
Java(TM) SE Runtime Environment (build 1.7.0_75-b13)
Java HotSpot(TM) Client VM (build 24.75-b04, mixed mode)
(09:07:37) Installation of Callback Assist Maintenance done.
(09:07:37) Check /opt/Avaya/callbackassist/maintenance/logs/execution.log file for Callback Assist
Maintenance service startup details.
(09:07:37) Installation of Callback Assist Engine done.
(09:07:37) Check /opt/Avaya/callbackassist/engine/logs/execution.log file for Callback Assist Engine service
startup details.
(09:07:37) 32 bit Architecture detected ...
(09:07:37) Installing file server for Red Hat 6.x 32 bits
(09:07:38) Local Ip Address: 135.122.99.108
```

```
(09:07:38) INFO: Using hostname/FQDN: server108.avayacba.com to run the file-server.
(09:07:38) Configuring File Server IP Address...
(09:07:38) Done.
(09:07:38) Changing SELinux context to some CBA files...
(09:07:38) Waiting for the Database Schema to be created or updated...
Migration successful
(09:07:43) Database Schema was successfully created or updated.
(09:07:43) Platform successfully set.
(09:07:43) Storing Release version and Build Number...
(09:07:43) Release version and Build number successfully stored.
(09:07:43) Running Platform dependant changes...
Migration successful
(09:07:44) Database Schema was successfully updated.
(09:07:44) Installing weblm Tomcat service (tomcat-weblm)...
(09:07:44) Installation of tomcat-weblm done.
(09:07:44) Installing adminapp Tomcat service (tomcat-adminapp)...
(09:07:44) Installation of tomcat-adminapp done.
(09:07:44) Installing ddapps Tomcat service (tomcat-ddapps)...
(09:07:44) Installation of tomcat-ddapps done.
(09:07:44) Installing webcallback Tomcat service (tomcat-webcallback)...
(09:07:44) Installation of tomcat-webcallback done.
(09:07:44) Deploying Applications...
(09:07:44) Moving tomcat realm related jar files
(09:07:44) Installation of Tomcat instances done.
(09:07:44) Setting ownership to callback user.
Starting Callback Assist Engine...
Callback Assist Engine Started.                    [  OK  ]
Callback Assist Engine ( pid 13998 ) is running...
Starting Callback Assist Maintenance...
Callback Assist Maintenance Started.               [  OK  ]
Callback Assist Maintenance ( pid 14064 ) is running...
Starting tomcat-weblm...                           [  OK  ]
```

```
tomcat-weblm ( pid 14147 ) is running...
Starting tomcat-adminapp...                    [  OK  ]
tomcat-adminapp ( pid 14230 ) is running...
Starting tomcat-webcallback...                 [  OK  ]
tomcat-webcallback ( pid 14319 ) is running...
Starting tomcat-ddapps...                      [  OK  ]
tomcat-ddapps ( pid 14427 ) is running...
Starting Callback Assist File Server...
Callback Assist File Server Started.           [  OK  ]
Callback Assist File Server ( 14541 ) is running...    [  OK  ]
Callback Assist File Server Ping test          [  OK  ]
Callback Assist File Server Read/Write cycle Test      [  OK  ]
Callback Assist File Server Ring Status is Up          [  OK  ]
Callback Assist File Server is joined to a cluster     [  NO  ]
If this is not a HA deployment, then disregard this warning.
(09:08:11) [ Installation of Avaya Callback Assist (AACC/CM (SIP Based)) completed. ]


*********************************************************
ACTION REQUIRED on Time Zone Configuration
*********************************************************
The default Time Zone of CBA is UTC. If your system requires a different Time Zone
you must manually configure it in the Global Settings of the Admin Portal.
*********************************************************
```

Managing status of Callback services

You can start, stop, and find status of all the callback processes by using the service System V init script as follows:

`service <service_name> {start | stop | status}`
where <service_name> is the name of a CBA process.

Service names of Callback Assist processes or components are as follows:

On CBA Linux server:

- cba-postgresql (Callback database)
- cbamaint (Callback database Maintenance processes)
- tomcat-adminapp (Callback Admin Web Tomcat Application Server)
- tomcat-ddapps (VoiceXML Tomcat Application Server)
- tomcat-webcallback (Webservices Tomcat Application Server)
- cbaengine (Callback Engine)
- cba-fileserver (Callback File Server)

To query the status of Callback Assist database, use the following trace:

```
# service cba-postgresql status
```

To start the Tomcat Application Server where the Orchestration Designer applications are hosted, use the following trace:

```
# service tomcat-ddapps start
```

To stop the Callback Engine, use the following trace:

```
# service cbaengine stop
```

Also, there is a support script which can be used to start or stop all the CBA services at once.
This script is located at: */opt/Avaya/callbackassist/support/callbackservices.sh*, assuming that CBA is installed at the default location */opt/Avaya/callbackassist*.
Use the script as follows:

```
# /opt/Avaya/callbackassist/support/callbackservices.sh
{start|stop|restart|status}
```

For example, to query the status of Callback Assist components:

```
[root@RHEL6232CBA support]# ./callbackservices.sh status
pg_ctl: server is running (PID: 32649)
/opt/Avaya/callbackassist/PostgreSQL/9.1/bin/postgres "-D"
"/opt/Avaya/callbackassist/PostgreSQL/9.1/data"
Callback Assist Maintenance ( pid 702 ) is running...
Callback Assist Engine ( pid 617 ) is running...
tomcat-ddapps ( pid 962 ) is running...
tomcat-adminapp ( pid 781 ) is running...
tomcat-webcallback ( pid 865 ) is running...
Callback Assist File Server ( 1071 ) is running... [ OK ]
Callback Assist File Server Ping test [ OK ]
```

```
Callback Assist File Server Read/Write cycle Test [ OK ]
Callback Assist File Server Ring Status is Up [ OK ]
Callback Assist File Server is joined to a cluster [ NO ]
If this is not a HA deployment, then disregard this warning.
[root@RHEL6232CBA support]#
```

Post-installation steps

This section describes the minimum configuration required on Callback Assist software to start placing calls.

Callback Assist Web Administration

License Installation

For more information, see the License Installation section in this document.

Login to Administration Interface

The Administration interface is accessible from the URL http://<server-hostname>/admin (uses port tcp/80).

Default log in credential is user ***admin*** and password ***123456*** as shown below:

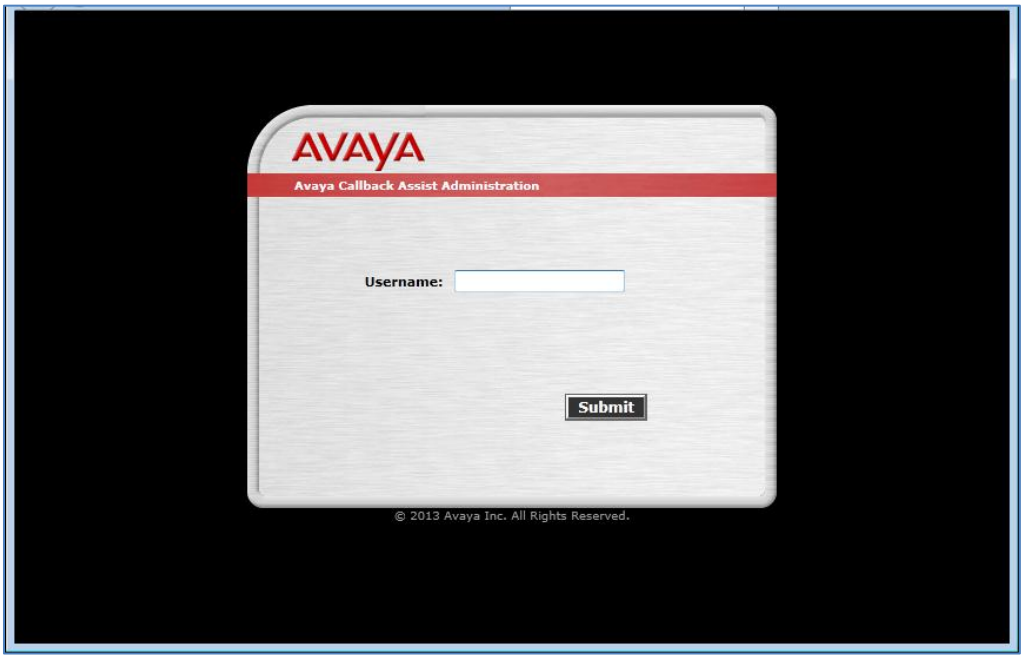


Figure 73 - Avaya Callback Administration Login Screen

After login into the application for the first time the user will be prompted to change the admin user password, as follows:

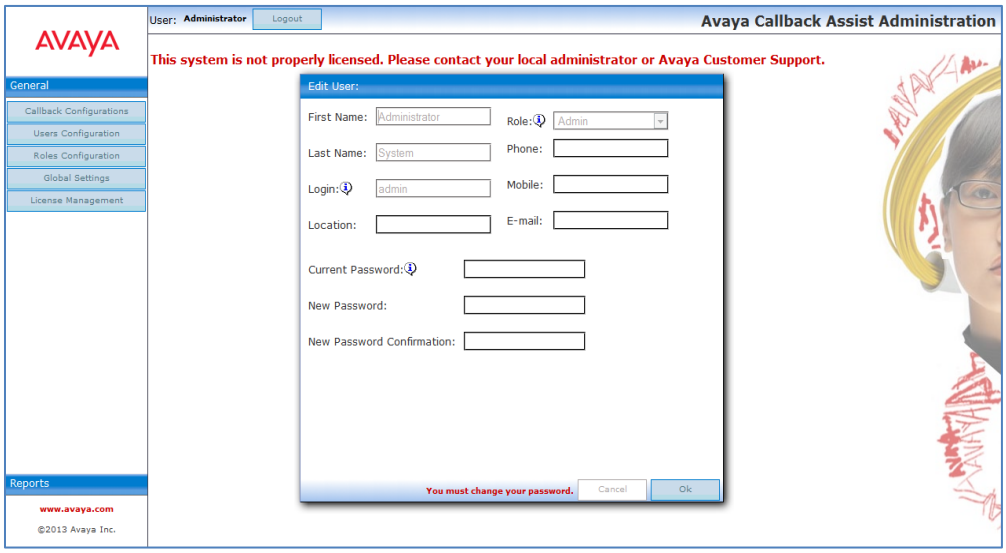


Figure 74 - Avaya Callback Assist – Change Password Screen

Enter new password and retype it on the confirmation field in order to complete the password change procedure.

Configuring license

To configure the WebLM Server where a Callback Assist license file is installed:

1. Go to **License Management** on the left navigation pane.
2. On the **License Management** page, enter the WebLM License Server URL in the following format: https://\[Host\]:\[Port\]/WebLM/LicenseServer.
3. Enter the Dialog Designer WebLM License Server URL in the following format: https://\[Host\]:\[Port\]/WebLM/LicenseServer. The purpose of the Dialog Designer URL here is to display the Dialog Designer license status only and it will not affect CBA operations if not updated
4. Specify the number of CBA ports to be used for the installed platform based on your license.
5. Click the **Update** button to update the **License Management** page with the valid the WebLM URL and port usage information.

Example 1: The **License Management** page looks as follows when the currently installed Callback Assist version does not have a valid license file.

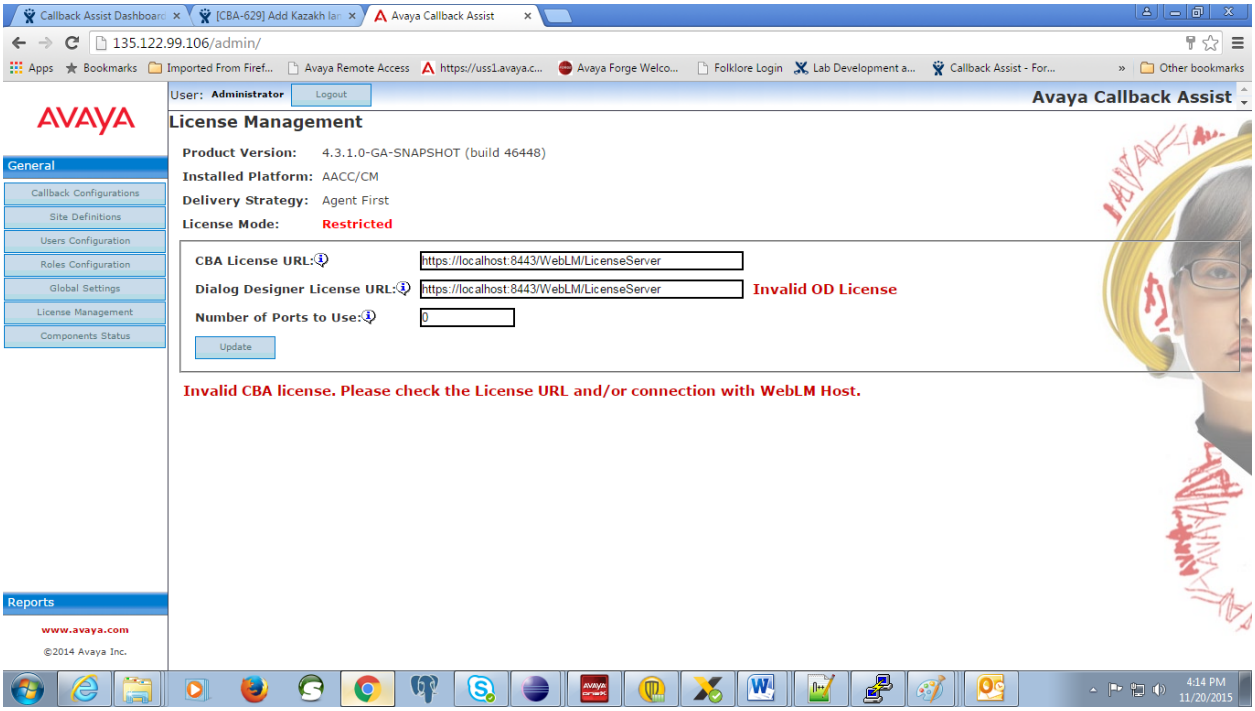


Figure 75 - License Management Screen - Invalid License

Example 2: The **License Management** page looks as follows when the currently installed Callback Assist version has a valid license file.

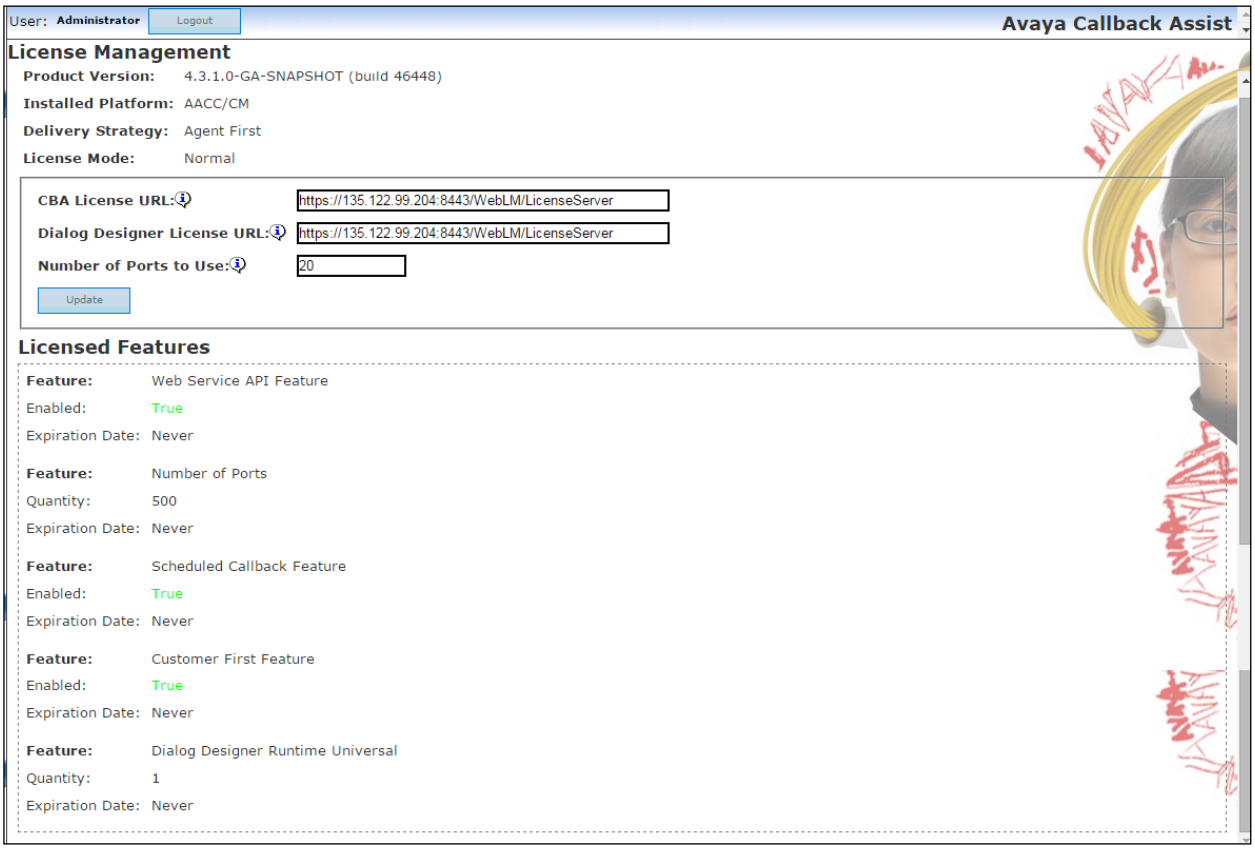


Figure 76 - License Management Screen - Valid License

Adding a custom certificate to connect to the WebLM Server

If the WebLM Server has a certificate file that is not imported in the keystore that CBA uses to validate the WebLM Server, you must add the certificate file to the keystore manually.

1. Save the certificate file in the DER format in your local machine by performing the following steps:
   Note: The procedure to add a certificate might vary based on your Web browser. The following procedure is for Internet Explorer 7.x.
   a. Open your Web browser and type the required WebLM License Server URL.
   b. Right click on the Web page and select **Properties** from the shortcut menu.
   c. On the **Properties** dialog box, click **Certificates**.
   d. Click **Details**.
   e. Click **Copy to File**.
   f. On the **Certificate Export Wizard**, under **Export File Format**, select the **DER encoded binary X.509 (.CER)** option.
   g. Click **Next** and then click **OK**.
   The system saves the certificate file with a *.CER* extension in your local machine.
2. Locate the certificate file you have just saved and copy it to the CBA Server.
3. Locate the *trusted_weblm_certs.jks* file in the following locations in the CBA Server:
   - `<CBA_HOME>/engine/lib`
   - `<CBA_HOME>/apache-tomcat-6.0.18-adminapp/webapps/admin/WEB-INF/lib`
   - `<CBA_HOME>/apache-tomcat-6.0.18-ddapps/lib`
   - `<CBA_HOME>/apache-tomcat-6.0.18-webcallback/webapps/webcallback/WEB-INF/lib`
4. Go to each of these locations and then run the following command:
   `<CBA_HOME>/jdk1.6.0_17/bin/keytool -import -alias customweblm -keystore trusted_weblm_certs.jks -file <certificate filename>`

   where <`certificate filename`> is the certificate file that you copied from your machine to the Callback Assist server.
5. Enter `password` when the system prompts you for a password.
   The system prints the certificate information.
6. Enter `yes` when the system asks whether you trust the certificate.
   The system displays a confirmation message that the certificate is added to the keystore.

   Important:
   To add another certificate to the keystore, change the alias name when running the keytool command.

When you update CBA, change the platform, or perform any other activity that involves running the installer script, the system overwrites the keystore. So you must add the required certificate again following this procedure.

Configuring Global Settings

Configuring the IVR tab

Go to **Global Settings > IVR** tab.

The system displays the **IVR** page with the IVR parameters configuration options as follows:

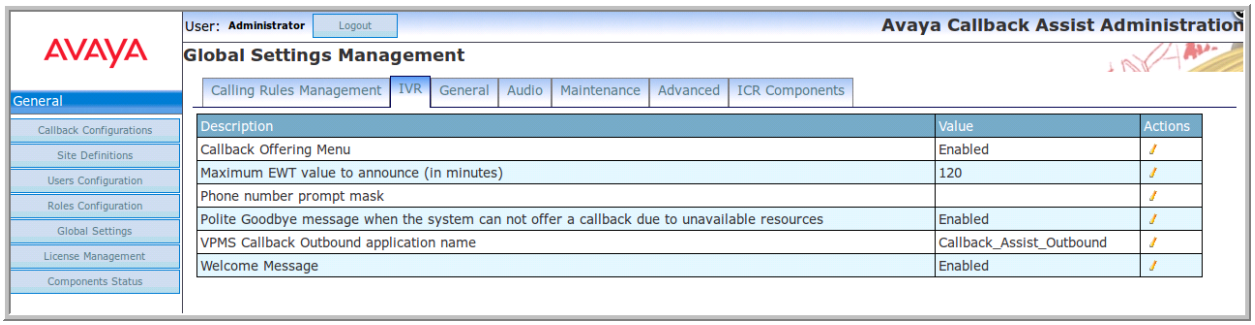


Figure 77 - IVR tab configuration screen

Configuring the General Tab

Go to **Global Settings > General** tab.

The system displays the **General** page with the default General parameters as follows:

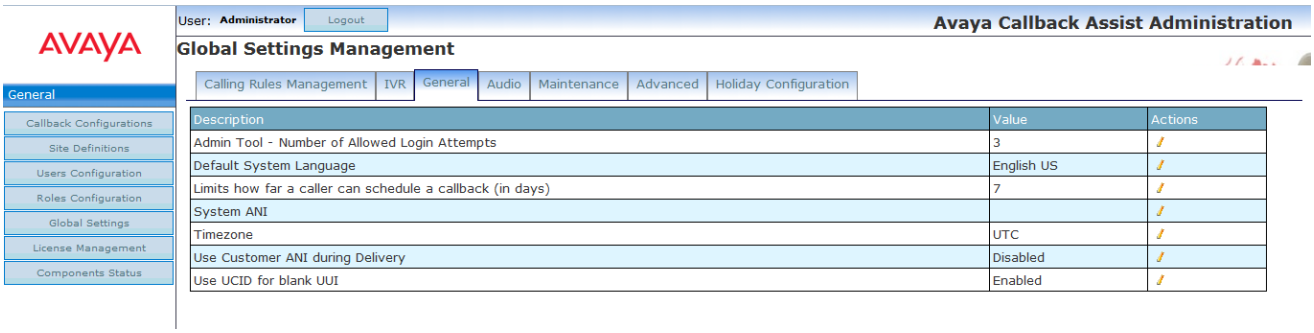


Figure 78 - General tab configuration screen

- ***System ANI***: Number used as ANI value for outbound calls to customers. This global parameter can be overridden per configuration basis, for more details see the *Administering Avaya Callback Assist* guide.

Use default values for the rest of the parameters. For more information on this operation, see the *Administering Avaya Callback Assist* guide.

Configuring the Audio tab

Go to **Global Settings > Audio** tab.

The system displays the **Audio** page with the audio parameters configuration options as follows:

Figure 79 - Audio tab configuration screen

- ***Storage URL***: Change [Host] mask with Callback Assist Server IP Address or hostname.
  For more information on this operation, see the *Administering Avaya Callback Assist* guide.

Configuring Site Definitions

The option “Site Definitions” would be enabled if the CBA installation has been done with site being enabled. The Site Definitions are required to be configured if using just one Experience Portal platform or if also having several ones. Below steps show minimal configuration for one Site only.

Go to Site Definitions and follow these steps:

1. Edit existing default site definition by clicking on pencil icon.
2. A window with site name text box and a button “Add New” for adding primary/auxiliary EPM will appear.
3. Optionally change ***Name*** field and it should be unique.
4. Click on “Add New” button to add primary EPM.
5. Enter ***Outbound Web Service IP Address*** of Primary EPM field.
6. Enter ***Outbound Web Service User*** defined on EPM. The User parameter must match the Outcall credentials administered on EPM Settings on Experience Portal Web Portal.
7. Enter ***Outbound Web Service Password*** defined on EPM. The User parameter must match the Outcall credentials administered on EPM Settings on Experience Portal Web Portal.
8. Click on OK button to save primary EPM detail
9. Optionally, if auxiliary EPMs are present, add the multiple auxiliary EPMs by clicking on “Add New” button and enter the ***outbound web service IP Address, user name*** and ***password***.
10. Click on ***OK*** button to save the auxiliary EPM detail.

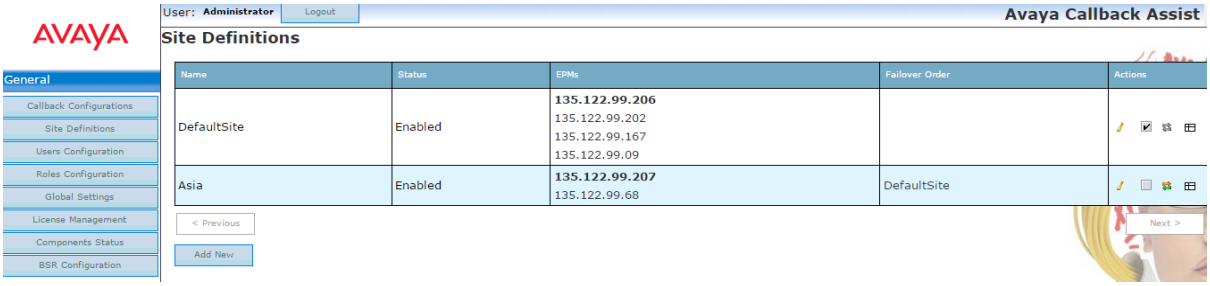


Figure 80 - Sites Definitions screen

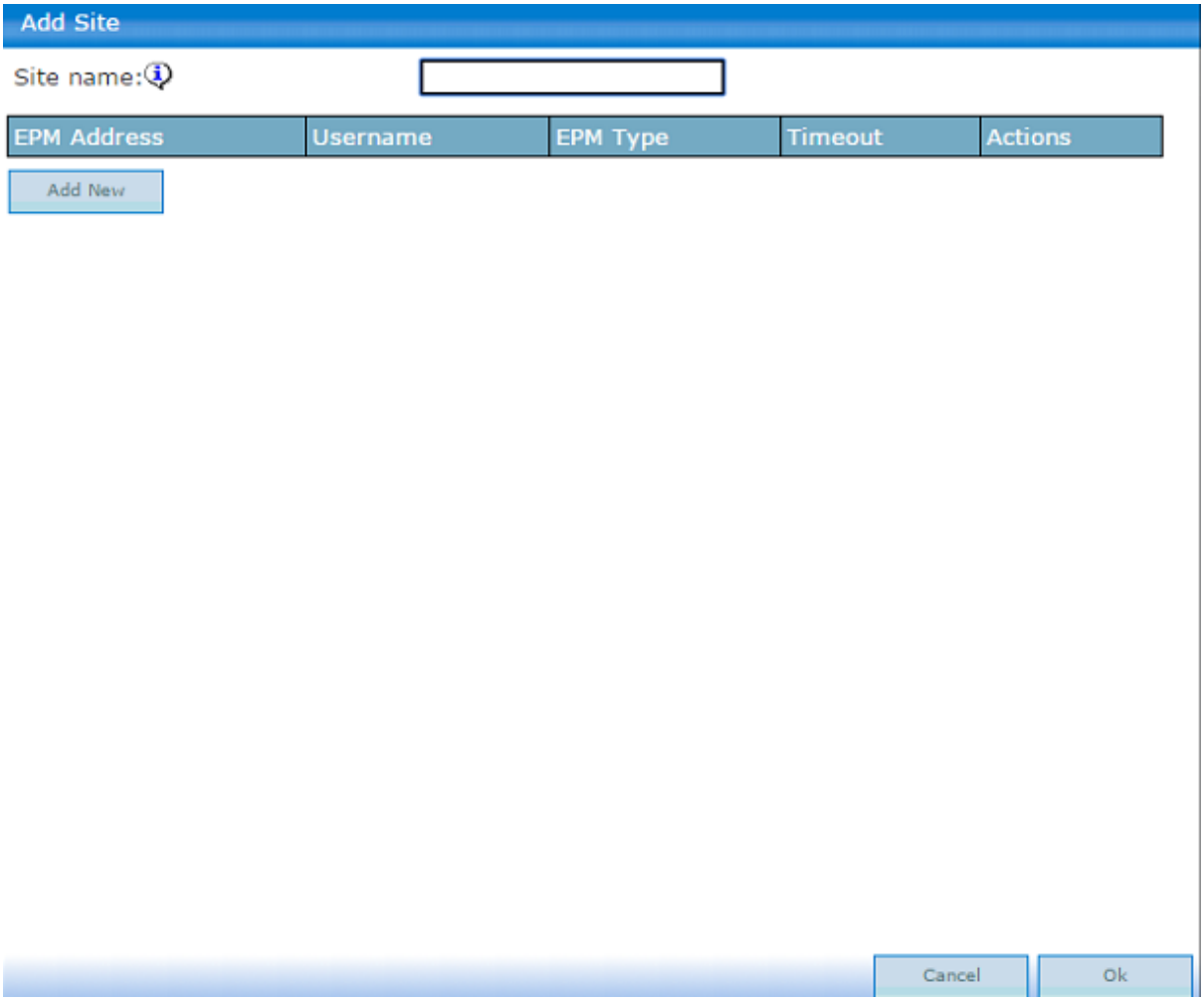


Figure 81 - Add New Site screen

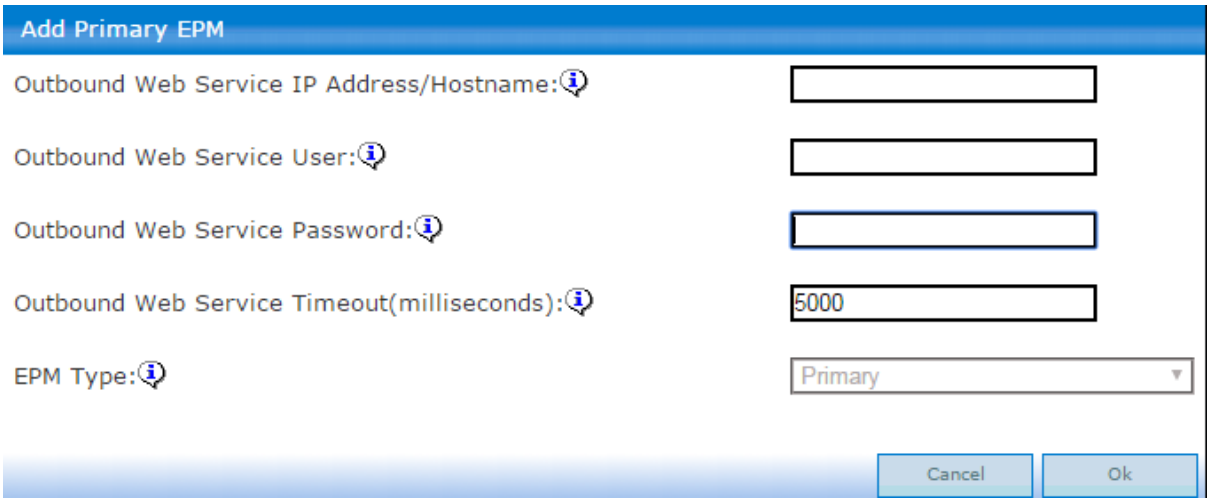


Figure 82 - Add Primary EPM screen

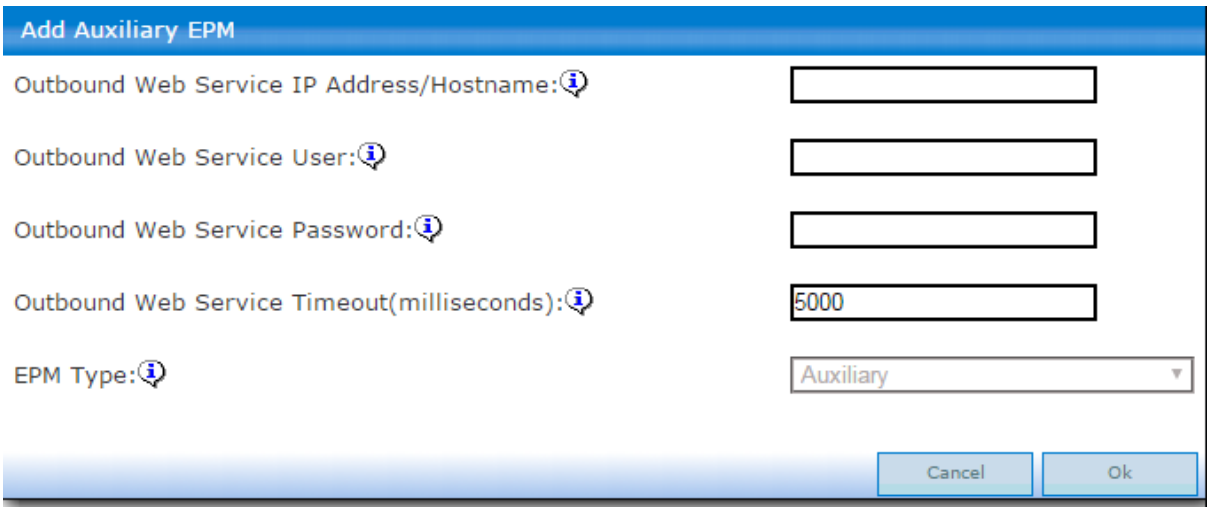


Figure 83 - Add Secondary EPM screen

Configuring Zone Definitions

The option “Zone Definitions” would be enabled if the CBA installation has been done with zone being enabled. The Zone Definitions are required to be configured if using just one Experience Portal platform or if also having several ones. The primary EPM detail of the default zone would have been given during the CBA installation itself. Below steps show minimal configuration to add auxiliary EPMS for a Zone.

Go to Zone Definitions and follow these steps:

1. Edit existing default zone definition by clicking on pencil icon.
2. A window with zone name text box and a button “Add New” for adding auxiliary EPM will appear.
3. Select zone name from zone name drop down.
4. Click on “Add New” button to add auxiliary EPM.
5. Enter ***Outbound Web Service IP Address*** of auxiliary EPM field.
6. Enter ***Outbound Web Service User*** defined on EPM. The User parameter must match the Outcall credentials administered on EPM Settings on Experience Portal Web Portal.
7. Enter ***Outbound Web Service Password*** defined on EPM. The User parameter must match the Outcall credentials administered on EPM Settings on Experience Portal Web Portal.
8. Click on OK button to save auxiliary EPM detail.
9. Optionally, if multiple auxiliary EPMs are present, add those by clicking on “Add New” button and enter the ***outbound web service IP Address, user name*** and ***password*** and click on OK button to save it.

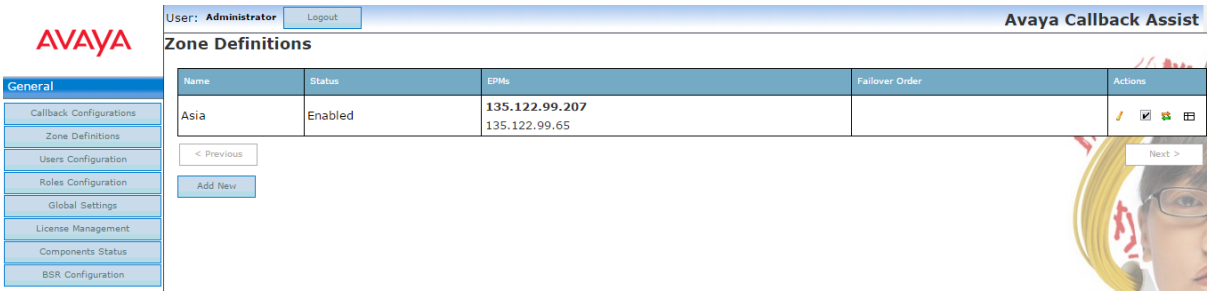


Figure 84 - Zone Definitions screen

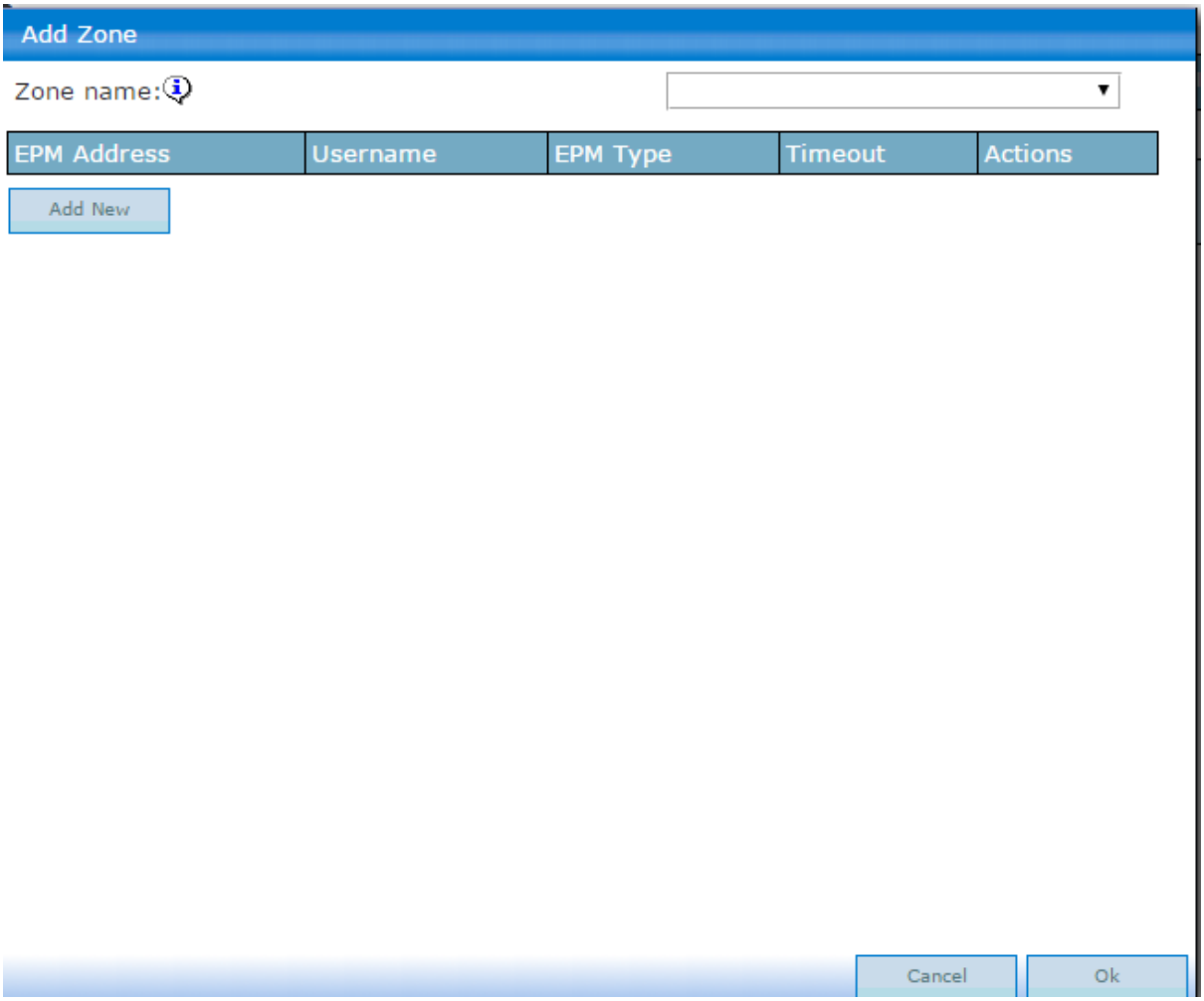


Figure 85 - Add New Zone screen

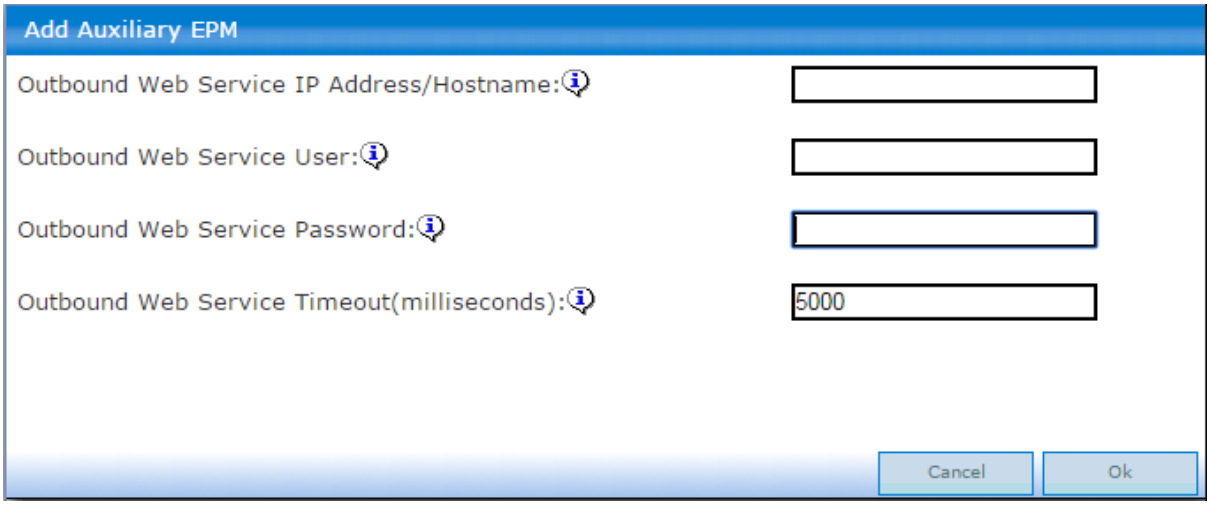


Figure 86 - Add Auxiliary EPM screen

Defining the Global Time Zone

After fresh installation or upgrade of Callback Assist, the system displays a warning message as follows:

```
 ************************************************************
ACTION REQUIRED on Time Zone Configuration
************************************************************
The default Time Zone of CBA is UTC. If your system requires a different
Time Zone
you must manually configure it in the Global Settings of the Admin Portal.
************************************************************
```

This means that the time zone of the Callback Assist server, by default, is set as UTC. It is extremely important to configure this value, as it will affect the behavior of Callback Assist, given that offering and delivery algorithms are based on this time zone.

If you have just performed an, and you want Callback Assist to behave exactly as it did before upgrading, you must set the Calback Assist Global Time Zone as it is in the server's Time Zone. You can check the server's time zone by typing the following command in the server's terminal:

`date +%Z`

```
[root@denpsqacba64 opt]# date +%Z
MDT
[root@denpsqacba64 opt]#
```

In this case, the output of the command is MDT, so you must configure the Callback Assist Global Time Zone with a Time zone that meets the Mountain Day light Time, such as "America/Denver".

To change the Global Time Zone, you must log into the CBA Web Administration application, and go the **Global Settings** > **General** tab.

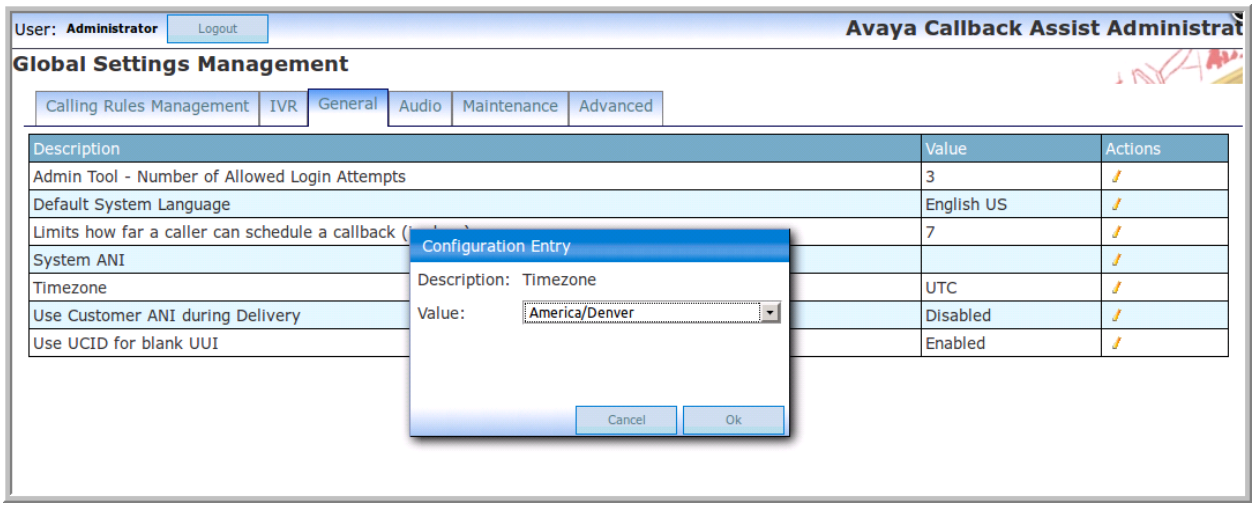


Figure 87 - Timezone screen in Global Settings page

For more information on the Global Time Zone, see the *Global Settings* section in the *Administering Avaya Callback Assist* guide

Creating a new Callback Configuration

The following steps shows minimum required fields to create a Voice Configuration for Agent First Strategy. Additional fields may be needed based on implementation requirements; if that's the case see *Administering Callback Assist Guide* for further details.

Go to Callback Configurations and follow these steps:

- Click on ***Add New***.
- Select ***Voice*** configuration type and click on ***Next…***. Other types depend on the license features available.
- The ***Add Voice Callback Configuration*** window is displayed.
- Click on ***General Tab*** and, as minimum, configure required fields shown below:
  - ***i. Name:*** a simple name for this configuration
  - *ii.* ***DNIS:*** Initial VDN created on section *Creating Incoming Calls Vector and Initial VDN*
  - iii. ***Agent Queue:*** Select Call Center Application from drop-down list.
  - iv. ***Error CDN***
- Click on Availability Tab and configure required fields:
  - i. ***Time Cushion:*** This is a time space between the call and the first schedule slot to be offered

ii. ***Number of Offered Slots:*** Number of callback timeslots to be offered to the callers.
iii. ***Time Zone Message:*** specify the time zone message to be played while offering schedule callback slots.
iv. Select current day of week and configure, as minimum, current time of day as an available slot.

- Click on ***Customer Tab*** and, as minimum, configure required fields shown below:
  i. ***Welcome Message:*** specify the first message to be played to the caller.
  ii. ***Goodbye Message:*** specify the message when caller leaves gracefully.
  iii. ***Customer WTA Message:*** specify the message to repeat when caller is waiting on queue.

  Note: *There are sample messages located on Callback Assist installation directory under /opt/Avaya/callbackassist/apache-tomcat-ddapps/webapps/CBAPhrases/samples/*

- Click on ***Agent Tab***
  i. ***Agent WTA Message:*** specify the message to play to agent when outbound call is in progress.

  Note: *There are sample messages located on Callback Assist installation directory under /opt/Avaya/callbackassist/apache-tomcat-ddapps/webapps/CBAPhrases/samples/*

- Click on ***Voice Questions Tab*** and configure the following Phone Number question fields:
  i. ***Question Message:*** message to request phone number to caller.
  ii. ***Agent Message:*** message to playback phone number to Agent.
  iii. ***No Input Message:*** message to play when caller does not enter any value.
  iv. ***No Match Message:*** message to play when caller enters invalid value.
  v. Click ***OK*** on Question window.

  Note: There are sample messages located on Callback Assist installation directory under /opt/Avaya/callbackassist/apache-tomcat-ddapps/webapps/CBAPhrases/samples/

- Click OK button on ***Add Voice Callback Configuration*** window.

Configuring Avaya Orchestration Designer

Refer *Configuring Avaya Orchestration Designer* section.

Post Installation

License Installation

Login to WebLM License Server

Starting from Callback Assist 4.1.6 Avaya Web License Manager (WebLM) is installed along with Callback Assist Package.

The integrated WebLM Server interface is accessible from the URL

http://<server-hostname>:8443/WebLM/index.jsp.

The system displays the certificate dialog box. The dialog box informs you that the application is running over HTTPS (secured HTTP). Click "Yes" to accept the certificate. From this point, all the communication between the browser and the server will be over https.

Default log in credential is user ***admin*** and password ***weblmadmin*** as shown below:

Figure 88 - Avaya WebLM – Login Screen

After login into the application for the first time the user will be prompted to change the admin user password, as follows:

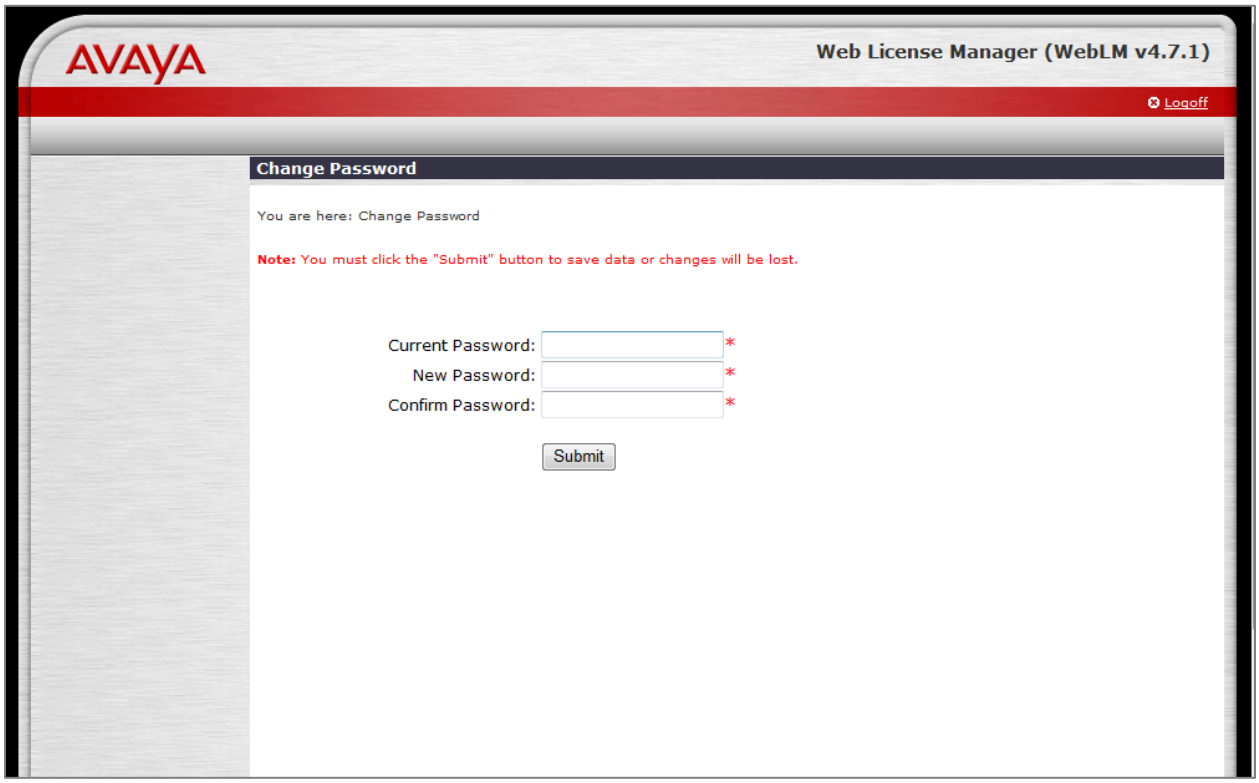


Figure 89 - Avaya WebLM - Change Password Screen

After the password changed, the application forces to logout and login again with the changed password.

Installing Callback Assist License

To add Callback Assist license in WebLM server follow the below steps:

1. Go to **Install License** on the left navigation pane.
2. Click the **Browse** button to select the valid license file
3. Click the **Install** button to submit the new license.

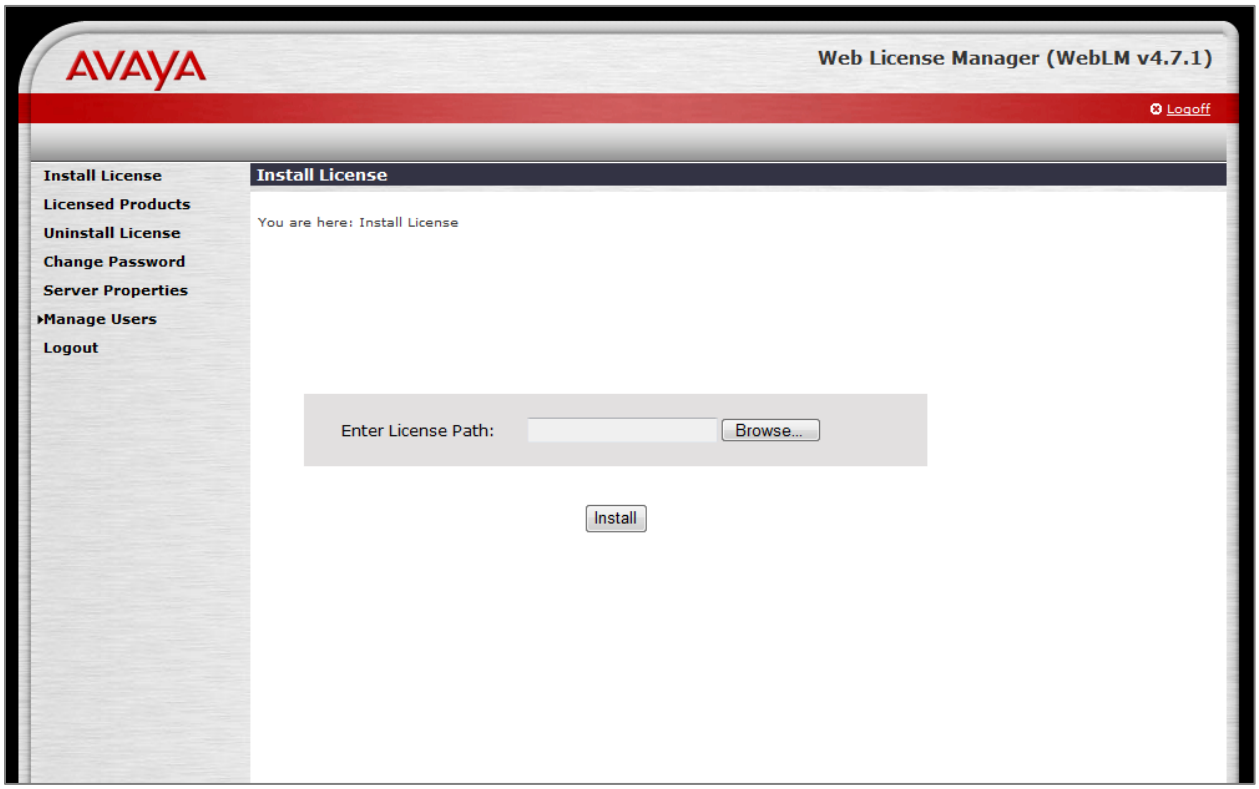


Figure 90 - Avaya WebLM – Install License

After the successful installation of Callback Assist license we can see the license in the left navigation under Licensed Products menu.

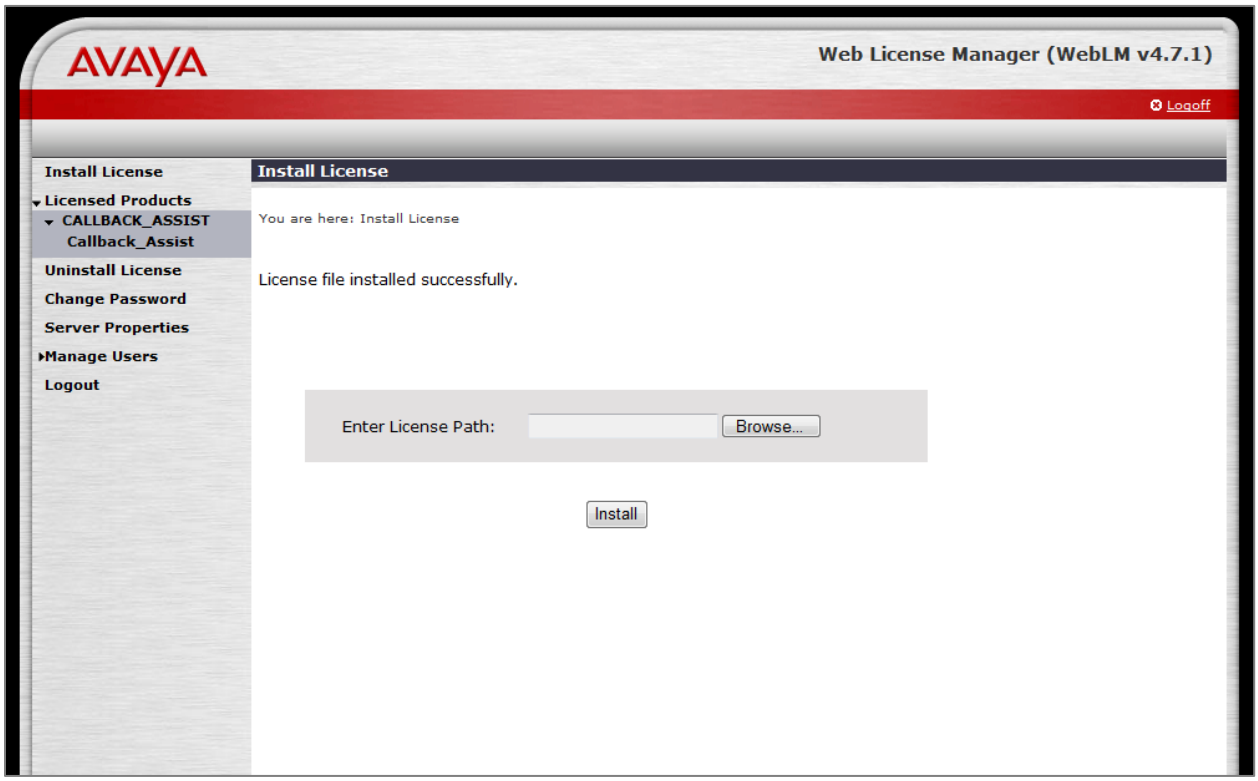


Figure 91 - Avaya WebLM – Callback Assist License Installed

Installing Callback Assist in a High Availability mode

The CBA installer offers the following options at the start when you run the installer:

1. Callback Assist Core Components with external Database
2. PostgreSQL Database Server for Callback Assist
3. Callback Assist Single Server deployment (Core Components & DB)

If you choose to install CBA by selecting option 1, CBA installs the Callback Assist Core components with an external and already installed CBA database. This is a High Availability mode of installation.

If you choose to install CBA by selecting option 2, CBA installs the PostgreSQL Database Server for Callback Assist. This Database will be used in a High Availability mode by all the CBA Servers which will be installed.

If you choose to install CBA by selecting option 3, CBA installs the Callback Assist Core Components and PostgreSQL Database Server in a single server. This is a Single-Server mode of installation.

To install CBA in a High Availability mode, the PostgreSQL Database Server for Callback Assist is the first server that needs to be installed.

Installing PostgreSQL Database Server for Callback Assist

To install the PostgreSQL Database for Callback Assist, perform the following steps:

1. On a Linux shell logged on as root, go to the directory where the CBA installer file is located.
2. Run the following command to extract the files to a temporary folder in your system: tar -xvf callbackassist<version>.tar. One of the extracted files is callback-install.sh which is the main installation script.
3. Run the following command to start the installation process: ./callback-install.sh
   The system writes the command output of the scripts to the standard output and to the callback-install.log file in your system. Then the system prompts you to choose the required platform for CBA installation.
4. Select [**2)**] as the required installation mode from the following options:
   a. Callback Assist Core Components with external Database
   b. PostgreSQL Database Server for Callback Assist
   c. Callback Assist Single Server deployment (Core Components & DB)
5. Select the directory in which PostgreSQL Database for Callback Assist is going to be installed.

The following example shows the script output of the callback-install.sh script for a PostgreSQL Database Server for Callback Assist:

```
[root@server123 4.3.1.0]# ./callback-install.sh
************************************************************
Avaya Callback Assist Installer
************************************************************
(03:29:05) [ No Callback Assist Components were found. ]
(03:29:05) Please choose the Components to be Installed on this server:
1) Callback Assist Core Components with external Database
2) PostgreSQL Database Server for Callback Assist
3) Callback Assist Single Server deployment (Core Components & DB)
#? 2
(03:29:08) [ About to Install PostgreSQL Database Server for Callback
Assist ]
Please, enter the base directory where Avaya Callback Assist will be
installed (default directory /opt):
Directory where Avaya Callback Assist is going to be installed:
/opt/Avaya/callbackassist


(03:29:10) Checking if there is enough disk space...
(03:29:11) Available disk space is enough.


(03:29:12) Unpackaging distribution file callbackassist.package...


(03:39:02) Creating callback Group...
(03:39:02) Creating callback User...
(03:39:04) 32 bit Architecture detected ...
(03:39:04) Using 32 bit PostgreSQL Installer...
(03:39:04) Installing PostgreSQL Server, this step may take several
minutes...
(03:42:10) Creating 'callback' PostgreSQL user...
(03:42:10) Creating 'callback' database...
(03:42:10) 'callback' database created.
server signaled
```

```
(03:42:10) Restarting database to reset max_connections
Restarting PostgreSQL 9.4:
waiting for server to shut down.... done
server stopped
waiting for server to start.... done
server started
PostgreSQL 9.4 restarted successfully
(03:42:12) Signaling Postgresql postmaster...
(03:42:12) Done.
(03:42:12) Setting ownership to callback user.

(03:42:13) [ Installation of PostgreSQL Database Server for Callback
Assist completed. ]
```

Important: You must install PosgreSQL Database Server using the Callback Assist installer binaries only and not the PosgreSQL official binaries. This is due to the fact that Callback Assist installer binaries create all the required users, permissions and schemas, and automatically configure all the required settings, for example port and time zone settings. The correct functionality, version, and configuration will be validated while installing Callback Assist Core Components with external Database.

For more information on PostgreSQL Database replication, see the *Avaya Callback Assist Application Notes for PostgreSQL Replication* guide.

Installing Callback Assist Core Components with external Database

After you install the PostgreSQL Database Server for Callback Assist, install at least two CBA Servers (Master and Slave).

Before installing Callback Assist Core Components with external Database, make sure that you have the PostgreSQL Database Server IP Address or host name, and the IP Address of any already installed CBA Server, as they would be required during the installation.

Perform the following tasks to install Callback Assist Core Components with external Database in a HA mode:

1. On a Linux shell logged on as root, go to the directory where the CBA installer file is located.
2. Run the following command to extract the files to a temporary folder in your system:
   `tar -xvf callbackassist<version>.tar.`
   One of the extracted files is callback-install.sh which is the main installation script.
3. Run the following command to start the installation process:
   `./callback-install.sh`
   The system writes the command output of the scripts to the standard output and to the callback-

install.log file in your system. Then the system prompts you to choose the required platform for CBA installation.

4. Select [**1)**] as the required installation mode from the following options:
    a. Callback Assist Core Components with external Database
    b. PostgreSQL Database Server for Callback Assist
    c. Callback Assist Single Server deployment (Core Components & DB)
5. Enter the installation directory where to install Callback Assist or just press ***Enter*** to use default location (default directory "/opt").
6. Select SIP as the installation platform.

    Note: Callback Assist does not support HA installation on CTI (H.323) and AACC/CM platforms.

For example, to install Callback Assist on a SIP platform, select [**2)**] at the prompt:
Select [**2)**] SIP as the platform type from the following options and press Enter.

1. CTI (H.323)
2. SIP
3. AACC/CM (SIP Based)

7. Choose desired AAEP system type from the following options and press Enter. (Default is zone).

    Note: This option is applicable only for SIP platform. For site, no additional inputs are required. For zone, user will be asked to provide *Default Zone Name, Primary EPM Information (EPM Address, Username, Password)* as inputs.

    1. Site
    2. Zone
    3. Skip (if already chosen in another HA server)

8. Choose desired Strategy from the following options and press Enter.
    1. Agent First
    2. Customer First Phantom Pool Strategy
    3. Customer First Priority Queuing Strategy

The system starts installing SIP platform with the desired strategy.

9. Choose the authentication type
    1. Internal
    2. External (Open ID ex: Google)

If the authentication type is selected as External then follow the CBA Installation as External Authentication section for more details.

10. Choose whether you need Local Web LM server or not. If you decide to use the Local Web LM server as license server then select “yes”, otherwise “no”.

    Note: If you select “no” at the time of installation and later if you decide to use Local Web LM server as a license server; then the WebLM (tomcat-weblm) service can be restored by running the script reinstallservices.sh from <CBA_INSTALLATION_LOCATION>/support folder.

11. Enter the IP address or the host name of the external Database Server at the prompt.
12. Enter *yes* to confirm the IP address or the host name you entered in the previous step.
13. Enter *yes* if any other instance of Callback Assist is sharing the same external database.
    Note: In a HA deployment, enter No in this step only if this is the very first Callback Assist server with external Database being installed. For all the following Callback Assist installations you must enter *yes* in this step.
14. Enter the IP Address of any Callback Assist server already installed that is sharing the Database Server entered above.
    In this step, Callback Assist automatically joins other Callback Assist 's File Server, being able to replicate all the audio files.
    Note: If there are several Callback Assist servers already installed, it does not matter which Callback Assist server IP Address you specify in this step if all previous Callback Assist servers were joined and are now up and running. This step will be skipped if you entered *No* in the previous step, that is, if this is the first Callback Assist Server being installed and there is no other Callback Assist server sharing the PostgreSQL Database.

    The procedures after this step are similar to the single server installation procedures. For more information, see the respective installation procedures based on SIP platform. After completing the installation, the system displays an installation successful message.

The following example shows the script output of the callback-install.sh script for a Callback Assist installation with an external Database in a SIP environment:

```
[root@cbadb81 4.4.0.0]# ./callback-install.sh


*********************************************************

Avaya Callback Assist Installer

*********************************************************


(09:10:54) [ No Callback Assist Components were found. ]

(09:10:54) Please choose the Components to be Installed on this server:


1) Callback Assist Core Components with external Database
```

```
2) PostgreSQL Database Server for Callback Assist

3) Callback Assist Single Server deployment (Core Components & DB)

4) Webcallback & DDApps components for Callback Assist

#? 1


(09:10:56) [ About to Install Callback Assist Core Components with
external Database ]


Please, enter the base directory where Avaya Callback Assist will be
installed (default directory /opt):


Directory where Avaya Callback Assist is going to be installed:
/opt/Avaya/callbackassist


(09:10:56) Checking if there is enough disk space...

(09:10:56) Available disk space is enough.


(09:10:56) Please choose a Platform type for Callback Assist
Application to be Installed from the options below:

1) CTI (H.323)

2) SIP

3) AACC/CM (SIP Based)

#? 2


(09:11:02) Please choose AAEP System type (Default: Zone):

1) Site

2) Zone

3) Skip (if already chosen in another HA server)

#? 2
```

```
(09:11:04) Please provide following information for Default Zone

Default Zone Name: default zone

Primary EPM IPAddress/HostName: 135.122.99.98

Primary EPM User Name: admin

Primary EPM Password: *******

(09:11:21) Please choose the Delivery Strategy:
1) Agent First Strategy
2) Customer First Phantom Pool Strategy
3) Customer First Priority Queuing Strategy
#? 1

(09:11:23) Please choose an Authentication Type:
1) Internal
2) External (OpenId ex: Google)
#? 1

(09:11:24) Are you going to use local WebLM service? (yes/no) : n

(09:11:26) [ Local WebLM server will not be added as a daemon service.
]
```

```
(09:11:26) [ Internal has been selected as an Authentication Type. ]


(09:11:26) [ Agent First Strategy has been selected as the Delivery
Strategy. ]


(09:11:26) [ Zone based AAEP system has been selected for installation.
]


(09:11:26) [ SIP Platform has been selected for installation. ]


(09:11:26) Unpackaging distribution file callbackassist.package...


(09:11:35) Creating callback Group...

(09:11:35) Creating callback User...

(09:11:35) Performing JDK silent install...

(09:11:39) JDK installed.

java version "1.7.0_75"

Java(TM) SE Runtime Environment (build 1.7.0_75-b13)

Java HotSpot(TM) Server VM (build 24.75-b04, mixed mode)


(09:11:39) Please enter the Callback Assist Database Server IP Address
or Host name: 135.122.99.103

(09:12:04) You've entered 135.122.99.103. Is this correct? (Yes/No) y

(09:12:06) Checking PostgreSQL Server version on 135.122.99.103...

(09:12:06) PostgreSQL version is OK.

(09:12:06) Checking for 'callback' User and Database on
135.122.99.103...

(09:12:07) User and Database are OK.
```

```
(09:12:07) Is there any other CBA instance sharing this same Database?
(Yes/No) n

(09:12:09) Installation of Callback Assist Maintenance done.

(09:12:09) Check
/opt/Avaya/callbackassist/maintenance/logs/execution.log file for
Callback Assist Maintenance service startup details.

(09:12:09) Installation of Callback Assist Engine done.

(09:12:09) Check /opt/Avaya/callbackassist/engine/logs/execution.log
file for Callback Assist Engine service startup details.

(09:12:09) 64 bit Architecture detected ...

(09:12:09) Installing file server for Red Hat 6.x 64 bits

(09:12:10) Local Ip Address: 135.122.99.81

(09:12:10) INFO: Using hostname/FQDN: cbadb81.avayacba.com to run the
file-server.

(09:12:10) Configuring File Server IP Address...

(09:12:10) Done.

(09:12:10) Changing SELinux context to some CBA files...

(09:12:10) Waiting for the Database Schema to be created or updated...

Migration successful

(09:20:19) Database Schema was successfully created or updated.

(09:20:20) Platform successfully set.

(09:20:21) AAEP System type successfully set.

(09:20:22) Primary EPM is successfully added for Default Zone.

(09:20:23) Deployment type successfully set.

(09:20:23) Storing Release version and Build Number...

(09:20:24) Release version and Build number successfully stored.

(09:20:24) Running Platform dependant changes...

Migration successful

(09:20:27) Database Schema was successfully updated.
```

```
(09:20:27) Running Deployment Type changes...

Migration successful

(09:20:36) Database Schema was successfully updated.

(09:20:36) Updating authentication type into database ...

(09:20:37) Installing weblm Tomcat service (tomcat-weblm)...

(09:20:37) Installation of tomcat-weblm done.

(09:20:37) Installing adminapp Tomcat service (tomcat-adminapp)...

(09:20:37) Installation of tomcat-adminapp done.

(09:20:37) Installing ddapps Tomcat service (tomcat-ddapps)...

(09:20:37) Installation of tomcat-ddapps done.

(09:20:37) Installing webcallback Tomcat service (tomcat-
webcallback)...

(09:20:37) Installation of tomcat-webcallback done.

(09:20:37) Deploying Applications...

(09:20:37) Moving tomcat realm related jar files

(09:20:37) Installation of Tomcat instances done.

(09:20:37) Installing BSR Server.  This operation may take several
minutes...

(09:20:37) Local Ip Address: 135.122.99.81

(09:20:38) Setting ownership to callback user.

Starting Callback Assist Engine...

Callback Assist Engine Started.                            [  OK  ]

Callback Assist Engine ( pid 28064 ) is running...

Starting Callback Assist Maintenance...

Callback Assist Maintenance Started.                       [  OK  ]

Callback Assist Maintenance ( pid 28143 ) is running...

Service tomcat-weblm is not present. Nothing to do

Starting tomcat-adminapp...                                [  OK  ]
```

```
tomcat-adminapp ( pid 28241 ) is running...

Starting tomcat-webcallback...                              [  OK  ]

tomcat-webcallback ( pid 28336 ) is running...

Starting tomcat-ddapps...                                   [  OK  ]

tomcat-ddapps ( pid 28432 ) is running...

Starting CBA BSR Server service...                          [  OK  ]

CBA BSR Server ( pid 28522 ) is running...

Starting Callback Assist File Server...

Callback Assist File Server Started.                        [  OK  ]

Callback Assist File Server ( 28677 ) is running...         [  OK  ]

Callback Assist File Server Ping test                       [  OK  ]

Callback Assist File Server Read/Write cycle Test           [  OK  ]

Callback Assist File Server Ring Status is Up               [  OK  ]

Callback Assist File Server is joined to a cluster          [  NO  ]

If this is not a HA deployment, then disregard this warning.


(09:21:09) [ Installation of Avaya Callback Assist (SIP - Agent First
Strategy) completed. ]




*********************************************************

ACTION REQUIRED on Time Zone Configuration

*********************************************************

The default Time Zone of CBA is UTC. If your system requires a
different Time Zone

you must manually configure it in the Global Settings of the Admin
Portal.

*********************************************************
```

```
[root@cbadb81 4.4.0.0]#
```

As a reference, in the installation output shown above there were 3 servers involved:

- PostgreSQL Database server *[135.122.60.123]*
- Callback Assist Server with external Database (already installed) *[135.122.99.80]*
- Callback Assist Server with external Database *[135.122.60.125]*

Installing Webcallback and DD Apps Components

This section guides you through the steps to install and configure Webcallback and DD Apps components in a separate server for scalability, including the following:

- Software Installation Steps
- Post Installation Steps

Checklist for setting up the system for Callback Assist

Prerequisites:

Before you set up the system to install Webcallback and DD Apps components, make sure that you meet the following requirements:

- CBA is already installed in HA mode and functional.
- Verify the Linux Shared Memory

Installing Webcallback and DD Apps

Avaya Callback Assist Installation is based on Linux shell scripts. The installer of Avaya Callback Assist is a single tar file whose name has the form callbackassist-<version>.tar (for example: callbackassist-4.3.1.0-GA.tar, where 4.3.1.0-GA is the version label). All Avaya Callback Assist components run as Linux daemons registered under /etc/init.d.

The callback-install.sh script installs all the available components in the server where the script is run, and prepares most of the configuration files automatically.

The idea behind this requirement is to increase the number of concurrent request that can be handled via the WS API (offering phase only) and Voice UI (offering and delivery phases).

Apart from HA installation, it is assumed that this installation (Webcallback and DD Apps components) will be done in two separate servers at maximum.

Software Installation Steps

1. On a Linux shell logged on as root, go to the directory where the CBA installer file is located.
2. Run the following command to extract the files to a temporary folder in your system:
   `tar -xvf callbackassist-<version>.tar`
   One of the extracted file is callback-install.sh which is the main installation script.
3. Run the following command to start the installation process:
   `./callback-install.sh`
   The system writes the command output of the scripts to the standard output and to the callback-install.log file in your system. Then the system prompts you to choose the installation type.
4. Select [**4**) ***Webcallback & DDApps installation for Callback Assist)*** as the installation mode.
5. Enter the installation directory where to install Callback Assist or just press ***Enter*** to use default location (default directory "/opt").

Step by step output of the *callback-install.sh* script:

```
[root@cbfsipcba1 4.3.2.0-GA-Latest]# ./callback-install.sh

************************************************************
Avaya Callback Assist Installer
************************************************************



(02:16:22) [ No Callback Assist Components were found. ]



(02:16:22) Please choose the Components to be Installed on this server:

1) Callback Assist Core Components with external Database
2) PostgreSQL Database Server for Callback Assist
3) Callback Assist Single Server deployment (Core Components & DB)
4) Webcallback & DDApps components for Callback Assist
#? 4

(02:16:24) [ About to Install Webcallback & DDApps components for Callback
Assist ]



Please, enter the base directory where Avaya Callback Assist will be
installed (default directory /opt):

Directory where Avaya Callback Assist is going to be installed:
/opt/Avaya/callbackassist

(02:16:26) Checking if there is enough disk space...
(02:16:26) Available disk space is enough.

(02:16:26) Unpackaging distribution file callbackassist.package...

(02:16:33) Creating callback Group...
```

```
(02:16:40) Creating callback User...
(02:16:40) Performing JDK silent install...
(02:16:43) JDK installed.
java version "1.7.0_75"
Java(TM) SE Runtime Environment (build 1.7.0_75-b13)
Java HotSpot(TM) Server VM (build 24.75-b04, mixed mode)

(02:16:43) Please enter the Callback Assist Database Server IP Address or
Host name: 135.122.60.125
(02:16:47) You've entered 135.122.60.125. Is this correct? (Yes/No) y
(02:16:48) Checking PostgreSQL Server version on 135.122.60.125...
(02:16:49) PostgreSQL version is OK.
(02:16:49) Checking for 'callback' User and Database on 135.122.60.125...
(02:16:49) User and Database are OK.

(02:16:49) Installing ddapps Tomcat service (tomcat-ddapps)...
(02:16:49) Installation of tomcat-ddapps done.
(02:16:49) Installing webcallback Tomcat service (tomcat-webcallback)...
(02:16:49) Installation of tomcat-webcallback done.
(02:16:49) Deploying Applications...
(02:16:49) Installation of Tomcat instances (DDApps and Webcallback) done.
(02:16:49) Setting ownership to callback user.
Starting tomcat-webcallback...                              [  OK  ]
tomcat-webcallback ( pid 31176 ) is running...
Starting tomcat-ddapps...                                   [  OK  ]
tomcat-ddapps ( pid 31404 ) is running...

(02:16:52) [ Installation of Webcallback & DDApps components completed. ]

**************************************************************************
ACTION REQUIRED on Database Connection Configuration
**************************************************************************
```

```
Since this installation is going to use database connections from existing
HA installation,

you must manually adjust (decrease) database connections from respective
CBA components in HA servers.


Please follow the instructions given in the document to do this manual
configuration

**************************************************************************
```

Post Installation Steps

This section describes the steps which should be followed after installing Webcallback and DD Apps in a separate server.

Load Balancer Configuration

If you perform this installation with existing HA environment where you use Load Balancer, this new server's IP address has to be added to the existing Load Balancer configuration.

Refer Configuring the Load Balancer section and add the new server's IP Address to the existing list.

Otherwise, if you perform this installation with existing HA environment where you don't use Load Balancer OR with Single server environment, you must configure primary server's IP address in corresponding Admin UI, instead of 127.0.0.1. (**Global Settings → BSR Components → BSR Server Node IP Address/Hostname**)

Please refer *Administering Avaya Callback Assist* guide to know the page navigation.

Database Connections Configuration

In HA installation, the maximum database connections are configured for Webcallback and DD Apps components as below;

Server	Component	Database Connections
Server 1	CBA Web Services	50
	CBA IVR Dialogs	70
Server 2	CBA Web Services	50
	CBA IVR Dialogs	70

The total Database connections will not be increased for this new installation of Webcallback and DD Apps components. Hence, it is required to use set of connections from each server of existing HA installation for this new installation. Sample database connections distribution is given below;

Server	Component	Database Connections
Server 3	CBA Web Services	20 (10+10)
	CBA IVR Dialogs	30 (15+15)

Now, it is required to manually decrease the database connections of existing HA servers to tally the total number connections, as below.

Server	Component	Database Connections
Server 1	CBA Web Services	40 (50-10)
	CBA IVR Dialogs	55 (70-15)
Server 2	CBA Web Services	40 (50-10)
	CBA IVR Dialogs	55 (70-15)

Above database connections distribution is just an illustration. Customer can distribute in any way which suits their environment and load of each server.

Steps to change database connections:

1. Stop the services

```
service tomcat-ddapps stop
service tomcat-webcallback stop
```

2. Set max_connections property to new value, in below files

   /opt/Avaya/callbackassist/apache-tomcat-webcallback/lib/callbackDatabaseConfiguration.properties

   /opt/Avaya/callbackassist/apache-tomcat-ddapps/lib/callbackDatabaseConfiguration.properties

```
Command: sed -i "s|max_connections=[0-9]*|max_connections={new-value}|g" {file-name}
```

3. Start the services

```
service tomcat-ddapps start
service tomcat-webcallback start
```

Uninstalling Webcallback and DD Apps

To uninstall Webcallback and DD Apps components, you must log on as root user and run the callback-uninstall.sh script copied during the installation to <INSTALL-PATH>/Avaya/callbackassist.

The following is an example of an execution output of the above script:

```
[root@cbfsipcba1 callbackassist]# ./callback-uninstall.sh


************************************************************

 Avaya Callback Assist Uninstaller

************************************************************


*** WARNING: You are about to Uninstall Avaya Callback Assist.

*** Note that all Callback data will be LOST.


(01:25:47) Are you sure you want to Uninstall Avaya Callback Assist? (yes/no): y


(01:25:50) Installation of tomcat-ddapps found.

(01:25:51) Waiting for tomcat-ddapps process to terminate...

Stopping tomcat-ddapps...                  [  OK  ]

(01:25:54) Uninstallation of tomcat-ddapps done.

(01:25:55) Installation of tomcat-webcallback found.
```

```
(01:25:55) Waiting for tomcat-webcallback process to terminate...

Stopping tomcat-webcallback...                    [  OK  ]

(01:25:58) Uninstallation of tomcat-webcallback done.

(01:25:59) Callback Assist Engine is not installed.

(01:25:59) Callback Assist Maintenance is not installed.

(01:25:59) BSR Server is not installed.

(01:25:59) File Server is not installed.

(01:25:59) Callback Assist Database is not installed.



*** [ Uninstallation of Avaya Callback Assist done. ]
```

Reinstalling Webcallback and DD Apps

When you try to install the same version of Webcallback and DD Apps as the one already installed in your system, the installer detects that a reinstall is being performed and prompts you to either proceed or abort with the reinstall operation.

Note: When performing a reinstallation operation, the system preserves all the existing data stored in the database, the stored audio files, license files, and configuration files.

The following example shows the reinstallation output.

```
[root@cbfsipcba1 4.3.2.0-GA]# ./callback-install.sh



**********************************************************

Avaya Callback Assist Installer

**********************************************************



*** ATTENTION: Installation of Webcallback & DDApps components of Avaya Callback Assist was
detected.
```

```
***

*** The current installed version is 4.3.2.0-GA-SNAPSHOT, build 46530,

*** which is the same as the one you want to install.

***

*** Do you want to Reinstall Webcallback & DDApps components of Avaya Callback Assist? (yes/no): y

(01:12:05) Checking if there is enough disk space...

(01:12:05) Available disk space is enough.

Stopping tomcat-ddapps...                    [  OK  ]

Stopping tomcat-webcallback...                  [  OK  ]


(01:12:06) Copying temporary data (this operation may take a couple of minutes)...

(01:12:10) Previous Version 4.3.2.0-GA-SNAPSHOT backed up in: /opt/Avaya/backup

(01:12:11) Backing up CBA Phrases directory...

(01:12:16) CBA Phrases directory Successfully Backed up.

(01:12:16) Installation of tomcat-ddapps found.

(01:12:16) Uninstallation of tomcat-ddapps done.

(01:12:17) Installation of tomcat-webcallback found.

(01:12:17) Uninstallation of tomcat-webcallback done.

(01:12:17) Callback Assist Engine is not installed.

(01:12:17) Callback Assist Maintenance is not installed.

(01:12:17) BSR Server is not installed.


(01:12:17) Unpackaging distribution file callbackassist.package...


(01:12:28) Group callback already exists, so there is no need to create it.

(01:12:28) User callback already exists, so there is no need to create it.
```

```
(01:12:28) Performing JDK silent install...

(01:12:32) JDK installed.

java version "1.7.0_75"

Java(TM) SE Runtime Environment (build 1.7.0_75-b13)

Java HotSpot(TM) Server VM (build 24.75-b04, mixed mode)


(01:12:32) Please enter the Callback Assist Database Server IP Address or Host name: 135.122.99.106

(01:12:38) You've entered 135.122.99.106. Is this correct? (Yes/No) y

(01:12:40) Checking PostgreSQL Server version on 135.122.99.106...

(01:12:40) PostgreSQL version is OK.

(01:12:40) Checking for 'callback' User and Database on 135.122.99.106...

(01:12:40) User and Database are OK.


(01:12:40) Installing ddapps Tomcat service (tomcat-ddapps)...

(01:12:41) Installation of tomcat-ddapps done.

(01:12:41) Installing webcallback Tomcat service (tomcat-webcallback)...

(01:12:41) Installation of tomcat-webcallback done.

(01:12:41) Deploying Applications...

(01:12:41) Installation of Tomcat instances (DDApps and Webcallback) done.

(01:12:41) Setting ownership to callback user.

Starting tomcat-webcallback...                    [  OK  ]

tomcat-webcallback ( pid 15595 ) is running...

Starting tomcat-ddapps...                         [  OK  ]

tomcat-ddapps ( pid 15823 ) is running...


(01:12:43) Restoring Backed up CBA Phrases directory...
```

```
(01:12:45) CBA Phrases directory successfully restored.


(01:12:45) [ Installation of Webcallback & DDApps components completed. ]


********************************************************************************

ACTION REQUIRED on Database Connection Configuration

********************************************************************************

Since this installation is going to use database connections from existing HA installation,

you must manually adjust (decrease) database connections of Webcallback & DDApps components in
existing HA servers.


Please follow the instructions given in the document to do this manual configuration

********************************************************************************
```

Upgrading Webcallback and DD Apps

To perform an upgrade for Webcallback and DD Apps Deployment, you have to run the Callback Assist installer as you do during a fresh installation. During the upgrade, the Callback Assist installer automatically searches for any installed instance of Webcallback and DD Apps components in the system. After detecting a previous version of installation in the system, the installer informs about the current version and prompts you to confirm upgrading to the later version.

After you confirm the upgrade, Callback Assist installer backs up the previous version to protect against the possible failure of the upgrade process. After the backup, Callback Assist installer installs the new version components as it does with a fresh installation.

When the upgrade process is over, the system automatically restores all the configuration files, license files, and other required files from the previous version.

Performing Upgrade

Perform the following tasks for upgrading to the latest environment version:

1. On a Linux shell logged on as root, go to the directory where the CBA installer file is located.
2. Run the following command to extract the files to a temporary folder in your system:
   `tar -xvf callbackassist-<version>.tar`
3. Run the following command to start the installation process:
   `./callback-install.sh`
   The system detects that there is a previous version of Webcallback and DD Apps components installed, and asks if you want to upgrade it to a later version.
4. Type *Yes* to confirm upgrading to the later version and press Enter.
   The system starts installing the latest version based on your selection. After completing the installation, the system displays an installation successful message.

The following example shows an output of a Callback Assist v4.x.x.x upgrade. The example also shows an output of a *callback-install.sh* script detecting that there is a previous version of Webcallback and DD Apps components installed.

```
[root@cbfsipcba1 4.3.2.0-GA-Latest]# ./callback-install.sh


************************************************************

Avaya Callback Assist Installer

************************************************************


*** ATTENTION: Installation of Webcallback & DDApps components of Avaya
Callback Assist was detected.

***

*** The current installed version is 4.3.2.0-GA-SNAPSHOT, build 46495.

*** You are about to install 4.3.2.0-GA-SNAPSHOT, build 46497.

***

*** Do you want to proceed with an upgrade (Webcallback & DDApps
components of Avaya Callback Assist)? (yes/no): y

(02:25:39) Checking if there is enough disk space...

(02:25:39) Available disk space is enough.
```

```
Stopping tomcat-ddapps...                                  [  OK  ]

Stopping tomcat-webcallback...                             [  OK  ]


(02:25:40) Copying temporary data (this operation may take a couple of
minutes)...

(02:25:44) Previous Version 4.3.2.0-GA-SNAPSHOT backed up in:
/opt/Avaya/backup

(02:25:44) Backing up CBA Phrases directory...

(02:25:46) CBA Phrases directory Successfully Backed up.

(02:25:46) Installation of tomcat-ddapps found.

(02:25:46) Uninstallation of tomcat-ddapps done.

(02:25:47) Installation of tomcat-webcallback found.

(02:25:47) Uninstallation of tomcat-webcallback done.

(02:25:47) Callback Assist Engine is not installed.

(02:25:47) Callback Assist Maintenance is not installed.

(02:25:47) BSR Server is not installed.


(02:25:47) Unpackaging distribution file callbackassist.package...


(02:25:54) Group callback already exists, so there is no need to create
it.

(02:25:54) User callback already exists, so there is no need to create
it.

(02:25:54) Performing JDK silent install...

(02:25:58) JDK installed.

java version "1.7.0_75"

Java(TM) SE Runtime Environment (build 1.7.0_75-b13)

Java HotSpot(TM) Server VM (build 24.75-b04, mixed mode)
```

```
(02:25:58) Please enter the Callback Assist Database Server IP Address
or Host name: 135.122.60.125

(02:26:01) You've entered 135.122.60.125. Is this correct? (Yes/No) y

(02:26:02) Checking PostgreSQL Server version on 135.122.60.125...

(02:26:03) PostgreSQL version is OK.

(02:26:03) Checking for 'callback' User and Database on
135.122.60.125...

(02:26:03) User and Database are OK.



(02:26:03) Installing ddapps Tomcat service (tomcat-ddapps)...

(02:26:03) Installation of tomcat-ddapps done.

(02:26:03) Installing webcallback Tomcat service (tomcat-
webcallback)...

(02:26:03) Installation of tomcat-webcallback done.

(02:26:03) Deploying Applications...

(02:26:03) Installation of Tomcat instances (DDApps and Webcallback)
done.

(02:26:03) Setting ownership to callback user.

Starting tomcat-webcallback...                              [  OK  ]

tomcat-webcallback ( pid 4142 ) is running...

Starting tomcat-ddapps...                                   [  OK  ]

tomcat-ddapps ( pid 4369 ) is running...



(02:26:06) Restoring Backed up CBA Phrases directory...

(02:26:08) CBA Phrases directory successfully restored.



(02:26:08) [ Installation of Webcallback & DDApps components completed.
]
```

```
**************************************************************************
ACTION REQUIRED on Database Connection Configuration

**************************************************************************

Since this installation is going to use database connections from
existing HA installation,

you must manually adjust (decrease) database connections from
respective CBA components in HA servers.



Please follow the instructions given in the document to do this manual
configuration

**************************************************************************
```

Managing status of Callback services

You can start, stop, and find status of all the callback processes by using the service System V init script as follows:

service <service_name> {start | stop | status}
where <service_name> is the name of a CBA 4.1.x process.

Service names of Callback Assist processes or components are as follows:

On CBA Linux server:

- cba-bsrserver (BSR Server process)
- cba-postgresql (Callback database)
- cbamaint (Callback database Maintenance processes)
- tomcat-adminapp (Callback Admin Web Tomcat Application Server)
- tomcat-ddapps (VoiceXML Tomcat Application Server)
- tomcat-webcallback (Webservices Tomcat Application Server)
- cbaengine (Callback Engine)
- cba-fileserver (Callback File Server)
- tomcat-weblm (WebLM Tomcat Application Server)

To query the status of Callback Assist database, use the following trace:

 # service cba-postgresql status

To start the Tomcat Application Server where the Orchestration Designer applications are hosted, use the following trace:

 # service tomcat-ddapps start

To stop the Callback Engine, use the following trace:

 # service cbaengine stop

Also, there is a support script which can be used to start or stop all the CBA services at once.
This script is located at: */opt/Avaya/callbackassist/support/callbackservices.sh*, assuming that CBA is installed at the default location */opt/Avaya/callbackassist*.
Use the script as follows:

 # /opt/Avaya/callbackassist/support/callbackservices.sh {start|stop|restart|status}

For example, to query the status of Callback Assist components:

```
[root@RHEL6232CBA support]# ./callbackservices.sh status
pg_ctl: server is running (PID: 32649)
```

```
/opt/Avaya/callbackassist/PostgreSQL/9.4/bin/postgres "-D"
"/opt/Avaya/callbackassist/PostgreSQL/9.4/data"
Callback Assist Maintenance ( pid 702 ) is running...
Callback Assist Engine ( pid 617 ) is running...
tomcat-ddapps ( pid 962 ) is running...
tomcat-adminapp ( pid 781 ) is running...
tomcat-webcallback ( pid 865 ) is running...
Callback Assist File Server ( 1071 ) is running... [ OK ]
Callback Assist File Server Ping test [ OK ]
Callback Assist File Server Read/Write cycle Test [ OK ]
Callback Assist File Server Ring Status is Up [ OK ]
Callback Assist File Server is joined to a cluster [ NO ]
If this is not a HA deployment, then disregard this warning.
[root@RHEL6232CBA support]#
```

Post-installation steps

The Administration interface is accessible from the URL http://<server-hostname>/admin (uses port tcp/80).

Default log in credential is user ***admin*** and password ***123456*** as shown below:

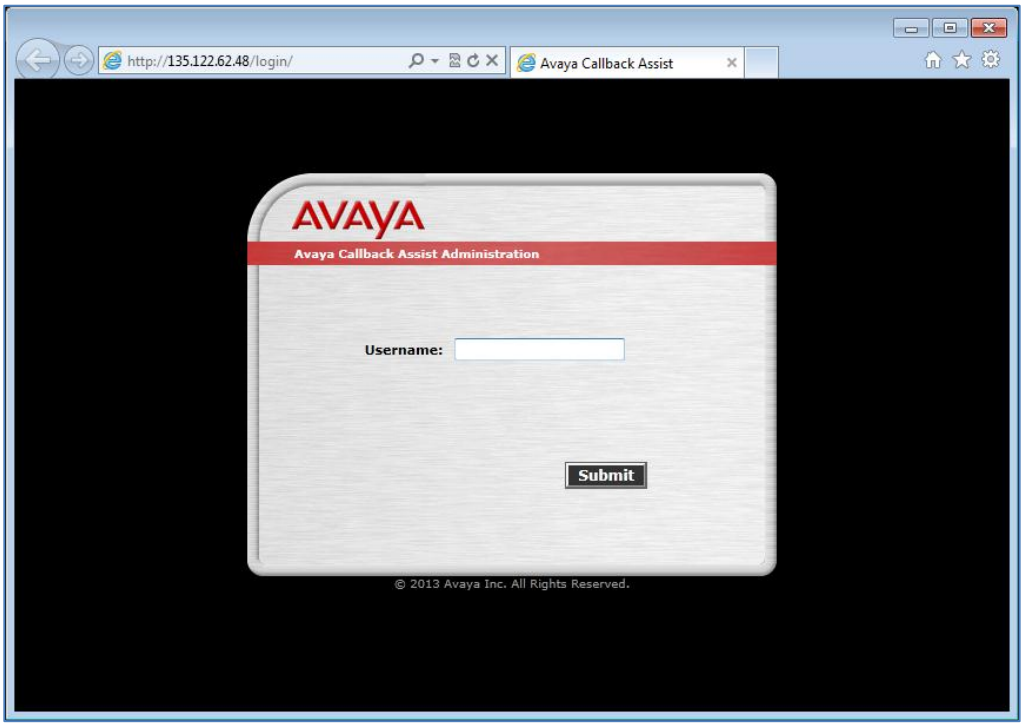


Figure 92 - Avaya Callback Administration Login Screen

After login into the application for the first time the user will be prompted to change the admin user password, as follows:

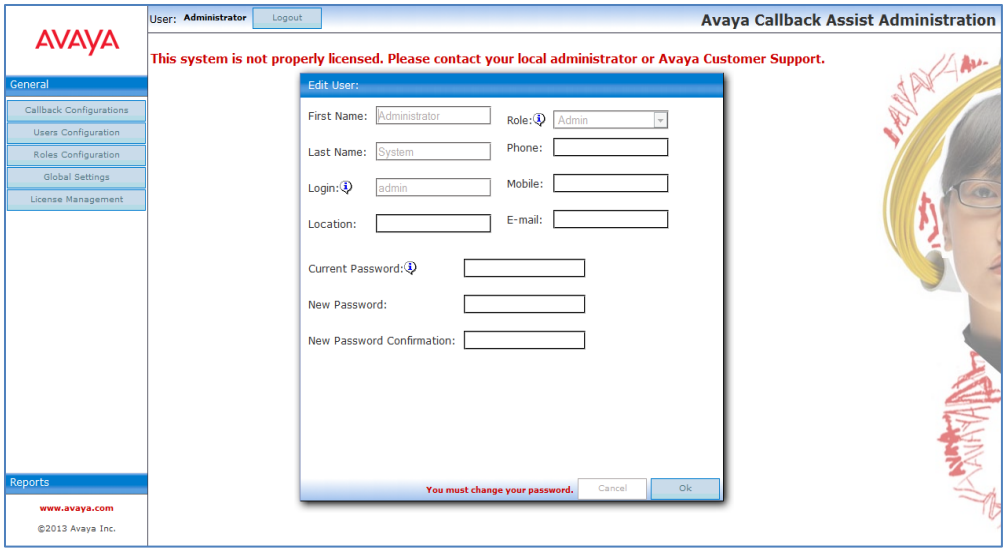


Figure 93 - Avaya Callback Assist – Change Password Screen

Enter new password and retype it on the confirmation field in order to complete the password change procedure.

Configuring license

To configure the WebLM Server where a Callback Assist license file is installed:

1. Go to **License Management** on the left navigation pane.
2. On the **License Management** page, enter the CBA WebLM License Server URL in the following format: https://\[Host\]:\[Port\]/WebLM/LicenseServer.
3. Enter the Dialog Designer WebLM License Server URL in the following format: https://\[Host\]:\[Port\]/WebLM/LicenseServer. The purpose of the Dialog Designer URL here is to display the Dialog Designer license status only and it will not affect CBA operations if not updated
4. Specify the number of CBA ports to be used for the installed platform based on your license.
5. Click the **Update** button to update the **License Management** page with the valid the WebLM URL and port usage information.

Example 1: The **License Management** page looks as follows when the currently installed Callback Assist version does not have a valid license file.

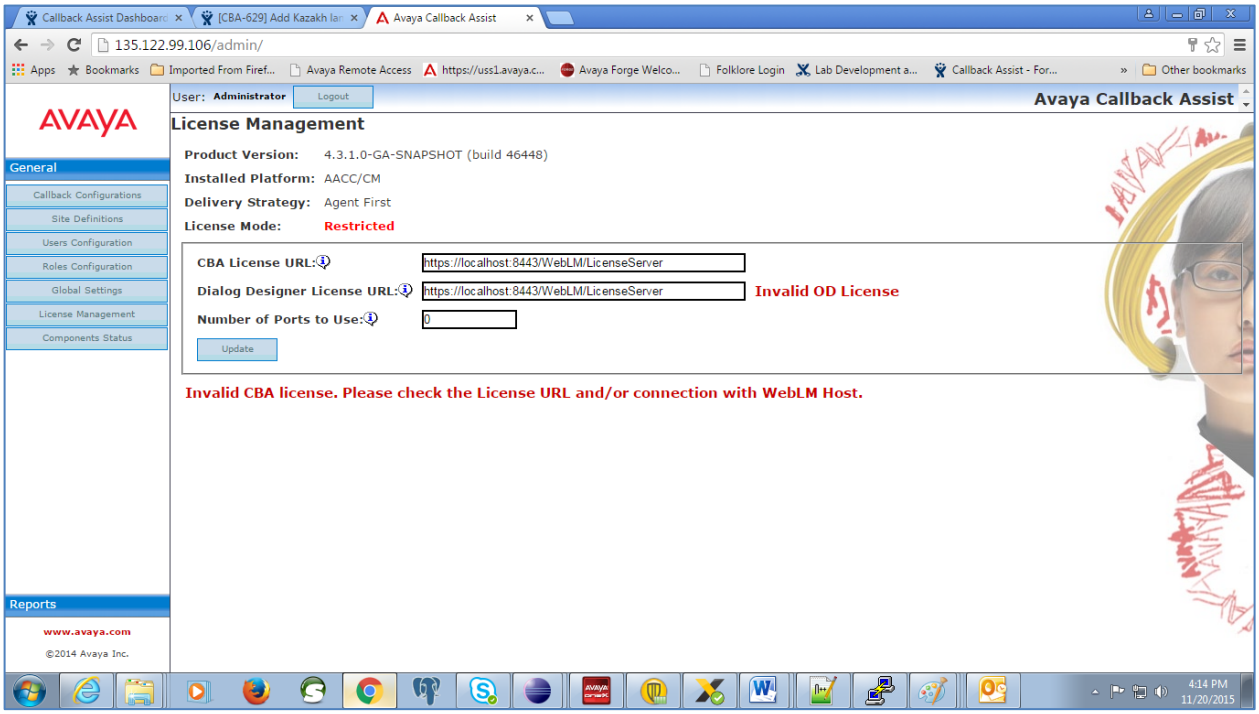


Figure 94 - License Management Screen - Invalid License

Example 2: The **License Management** page looks as follows when the currently installed Callback Assist version has a valid license file.

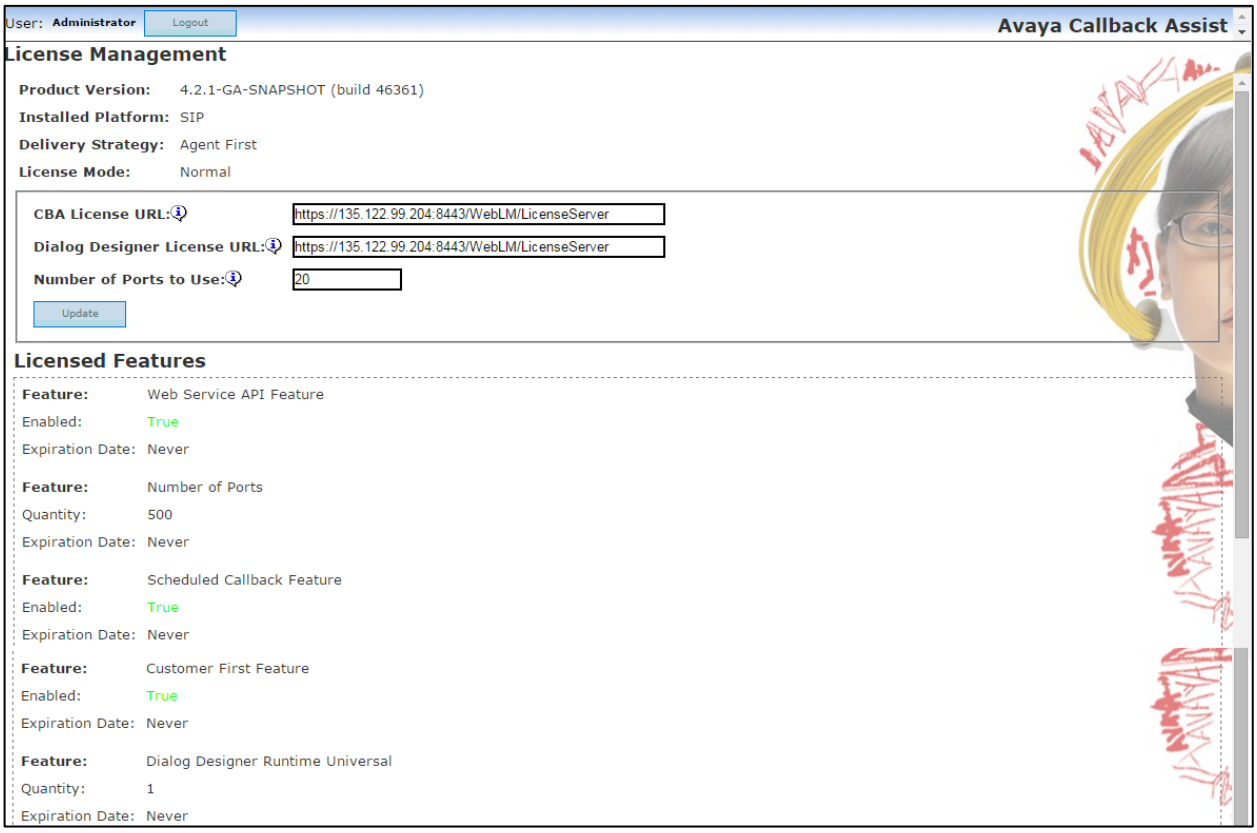


Figure 95 - License Management Screen - Valid License

Adding a custom certificate to connect to the WebLM Server

If the WebLM Server has a certificate file that is not imported in the keystore that CBA uses to validate the WebLM Server, you must add the certificate file to the keystore manually.

7. Save the certificate file in the DER format in your local machine by performing the following steps:
   Note: The procedure to add a certificate might vary based on your Web browser. The following procedure is for Internet Explorer 7.x.
   a. Open your Web browser and type the required WebLM License Server URL.
   b. Right click on the Web page and select **Properties** from the shortcut menu.
   c. On the **Properties** dialog box, click **Certificates**.
   d. Click **Details**.
   e. Click **Copy to File**.
   f. On the **Certificate Export Wizard**, under **Export File Format**, select the **DER encoded binary X.509 (.CER)** option.

g. Click **Next** and then click **OK**.
The system saves the certificate file with a *.CER* extension in your local machine.

8. Locate the certificate file you have just saved and copy it to the CBA Server.
9. Locate the *trusted_weblm_certs.jks* file in the following locations in the CBA Server:
   - <CBA_HOME>/apache-tomcat-ddapps/lib
10. Go to this location and then run the following command:
    <CBA_HOME>/jdk/bin/keytool -import -alias customweblm -keystore trusted_weblm_certs.jks -file <certificate filename>

    where <certificate filename> is the certificate file that you copied from your machine to the Callback Assist server.
11. Enter password when the system prompts you for a password.
    The system prints the certificate information.
12. Enter yes when the system asks whether you trust the certificate.
    The system displays a confirmation message that the certificate is added to the keystore.

    Important:
    To add another certificate to the keystore, change the alias name when running the keytool command.

Configuring Global Settings

Configuring the BSR Components tab (SIP environment)

On High Availability environments, multiple BSR Server process can be installed so it is recommended to configure a Load Balancer as the IP Address of BSR Server Node and distribute load between all BSR Servers.

Go to **Global Settings > BSR Components** tab.

The following parameters must be configured:

- BSR Server Node IP Address
- BSR Server Node Port Number
- BSR Web Service Password
- BSR Web Service User
- BSR Web Service Query Timeout (optional)

The IP address, port number, user and password referred to in the above image are the BSR Server Node IP address, port number, user and password respectively, entered during the BSR Server Node installation.

The following page shows an example of the configuration:

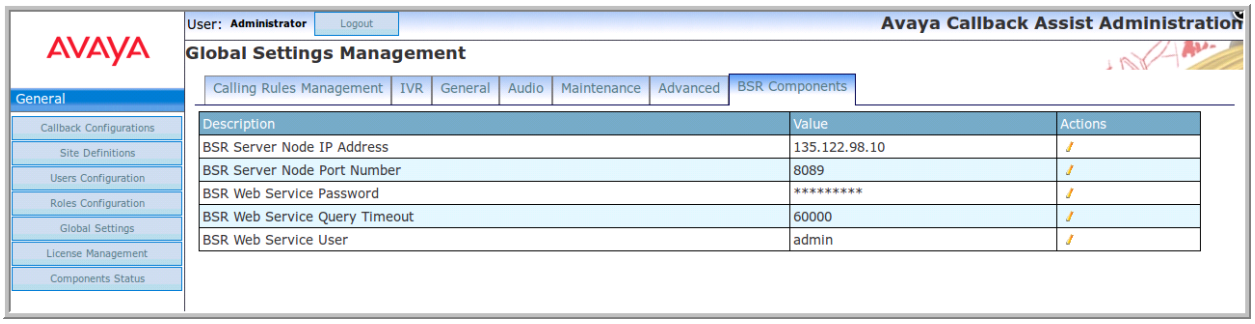


Figure 96 - BSR Components tab configuration screen

Configuring the Audio tab

On High Availability environments, multiple Callback Assist Servers can be installed so it is recommended to configure a Load Balancer as the IP Address of Storage URL and distribute load between all Callback Assist File servers.

Go to **Global Settings > Audio** tab.
The system displays the **Audio** page with the audio parameters configuration options as follows:

Figure 97 - Audio tab configuration screen

IP Address/Port: Callback Assist Application Server IP address/Admin Web App Port number.
For more information on this operation, see the *Administering Avaya Callback Assist* guide.

Adding Site Definitions

For SIP Platform, at least one Site definition must be configured.

If the CBA installation has been done with site being enabled, then “Site Definitions” link would come. The system displays the Site Definitions page as follows.

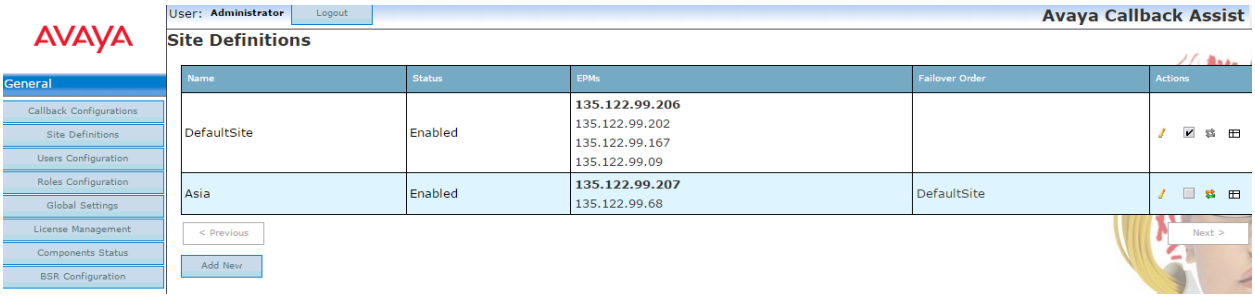


Figure 98 - Site Definitions screen

Each site can be configured with one primary EPM and multiple Auxiliary EPMs. Primary EPM has been shown as first EPM in the column EPMs followed by the auxiliary EPMs.

To add a new Site definition, go to **Site Definitions** > **Add new**. The system displays the page “Add Site” as follows.

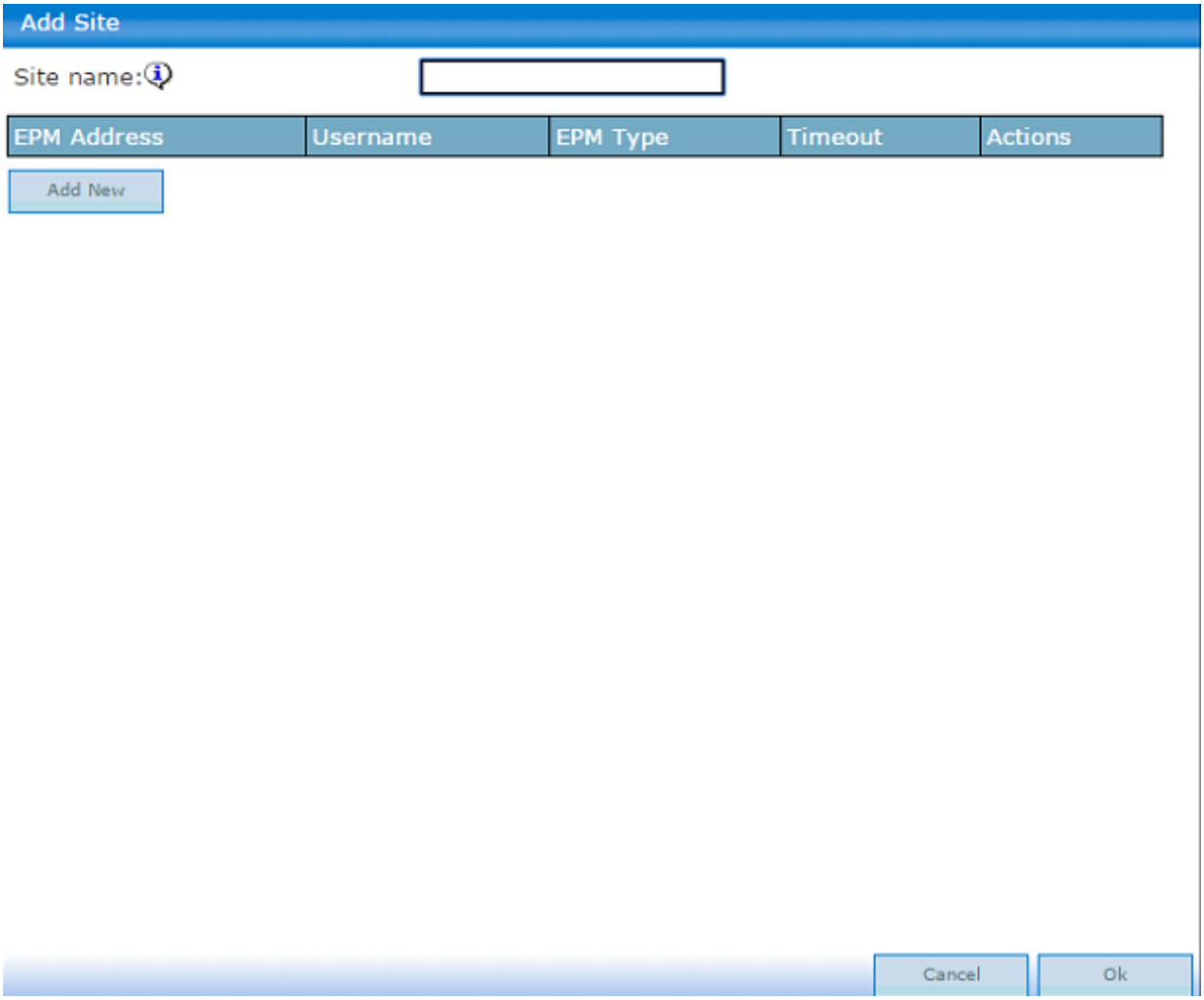


Figure 99 - Add Site screen

Site name has to be entered and it has to be unique. On click of “Add New” button in this page, new primary/auxiliary EPM can be added.

Primary EPM could be added as first EPM while adding a new site definition after which multiple auxiliary EPMs could be added. Primary EPM once added could be only edited and the system will not allow to remove it.

The system will display the page “Add Primary EPM” as follows.

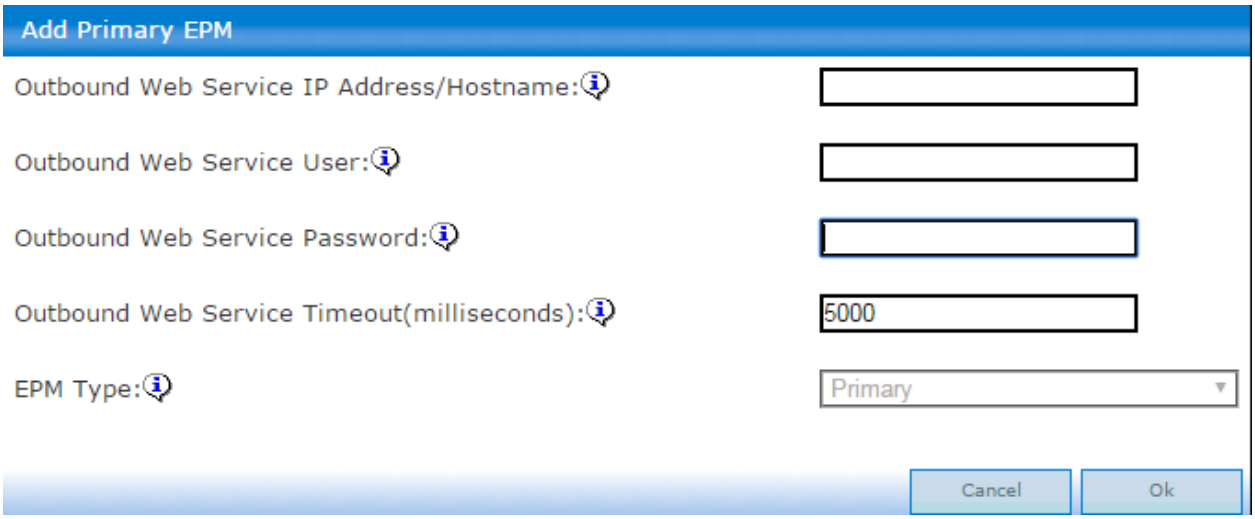


Figure 100 - Add Primary EPM screen

In this page, EPM type has been preselected as “Primary” and the timeout field has been filled with default value 5000 and it can be modified again. The IP Address in the EPM Outbound Web Service IP Address is the IP Address of the Experience Portal Server. The User and Password parameters must match the Outcall credentials administered on EPM Settings on Experience Portal Web Administration Portal

.

The system will display the page “Add Auxiliary EPM” as follows.

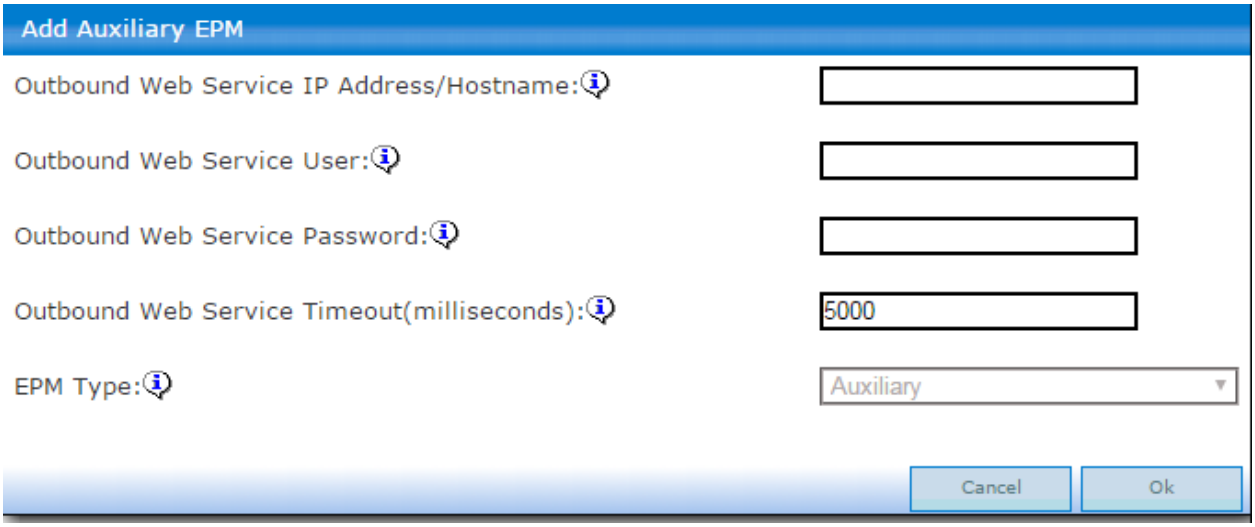


Figure 101 - Add Auxiliary EPM screen

The system will display the page as follows once the primary and auxiliary EPM has been added.

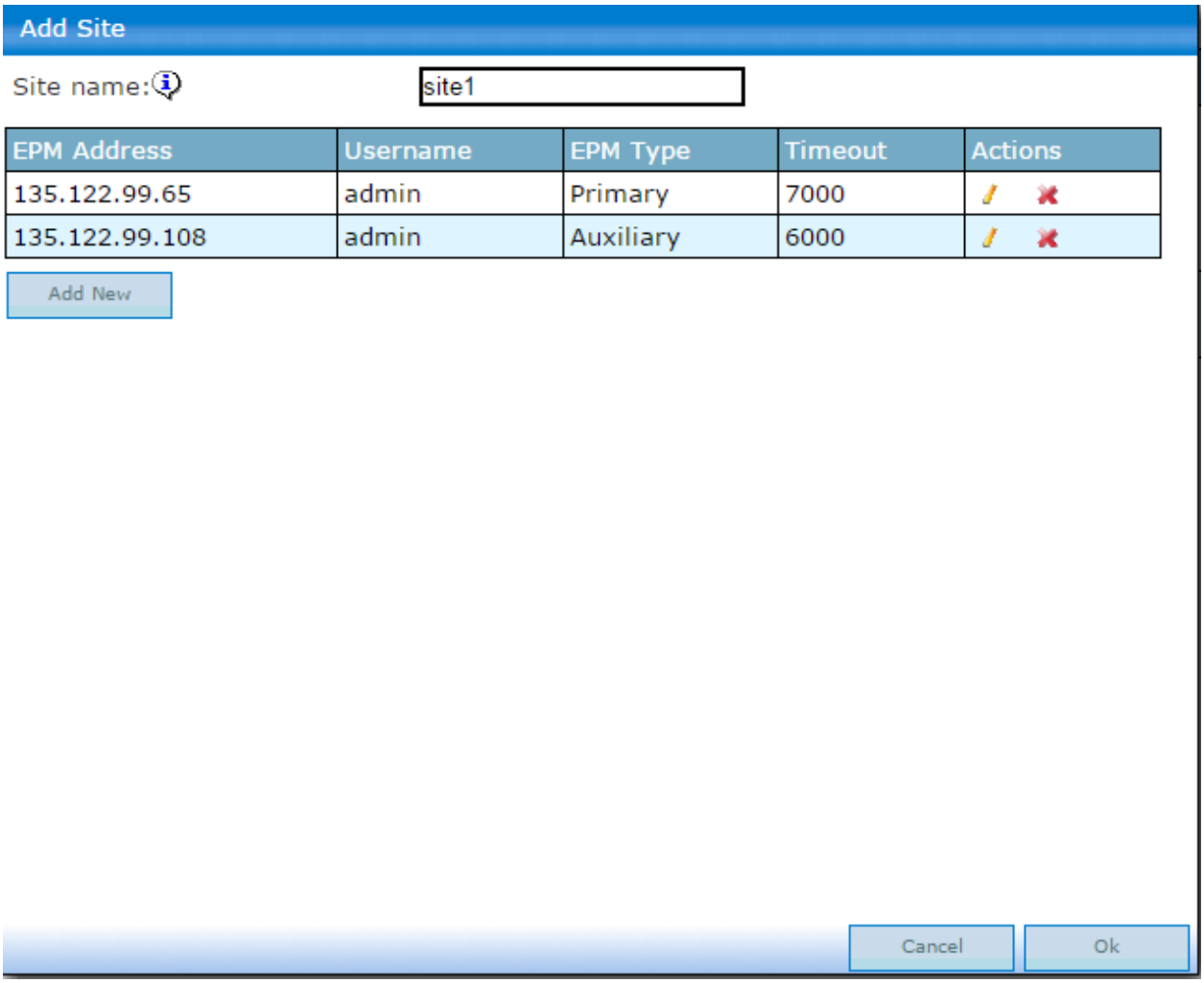


Figure 102 - Available EPMs screen

The primary or auxiliary EPM details can be modified by clicking on Edit button in the column “Actions”. The Auxiliary EPM could be removed by clicking on remove button in the column “Actions”.

The system displays the page Edit EPM as follows.

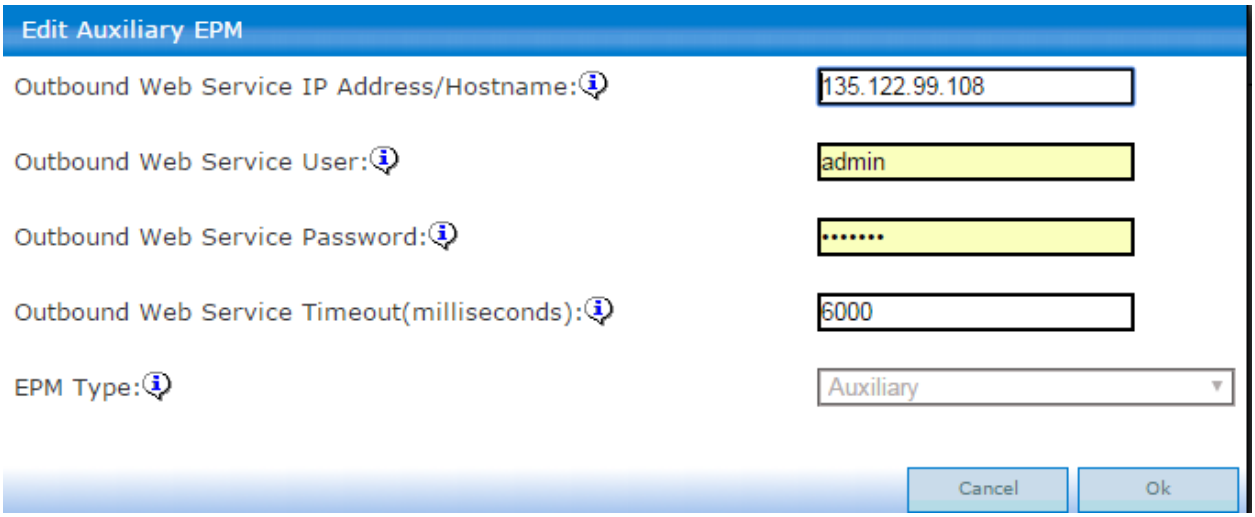


Figure 103 - Edit Auxiliary EPM screen

In this page, any field can be edited except the “EPM Type” field and could be saved by clicking on “OK” button.

Adding Zone Definitions

If the CBA installation has been done with zone being enabled, then the primary EPM details like Primary IP Address, username and password would have been given in the installation itself.

The system displays the Zone Definitions page as follows.

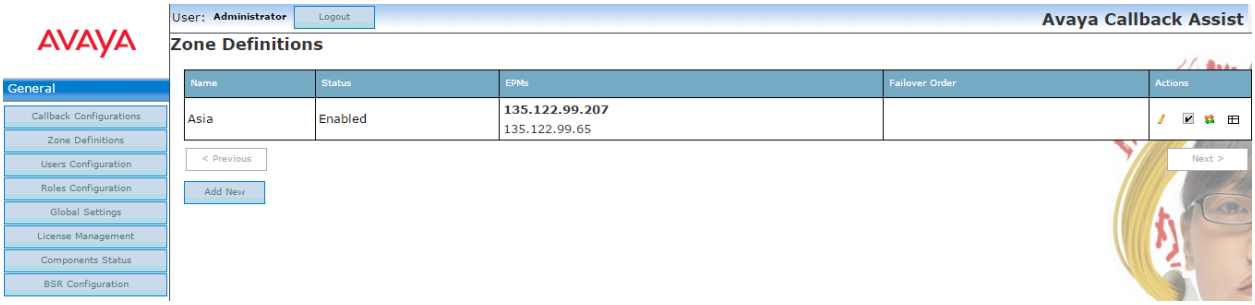


Figure 104 - Zone Definitions screen

The default zone which has been added during installation would be having one primary EPM and multiple auxiliary EPMs could be added further. The new zone definition can be configured with only auxiliary EPMs. And not with primary EPM. The default zone is the zone which has been configured during installation and the other zone definitions whichever are created from Admin UI could not be set as default from Admin UI.

To add a new Zone definition, go to **Zone Definitions** > **Add new**. The system displays the page “Add Zone” as follows.

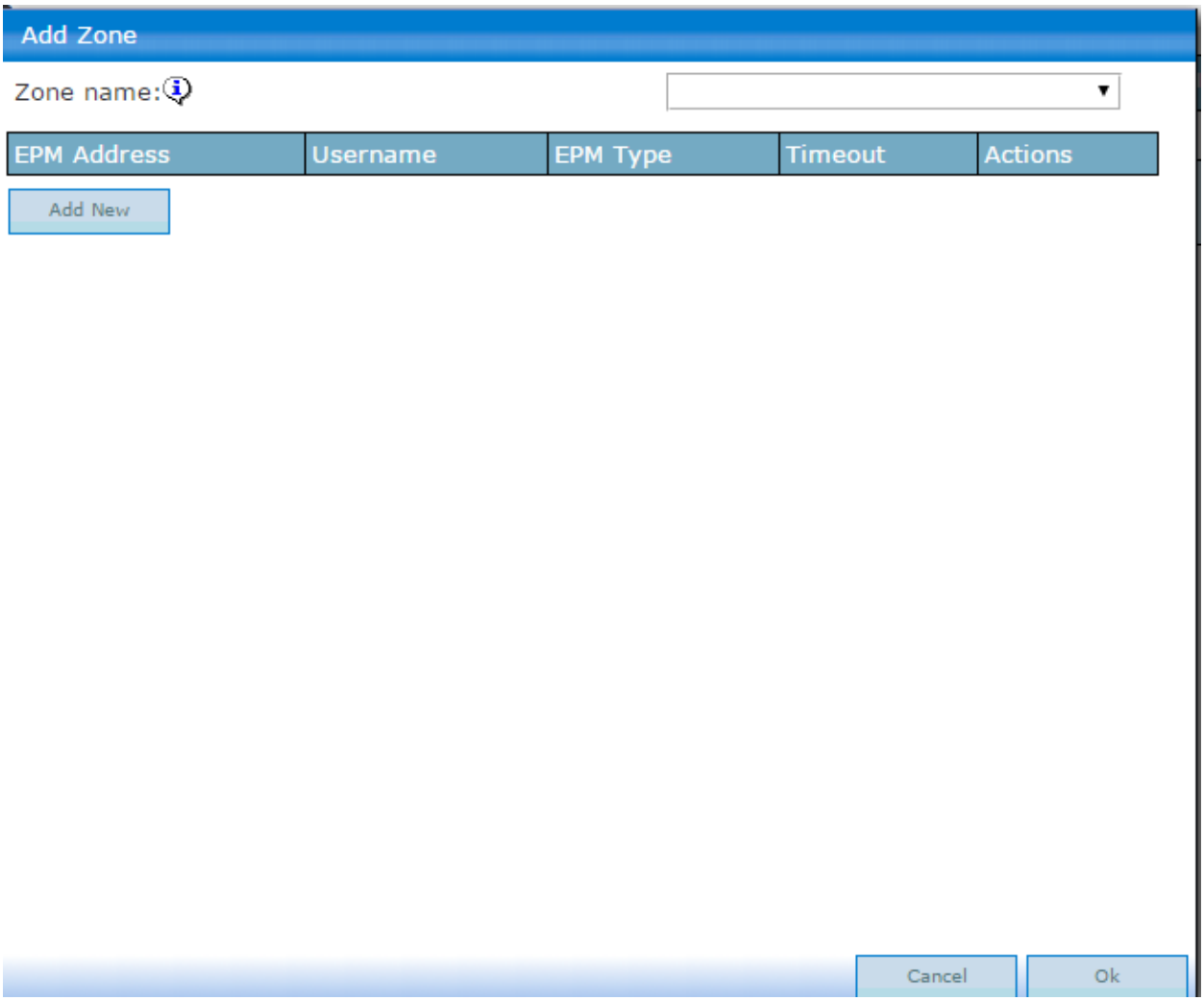


Figure 105 - Add Zone screen

Zone name has to be selected from the drop down “Zone name”. On click of “Add New” button in this page, new auxiliary EPM can be added.

The system will display the page “Add Auxiliary EPM” as follows.

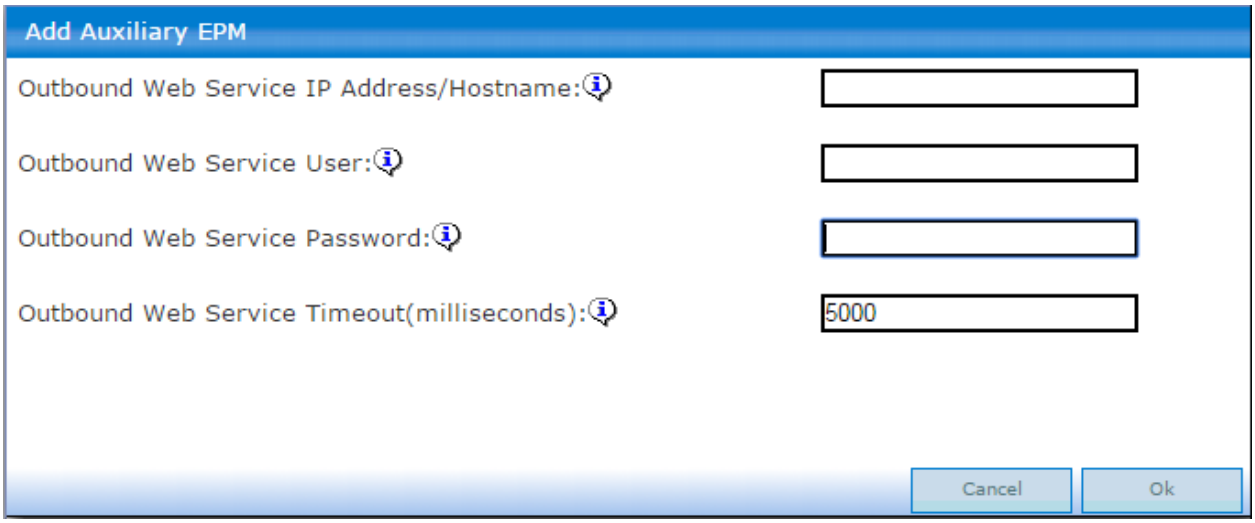


Figure 106 - Add Auxiliary EPM screen

In this page, the timeout field has been filled with default value 5000 and it can be modified again. The IP Address in the EPM Outbound Web Service IP Address is the IP Address of the Experience Portal Server. The User and Password parameters must match the Outcall credentials administered on EPM Settings on Experience Portal Web Administration Portal

The system will display the page as follows once the primary and auxiliary EPM has been added.

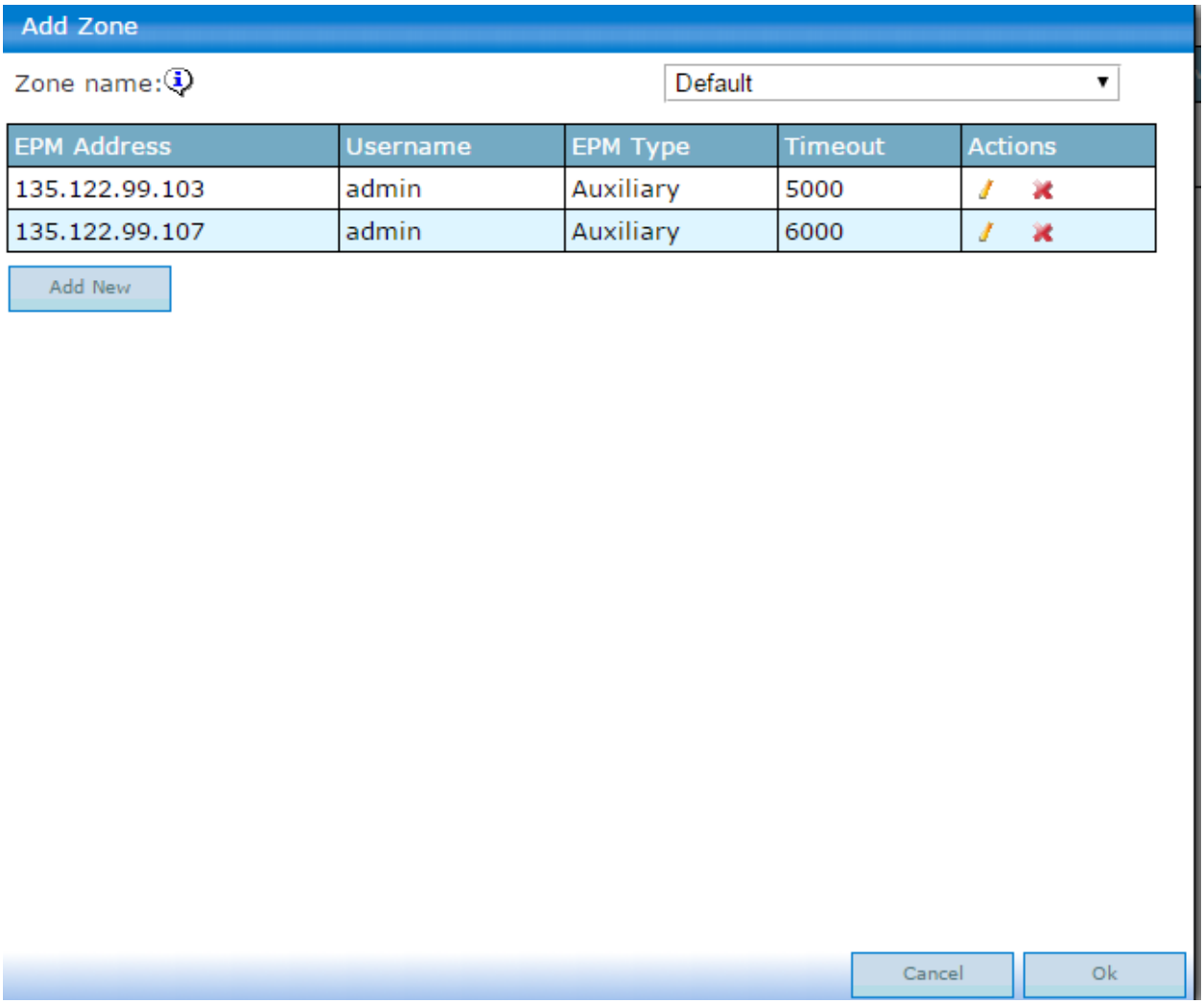


Figure 107 - Available Zones screen

The auxiliary EPM details can be modified by clicking on Edit button in the column “Actions”. The Auxiliary EPM could be removed by clicking on remove button in the column “Actions”.

The system displays the page Edit EPM as follows.

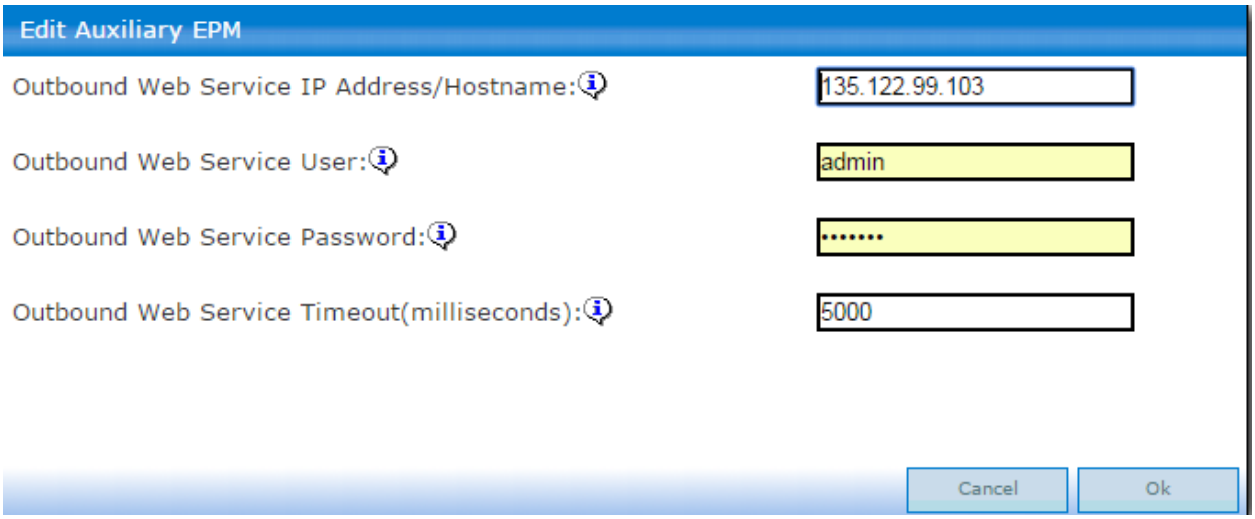


Figure 108 - Edit Auxiliary EPM screen

In this page, any field can be edited and the changes can be saved by clicking on “OK” button.

Defining the Global Time Zone

This configuration applies to all Platforms and Delivery Strategies.

After fresh installation or upgrade of Callback Assist, the system displays a warning message as follows:

```
 ********************************************************
ACTION REQUIRED on Time Zone Configuration
********************************************************
The default Time Zone of CBA is UTC. If your system requires a different
Time Zone
you must manually configure it in the Global Settings of the Admin Portal.
********************************************************
```

This means that the time zone of the Callback Assist server, by default, is set as UTC. It is extremely important to configure this value, as it will affect the behavior of Callback Assist, given that offering and delivery algorithms are based on this time zone.

If you have just performed an, and you want Callback Assist to behave exactly as it did before upgrading, you must set the Calback Assist Global Time Zone as it is in the server's Time Zone. You can check the server's time zone by typing the following command in the server's terminal:

date +%Z

```
[root@denpsqacba64 opt]# date +%Z
MDT
[root@denpsqacba64 opt]#
```

In this case, the output of the command is MDT, so you must configure the Callback Assist Global Time Zone with a Time zone that meets the Mountain Day light Time, such as "America/Denver".

To change the Global Time Zone, you must log into the CBA Web Administration application, and go the **Global Settings** > **General** tab.

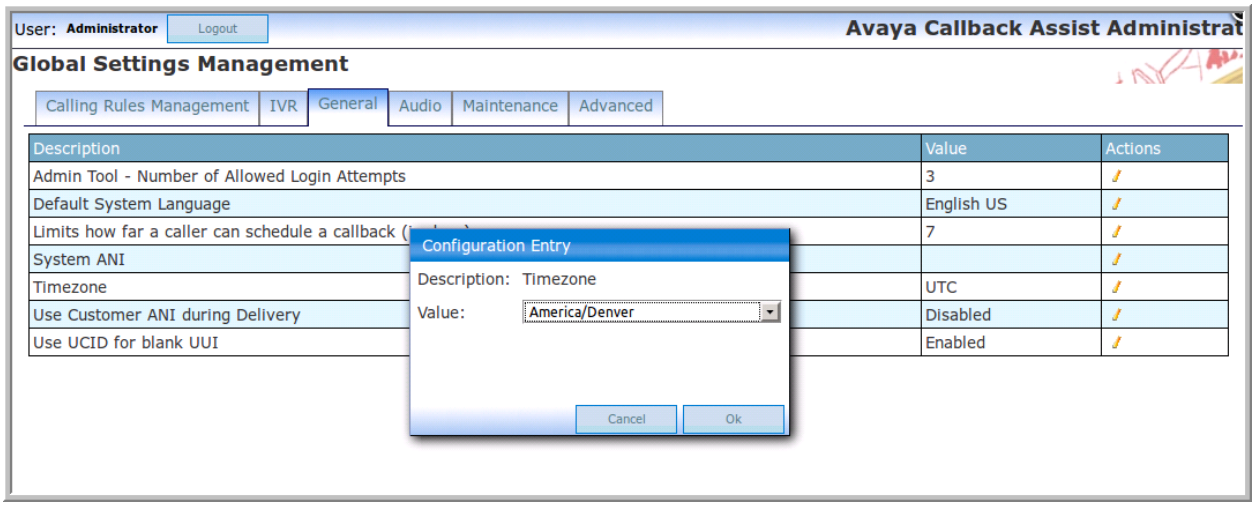


Figure 109 - Timezone screen in Global Settings page

For more information on the Global Time Zone, see the *Global Settings* section in the *Administering Avaya Callback Assist* guide.

Configuring Load Balancer

You can configure a Load Balancer be used with some of the Callback Assist components. The Callback Assist components which support to be behind a Load Balancer are the Admin Portal, the Web Services, the File Server, the BSR, and the IVR applications. Note that neither the Maintenance application nor the Engine application support to be behind a Load Balancer, given that there can only be one Master Maintenance Server and one Master Engine Server at a time.

The Web Services application, the BSR, and the File Server do not need any specific setting or strategy to be configured. As of the Admin Portal and the IVR applications, there are a few considerations to take into account in order to configure a Load Balancer to be in front of these applications.

Configuring a Load Balancer for the Callback Assist Web Administration application

You must configure Load Balancer to use cookie preservation for maintaining session affinity between the client and the server. The system offers several options for maintaining cookies, being most common one to configure Load Balancer to preserve the ***JSESSIONID*** cookie as this cookie is defined in the Callback Assist

Web Administration application. This strategy of preserving a cookie is called as Cookie Learning, and the system uses this strategy when you configure Load Balancer to learn the application cookie.

Configuring a Load Balancer for the IVR applications

The only consideration to take into account for configuring a Load Balancer to be used with the IVR applications is that the Load Balancer must be configured to use cookie preservation in order to maintain session affinity between client and server. There are many options for "cookie" but as the IVR applications have the ___***DDSESSIONID*** cookie already defined by the server, the Load Balancer just needs to preserve it.

This strategy is called "Cookie Learning", and it is used when the Load Balancer is configured to learn the application

Changing Linux box Time Zone

Given that Callback Assists uses most of its time stamps in UTC as each Callback Assist server might have different Time Zones, you must be careful in changing the Linux box time zone as it could affect the Callback Assist behavior.

The following is a list of considerations before you change any Callback Assist Linux box time zones.

1. Stop all the Callback Assist services.
   This step is extremely important as the server's current time zone is read by the JVM when Callback Assist services start.
   To stop all the Callback Assist services, you can use the support script and run the following command from the command line:
   `/opt/Avaya/callbackassist/support/callbackservices.sh stop`
   The command assumes that Callback Assist is installed at the default location.
2. Change the server's time zone.
   Note: There are several methods for changing the server's time zone, and while using any of the methods, you must configure the same time zone in each case for the system.
   For example, if you have configured America/Los Angeles time zone by editing the */etc/sysconfig/clock* file, but there is a line in the */etc/bashrc* file which exports the environment variable TZ with a different time zone, then that could lead to inconsistencies, and then Callback Assist will not work properly.
3. Restart the Callback Assist services.
   You can use the support script to restart all the Callback Assist services at the same time by running the following command from the command line:
   /opt/Avaya/callbackassist/support/callbackservices.sh start
   The command assumes that Callback Assist is installed at the default location.

Note: These considerations are applicable only if you plan to change the Callback Assist Linux box time zones. These considerations do not apply if you plan to change the Callback Assist Global Time Zone.
For more information on how to change the Callback Assist Global Time Zone, see the Global Settings section in the **Administering Callback Assist** guide.

Installing Callback Assist in Dual Server mode

Important: Dual Server deployment is supported from CBA release 4.3.0.0 or later versions.

The CBA installer offers the following options at the start when you run the installer:

1. Callback Assist Core Components with external Database
2. PostgreSQL Database Server for Callback Assist
3. Callback Assist Single Server deployment (Core Components & DB)

If you choose to install CBA by selecting option 1, CBA installs the Callback Assist Core components with an external and already installed CBA database. This is a High Availability / Dual Server mode of installation.

If you choose to install CBA by selecting option 2, CBA installs the PostgreSQL Database Server for Callback Assist. This Database will be used in a High Availability / Dual Server mode by all the CBA Server(s) which will be installed.

If you choose to install CBA by selecting option 3, CBA installs the Callback Assist Core Components and PostgreSQL Database Server in a single server. This is a Single-Server mode of installation.

To install CBA in dual server mode, the PostgreSQL Database Server for Callback Assist (option #2) is the first server that needs to be installed followed by one CBA Application server (option #1).

Installing PostgreSQL Database Server for Callback Assist

To install the PostgreSQL Database for Callback Assist, follow the steps mentioned in section "Installing PostgreSQL Database Server for Callback Assist".

Installing Callback Assist Core Components with external Database

After you install the PostgreSQL Database Server for Callback Assist, you can start installing a CBA Server.

Before installing Callback Assist Core Components with external Database, make sure that you have the PostgreSQL Database Server IP Address or host name, as they would be required during the installation.

Perform the following tasks to install Callback Assist Core Components with external Database in a Dual Server mode:

1. On a Linux shell logged on as root, go to the directory where the CBA installer file is located.
2. Run the following command to extract the files to a temporary folder in your system:
   `tar -xvf callbackassist<version>.tar.`
   One of the extracted files is callback-install.sh which is the main installation script.
3. Run the following command to start the installation process:
   `./callback-install.sh`
   The system writes the command output of the scripts to the standard output and to the callback-

install.log file in your system. Then the system prompts you to choose the required platform for CBA installation.

4. Select [**1)**] as the required installation mode from the following options:

    1. Callback Assist Core Components with external Database
    2. PostgreSQL Database Server for Callback Assist
    3. Callback Assist Single Server deployment (Core Components & DB)

5. Enter the installation directory where to install Callback Assist or just press ***Enter*** to use default location (default directory "/opt").

6. Select SIP as the installation platform.

    Note: Callback Assist does not support Dual Server deployment on CTI (H.323) and AACC/CM platforms.

    For example, to install Callback Assist on a SIP platform, select [**2)**] at the prompt:
    Select [**2)**] SIP as the platform type from the following options and press Enter.

    1. CTI (H.323)
    2. SIP
    3. AACC/CM (SIP Based)

7. Choose desired AAEP system type from the following options and press Enter. (Default is zone).

    Note: This option is applicable only for SIP platform. For site, no additional inputs are required. For zone, user will be asked to provide *Default Zone Name, Primary EPM Information (EPM Address, Username, Password)* as inputs. Option 3 (Skip) is not applicable for dual server deployment, since there won't any existing CBA application server installed previously.

    1. Site
    2. Zone
    3. Skip (if already chosen in another HA server)

8. Choose desired Strategy from the following options and press Enter.

    1. Agent First
    2. Customer First Phantom Pool Strategy
    3. Customer First Priority Queuing Strategy

    The system starts installing SIP platform with the desired strategy.

9. Choose the authentication type

    1. Internal
    2. External (Open ID ex: Google)

    If the authentication type is selected as External then follow the CBA Installation as External Authentication section for more details.

10. Choose whether you need Local Web LM server or not. If you decide to use the Local Web LM server as license server then select “yes”, otherwise “no”.

    Note: If you select “no” at the time of installation and later if you decide to use Local Web LM server as a license server; then the WebLM (tomcat-weblm) service can be restored by running the script reinstallservices.sh from <CBA_INSTALLATION_LOCATION>/support folder.

11. Enter the IP address or the host name of the external Database Server at the prompt.
12. Enter *yes* to confirm the IP address or the host name you entered in the previous step.
13. Enter *no* if any other instance of Callback Assist is sharing the same external database.

    The procedures after this step are similar to the single server installation procedures. For more information, see the respective installation procedures based on SIP platform. After completing the installation, the system displays an installation successful message.

The following example shows the script output of the callback-install.sh script for a Callback Assist installation with an external Database in a SIP environment:

```
[root@server205 4.3.2]# ./callback-install.sh

************************************************************

Avaya Callback Assist Installer

************************************************************

(05:09:48) [ No Callback Assist Components were found. ]

(05:09:48) Please choose the Components to be Installed on this server:


1) Callback Assist Core Components with external Database

2) PostgreSQL Database Server for Callback Assist

3) Callback Assist Single Server deployment (Core Components & DB)

4) Webcallback & DDApps components for Callback Assist

#? 1


(05:11:51) [ About to Install Callback Assist Core Components with
external Database ]


Please, enter the base directory where Avaya Callback Assist will be
installed (default directory /opt):
```

```
Directory where Avaya Callback Assist is going to be installed:
/opt/Avaya/callbackassist


(05:11:53) Checking if there is enough disk space...

(05:11:53) Available disk space is enough.


(05:11:53) Please choose a Platform type for Callback Assist
Application to be Installed from the options below:

1) CTI (H.323)

2) SIP

3) AACC/CM (SIP Based)

#? 2


(05:12:21) Please choose AAEP System type (Default: Zone):

1) Site

2) Zone

3) Skip (if already chosen in another HA server)

#? 1


(05:12:35) Please choose the Delivery Strategy:

1) Agent First Strategy

2) Customer First Phantom Pool Strategy

3) Customer First Priority Queuing Strategy

#? 1


(05:13:06) Please choose an Authentication Type:

1) Internal
```

```
2) External (OpenId ex: Google)

#? 1


(05:13:08) Are you going to use local WebLM service? (yes/no) : yes


(05:13:11) [ Local WebLM server will be added as a daemon service. ]


(05:13:11) [ Internal has been selected as an Authentication Type. ]

(05:13:11) [ Agent First Strategy has been selected as the Delivery
Strategy. ]

(05:13:11) [ Site based AAEP system has been selected for installation.
]

(05:13:11) [ SIP Platform has been selected for installation. ]

(05:13:11) Unpackaging distribution file callbackassist.package...


(05:13:23) Creating callback Group...

(05:13:23) Creating callback User...

(05:13:24) Performing JDK silent install...

(05:13:28) JDK installed.

java version "1.7.0_75"

Java(TM) SE Runtime Environment (build 1.7.0_75-b13)

Java HotSpot(TM) Server VM (build 24.75-b04, mixed mode)


(05:13:28) Please enter the Callback Assist Database Server IP Address
or Host name: 10.130.124.27

(05:14:41) You've entered 10.130.124.27. Is this correct? (Yes/No) yes

(05:14:42) Checking PostgreSQL Server version on 10.130.124.27...

(05:14:43) PostgreSQL version is OK.
```

```
(05:14:43) Checking for 'callback' User and Database on
10.130.124.27...

(05:14:43) User and Database are OK.



(05:14:43) Is there any other CBA instance sharing this same Database?
(Yes/No) no

(05:14:46) Installation of Callback Assist Maintenance done.

(05:14:46) Check
/opt/Avaya/callbackassist/maintenance/logs/execution.log file for
Callback Assist Maintenance service startup details.

(05:14:46) Installation of Callback Assist Engine done.

(05:14:46) Check /opt/Avaya/callbackassist/engine/logs/execution.log
file for Callback Assist Engine service startup details.

(05:14:46) 64 bit Architecture detected ...

(05:14:46) Installing file server for Red Hat 6.x 64 bits

(05:14:47) Local Ip Address: 135.122.99.205

(05:14:47) INFO: Using hostname/FQDN: server205.avayacba.com to run the
file-server.

(05:14:47) Configuring File Server IP Address...

(05:14:47) Done.

(05:14:47) Changing SELinux context to some CBA files...

(05:14:47) Waiting for the Database Schema to be created or updated...

Migration successful

(05:14:51) Database Schema was successfully created or updated.

(05:14:51) Platform successfully set.

(05:14:52) AAEP System type successfully set.

(05:14:52) Deployment type successfully set.

(05:14:52) Storing Release version and Build Number...

(05:14:52) Release version and Build number successfully stored.

(05:14:52) Running Platform dependant changes...
```

```
Migration successful

(05:14:53) Database Schema was successfully updated.

(05:14:53) Running Deployment Type changes...

Migration successful

(05:14:54) Database Schema was successfully updated.

(05:14:54) Updating authentication type into database ...

(05:14:54) Installing weblm Tomcat service (tomcat-weblm)...

(05:14:54) Installation of tomcat-weblm done.

(05:14:54) Installing adminapp Tomcat service (tomcat-adminapp)...

(05:14:54) Installation of tomcat-adminapp done.

(05:14:54) Installing ddapps Tomcat service (tomcat-ddapps)...

(05:14:54) Installation of tomcat-ddapps done.

(05:14:54) Installing webcallback Tomcat service (tomcat-
webcallback)...

(05:14:55) Installation of tomcat-webcallback done.

(05:14:55) Deploying Applications...

(05:14:55) Moving tomcat realm related jar files

(05:14:55) Installation of Tomcat instances done.

(05:14:55) Installing BSR Server.  This operation may take several
minutes...

(05:14:55) Local Ip Address: 135.122.99.205

(05:14:55) Setting ownership to callback user.

Starting Callback Assist Engine...

Callback Assist Engine Started.                              [  OK  ]

Callback Assist Engine ( pid 24935 ) is running...

Starting Callback Assist Maintenance...

Callback Assist Maintenance Started.                         [  OK  ]

Callback Assist Maintenance ( pid 25016 ) is running...
```

```
Starting tomcat-weblm...                                    [  OK  ]

tomcat-weblm ( pid 25107 ) is running...

Starting tomcat-adminapp...                                 [  OK  ]

tomcat-adminapp ( pid 25206 ) is running...

Starting tomcat-webcallback...                              [  OK  ]

tomcat-webcallback ( pid 25334 ) is running...

Starting tomcat-ddapps...                                   [  OK  ]

tomcat-ddapps ( pid 25434 ) is running...

Starting CBA BSR Server service...                          [  OK  ]

CBA BSR Server ( pid 25521 ) is running...

Starting Callback Assist File Server...

Callback Assist File Server Started.                        [  OK  ]

Callback Assist File Server ( 25690 ) is running...         [  OK  ]

Callback Assist File Server Ping test                       [  OK  ]

Callback Assist File Server Read/Write cycle Test           [  OK  ]

Callback Assist File Server Ring Status is Up               [  OK  ]

Callback Assist File Server is joined to a cluster          [  NO  ]

If this is not a HA deployment, then disregard this warning.

(05:15:25) [ Installation of Avaya Callback Assist (SIP - Agent First
Strategy) completed. ]



*********************************************************

ACTION REQUIRED on Time Zone Configuration

*********************************************************

The default Time Zone of CBA is UTC. If your system requires a
different Time Zone

you must manually configure it in the Global Settings of the Admin
Portal.

*********************************************************
```

As a reference, in the installation output shown above there were 2 servers involved:

- PostgreSQL Database server *[10.130.124.27]*
- Callback Assist Server with external Database *[135.122.99.205]*

Post-installation steps

Follow the steps mentioned in section "Post-installation Steps".

Changing the existing installation platform

You can install Callback Assist on one of the three platforms as follows:

- CTI (H.323)
- SIP
- AACC/CM (SIP Based)

You can change the current platform anytime as required by reinstalling the same version of CBA available in your system. If there is a later version of CBA available, you can run the latest installer and choose the required platform.

For example, if your current installation is Callback Assist 4.1, SIP (Customer First Priority Queuing), where SIP is the Platform and Customer First Priority Queuing is the Delivery Strategy, you can change the platform to CTI or AACC/CM or SIP with Agent First or Customer First Phantom Pool as the Delivery Strategy, and also you can change the authentication type from internal to external or vice versa. When you reinstall CBA, select the required platform (and the required Delivery Strategy if the platform is SIP) during the prompt.

Perform the following tasks to change the existing installation platform:

1. On a Linux shell logged on as root, go to the directory where the Callback Assist installer file is located. This can be the existing installer or the latest version of the installer.
2. Run the following command to extract the files to a temporary folder in your system:
   `tar -xvf callbackassist<version>.tar`
   One of the extracted file is callback-install.sh which is the main installation script.
3. Run the following command to start the installation process:
   `./callback-install.sh`
4. Choose to install Callback Assist Single Server deployment (Core Components and DB) (option #3).
   The system detects that there is already a Callback Assist installed, and asks if you want to update or reinstall.
5. Type *Yes* to confirm updating or reinstalling Callback Assist.
   The system prompts you to choose the required platform for Callback Assist installation.
6. Select the new required platform type from the following options and press Enter.

   1) CTI (H.323)

   2) SIP

   3) AACC/CM (SIP Based)

(Optional) If you select SIP as the Platform type, the system asks you to choose a Delivery Strategy

Select the required SIP environment from the following options and press Enter.

1) Agent First

2) Customer First Phantom Pool

3) Customer First Priority Queuing

7. Choose the authentication type
    1. Internal
    2. External (Open ID ex: Google)

If the authentication type is selected as External then follow the CBA Installation as External Authentication section for more details.

The system starts installing CBA in the platform you selected. After completing the installation, the system displays an installation successful message.

Note: This method of changing the existing platform is valid only for a Single Server deployment. For changing the existing platform in a High Availability deployment, contact Avaya Professional Services or Avaya Support.

Uninstalling Callback Assist

To uninstall Callback Assist, you must log on as root user and run the callback-uninstall.sh script copied during the installation to <INSTALL-PATH>/Avaya/callbackassist.

The following is an example of an execution output of the above script for a Single Server deployment:

```
[root@RHEL60EPAEP dev]# /opt/Avaya/callbackassist/callback-uninstall.sh
************************************************************
Avaya Callback Assist Uninstaller
************************************************************
*** WARNING: You are about to Uninstall Avaya Callback Assist.
*** Note that all Callback data will be LOST.
(16:32:52) Are you sure you want to Uninstall Avaya Callback Assist?
(yes/no): yes
(16:32:53) Installation of tomcat-adminapp found.
(16:32:53) Waiting for tomcat-adminapp process to terminate...
Stopping tomcat-adminapp... [ OK ]
(16:32:56) Uninstallation of tomcat-adminapp done.
(16:32:56) Installation of tomcat-ddapps found.
(16:32:56) Waiting for tomcat-ddapps process to terminate...
Stopping tomcat-ddapps... [ OK ]
(16:32:58) Uninstallation of tomcat-ddapps done.
(16:32:59) Installation of tomcat-webcallback found.
(16:32:59) Waiting for tomcat-webcallback process to terminate...
Stopping tomcat-webcallback... [ OK ]
(16:33:01) Uninstallation of tomcat-webcallback done.
(16:33:01) Installation of Callback Assist Engine found.
Stopping Callback Assist Engine...
Callback Assist Engine Stopped. [ OK ]
(16:33:01) Uninstallation of Callback Assist Engine done.
(16:33:01) Installation of Callback Assist Maintenance found.
Stopping Callback Assist Maintenance...
Callback Assist Maintenance Stopped. [ OK ]
(16:33:01) Uninstallation of Callback Assist Maintenance done.
```

```
(16:33:01) Installation of File Server found.
Stopping Callback Assist File Server...
Callback Assist File Server Stopped. [ OK ]
(16:33:04) Uninstallation of File Server done.
(16:33:04) Uninstalling Callback Assist Database...
(16:33:15) PostgreSQL directory successfully removed.
*** [ Uninstallation of Callback Assist Database done. ]
*** [ Uninstallation of Avaya Callback Assist done. ]
[root@RHEL60EPAEP dev]#
```

The uninstallation process for a High Availability deployment is quite similar to a Single Server deployment, as the procedures to uninstall either the Callback Assist components with external Database, or the PostgreSQL Database Server for Callback Assist are the same as in a Single Server deployment. The Callback Assist uninstaller script automatically detects which Callback Assist components are installed and will uninstall them accordingly.

Important: When uninstalling a server with Callback Assist Components with external Database, if the File Server was joined with another Callback Assist server, then all the audio files hosted in the server that is being uninstalled will be handed off to the other Callback Assist servers. This process might take some time. Aborting the uninstallation process while Callback Assist File Server is handing off its audio files will lead to other Callback Assist servers being in an inconsistent state. Contact Avaya Support if the uninstallation process did not finish gracefully in a High Availability deployment.

Reinstalling Callback Assist

When you try to install the same version of Callback Assist as the one already installed in your system, the installer detects that a reinstall is being performed and prompts you to either proceed or abort with the reinstall operation.

Note: When performing a reinstallation operation, the system preserves all the existing data stored in the database, the stored audio files, license files, and configuration files.

You cannot reinstall the Callback Assist Components with external Database, as you can reinstall only a Single server instance. If you want to reinstall Callback Assist Components with external Database, contact Avaya Professional Services or Avaya Support.

The following example shows the initial part of the reinstallation output. The later part is similar to the installation output as shown in the earlier section of this document.

```
[root@RHEL60EPAEP dev]# ./callback-install.sh
 ************************************************************
Avaya Callback Assist Installer
************************************************************
*** ATTENTION: A Single Server deployment of Avaya Callback Assist was
detected.
*** The current installed version is 4.3.0.0-GA, build 46426.
*** You are about to install 4.3.1.0-GA-SNAPSHOT, build 46448.
***
*** The current configured Platform is SIP (Customer First Priority
Queuing Strategy) - Authentication Type: Internal
***
*** Do you want to proceed with an upgrade? (yes/no): (22:44:17) Checking
if there is enough disk space...
(22:44:17) Available disk space is enough.


(22:44:17) Please choose a Platform type for Callback Assist Application
to be Installed from the options below:
1) CTI (H.323)
2) SIP
3) AACC/CM (SIP Based)
#? 2
 (14:12:58) Please choose the Delivery Strategy:
1) Agent First Strategy
2) Customer First Phantom Pool Strategy
3) Customer First Priority Queueing Strategy
```

```
#? 1
```

Stop and Start Callback Assist in HA mode

Stop Callback Assist in HA mode

1. Stop standby CBA Server
   /sbin/service callbackassist stop
2. Stop Master CBA Server
   /sbin/service callbackassist stop
3. Stop Secondary CBA Database Server (applicable if replication is setup)
   /sbin/service cba-postgresql stop
4. Stop Primary CBA Database Server
   /sbin/service cba-postgresql stop

Start Callback Assist in HA mode

1. Start Primary CBA Database Server
   /sbin/service cba-postgresql start
2. Start Secondary CBA Database Server (applicable if replication is setup)
   /sbin/service cba-postgresql start
3. Start Master CBA Server
   /sbin/service callbackassist start
4. Start standby CBA Server
   /sbin/service callbackassist start

Backup and Rollback Callback Assist

This section enlists the steps required to take backup of the existing Callback Assist installation and rollback procedure of backup files.

Backup

The callback Assist backup is performed at the file system level. The entire Callback Assist installation directory is saved as a tar file.

1. On a Linux shell logged on as root user create a temporary backup directory where the Callback Assist backup tar file is saved
   mkdir /tmp/cba_backup

2. Stop all CBA services (For HA deployments refer Stop Callback Assist in HA mode)

/sbin/service callbackassist stop

3. Create tar file of the entire Callback Assist installation directory (replace YYYYMMDD with current date)
   tar -czvf /tmp/cba_backup/cba_bkp_YYYYMMDD.tar.gz /opt/Avaya/callbackassist

4. The cba_bkp_YYYYMMDD.tar.gz file is created in /tmp/cba_backup/ folder

5. Start Callback Assist (For HA deployments refer Start Callback Assist in HA mode)
   /sbin/service callbackassist start

Note:

On a HA Deployment perform these steps on all the servers including database servers.

Rollback

Perform the below steps if Callback Assist needs to be rolled back to a previously installed version/state.

Prerequisites

- Backup files are already available
- Existing Callback Assist installation on all the servers are completely uninstalled.

Rollback Steps

1. Navigate to /opt/Avaya directory and extract the backup tar file cba_bkp_YYYYMMDD.tar.gz
   cd /opt/Avaya
   tar -zxvf /tmp/cba_backup/cba_bkp_YYYYMMDD.tar.gz --directory /

2. Edit the reinstallservices script file. (Skip this step for 4.1.8 and higher versions)
   vi /opt/Avaya/callbackassist/support/reinstallservices.sh

   Add the below lines at the end of the file and save the file

```
# Restoring BSR Server Daemon
BSRSERVER_BIN=$CBA_INSTALLATION_DIR/bsr-server/bin/ICRBSRServer.sh
if [[ -f $BSRSERVER_BIN && ! -f /etc/init.d/$DAEMON_BSR_SERVER ]]; then
        ln -s $BSRSERVER_BIN /etc/init.d/$DAEMON_BSR_SERVER
        /sbin/chkconfig --add $DAEMON_BSR_SERVER
        echo "BSR Server Daemon Restored."
fi
# Create callback group and user
createCallbackUser $CBA_INSTALLATION_DIR
```

3. Reinstall the callbackassist services
   cd /opt/Avaya/callbackassist/support
   ./reinstallservices.sh

4. On a HA Deployment repeat the steps 1 & 2 on all the servers before proceeding to the next step

5. Start all CBA services (For HA deployments refer Start Callback Assist in HA mode)
   /sbin/service callbackassist start

Persona Administration

Callback Assist offers the ability to add different personas to the IVR application languages. With this feature the language audio recordings in the IVR applications can be delivered using different personas. The default audio recordings provided by callback assist out of the box for the existing languages are the default personas for the languages. After the required personas are created, the persona can be assigned to callback configurations through the Callback Administration Web interface.

It is the responsibility of the customers to re-record all the audio files in the default persona of the selected language with the new persona recording and add it to Callback Assist.

Default Persona

All languages supported by callback assist have a default persona. It contains the pre-recorded audio files provided by Callback Assist for the languages.
The audio files of the default persona of the languages are available under the following directories.

<callback-install-directory>/apache-tomcat-ddapps/webapps/CBAPhrases/custom/customer/<language-name>/<locale>/default

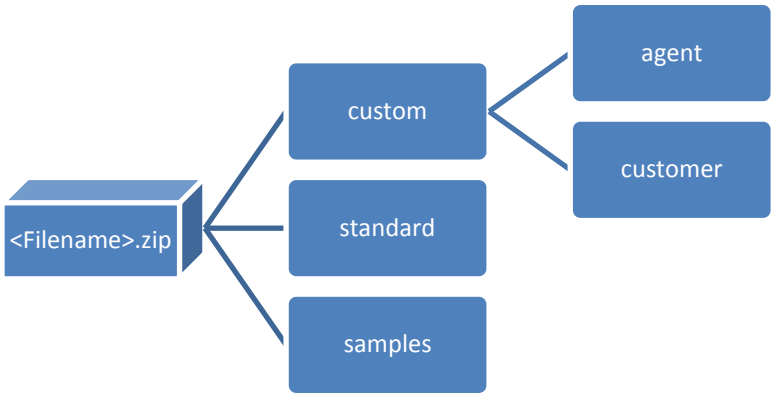
<callback-install-directory>/apache-tomcat-ddapps/webapps/CBAPhrases/custom/agent/<language-name>/<locale>/default

<callback-install-directory>/apache-tomcat-ddapps/webapps/CBAPhrases/standard/<language-name>/<locale>/default

<callback-install-directory>/apache-tomcat-ddapps/webapps/CBAPhrases/samples/<language-name>/<locale>/default

Persona Package file

The persona package file is a zip file which bundles all the required audio files for a persona. This file is used to create a new persona or update an existing persona.
To create a Persona Package file, all audio files in the default persona directories of a selected language should be re-recorded with the same filenames and packaged as a zip file in the predefined format mentioned below.
The zip file should have the following directory structure with the corresponding audio files. Note that the directory names should be in lowercase.



custom/agent - Directory for the agent messages
custom/customer - Directory for the customer messages
standard - Directory for the audio files used by localization bundle.
samples - Optional directory and is a backup place holder for messages uploaded through admin UI.

Add Persona

A new persona for a language can be created using the **administerpersona.sh** shell script located in the <callback-installation>/support folder.

Steps to add a new persona

Persona package file with the recordings should be available before running the following steps.
Refer Persona Package file for more information.

1. In the CBA server Linux console logged on as root user, run the **administerpersona.sh** script
2. Select **1) Add or Update Persona** in the "**Choose an action**" menu
3. In the "**Select language**" menu enter the corresponding numeric value of the language
4. Enter the name of the persona that needs to be created when the script prompts "**Enter persona Name :**"
5. Enter the database server IP address/hostname when the script prompts "**Please enter the Callback Assist Database Server IP Address or Host name:**" . This step is applicable only for HA deployments.
6. Enter the file path of the zip file containing the audio files for "**Enter persona filename/path :**"
7. The script will add a new database entry and extract the contents of the zip file to the required folders.
8. On a HA deployment repeat the above steps on both the CBA servers.

A sample script output is shown below

```
[root@RHEL5364MCES support]# ./administerPersona.sh
********* Callback Assist - Persona Administration Utility*************
```

```
(01:13:04) Choose an action
1) Add or Update Persona
2) Remove Persona
#? 1
(01:13:07) [ Add or Update Persona has been selected ]
(01:13:07) Select language:
1) English US                10) Polish
2) English United Kingdom    11) Russian
3) Spanish (Latin American)  12) Turkish
4) German                    13) Cantonese Chinese
5) French                    14) Mandarin Chinese
6) Portuguese (Brazilian)    15) Arabic
7) Japanese                  16) Korean
8) Italian                   17) Custom
9) Dutch
#? 10
(01:13:10) [ Polish has been selected as the language ]
(01:13:10) Enter persona Name : male
(01:13:13) Database component not present.
(01:13:13) Please enter the Callback Assist Database Server IP Address or Host name:
135.122.99.206
(01:13:21) You've entered 135.122.99.206. Is this correct? (Yes/No) yes
(01:13:23) Enter persona filename/path : polish-male.zip
(01:13:32) Extracting Persona Files
(01:13:32) Creating new Database entry for language Polish and persona male
(01:13:33) Creating directories for language Polish and persona male
(01:13:33) Copying new standard files
(01:13:33) Copying new customer files
(01:13:33) Copying new agent files
(01:13:33) Copying new sample files
(01:13:33) The persona audio files are deployed. Script completed.
[root@RHEL5364MCES support]#
```

Update Persona

The audio files of an existing persona can be updated using the **administerpersona.sh** shell script.

Steps to update an existing persona

Persona package file with the recordings should be available before running the following steps.
Refer Persona Package file for more information.

1. In the CBA server Linux console logged on as root user, run the **administerpersona.sh** script
2. Select **1) Add or Update Persona** in the "Choose an action" menu
3. In the "**Select language**" menu enter the corresponding numeric value of the language
4. Enter the name of the persona that needs to be updated when the script prompts "**Enter persona Name :**". Note that if a different persona name is entered the script will create a new persona.
5. Enter the database server IP address/hostname when the script prompts "**Please enter the Callback Assist Database Server IP Address or Host name:**" . This step is applicable only for HA deployments.
6. Enter the file path of the zip file containing the audio files for "**Enter persona filename/path :**"
7. Enter **yes** when the script prompts "**Persona <persona name> directory exists for language <language name>. All existing files will be removed and new files will be copied. Do you want to proceed? (yes/no) :**"
8. The script will create a backup of the old audio files as a zip file in the path **<callback-installation>/support/persona_backup/<language name>-<persona name>.zi**p and replace the existing audio files in callback installation with the new files from the zip file.
9. On a HA deployment repeat the above steps on both the CBA servers.

A sample script output is shown here.

```
[root@RHEL5364MCES support]# ./administerPersona.sh
********* Callback Assist - Persona Administration Utility*************
(01:19:24) Choose an action
1) Add or Update Persona
2) Remove Persona
#? 1
(01:19:26) [ Add or Update Persona has been selected ]
(01:19:26) Select language:
1) English US              10) Polish
2) English United Kingdom  11) Russian
3) Spanish (Latin American) 12) Turkish
4) German                  13) Cantonese Chinese
```

5) French 14) Mandarin Chinese

6) Portuguese (Brazilian) 15) Arabic

7) Japanese 16) Korean

8) Italian 17) Custom

9) Dutch

#? 10

(01:19:29) [Polish has been selected as the language]

(01:19:29) Enter persona Name : male

(01:19:32) Database component not present.

(01:19:32) Please enter the Callback Assist Database Server IP Address or Host name: 135.122.99.206

(01:19:44) You've entered 135.122.99.206. Is this correct? (Yes/No) yes

(01:19:49) Enter persona filename/path : polish-male.zip

(01:20:00) Extracting Persona Files

(01:20:00) Database entry already exists for language Polish and persona male

(01:20:00) Persona male directory exists for language Polish. All existing files will be removed and new files will be copied. Do you want to proceed? (yes/no) : yes

(01:20:06) Creating a backup of existing files for language Polish and persona male

(01:20:06) Backup zip file polish-male.zip created in /opt/Avaya/callbackassist/support/persona_backup

(01:20:06) Removing existing standard files

(01:20:06) Removing existing customer files

(01:20:06) Removing existing agent files

(01:20:06) Removing existing sample files

(01:20:06) Creating directories for language Polish and persona male

(01:20:06) Copying new standard files

(01:20:06) Copying new customer files

(01:20:06) Copying new agent files

(01:20:06) Copying new sample files

(01:20:06) The persona audio files are deployed. Script completed.

[root@RHEL5364MCES support]#

Remove Persona

Persona created for an existing language can be removed from Callback Assist using the **administerpersona.sh** shell script.

Steps to remove an existing persona

1. In the CBA server Linux console logged on as root user, run the **administerpersona.sh** shell script
2. Select **2) Remove Persona** option in the "**Choose an action**" menu
3. Select the language from which persona needs to be removed in the "**Select language**" menu.
4. Enter the name of the persona that needs to be deleted when the script prompts "**Enter persona Name :**".
5. Enter the database server IP address/hostname when the script prompts "**Please enter the Callback Assist Database Server IP Address or Host name:**" . This step is applicable only for HA deployments
6. Enter **yes** when "**Persona <persona name> in language <language name> will be removed including all files. Do you want to proceed? (yes/no) :**" is prompted.
7. The script will create a backup of the audio files as a zip file in the path **<callback-installation>/support/persona_backup/<language name>-<persona name>.zip**
8. The script will remove the persona from DB and the corresponding persona audio directories from the file system.
9. On a HA deployment repeat the above steps on both the CBA servers.

A sample script output is shown below.

```
[root@RHEL5364MCES support]# ./administerPersona.sh
********* Callback Assist - Persona Administration Utility*************
(01:22:47) Choose an action
1) Add or Update Persona
2) Remove Persona
#? 2
(01:22:49) [ Remove Persona has been selected ]
(01:22:49) Select language:
1) English US              10) Polish
2) English United Kingdom  11) Russian
3) Spanish (Latin American) 12) Turkish
4) German                  13) Cantonese Chinese
5) French                  14) Mandarin Chinese
6) Portuguese (Brazilian)  15) Arabic
7) Japanese                16) Korean
```

```
8) Italian              17) Custom
9) Dutch
#? 10
(01:22:51) [ Polish has been selected as the language ]
(01:22:51) Enter persona Name : male
(01:22:53) Database component not present.
(01:22:53) Please enter the Callback Assist Database Server IP Address or Host name: 135.122.99.206
(01:23:02) You've entered 135.122.99.206. Is this correct? (Yes/No) yes
(01:23:05) Persona male in language Polish will be removed including all files.
 Do you want to proceed? (yes/no) : yes
(01:23:09) Creating a backup of existing files for  language Polish and persona male
(01:23:09) Backup zip file polish-male.zip created in
/opt/Avaya/callbackassist/support/persona_backup
(01:23:10) Removing persona male for language Polish from Database
(01:23:10) Persona male for language Polish Removed from Database.

(01:23:10) Removing existing standard files
(01:23:10) Removing existing customer files
(01:23:10) Removing existing agent files
(01:23:10) Removing existing sample files
(01:23:10) Persona male removed from language Polish.
(01:23:10) Script completed
[root@RHEL5364MCES support]#
```

Assigning Persona to Callback Configurations

Refer the **Callback Configurations -> Customer Tab** and **Callback Configurations -> Agent Tab** sections of the "*Administering Avaya Callback Assist"* user guide.

Zoning Administration

Callback Assist will support migrating from a multiple AAEP systems configuration (i.e. multiple sites) to a single AAEP 7.0.1 system using multiple zones from version 4.4. Upgrading to 4.4 will also preserve previous site configurations.

Customer will be able to migrate between Sites and Zones at any point. New script (*administerZone.sh*) is shipped along with the installer and will be available on the *<callback-installation>/support* directory to perform the migration. This migration is applicable only for SIP platform.

Sites to Zones Migration

If CBA is installed newly with version 4.4 by choosing *"Site"* as AAEP system type, or if CBA is upgraded to version 4.4 from any of previous version, and customer plans to use *"Zoning feature"*, this migration script will help to migrate from sites to zones.

Steps to do Migration

1. In the CBA server Linux console logged on as root user, run the **administerZone.sh** script
2. Script will find current configuration (Site or Zone) and request user to confirm for doing the migration.
3. If user confirms the migration, current *"Default site name"* and *"Primary EPM address"* will be displayed. User can either choose to update this information along with migration, or keep the same information during migration.
4. If there is no issue at backend, migration will be done and success message will be displayed along with few notes to the administrator.
5. Following tasks are performed by this migration;
    - CBA configuration will be updated to use *"Zones"*.
    - All sites will be converted to zones and their status will be made as disabled. Administrator can use web admin to enable required zones later.
    - Primary EPMs of all non-default sites will be changed as auxiliary. Hence, all zones will have only auxiliary EPMs, and only Default zone will have Primary EPM.
6. User needs to logout from web admin and login again to see the migration changes.
7. After this migration, administrator needs to edit and save each zone through web admin to update *"VPMS Zone Id"* into database from Voice Portal Web services.
8. Administrator can add new zones and edit existing zone information through web admin.
9. CBA services should be restarted after migration.

    service callbackassist stop

    service callbackassist start

A sample script output is shown below

```
[root@cbadb81 support]# ./administerZone.sh

*** CBA is currently configured to use AAEP Sites ***
***
*** Do you want to migrate from Site to Zone? (yes/no):
y

*** Default Zone will have following information: Zone Name [ default site ] and Primary EPM
[ 135.122.99.207 ]. Do you want to change this? (yes/no):
y

Default Zone Name: default zone

Primary EPM IPAddress/HostName: 135.122.99.207

Primary EPM User Name: admin

Primary EPM Password: *******


(06:28:59) [ CBA configuration is updated to use AAEP Zones. ]

**************************** ATTENTION ****************************************
 a) Login to web admin; and edit, save each zone to assign VPMS Zone Id.
 b) All zones will be in disabled state after migration. Enable required zones through web admin.
 c) You can also add new zone, edit existing zone through web admin to match your
configuration.
 d) User provided zone information is not validated with Voice Portal system.
*******************************************************************************
[root@cbadb81 support]#
```

Zones to Sites Migration

If CBA is installed newly with version 4.4 by choosing *"Zone"* as AAEP system type, or if CBA is upgraded to version 4.4 from any of previous version and migrated to use *"Zones"*, and later if customer plans to go back using *"AAEP Sites"*, this migration script will help to migrate from Zones to Sites.

Steps to do Migration

1. In the CBA server Linux console logged on as root user, run the ***administerZone.sh*** script.
2. Script will find current configuration (Site or Zone) and request user to confirm for doing the migration.
3. If user confirms the migration, current *"Default zone name"* and *"Primary EPM address"* will be displayed. User can either choose to update this information along with migration, or keep the same information during migration.
4. Each non-default zone will be displayed with their Auxiliary EPM list; user should choose one EPM as primary from the list. Since Site should have one primary EPM, this step is mandatory.
5. If there is no issue at backend, migration will be done and success message will be displayed.
6. Following tasks are performed by this migration;
   - CBA configuration will be updated to use *"Sites"*.
   - All zones will be converted to sites.
   - One auxiliary EPM will be updated as *"Primary"* for each non-default site.
7. User needs to logout from web admin and login again to see the migration changes.
8. Administrator can add new sites, and edit existing site information through web admin.
9. CBA services should be restarted after migration.

   service callbackassist stop

   service callbackassist start

A sample script output is shown below

```
[root@cbadb81 support]# ./administerZone.sh



*** CBA is currently configured to use AAEP Zones ***
***
*** Do you want to migrate from Zone to Site? (yes/no):
y

*** Default Site will have following information: Site Name [ default zone ] and Primary EPM
```

```
[135.122.99.207 ]. Do you want to change this? (yes/no):
y

Default Site Name: default site

Primary EPM IPAddress/HostName: 135.122.99.207

Primary EPM User Name: admin

Primary EPM Password: *******

*** One EPM should be selected as primary for each non-default site from following auxiliary EPM list.

Site name (site 1) >> Auxilairy EPM List [135.122.99.99,135.122.99.98]:
#? 135.122.99.99

Site name (site 2) >> Auxilairy EPM List [135.122.99.81,135.122.99.80]:
#? 135.122.99.81

(07:07:48) [ CBA configuration is updated to use AAEP Sites. ]

********** ATTENTION **********
 You can add new site, edit existing site through web admin to match your configuration.
**********

[root@cbadb81 support]#
```

Upgrading Callback Assist

Avaya Callback Assist offers you the ability to upgrade to a later version of Callback Assist by preserving all your current data (Callback Requests, Callback Configurations, Reports data, and so on) in the Callback Assist Database.

Callback automatically modifies the schema of the database to the later version during the upgrade process and performs the required tasks to let Callback use all the previous configurations with the latest version.

Important:

- Create a backup of the existing installation before upgrading
- Follow the Post-Upgrade steps as required

Upgrading Single Server Deployment

To perform an upgrade in a Single Server Deployment, you have to run the Callback installer as you do during a fresh installation. During the upgrade, the Callback installer automatically searches for any installed instance of Callback in the system. After detecting a previous version of Callback in the system, the installer informs about the current version and platform of Callback and prompts you to confirm upgrading to the later version.

After you confirm the upgrade, Callback backs up the previous Callback version to protect against the possible failure of the upgrade process. After the backup, Callback installs the new version components as it does with a fresh installation.

When upgrading from Callback Assist v3.x or v4.0.x, the PostgreSQL Database version is upgraded from v8.3 to v9.1. This is due to High Availability capabilities in the new version.

Starting from Callback Assist v4.3, the PostgreSQL Database version is upgraded to v9.4. This is due to security vulnerabilities concerns.

The Callback Assist installer script will detect old PostgreSQL version and will automatically upgrade to the new one, preserving all the current data.

When the Callback Assist upgrade process is over, the system automatically restores all the configuration files, license files, audio files, and other required files from the previous version.

It is allowed to upgrade to a newer version and also changing the Platform (CTI or SIP) and/or Delivery Strategy (Agent First or Customer First) and/or Authentication Type to be used, but be aware that it will be required to configure Platform and/or Delivery strategy specific settings.

Performing the upgrade

Perform the following tasks for upgrading to the latest environment version:

1. On a Linux shell logged on as root, go to the directory where the CBA installer file is located.
2. Run the following command to extract the files to a temporary folder in your system:
   `tar -xvf callbackassist-<version>.tar`
3. Run the following command to start the installation process:
   `./callback-install.sh`
   The system detects that there is a previous version of CBA installed, and asks if you want to upgrade it to a later version.
4. Type *Yes* to confirm upgrading to the later version and press Enter.
   The system prompts you to choose the required platform for the CBA installation.

5. Choose the required platform for the CBA installation and press Enter.
   The system starts installing the latest version based on your selection. After completing the installation, the system displays an installation successful message.

The following example shows an output of a Callback Assist v4.x.x.x upgrade. The example also shows an output of a *callback-install.sh* script detecting that there is a previous version of CBA installed.

```
[root@server206 4.3.1.0]# ./callback-install.sh

*********************************************************

Avaya Callback Assist Installer

*********************************************************

*** ATTENTION: A Single Server deployment of Avaya Callback Assist was
detected.

*** The current installed version is 4.3.0.0-GA, build 46426.

*** You are about to install 4.3.1.0-GA, build 46464.

***

*** The current configured Platform is SIP (Agent First Strategy) -
Authentication Type: Internal

***

*** Do you want to proceed with an upgrade? (yes/no): yes

(06:09:27) Checking if there is enough disk space...

(06:09:27) Available disk space is enough.


(06:09:27) Please choose a Platform type for Callback Assist
Application to be Installed from the options below:

1) CTI (H.323)

2) SIP

3) AACC/CM (SIP Based)

#? 2


(06:09:32) Please choose the Delivery Strategy:

1) Agent First Strategy
```

```
2) Customer First Phantom Pool Strategy

3) Customer First Priority Queuing Strategy

#? 1


(06:09:34) Please choose an Authentication Type:

1) Internal

2) External (OpenId ex: Google)

#? 1


(06:09:36) Are you going to use local WebLM service? (yes/no) : yes

(06:09:38) [ Local WebLM server will be added as a daemon service. ]

(06:09:38) [ Internal has been selected as an Authentication Type. ]

(06:09:38) [ Agent First Strategy has been selected as the Delivery
Strategy. ]

(06:09:38) [ SIP Platform has been selected for installation. ]

Stopping Callback Assist Engine...

Callback Assist Engine Stopped.                              [  OK  ]

Stopping Callback Assist Maintenance...

Callback Assist Maintenance Stopped.                         [  OK  ]

Stopping tomcat-adminapp...                                  [  OK  ]

Stopping tomcat-ddapps...                                    [  OK  ]

Stopping tomcat-webcallback...                               [  OK  ]

Stopping tomcat-weblm...                                     [  OK  ]

Stopping CBA BSR Server service...                           [  OK  ]

(06:09:51) Copying temporary data (this operation may take a couple of
minutes)...

(06:10:12) Previous Version 4.3.0.0-GA backed up in: /opt/Avaya/backup

(06:10:12) Backing up Engine Configuration Files...
```

```
(06:10:12) Engine Configuration Files Successfully Backed up.

(06:10:12) Backing up WebLM User Configuration File...

(06:10:12) Backing up WebLM Usage History File...

(06:10:12) Backing up WebLM Server Properties File...

(06:10:12) WebLM Configuration Files Successfully Backed up.

(06:10:12) Backing up CBA Phrases directory...

(06:10:14) CBA Phrases directory Successfully Backed up.

(06:10:15) Installation of tomcat-adminapp found.

(06:10:15) Uninstallation of tomcat-adminapp done.

(06:10:15) Installation of tomcat-ddapps found.

(06:10:15) Uninstallation of tomcat-ddapps done.

(06:10:15) Installation of tomcat-webcallback found.

(06:10:15) Uninstallation of tomcat-webcallback done.

(06:10:15) Installation of tomcat-weblm found.

(06:10:15) Uninstallation of tomcat-weblm done.

(06:10:15) Installation of Callback Assist Engine found.

(06:10:15) Uninstallation of Callback Assist Engine done.

(06:10:16) Installation of Callback Assist Maintenance found.

(06:10:16) Uninstallation of Callback Assist Maintenance done.

(06:10:16) Installation of BSR Server found.

CBA BSR Server is stopped

(06:10:16) Uninstallation of CBA BSR Server done.



(06:10:16) Unpackaging distribution file callbackassist.package...

(06:10:42) Group callback already exists, so there is no need to create
it.

(06:10:42) User callback already exists, so there is no need to create
it.
```

```
(06:10:42) 32 bit Architecture detected ...

(06:10:42) Database is already installed and it is in its latest
version.

(06:10:42) Restarting database to reset max_connections

Restarting PostgreSQL 9.4:

waiting for server to shut down.... done

server stopped

waiting for server to start.... done

server started

PostgreSQL 9.4 restarted successfully

(06:10:44) Signaling Postgresql postmaster...

(06:10:44) Done.

(06:10:44) Setting ownership to callback user.

(06:10:44) Performing JDK silent install...

(06:10:47) JDK installed.

java version "1.7.0_75"

Java(TM) SE Runtime Environment (build 1.7.0_75-b13)

Java HotSpot(TM) Server VM (build 24.75-b04, mixed mode)

(06:10:48) Installation of Callback Assist Maintenance done.

(06:10:48) Check
/opt/Avaya/callbackassist/maintenance/logs/execution.log file for
Callback Assist Maintenance service startup details.

(06:10:48) Installation of Callback Assist Engine done.

(06:10:48) Check /opt/Avaya/callbackassist/engine/logs/execution.log
file for Callback Assist Engine service startup details.

(06:10:48) File Server is already installed.

(06:10:48) Waiting for the Database Schema to be created or updated...

Migration successful

(06:10:51) Database Schema was successfully created or updated.
```

```
(06:10:51) Platform successfully set.

(06:10:51) Deployment type successfully set.

(06:10:51) Storing Release version and Build Number...

(06:10:51) Release version and Build number successfully stored.

(06:10:51) Running Platform dependant changes...

Migration successful

(06:10:52) Database Schema was successfully updated.

(06:10:52) Running Deployment Type changes...

Migration successful

(06:10:53) Database Schema was successfully updated.

(06:10:53) Restoring Engine Configuration Files...

(06:10:53) Engine Configuration files successfully restored.

(06:10:53) Updating authentication type into database ...

(06:10:53) Installing weblm Tomcat service (tomcat-weblm)...

(06:10:53) Installation of tomcat-weblm done.

(06:10:53) Installing adminapp Tomcat service (tomcat-adminapp)...

(06:10:53) Installation of tomcat-adminapp done.

(06:10:53) Installing ddapps Tomcat service (tomcat-ddapps)...

(06:10:53) Installation of tomcat-ddapps done.

(06:10:53) Installing webcallback Tomcat service (tomcat-
webcallback)...

(06:10:53) Installation of tomcat-webcallback done.

(06:10:53) Deploying Applications...

(06:10:53) Moving tomcat realm related jar files

(06:10:54) Installation of Tomcat instances done.

(06:10:54) Restoring Backed up WebLM User Configuration File...

(06:10:54) Restoring Backed up WebLM Usage History File...

(06:10:54) Restoring Backed up WebLM Server Properties File...
```

```
(06:10:54) WebLM configuration files successfully restored.

(06:10:54) Installing BSR Server.  This operation may take several
minutes...

(06:10:54) Local Ip Address: 135.122.99.206

(06:10:54) Setting ownership to callback user.

Starting Callback Assist Engine...

Callback Assist Engine Started.                              [  OK  ]

Callback Assist Engine ( pid 6534 ) is running...

Starting Callback Assist Maintenance...

Callback Assist Maintenance Started.                         [  OK  ]

Callback Assist Maintenance ( pid 6609 ) is running...

Starting tomcat-weblm...                                     [  OK  ]

tomcat-weblm ( pid 6697 ) is running...

Starting tomcat-adminapp...                                  [  OK  ]

tomcat-adminapp ( pid 6793 ) is running...

Starting tomcat-webcallback...                               [  OK  ]

tomcat-webcallback ( pid 6909 ) is running...

Starting tomcat-ddapps...                                    [  OK  ]

tomcat-ddapps ( pid 7015 ) is running...

Starting CBA BSR Server service...                           [  OK  ]

CBA BSR Server ( pid 7103 ) is running...

Starting Callback Assist File Server...

Callback Assist File Server Started.                         [  OK  ]

Callback Assist File Server ( 7275 ) is running...           [  OK  ]

Callback Assist File Server Ping test                        [  OK  ]

Callback Assist File Server Read/Write cycle Test            [  OK  ]

Callback Assist File Server Ring Status is Up                [  OK  ]

Callback Assist File Server is joined to a cluster           [  NO  ]
```

```
If this is not a HA deployment, then disregard this warning.


(06:11:11) Restoring Backed up CBA Phrases directory...

(06:11:18) CBA Phrases directory successfully restored.

(06:11:19) [ Installation of Avaya Callback Assist (SIP - Agent First
Strategy) completed. ]

************************************************************

ACTION REQUIRED on Time Zone Configuration

************************************************************

The default Time Zone of CBA is UTC. If your system requires a
different Time Zone

you must manually configure it in the Global Settings of the Admin
Portal.

************************************************************
```

Upgrading High Availability Deployment

Steps to upgrade an existing Callback Assist installed in HA mode is similar to Upgrading Single Server deployment.

Checklist before upgrading

1. Create a backup of the existing installation.
2. Starting from Callback Assist v4.3 and later versions, the PostgreSQL Database server(s) must be upgraded to v9.4. Follow the steps Upgrade Database Servers(s) before upgrading the Callback Assist Core Components servers (Hot and Standby CBA Servers). Ignore, if the PostgreSQL database is already in latest version.
3. Do not stop any CBA servers at any moment during the upgrade.
4. Make sure that all the CBA servers are up and running and they are reachable through the network by the other.
5. If the Database Replication is already configured make sure it is working fine.
6. Back up and note down the below details before running the upgrade steps

Server Name	Server IP & Hostname
Master CBA Server	
Slave CBA Server	
Primary CBA Database Server	
Secondary CBA Database Server	

Backup SSL Key Store and configuration

This is applicable only if https is configured in Callback Assist. This needs to be performed on both CBA servers.

1. Create temporary folders
   mkdir /tmp/ssl/adminapp
   mkdir /tmp/ssl/ddapps
   mkdir /tmp/ssl/webservice

2. Take backup of server.xml files of the tomcat servers
   cp /opt/Avaya/callbackassist/apache-tomcat*adminapp/conf/server.xml /tmp/ssl/adminapp/
   cp /opt/Avaya/callbackassist/apache-tomcat*ddapps/conf/server.xml /tmp/ssl/ddapps/
   cp /opt/Avaya/callbackassist/apache-tomcat*webcallback/conf/server.xml /tmp/ssl/webservice/

Upgrade Database Server(s)

Callback Assist v4.3 and later versions, the PostgreSQL database server needs to be upgraded to 9.4, due to security/vulnerabilities concerns. Once the upgrade steps completed, we need to setup the replication again. This upgrade will not restore the database replication settings automatically.

1. Create a backup of the existing Database Server(s) installation. Skip this step if it is already done.
2. Stop the Callback Assist Master and Standby Application servers.
   /sbin/service callbackassist stop
3. Do not stop the Master Database Server during this upgrade.
4. Uninstall Callback Assist Standby Database server. Run the uninstall script and wait for the prompt to return.
   /opt/Avaya/callbackassist/callback-uninstall.sh
5. Go to Master Database Server and upgrade the Callback Assist (should be v4.3 and later). You have to run the Callback installer as you do during a fresh installation. Refer the section Installing Callback Assist Database on a separate server.
6. Once the installation completed the newer Postgresql v9.4 should be upgraded.

```
[root@localhost 4.2.1]# ./callback-install.sh


*********************************************************
Avaya Callback Assist Installer
*********************************************************


*** INFO: Attempting to upgrade an External Database Server. ***
*** PostgreSQL v9.1 was found.
*** You are about to install PostgreSQL v9.4.
***
*** Do you want to proceed with an upgrade? (yes/no): yes
```

```
(03:25:11) Unpackaging distribution file callbackassist.package...


(19:27:35) Group callback already exists, so there is no need to create
it.
(19:27:35) User callback already exists, so there is no need to create
it.
(19:27:35) 64 bit Architecture detected ...
(19:27:36) Checking if PostgreSQL Data Base needs to be upgraded...
(19:27:36) PostgreSQL v9.1 was found.
(19:27:36) Installing new PostgreSQL database...
(19:28:32) Done.
(19:28:32) Dumping and Restoring data from old to new database...
(19:28:33) Done.
(19:28:33) Uninstalling old Database...
(19:28:41) Done.
(19:28:41) Assigning old port to new database...
(19:28:41) Done.
(19:28:43) Restarting database to reset max_connections
Restarting PostgreSQL 9.4:



waiting for server to shut down.... done
server stopped
waiting for server to start.... done
server started
PostgreSQL 9.4 restarted successfully
(19:28:45) Signaling Postgresql postmaster...
(19:28:45) Done.
(19:28:45) Setting ownership to callback user.


(19:28:46) [ Installation of PostgreSQL Database Server for Callback
Assist completed. ]
```

7. Follow the Replication guide document **("Avaya Callback Assist Application Notes for PostgreSQL Replication")** to setup the Standby Database Server and Replication.
8. Start the Callback Assist Master and Standby Application servers.
   /sbin/service callbackassist start

Upgrade CBA Server(s)

To perform an upgrade in a HA Server Deployment, you have to run the Callback installer as you do during a fresh installation. During the upgrade, the Callback installer automatically searches for any installed instance of Callback in the system. After detecting a previous version of Callback in the system, the installer informs about the current version and platform of Callback and prompts you to confirm upgrading to the later version.

1. Create a backup of the existing Database Server(s) installation. Skip this step if it is already done.

2. Upgrade CBA Server(s) one by one. There is no specific order in which the CBA servers are upgraded. It doesn't matter if it is the master or the slave which is upgraded first.

3. After successful installation restore the previous configurations.

4. Run the RiakFileCheck utility to verify the audio files in the Riak File Sever. Refer Verify Audio Files in Riak File Server section.

Restore Configurations

Restore SSL Configurations

Applicable only if SSL configuration is enabled. Perform this on both the CBA servers.

1. Follow the instructions in the following sections of the *Avaya Callback Assist Application Notes for HTTPS Configuration document* and using the SSL configuration files taken as backup for reference.
   - Configuring SSL for the web administration application
   - Configuring SSL for Dialogs
   - Configuring SSL for Web services

Change Database MAX CONNECTIONS

This is required if upgrade is performed on Callback Assist 4.1.6 or lower versions.

1. Stop all services
   Refer Stop Callback Assist in HA mode.
2. Execute below instructions on both database servers, one at a time

   a) Edit postgresql.conf file
   vi /opt/Avaya/callbackassist/PostgreSQL/9.1/data/postgresql.conf

   b) Update below properties
   max_connections = 400
   effective_cache_size should be 2/3 of your available RAM

checkpoint_segments = 16
shared_buffers = 512MB (or 1/4 of your available RAM)

c) Save and exit the file
Escape + wq!

3. Start all services
Refer Start Callback Assist in HA mode.

Verify Audio Files in Riak File Server

The RiakFileCheck utility is used to check and verify the availability of audio files which were uploaded in the Riak File Server through the Administration UI. The utility is available under **<Callback Assist installation path>/support/riak-utility/** directory. This utility is run to ensure that the audio files in the File servers are not lost during the upgrade process and it should be run on both CBA servers.

1. Navigate to the riak-utility directory.
cd <Callback Assist installation path>/support/riak-utility/
2. Run the utility as below
java -jar RiakFileCheck.jar <fileServerIP> <databaseIP>
E.g. java -jar RiakFileCheck.jar 135.122.99.106 135.122.99.81
3. Sample output is shown below

```
2015-06-16 03:53:36,146 [main] INFO RiakFileCheck - No. of files found in database = 1515
2015-06-16 03:53:39,300 [main] INFO RiakFileCheck - Callback ConfigId - Name: 1 - Test1
2015-06-16 03:53:39,301 [main] INFO RiakFileCheck - Configuration Message Audio Files:
2015-06-16 03:53:39,301 [main] INFO RiakFileCheck - http:// 135.122.99.106:8098/riak/audios/851af8a0-6863-43b9-8dd8-7b83992fe2eb.wav [silence_1ms.wav] [OK]
2015-06-16 03:53:39,301 [main] INFO RiakFileCheck - http:// 135.122.99.106:8098/riak/audios/8d9f1dfa-930a-44f2-bcbf-0fe99c4407bd.wav [example_CF_name_noinput.wav] [OK]....
....
....
2015-06-16 03:53:53,545 [main] INFO RiakFileCheck -
**********************************************
2015-06-16 03:53:53,545 [main] INFO RiakFileCheck - No. of files available in database 135.122.99.81 = 1515
2015-06-16 03:53:53,545 [main] INFO RiakFileCheck - No. of files available in fileServer 135.122.99.106 = 1515
2015-06-16 03:53:53,545 [main] INFO RiakFileCheck - No. of files missing in fileServer 135.122.99.106 = 0
2015-06-16 03:53:53,545 [main] INFO RiakFileCheck -
**********************************************
```

4. The output can also be viewed in the log file named "datalayer.log". This log file will be created in the same directory of the RiakFileCheck.jar file.
5. The count of "**No. of files available in database**" and "**No. of files available in fileServer**" should match.

If they do not match and if the count "**No. of files missing in fileServer**" is not 0, File Server sync issue has occurred during the upgrade process.

6. If a File Server sync issue is identified, follow the steps in Restore Riak File Server section to restore the audio files.

Upgrading Dual Server Deployment

Steps to upgrade an existing Callback Assist installed in Dual Server deployment will be similar to Upgrading Single Server deployment.

Backup

Create a backup of the existing installations before upgrading callback assist.

Upgrade Database Server(s)

Follow the steps mentioned in section Upgrade Database Server(s) to complete this procedure.

Upgrade CBA Application Server

To perform an upgrade in a Dual Server Deployment, you have to run the Callback installer as you do during a fresh installation.

Perform the following tasks for upgrading to the latest environment version:

1. On a Linux shell logged on as root, go to the directory where the CBA installer file is located.
2. Run the following command to extract the files to a temporary folder in your system:
   `tar -xvf callbackassist-<version>.tar`
3. Run the following command to start the installation process:
   `./callback-install.sh`
4. Enter the IP address or the host name of the external Database Server at the prompt.
5. Enter ***yes*** to confirm the IP address or the host name you entered in the previous step.

   The system detects that there is a previous version of CBA installed, and asks if you want to upgrade it to a later version.
6. Type ***Yes*** to confirm upgrading to the later version and press Enter.
   The system prompts you to choose the required platform for the CBA installation.
7. Choose the required platform for the CBA installation and press Enter.
8. Choose desired Strategy from the following options and press Enter.
9. Choose desired authentication type and press Enter.
10. Choose whether you need Local Web LM server or not. If you decide to use the Local Web LM server as license server then select "yes", otherwise "no".

Note: If you select “no” at the time of installation and later if you decide to use Local Web LM server as a license server; then the WebLM (tomcat-weblm) service can be restored by running the script reinstallservices.sh from <CBA_INSTALLATION_LOCATION>/support folder.

11. Enter ***no*** if any other instance of Callback Assist is sharing the same external database.

    The system starts installing the latest version based on your selection. After completing the installation, the system displays an installation successful message.

The following example shows an output of a Callback Assist v4.x.x.x upgrade. The example also shows an output of a *callback-install.sh* script detecting that there is a previous version of CBA installed.

```
[root@server206 4.3.1.0]# ./callback-install.sh

************************************************************

Avaya Callback Assist Installer



************************************************************

Avaya Callback Assist Installer

************************************************************



*** INFO: Attempting to upgrade HA Installation of CBA Components. ***



(04:07:08) Please enter the Callback Assist Database Server IP Address
or Host name: 10.130.124.27

(04:07:24) You've entered 10.130.124.27. Is this correct? (Yes/No) yes

(04:07:26) Checking PostgreSQL Server version on 10.130.124.27...

(04:07:27) PostgreSQL version is OK.

(04:07:27) Checking for 'callback' User and Database on
10.130.124.27...

(04:07:27) User and Database are OK.



*** ATTENTION: HA deployment of Avaya Callback Assist was detected.

*** The current installed version is 4.3.3.0-GA, build 46566.

*** You are about to install 4.4.0.0-beta3, build 46774.
```

```
***

*** The current configured Platform is SIP (Agent First Strategy) -
Authentication Type: Internal

***

*** Do you want to proceed with an upgrade? (yes/no): yes

(04:07:39) Checking if there is enough disk space...

(04:07:39) Available disk space is enough.


(04:07:39) Please choose a Platform type for Callback Assist
Application to be Installed from the options below:

1) CTI (H.323)

2) SIP

3) AACC/CM (SIP Based)

#?

(04:07:49) Please choose the Delivery Strategy:

1) Agent First Strategy

2) Customer First Phantom Pool Strategy

3) Customer First Priority Queuing Strategy

#?

(04:07:50) Please choose an Authentication Type:

1) Internal

2) External (OpenId ex: Google)

#?

(04:07:51) Are you going to use local WebLM service? (yes/no) : yes

(04:08:03) [ Local WebLM server will be added as a daemon service. ]


(04:08:03) [ Internal has been selected as an Authentication Type. ]
```

```
(04:08:03) [ Agent First Strategy has been selected as the Delivery
Strategy. ]


(04:08:03) [ SIP Platform has been selected for installation. ]


Stopping Callback Assist Engine... OK

Callback Assist Engine Stopped.

Stopping Callback Assist Maintenance... OK

Callback Assist Maintenance Stopped.

Stopping tomcat-adminapp... OK

Stopping tomcat-ddapps... OK

Stopping tomcat-webcallback... OK

Stopping tomcat-weblm... OK

Stopping CBA BSR Server service...  OK


(04:08:16) Copying temporary data (this operation may take a couple of
minutes)...

(04:08:27) Previous Version 4.4.0.0-beta2 backed up in:
/opt/Avaya/backup

(04:08:28) Backing up Engine Configuration Files...

(04:08:28) Engine Configuration Files Successfully Backed up.

(04:08:28) Backing up WebLM User Configuration File...

(04:08:28) Backing up WebLM Usage History File...

(04:08:28) Backing up WebLM Server Properties File...

(04:08:28) WebLM Configuration Files Successfully Backed up.

(04:08:28) Backing up CBA Phrases directory...

(04:08:41) CBA Phrases directory Successfully Backed up.

(04:08:41) Installation of tomcat-adminapp found.

(04:08:41) Uninstallation of tomcat-adminapp done.
```

```
(04:08:41) Installation of tomcat-ddapps found.

(04:08:41) Uninstallation of tomcat-ddapps done.

(04:08:42) Installation of tomcat-webcallback found.

(04:08:42) Uninstallation of tomcat-webcallback done.

(04:08:42) Installation of tomcat-weblm found.

(04:08:42) Uninstallation of tomcat-weblm done.

(04:08:42) Installation of Callback Assist Engine found.

(04:08:42) Uninstallation of Callback Assist Engine done.

(04:08:46) Installation of Callback Assist Maintenance found.

(04:08:46) Uninstallation of Callback Assist Maintenance done.

(04:08:46) Installation of BSR Server found.

CBA BSR Server is stopped

(04:08:46) Uninstallation of CBA BSR Server done.


(04:08:46) Unpackaging distribution file callbackassist.package...


(04:09:02) Group callback already exists, so there is no need to create
it.

(04:09:02) User callback already exists, so there is no need to create
it.

(04:09:02) Performing JDK silent install...

(04:09:48) JDK installed.

java version "1.7.0_75"

Java(TM) SE Runtime Environment (build 1.7.0_75-b13)

Java HotSpot(TM) Server VM (build 24.75-b04, mixed mode)


(04:09:48) Database Server running on host: 10.130.124.27.
```

```
(04:09:48) Is there any other CBA instance sharing this same Database?
(Yes/No) no

(04:11:29) Installation of Callback Assist Maintenance done.

(04:11:29) Check
/opt/Avaya/callbackassist/maintenance/logs/execution.log file for
Callback Assist Maintenance service startup details.

(04:11:29) Installation of Callback Assist Engine done.

(04:11:29) Check /opt/Avaya/callbackassist/engine/logs/execution.log
file for Callback Assist Engine service startup details.

(04:11:29) File Server is already installed.

(04:11:29) Waiting for the Database Schema to be created or updated...

Migration successful

(04:11:33) Database Schema was successfully created or updated.

(04:11:33) Platform successfully set.

(04:11:33) Deployment type successfully set.

(04:11:33) Storing Release version and Build Number...

(04:11:33) Release version and Build number successfully stored.

(04:11:33) Running Platform dependant changes...

Migration successful

(04:11:34) Database Schema was successfully updated.

(04:11:34) Running Deployment Type changes...

Migration successful

(04:11:35) Database Schema was successfully updated.

(04:11:35) Restoring Engine Configuration Files...

(04:11:35) Engine Configuration files successfully restored.

(04:11:35) Updating authentication type into database ...

(04:11:35) Installing weblm Tomcat service (tomcat-weblm)...

(04:11:35) Installation of tomcat-weblm done.

(04:11:35) Installing adminapp Tomcat service (tomcat-adminapp)...
```

```
(04:11:35) Installation of tomcat-adminapp done.

(04:11:36) Installing ddapps Tomcat service (tomcat-ddapps)...

(04:11:36) Installation of tomcat-ddapps done.

(04:11:36) Installing webcallback Tomcat service (tomcat-
webcallback)...

(04:11:36) Installation of tomcat-webcallback done.

(04:11:36) Deploying Applications...

(04:11:36) Moving tomcat realm related jar files

(04:11:36) Installation of Tomcat instances done.

(04:11:36) Restoring Backed up WebLM User Configuration File...

(04:11:36) Restoring Backed up WebLM Usage History File...

(04:11:36) Restoring Backed up WebLM Server Properties File...

(04:11:36) WebLM configuration files successfully restored.

(04:11:36) Installing BSR Server.  This operation may take several
minutes...

(04:11:36) Local Ip Address: 135.122.99.205

(04:11:36) Setting ownership to callback user.

Starting Callback Assist Engine... OK

Callback Assist Engine Started.

Callback Assist Engine ( pid 6981 ) is running...

Starting Callback Assist Maintenance... OK

Callback Assist Maintenance Started.

Callback Assist Maintenance ( pid 7062 ) is running...

Starting tomcat-weblm... OK

Starting tomcat-adminapp... OK

tomcat-adminapp ( pid 7252 ) is running...

Starting tomcat-webcallback... OK

tomcat-webcallback ( pid 7371 ) is running...
```

```
Starting tomcat-ddapps... OK

tomcat-ddapps ( pid 7478 ) is running...

Starting CBA BSR Server service... OK

CBA BSR Server ( pid 7563 ) is running...

Starting Callback Assist File Server... OK

Callback Assist File Server Started.

Callback Assist File Server ( 7741 ) is running... OK

Callback Assist File Server Ping test OK

Callback Assist File Server Read/Write cycle Test OK

Callback Assist File Server Ring Status is Up OK

Callback Assist File Server is joined to a cluster OK

If this is not a HA deployment, then disregard this warning.


(04:12:13) Restoring Backed up CBA Phrases directory...

(04:12:19) CBA Phrases directory successfully restored.


(04:12:21) [ Installation of Avaya Callback Assist (SIP - Agent First
Strategy) completed. ]


*********************************************************

ACTION REQUIRED on Time Zone Configuration

*********************************************************

The default Time Zone of CBA is UTC. If your system requires a
different Time Zone

you must manually configure it in the Global Settings of the Admin
Portal.

*********************************************************
```

As a reference, in the installation output shown above there were 2 servers involved:

- PostgreSQL Database server *[10.130.124.27]*
- Callback Assist Server with external Database *[135.122.99.205]*

Restore Configurations

Restore SSL Configurations

Applicable only if SSL configuration is enabled. Perform this on both the CBA servers.

Follow the instructions in the following sections of the *Avaya Callback Assist Application Notes for HTTPS Configuration document* and using the SSL configuration files taken as backup for reference.

- Configuring SSL for the web administration application
- Configuring SSL for Dialogs
- Configuring SSL for Web services

Post-upgrade steps

After upgrading Callback Assist, assuming you haven't changed either Platform or Delivery Strategies, there is nothing much to configure in most of the cases except, you might be required to perform additional configurations in the Admin Web Page based on your upgraded version for the new features to be effective.

Restore Custom Audio Files

The steps below outline the procedure to restore the custom recorded audio files used in the previous Callback Assist installation to the newer version installed. This is not required if the previous installation was using the default audio files provided with the Callback Assist installer.

1. Stop dialogs tomcat
   service tomcat-ddapps stop

2. Navigate to the backup folder and extract the cba_bkp_YYYYMMDD.tar.gz file to a new folder
   cd /tmp/cba_backup/
   mkdir extracted
   tar -zxvf /tmp/cba_backup/cba_bkp_YYYYMMDD.tar.gz --directory extracted/

3. Copy the audio files from the backup directory to the new Callback Assist

For version below 4.2

cp /tmp/cba_backup/extracted/opt/Avaya/callbackassist/apache-tomcat*ddapps/webapps/CBAPhrases/standard/<language_name> /<locale>/* /opt/Avaya/callbackassist/apache-tomcat-ddapps/webapps/CBAPhrases/standard/<language_name>/<locale>/default

cp /tmp/cba_backup/extracted/opt/Avaya/callbackassist/apache-tomcat*ddapps/webapps/CBAPhrases/custom/customer/<language_name> /<locale>/* /opt/Avaya/callbackassist/apache-tomcat-

ddapps/webapps/CBAPhrases/custom/customer/<language_name>/<locale>/default

cp /tmp/cba_backup/extracted/opt/Avaya/callbackassist/apache-tomcat*ddapps/webapps/CBAPhrases/custom/agent/<language_name>/<locale>/* /opt/Avaya/callbackassist/apache-tomcat-ddapps/webapps/CBAPhrases/custom/agent/<language_name>/<locale>/default

For version above 4.2

cp /tmp/cba_backup/extracted/opt/Avaya/callbackassist/apache-tomcat*ddapps/webapps/CBAPhrases/standard/<language_name>/<locale>/default/* /opt/Avaya/callbackassist/apache-tomcat-ddapps/webapps/CBAPhrases/standard/<language_name>/<locale>/default

cp /tmp/cba_backup/extracted/opt/Avaya/callbackassist/apache-tomcat*ddapps/webapps/CBAPhrases/custom/customer/<language_name>/<locale>/default/* /opt/Avaya/callbackassist/apache-tomcat-ddapps/webapps/CBAPhrases/custom/customer/<language_name>/<locale>/default

cp /tmp/cba_backup/extracted/opt/Avaya/callbackassist/apache-tomcat*ddapps/webapps/CBAPhrases/custom/agent/<language_name>/<locale>/default/* /opt/Avaya/callbackassist/apache-tomcat-ddapps/webapps/CBAPhrases/custom/agent/<language_name>/<locale>/default

4. After copying all the files start dialogs tomcat
   service tomcat-ddapps start
5. Remove the extracted folder
   rm -rf /tmp/cba_backup/extracted

Restore Riak File Server

This is a workaround procedure to restore the Riak File server from the Callback Assist backup. This should be performed only if the Riak File server sync failed during upgrade process and confirmed by running the RiakFileCheck utility. This is done by replacing the complete file-server directory with the file-server directory from the backup.

1. Stop all Callback Assist services
   Refer Stop Callback Assist in HA mode.
2. Remove the complete file-server directory from the Callback Assist server

   rm -rf /opt/Avaya/callbackassist/file-server/

3. Navigate to the backup folder and extract the cba_bkp_YYYYMMDD.tar.gz file to a temporary directory

   cd /tmp/cba_backup/
   mkdir extracted
   tar -zxvf /tmp/cba_backup/cba_bkp_YYYYMMDD.tar.gz --directory extracted/

4. Copy the file-server directory from the backup to the Callback Assist installation directory

   cp -R /tmp/cba_backup/extracted/opt/Avaya/callbackassist/file-server /opt/Avaya/callbackassist/file-server

5. Repeat the steps 2 to 4 on both the Callback Assist servers

6. Start all Callback Assist services

   Refer Start Callback Assist in HA mode.

7. Remove the temporary extracted folder

   rm -rf /tmp/cba_backup/extracted

Customer First Callback Configuration changes for 4.1.6 or higher

In Customer First priority Queuing and Customer First Phantom Pool strategies, the field to upload the **Menu Options message** in the *Global Settings → Delivery Tab* has been split into two different fields in 4.1.6.

Menu Options with Reschedule Message

To specify the menu options with reschedule option during the callback delivery. For example: "Please press 1 to accept the call and wait for an agent, press 2 to be called back later or press 3 to cancel this callback."

Menu Options without Reschedule Message

To specify the menu options without reschedule option during the callback delivery. For example: "Please press 1 to accept the call and wait for an agent, press 2 to cancel this callback."

Upgrading Callback Assist to 4.1.6 or higher versions in Customer First Strategy requires reloading these wav files through the administration UI. The existing audio file uploaded to the **Menu Options Message** field will be retained in the **Menu Options with Reschedule Message** field after the upgrade. The **Menu Options without Reschedule Message** should be uploaded for all the Callback Configurations. The **Menu Options with Reschedule Message** need not be reloaded if it already has the correct message**.**

If there are large number of Callback Configurations, refer the **Bulk Upload Menu Options Messages** section of the *Administering Avaya Callback Assist* user guide to upload audio files to all the callback configurations using an upload utility**.**

Additional step when upgrading to 4.4.0.0 or higher in SIP/AACC Platform

Add the AAEP security certificate to CBA components by following the instructions here.

Additional Steps when upgrading to 4.1.4 or higher

One of the new features included on Callback Assist 4.1.4 is the migration of BSR Administration and BSR Server Node to be part of Callback Assist distribution, meaning that it is not required to run a separate installation script for BSR Admin or BSR Server Node after installing Callback Assist. The following changes have been performed:

- The BSR Admin UI has been incorporated under Callback Assist Admin UI and it is not required to install it EPM anymore.
- The BSR Server Node installation has been integrated on Callback Assist installation scripts and should not be install separately from Callback Assist. Existing BSR Server Node must be removed from Callback Assist Linux server.

These changes affect existing SIP installations, so it is required to migrate the existing configuration on BSR Admin Database to Callback Assist Database. In order to avoid manual migration, a separate script is provided with Callback Assist installation to migrate BSR Database to Callback Assist Database. After migration, the BSR Admin UI can be uninstalled from customer EPM (recommended).

The migration process consists of three steps:

- Migrating existing configuration data from BSR Administration to Callback Assist Database
- Uninstallation of BSR Administration deployed on Experience Portal EPM
- Uninstallation existing BSR Server Node from Callback Assist Server

Migrating existing configuration data from BSR Administration to Callback Assist Database

Pre-Requisites: The migration script will request EPM IP address to get the data from.

1. Locate Callback Assist installation folder. Default is: "/opt/Avaya/callbackassist"
2. Navigate to "support" directory at: ***cd /opt/Avaya/callbackassist/support***.
3. Create temporary folder by executing: ***mkdir temp***
4. Decompress file "bsr-db-migration.tar.gz" by execution: ***tar -xf bsr-db-migration.tar.gz -C ./temp/***
5. Navigate to temp directory: ***cd temp***
6. Execute "bsr-db-migration.sh" script: ***./bsr-db-migration.sh***
7. Enter ***EPM/VPMS IP address*** when request for source IP "***Please enter source IP (EPM/VPMS):***" (like 135.122.60.209).
8. Enter Callback Assist Server IP address or leave it blank for localhost (127.0.0.1) when requested for destination IP "***Please, enter destination IP (default is localhost):***".
9. Wait until script finishes. Check for "***Migration successful***" message on command line. Complete output:

```
[root@RHEL60EPAEP temp]# ./bsr-db-migration.sh
(15:14:14) CBA Installation Path obtained:  /opt/Avaya/callbackassist
Please enter source IP (EPM/VPMS): 135.122.99.204
Please, enter destination IP (default is localhost):
Destination IP is:  localhost
(15:17:10) Creating directory for data export  ./derbyExportDataDir
```

```
Stopping CBA BSR Server service...                    [  OK  ]
(15:17:15) Doing Export from  135.122.99.204
(15:17:16) Doing Import to  localhost
Migration successful
Starting CBA BSR Server service...                    [  OK  ]
[root@RHEL60EPAEP temp]#
```

10. Log on to Callback Assist Admin UI at **http://<CBA_IP>/admin.** Click on ***BSR Configuration*** menu option on the left of the screen.

11. Click on ***CBA BSR Call Center Applications Configuration*** and check that the data has been migrated and it is the same as the one shown on EPM/VPMS BSR UI.

Uninstallation of BSR Administration deployed on Experience Portal EPM

After the migration of BSR Database data to Callback Assist 4.1.4 Database it is recommended to uninstall existing BSR from EPM to avoid any mistakes or issues in the future.

Pre-Requisite: It is required to have BSR Admin installation scripts so if these files are not available on the EPM server anymore please download them from the internal site.

1. Open a SSH session to EPM server.

2. If login user is not “sroot” or “root”, change to root user by executing: ***su root***

3. Locate the BSR installation binaries used to install on BSR or download them (ICRAdmin-1.2.xx.x.zip).

4. Extract them if needed by executing: ***unzip ICRAdmin-1.2.xx.x.zip***

5. Navigate to ICRAdmin directory.

6. Grant execution permission to uninstall script. Choose appropriate file based on EPM version. For example, for AAEP 6.0 use “uninstallICRAdminAAEP60.sh”. To grant permissions, execute: ***chmod +x uninstallICRAdminAAEP60.sh***.

7. Run the uninstallation script: ***./uninstallICRAdminAAEP60.sh***.

8. Enter ***yes*** when requested to continue with uninstallation on “***Do you want to continue with the CBA BSR Admin uninstall?(yes/no)***”.

9. Wait till uninstallation finishes. Once completed, enter EPM admin to double-check that “BSR Configuration” option is not available anymore. See example of script output.

```
[root@RHEL60EPVPC ICRAdmin]# ./uninstallICRAdminAAEP60.sh
Welcome to CBA BSR Admin Uninstall Script
Installer Version -
*****WARNING****
This installer will override any configuration that was done to the files listed below setting them to the
original VPMS configurations. Make sure you re-apply any necessary configurations to these files if it is
needed.
/opt/Tomcat/tomcat/webapps/VoicePortal/WEB-INF/classes/config/menu.xml
/opt/Tomcat/tomcat/webapps/VoicePortal/WEB-INF/classes/config/features.xml
/opt/Tomcat/tomcat/webapps/VoicePortal/WEB-INF/classes/messages/menu.properties
/opt/Tomcat/tomcat/conf/server.xml
/etc/httpd/conf.d/vpms.conf

Do you want to continue with the CBA BSR Admin uninstall?(yes/no) yes
Shutting down VPMS...
Stopping individual components:
Stopping Tomcat.............Counter: 10. Tomcat is not running: 0
Tomcat not shut down gracefully; forceful shut down being enacted
Will kill tomcat PIDs: 2081
Stopping SL..... successful
Stopping ActiveMQ..... successful

VPMS Shutdown Status:                    [  OK  ]
VPMS was shut down
Removing CBA BSR Administration GUI from VPMS ...
Stopping httpd:                          [  OK  ]
Starting httpd:                          [  OK  ]
Starting VPMS Services...
Starting individual components:
Starting and checking ActiveMQ at Fri Oct 4 19:36:24 MDT 2013....
... successful. ActiveMQ is ready at Fri Oct 4 19:36:43 MDT 2013
Starting and checking SL at Fri Oct 4 19:36:43 MDT 2013...
... successful. SL is ready at Fri Oct 4 19:36:47 MDT 2013
Starting and checking Tomcat at Fri Oct  4 19:36:47 MDT 2013....
Curl Counter: 0 Tomcat ready: 1
... successful. Tomcat is ready at Fri Oct 4 19:37:23 MDT 2013

VPMS Start Status:                       [  OK  ]
Looking for CBA BSR Administration DB Service installation in this machine...
Stopping ICR Admin DB Service...         [  OK  ]
CBA BSR Administration DB Service removed!
Done. CBA BSR Administration GUI successfully uninstalled!
[root@RHEL60EPVPC ICRAdmin]#
```

Uninstallation existing BSR Server Node from Callback Assist Server

These steps remove existing BSR Server Node installed on same server as Callback Assist. As Callback Assist 4.1.4 already installs the BSR Server Node and manages it, the old/existing one is not required anymore.

Note: There are two ways to uninstall BSR Server Node from the server but here only the manual one is covered in detail.

1. On the Callback Assist Server, navigate to BSR parent directory by executing: ***cd /opt/Avaya/ICR***.
2. Double check that the existing BSR Server node is stopped by executing: ***/sbin/service ICRBSRServer status***.
3. If running, stop if by executing: ***/sbin/service ICRBSRServer stop***
4. Remove the service registration from Linux by executing: ***chkconfig --del ICRBSRServer***
5. Remove the symbolic link to startup script by executing: ***rm -rf /etc/init.d/ICRBSRServer***
6. Navigate one directory back.
7. Remove the entire ICR directory by executing: ***rm -rf ICR***.

Examples of High Availability Installation and configuration

This section provides some examples of all the procedures required to configure a highly available Callback Assist environment, and also on operating with multiple Experience Portal systems.

Environment

The required environment is as follows:

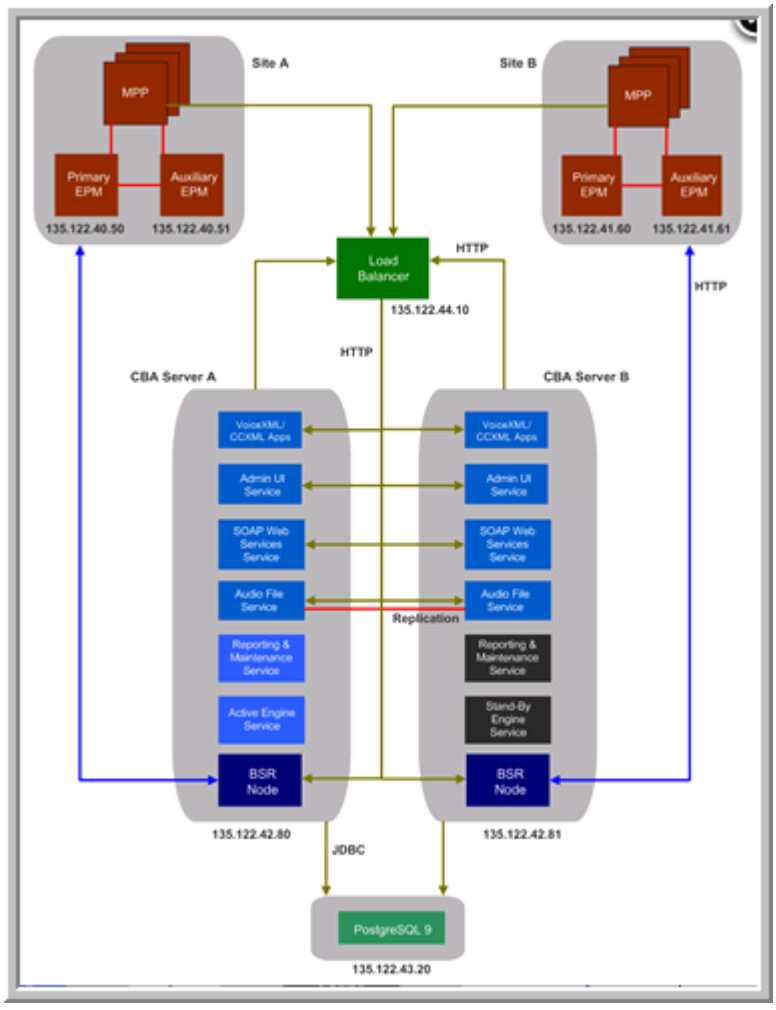


Figure 110 - Environment with sample IP addresses

Installing Callback Assist

Installing Callback Assist Database on a separate server

To install the PostgreSQL Database for Callback Assist, perform the following tasks:

1. Log on as root user in the 135.122.43.20 server.
2. Go to the directory where the Callback Assist installer file is located.
3. Run the following command to extract the files to a temporary folder in your system:
   `tar -xvf callbackassist<version>.tar`
   One of the extracted files is *callback-install.sh* which is the main installation script.
4. Run the following command to start the installation process:
   `./callback-install.sh`
   The system writes the command output of the scripts to the standard output and to the *callback-install.log* file in your system. Then, the system prompts you to choose the required platform for Callback Assist installation.
5. Select [**2)**] as the required installation mode from the following options:
   a. Callback Assist Components with external Database
   b. PostgreSQL Database Server for Callback Assist
   c. Callback Assist Components and PostgreSQL Database Server
6. Select the directory in which PostgreSQL Database for Callback Assist is going to be installed, or press Enter to install to the default of **/opt**

Installing Callback Assist server

Installing the first Callback Assist server

Perform the following tasks to install Callback Assist components in the first server:

1. Log on as root user in the 135.122.42.80 server.
2. Go to the directory where the Callback Assist installer file is located.
3. Run the following command to extract the files to a temporary folder in your system:
   `tar -xvf callbackassist<version>.tar`
   One of the extracted files is *callback-install.sh* which is the main installation script.
4. Run the following command to start the installation process:
   `./callback-install.sh`
   The system writes the command output of the scripts to the standard output and to the *callback-install.log* file in your system. Then, the system prompts you to choose the required platform for Callback Assist installation.
5. Select [**1)**] as the required installation mode from the following options:

a. Callback Assist Components with external Database

b. PostgreSQL Database Server for Callback Assist

c. Callback Assist Components and PostgreSQL Database Server

6. Enter the installation directory where to install Callback Assist or just press ***Enter*** to use default location (default directory "/opt").

7. Select [**1)**] Agent First as the delivery strategy from the options and press Enter.

4. CTI (H.323)

5. SIP

8. Choose desired Strategy from the following options and press Enter.

1. Agent First

2. Customer First Phantom Pool Strategy

3. Customer First Priority Queuing Strategy

The system starts installing SIP platform with the desired strategy.

9. Choose the authentication type

1. Internal

2. External (Open ID ex: Google)

If the authentication type is selected as External then follow the CBA Installation as External Authentication section for more details.

10. Choose whether you need Local Web LM server or not. If you decide to use the Local Web LM server as license server then select "yes", otherwise "no".

Note: If you select "no" at the time of installation and later if you decide to use Local Web LM server as a license server; then the WebLM (tomcat-weblm) service can be restored by running the script reinstallservices.sh from <CBA_INSTALLATION_LOCATION>/support folder.

11. Enter 135.122.43.20 as the IP address of the external Database Server at the prompt.

12. Enter *yes* to confirm the IP address you entered in the previous step.

13. Enter *no* when asked if any other instance of Callback Assist is sharing the same external database.

Installing the second Callback Assist server

Perform the following tasks to install Callback Assist Components in the second server:

1. Log on as root user in the 135.122.42.81 server.

2. Go to the directory where the CBA installer file is located.

3. Run the following command to extract the files to a temporary folder in your system:
   `tar -xvf callbackassist<version>.tar`
   One of the extracted files is callback-install.sh which is the main installation script.

4. Run the following command to start the installation process:
   `./callback-install.sh`
   The system writes the command output of the scripts to the standard output and to the *callback-install.log* file in your system. Then, the system prompts you to choose the required platform for Callback Assist installation.
5. Select [**1)**] as the required installation mode from the following options:
   a. Callback Assist Components with external Database
   b. PostgreSQL Database Server for Callback Assist
   c. Callback Assist Components and PostgreSQL Database Server
6. Enter the installation directory where to install Callback Assist or just press ***Enter*** to use default location (default directory "/opt").
7. Select [**1)**] Agent First as the delivery strategy from the options and press Enter.
   6. CTI (H.323)
   7. SIP
8. Choose desired Strategy from the following options and press Enter.
   1. Agent First
   2. Customer First Phantom Pool Strategy
   3. Customer First Priority Queuing Strategy

   The system starts installing SIP platform with the desired strategy.
9. Choose the authentication type
   1. Internal
   2. External (Open ID ex: Google)

   If the authentication type is selected as External then follow the CBA Installation as External Authentication section for more details.
10. Choose whether you need Local Web LM server or not. If you decide to use the Local Web LM server as license server then select "yes", otherwise "no".

    Note: If you select "no" at the time of installation and later if you decide to use Local Web LM server as a license server; then the WebLM (tomcat-weblm) service can be restored by running the script reinstallservices.sh from <CBA_INSTALLATION_LOCATION>/support folder.
11. Enter 135.122.43.20 as the IP address of the external Database Server at the prompt.
12. Enter *yes* to confirm the IP address you entered in the previous step.
13. Enter *yes* when asked if any other instance of Callback Assist is sharing the same external database.

14. Enter 135.122.42.80 as the IP address of the existing Callback Assist instance sharing the same database.
15. Enter *Yes* to confirm the IP address you entered in the previous step.

Note: If you have more than two Callback Assist servers, repeat the same procedure for installing the second Callback Assist server for all the servers.

Configuring the Load Balancer

The load balancer should include at minimum the following configuration:

Purpose	Inbound Port	Destination IP Addresses	Destination Port	Use Session Affinity
CBA Web Administration	80	135.122.42.80, 135.122.42.81	80	Yes
CBA IVR Dialogs	8080	135.122.42.80, 135.122.42.81	8080	Yes
CBA Web Services	8081	135.122.42.80, 135.122.42.81	8081	No
CBA Audio Storage	8098	135.122.42.80, 135.122.42.81	8098	No
BSR Server Node Web Services	8089	135.122.42.80, 135.122.42.81	8089	No

Configuring Callback Assist

Configuring the Audio Storage Server

1. Log on to the Callback Assist Web Administration application.
2. Go to **Global Settings** > **Audio** tab.
3. Click on the pencil icon to the edit the **Storage URL** parameter.

4. Enter *http://135.122.44.10:8098/riak* as the parameter value.

Configuring Sites

The Site Definitions option would be enabled if the CBA installation has been done with site being enabled. Before configuring the sites that will be used by Callback Assist, you have to decide which Experience Portal system will be the default site. The default site will primarily be used if, when adding callback request through the Callback Assist Web services interface, no preferred site for the delivery of the callback request is given. In such situations, the default site is used to deliver the callback request. In this example, site A is selected as the default site.

Also, before proceeding with the configuration, the failover behavior of the sites has to be analyzed. In an environment with 2 sites, each site should fail over to the other. So, in the example, site A will fail over to site B and vice versa. If more sites will be configured in Callback Assist, the failover behavior may be more complex and system capacity and network topology will determine how the failover is configured.

Configuring the first site

1. Log on to the Callback Assist Web Administration application.
2. Go to **Site Definitions**.
3. Click on the pencil icon to edit the default site.
4. A window with site name text box and a button “Add New” for adding primary/auxiliary EPM will appear.
5. Optionally change ***Name*** field and it should be unique.
6. Click on “Add New” button to add primary EPM.
7. Enter ***Outbound Web Service IP Address*** (135.122.40.50) of Primary EPM field.
8. Enter ***Outbound Web Service User*** defined on EPM. The User parameter must match the Outcall credentials administered on EPM Settings on Experience Portal Web Portal.
9. Enter ***Outbound Web Service Password*** defined on EPM. The User parameter must match the Outcall credentials administered on EPM Settings on Experience Portal Web Portal.
10. Click on OK button to save primary EPM detail
11. Optionally, if auxiliary EPMs are present, add the multiple auxiliary EPMs by clicking on “Add New” button and enter the ***Outbound web service IP Address, user name*** and ***password***.

Configuring the second site

1. Log on to the Callback Assist Web Administration application.
2. Go to **Site Definitions**.
3. Click on the **Add New** icon to create a new site definition.
4. Enter the site name.

5. Enter ***Outbound Web Service IP Address*** (135.122.41.60) of Primary EPM field.
6. Enter ***Outbound Web Service User*** defined on EPM. The User parameter must match the Outcall credentials administered on EPM Settings on Experience Portal Web Portal.
7. Enter ***Outbound Web Service Password*** defined on EPM. The User parameter must match the Outcall credentials administered on EPM Settings on Experience Portal Web Portal.
8. Click on OK button to save primary EPM detail
9. Optionally, if auxiliary EPMs are present, add the multiple auxiliary EPMs by clicking on "Add New" button and enter the ***outbound web service IP Address, user name*** and ***password***.

Configuring Failover

1. Log on to the Callback Assist Web Administration application.
2. Go to **Site Definitions**.
3. Click on the table icon to edit the failover sequence for Site A.
4. Select Site B from the **Available Sites** list.
5. Click on the right arrow button, Site B should be moved to the Selected Sites list.
6. Click **OK**.
7. Click on the table icon to edit the failover sequence for Site B.
8. Select Site A from the **Available Sites** list.
9. Click on the right arrow button, Site A should be moved to the Selected Sites list.
10. Click **OK**.

Configuring Zones

The Zone Definitions option would be enabled if the CBA installation has been done with zone being enabled. There will be only one primary EPM in all the zone definitions and that should have been given during installation itself and that zone will be default zone. Before configuring the zones that will be used by Callback Assist, you have to decide which Experience Portal system will be the default zone. The default zone will primarily be used if, when adding callback request through the Callback Assist Web services interface, no preferred zone for the delivery of the callback request is given. In such situations, the default zone is used to deliver the callback request. In this example, zone A is selected as the default zone.

Also, before proceeding with the configuration, the failover behavior of the zones has to be analyzed. In an environment with 2 zones, each zone should fail over to the other. So, in the example, zone A will fail over to zone B and vice versa. If more zones will be configured in Callback Assist, the failover behavior may be more complex and system capacity and network topology will determine how the failover is configured.

Configuring the first zone

1. Log on to the Callback Assist Web Administration application.

2. Go to **Zone Definitions**.
3. Click on the pencil icon to edit the default zone.
4. A window with zone name drop down box and a button “Add New” for adding primary/auxiliary EPM will appear.
5. Select zone name from drop down and that should be unique
6. Click on “Add New” button to add auxiliary EPM.
7. Enter ***Outbound Web Service IP Address*** of auxiliary EPM field.
8. Enter ***Outbound Web Service User*** defined on EPM. The User parameter must match the Outcall credentials administered on EPM Settings on Experience Portal Web Portal.
9. Enter ***Outbound Web Service Password*** defined on EPM. The User parameter must match the Outcall credentials administered on EPM Settings on Experience Portal Web Portal.
10. Click on OK button to save auxiliary EPM detail
11. Optionally, if multiple auxiliary EPMs are present, add those by clicking on “Add New” button and enter the ***outbound web service IP Address, user name*** and ***password*** and click on OK button to save it.

Configuring the second zone

1. Log on to the Callback Assist Web Administration application.
2. Go to **Zone Definitions**.
3. Click on the **Add New** button to create a new zone definition.
4. A window with zone name drop down box and a button “Add New” for adding auxiliary EPM will appear.
5. Select the zone name from zone name dropdown.
6. Click on **Add New** button to add auxiliary EPM.
7. Enter ***Outbound Web Service IP Address*** (135.122.41.60) of auxiliary EPM field.
8. Enter ***Outbound Web Service User*** defined on EPM. The User parameter must match the Outcall credentials administered on EPM Settings on Experience Portal Web Portal.
9. Enter ***Outbound Web Service Password*** defined on EPM. The password parameter must match the Outcall credentials administered on EPM Settings on Experience Portal Web Portal.
10. Click on OK button to save auxiliary EPM detail

Configuring Failover

1. Log on to the Callback Assist Web Administration application.
2. Go to **Zone Definitions**.
3. Click on the table icon to edit the failover sequence for Zone A.
4. Select Zone B from the **Available Zones** list.

5. Click on the right arrow button, Zone B should be moved to the Selected Zones list.
6. Click **OK**.
7. Click on the table icon to edit the failover sequence for Zone B.
8. Select Zone A from the **Available Zones** list.
9. Click on the right arrow button, Zone A should be moved to the Selected Zones list.
10. Click **OK**.

Adding Callback Configurations

Callback configurations do not have any restrictions when working on a High Availability environment. Only consideration, due to be operating with 2 sites, is the DNIS management, as detailed in Configuring Experience Portal.

Configuring BSR

Configuring Call Center Applications

There is no particular restrictions for this configuration in a HA environment. The only consideration is that all BSR Nodes must share the same configuration, so the BSR Administration applications in site A and in Site B and in the primary as well as the auxiliary EPM must have the exact same configuration for call center applications, business hours, and holidays. You must replicate this configuration manually in all the EPM in 135.122.40.50, 135.122.40.51, 135.122.41.60, and 135.122.41.61.

Configuring Callback Assist Access

1. Log on to EPM in 135.122.40.50.
2. Go to CBA BSR Configuration.
3. Go to Callback Assist Configuration.
4. In CCXML URL, enter http://135.122.44.10:8080/CBAScripts/ccxml/cbaCallControl.ccxml.
5. Repeat this process in EPM in 135.122.40.51, 135.122.41.60 and 135.122.41.61.

Configuring Experience Portal

Before configuring the Experience Portal systems, decide about how the DNIS will be managed. Two options can be as follows:

- **DNIS are unique across all the sites:** In this situation, a DNIS that is defined in site A is not replicated in site B. When site A is not available, the traffic that would normally be routed to a DNIS in site A, has to be routed to a different DNIS in site B. This would normally be managed through Session Manager configuration. To handle this difference in the DNIS, and as each callback configuration has a single DNIS, you must create distinct callback configurations.
  For example, consider that in normal conditions calls arrive to site A to DNIS 5000 and if this fails then the traffic is routed to DNIS 6000 in site B. Then a callback configuration has to be created for DNIS 5000 to handle normal traffic; and another configuration with DNIS 6000 has to be created to handle traffic in the event that site A goes down.
- **DNIS are shared among sites:** In this configuration, the same DNIS is created in all site and traffic is routed based on preference. That is, all traffic for each DNIS is configured to go to specific sites. If this fails, then calls are routed to the same DNIS in another site. In this scenario, a single callback configuration can handle all traffic coming from any site for the same DNIS.

Adding the Applications

- ***Adding the Inbound Application***

1. Log on to the EPM in 135.122.40.50 as an administrator.
2. Go to **System Configurations** > **Applications**.
3. Click on the **Add** button.
   The system displays the page to add a new application.
4. Enter a proper application name, like CBA Inbound.
5. In the **Type** dropdown menu, select CCXML.
6. In the **CCXML URL** field, enter http://135.122.44.10:8080/CBAScripts/cbaCallControl
7. In the **Application Launch** field, select **Inbound** and add the required DNIS.
8. Perform any additional required configuration, like selecting the UUI mode.
9. Repeat the same procedure in the other site, logging on to the EPM in 135.122.41.60.

- ***Adding the Outbound Application***

1. Log on to the EPM in 135.122.40.50 as an administrator.
2. Go to **System Configurations** > **Applications**.
3. Click on the **Add** button.
   The system displays the page to add a new application.
4. Enter a proper application name, like CBA Outbound.
5. In the **Type** dropdown menu, select CCXML.

   In the **CCXML URL** field, enter http://135.122.44.10:8080/CBAScripts/cbaCallControl
6. In **Application Launch**, select **Outbound**.
7. Perform any additional required configuration, like selecting the UUI mode.
8. Repeat the same procedure in the other site, logging on to the EPM in 135.122.41.60

Miscellaneous

This section covers additional configuration required not related to a specific platform or strategy.

Security-Enhanced Linux (SELinux)

Callback Assist supports enabling Security-Enhanced Linux (SELinux).

During the installation, the Callback Assist installer detects if SELinux is enabled, and if it is, it changes the file context to a Callback Assist File Server file, which requires text relocation.
On the contrary, if SELinux is disabled, Callback Assist will continue installation normally.

After installation, if you decide to enable SELinux, Callback Assist File Server would not work properly as one of its files will require text relocation, which will be forbidden by SELinux. To change the file context to the Callback Assist File Server file that requires text relocation, run the following command, if you have installed Callback Assist at: */opt/Avaya/callbackassist*.

```
[root@RHEL54Nacho ~]# chcon -t textrel_shlib_t
'/opt/Avaya/callbackassist/file-server/lib/erlang_js-
1.0.1/priv/erlang_js_drv.so'
```

When you run the command, Callback Assist File Server will start work properly again.

Invoking CBASIPOffer as VXML sub dialog Sample

To ease integration of the CBASIPOffer application into an existing IVR application running on a different tomcat, CBA provides source code of a sample OD application which invokes the CBASIPOffer as a VXML sub dialog. The source code is available as **CBAExternalVXMLLauncherSample.tar** under **<callback-installation-directory>/support** directory. This can be used as a reference while developing the IVR application.

This is developed using OD release 06.00.13.01.

Import CBAExternalVXMLLauncherSample.tar as an existing project into an AAOD eclipse workspace. Export the war file as a speech project and deploy it in a tomcat with OD runtime libraries.

The CBAExternalVXMLLauncherSample application receives the test data from Configurable Application Variables listed in the table below.

Call Center Id	The callcenterId parameter passed to the CBSIPOffer application
CBA Offer Application URL	The URL of the CBASIPOffer application. Ex. **http://<CBA_FQDN>:8080/CBASIPOffer/Start**

Callback Configuration DNIS	The dnis input parameter passed to CBASIPoffer application.
Emergency Destination	The destination to transfer calls when CBA does not return a queuing VDN due to internal errors.
Vpms Address	The vpmsAddress passed to the CBASIPOffer application. Mandatory if site based deployment is used.
Zone ID	vpmsZoneId parameter passed as input to the CBASIPOffer application. Mandatory if zone based deployment is used.

See Experience Portal configuration of CBAExternalVXMLLauncherSample application below.

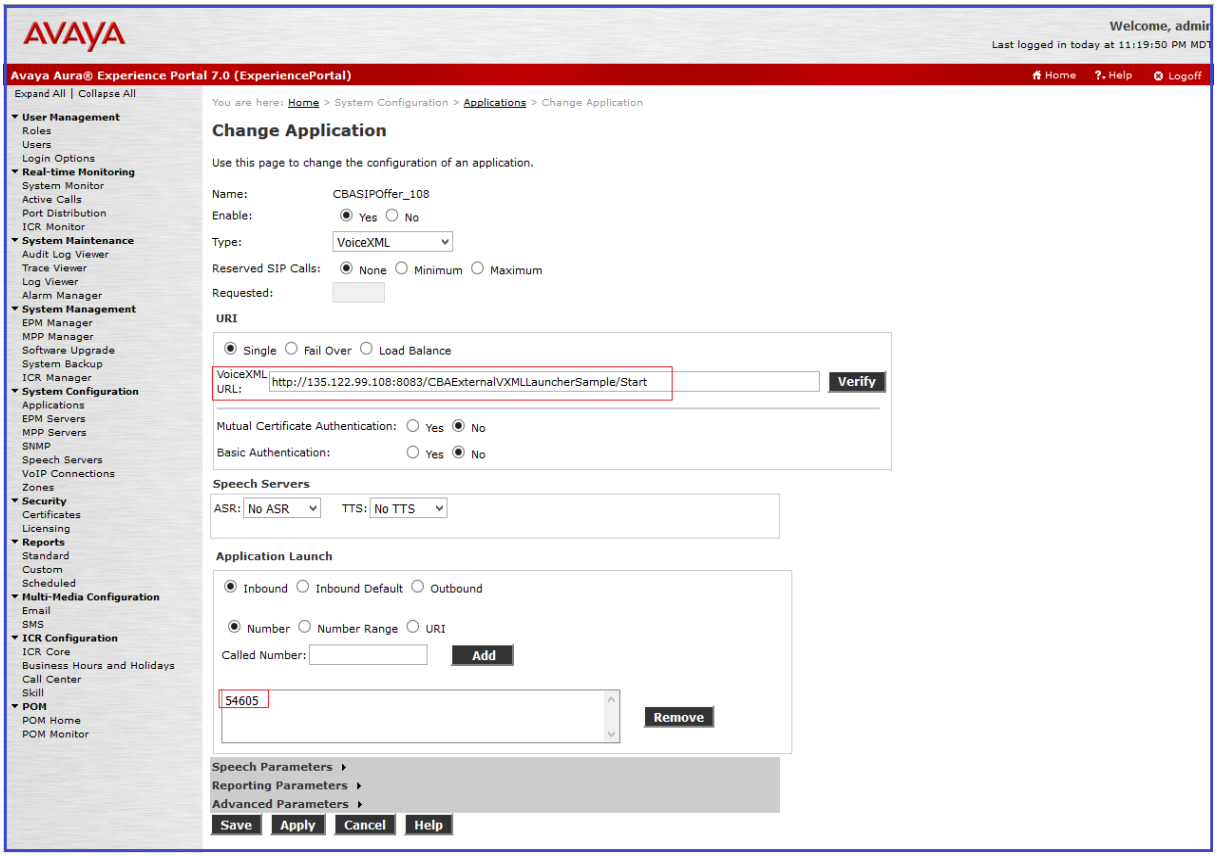


Figure 111 - Adding CBAExternalVXMLLauncherSample to Experience Portal

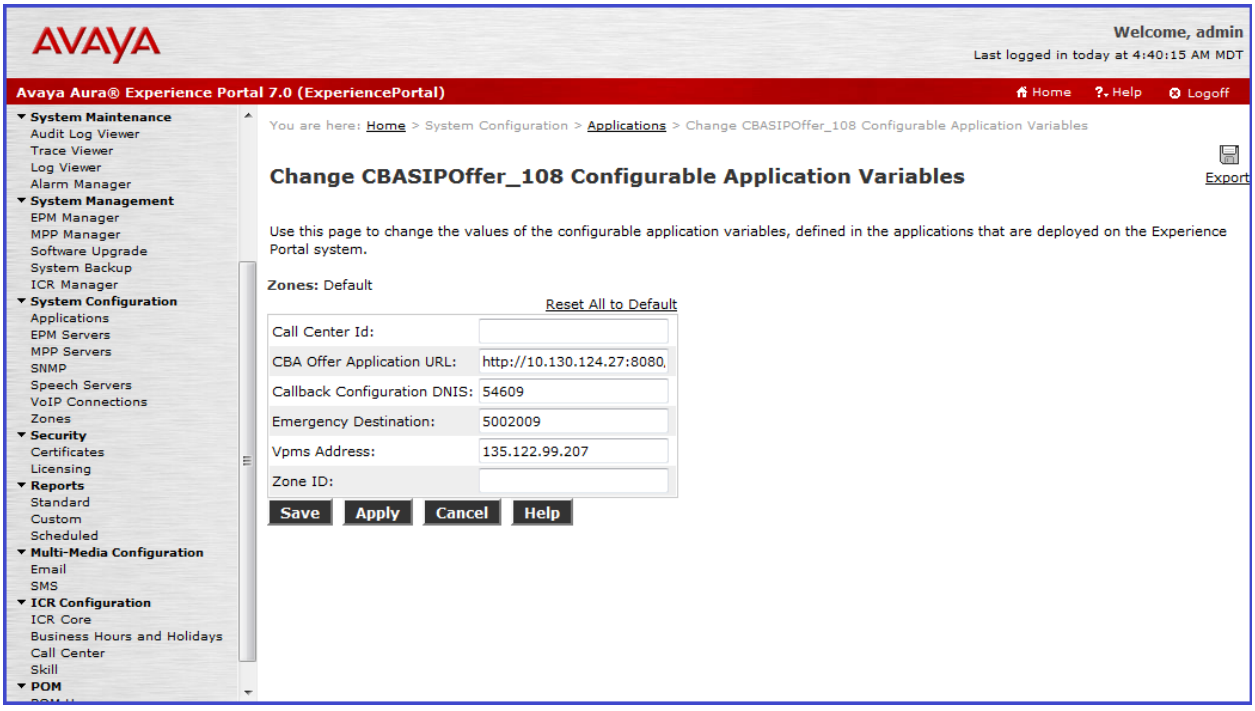


Figure 112 - CBAExternalVXMLLauncherSample CAV Configuration

Database Tuning

The following properties can be changed for better database performance.
<CALLBACK_INSTALLATION_DIRECTORY>PostgreSQL /9.4/data/postgresql.conf

Property	Default value	Recommended value
effective_cache_size	128MB	2/3 of available RAM
checkpoint_segments	3	16
shared_buffers	32MB	512M (or 1/4 of your available RAM

Note:
In a single server deployment ensure that enough memory is available for other CBA components before allocating additional memory to Database.

Increase JVMs heap size for CBA components

Note: Adjust the settings only if the CBA system has more than the minimum hardware requirement, especially the RAM size.

If you want to override JVM's heap size & perm gen space, you can set the VM options to each components [Refer the below table to add this setting]. Except CBA web services (Xmx 512 MB) and dialog designer (Xmx 1024 MB) applications, by default each JVM uses Heap Size 256 MB + Perm Size 64 MB + Non-Heap memory XX MB (each component will differ). Adjust the setting carefully based on Free RAM size.

Example:

```
JAVA_OPTS="$JAVA_OPTS -Xmx512m -Xms512m -XX:PermSize=128m -XX:MaxPermSize=128m"
```

Xmx = specifies the maximum memory allocation pool for a Java Virtual Machine (JVM)

Xms = specifies the initial memory allocation pool for a Java Virtual Machine (JVM)

PermSize = Size of the Permanent Generation

MaxPermSize = Size of the maximum Permanent Generation

Default Installation Directory: /opt/Avaya/callbackassist/

Component Name	**Directory Name** **[Installation Directory]/**	**File To Edit**
CBA Administration	apache-tomcat-adminapp/bin	catalina.sh
Dialog Designer Applications	apache-tomcat- ddapps/bin	catalina.sh
CBA Web services	apache-tomcat- webcallback/bin	catalina.sh
BSR Server	bsr-server/bin	ICRBSRServer.sh
CBA Engine	engine/lib	cbaengine-start.sh
Maintenance Process	maintenance/bin	cbamaint-start.sh
Web LM	apache-tomcat-6.0.29-weblm/bin	catalina.sh

Note*:* After adding this VM options, the corresponding service needs to be restarted. The components that would benefit for fine tuning are Dialogs, Web services and Engine for handling the call traffics.

Database Local Cache Setup

Starting from release 4.2, Avaya Callback Assist (CBA) has made available entity and query cache, which greatly improves the read from database times. By default the caching is disabled in CBA and to enable it is necessary to define a value greater than 0 for the property **`cache_entries_time_to_live_in_seconds`** in the following files:

- **<CBA_INSTALLATION_FOLDER>/engine/lib/callbackDatabaseConfiguration.properties**

- **<CBA_INSTALLATION_FOLDER>/apache-tomcat-ddapps/lib/callbackDatabaseConfiguration.properties**
- **<CBA_INSTALLATION_FOLDER>/apache-tomcat-adminapp/lib/callbackDatabaseConfiguration.properties**
- **<CBA_INSTALLATION_FOLDER>/maintenance/lib/callbackDatabaseConfiguration.properties**
- **<CBA_INSTALLATION_FOLDER>/apache-tomcat-webcallback/lib/callbackDatabaseConfiguration.properties**

Besides the **cache_entries_time_to_live_in_seconds** property it is possible to control the maximum number of cached entries. A sample is provided below setting the TTL of cached entries to 5 minutes and 1000 for maximum cached entries. CBA uses LRU (Least Frequently Used) as the memory store eviction policy.

```
#TTL of cached entries in seconds
cache_entries_time_to_live_in_seconds=300
#Max Number of cached entries in the local heap
cache_max_entries=1000
```

Connection Pool and Caching Settings for CBA Components

Engine

File Location: <CBA_INSTALLATION_LOCATION>/engine/lib/callbackDatabaseConfiguration.properties

Setting	Meaning	Default value	Remarks	Mandatory /Optional
min_connections	Minimum number of Connections a pool will maintain at any given time.	2	Please ensure that min_connections <= max_connections	Mandatory
max_connections	Maximum number of Connections a pool will maintain at any given time	50	Please ensure that min_connections <= max_connections	Mandatory
connection_idle_test_period	If this is a number greater than 0, c3p0 will test all idle, pooled but unchecked-out connections, every this number of seconds	600		Mandatory
acquire_increment	Determines how many connections at a time c3p0 will try to acquire when the pool is exhausted	2		Mandatory
max_statements	The size of c3p0's global PreparedStatement cache.	0	If both maxStatements and maxStatementsPerConne	Mandatory

			ction are zero, statement caching will not be enabled.	
max_idle_time	Seconds a Connection can remain pooled but unused before being discarded.	0	Zero means idle connections never expire	Optional
acquire_retry_attempts	Defines how many times c3p0 will try to acquire a new Connection from the database before giving up.	2	If this value is less than or equal to zero, c3p0 will keep trying to fetch a Connection indefinitely.	Optional
acquire_retry_delay	Milliseconds, time c3p0 will wait between acquire attempts	2000		Optional
checkoutTimeout	The number of milliseconds a client calling getConnection() will wait for a Connection to be checked-in or acquired when the pool is exhausted.	3000	Zero means wait indefinitely. Setting any positive value will cause the getConnection() call to time-out and break with an SQLException after the specified number of milliseconds.	Optional
testConnectionOnCheckout	If true, an operation will be performed at every connection checkout to verify that the connection is valid.	false	Be sure to set an efficient **preferredTestQuery** or **automaticTestTable** if you set this to true. **Performing the (expensive) default Connection test on every client checkout will harm client performance**. Testing Connections in checkout is the simplest and most reliable form of Connection testing, **but for better performance, consider verifying connections periodically using idleConnectionTestPeriod.**	Optional
cache_entries_time_to_live_in_seconds	TTL of cached entries in seconds	0	Zero means that the 2nd level cache of Callback is disabled.	Optional
cache_max_entries	Max Number of cached entries in	1000		Optional

	the local heap			
application_name	Specifies the name of the application that is using the connection. This allows a database administrator to see what applications are connected to the server and what resources they are using through views like pgstatactivity.	engine		Optional
socketTimeout	The timeout value used for socket read operations. If reading from the server takes longer than this value, the connection is closed.	60000	This can be used as both a brute force global query timeout and a method of detecting network problems.The timeout is specified in seconds and a value of zero means that it is disabled.	Optional
tcpKeepAlive	Enable or disable TCP keep-alive probe. The default is false.	false		Optional
connection_timeout	The timeout value used for socket connect operations. If connecting to the server takes longer than this value, the connection is broken.	0	The timeout is specified in seconds and a value of zero means that it is disabled	Optional

DD Apps

File Location: <CBA_INSTALLATION_LOCATION>/apache-tomcat-ddapps/lib/callbackDatabaseConfiguration.properties

Setting	Meaning	Default value	Remarks	Mandatory /Optional
min_connections	Minimum number of Connections a pool will maintain at any given time.	2	Please ensure that min_connections <= max_connections	Mandatory
max_connections	Maximum number of Connections a pool will maintain at any given time	50	Please ensure that min_connections <= max_connections	Mandatory

connection_idle_test_period	If this is a number greater than 0, c3p0 will test all idle, pooled but unchecked-out connections, every this number of seconds	600		Mandatory
acquire_increment	Determines how many connections at a time c3p0 will try to acquire when the pool is exhausted	2		Mandatory
max_statements	The size of c3p0's global PreparedStatement cache.	0	If both maxStatements and maxStatementsPerConnection are zero, statement caching will not be enabled.	Mandatory
max_idle_time	Seconds a Connection can remain pooled but unused before being discarded.	0	Zero means idle connections never expire	Optional
acquire_retry_attempts	Defines how many times c3p0 will try to acquire a new Connection from the database before giving up.	2	If this value is less than or equal to zero, c3p0 will keep trying to fetch a Connection indefinitely.	Optional
acquire_retry_delay	Milliseconds, time c3p0 will wait between acquire attempts	2000		Optional
checkoutTimeout	The number of milliseconds a client calling getConnection() will wait for a Connection to be checked-in or acquired when the pool is exhausted.	3000	Zero means wait indefinitely. Setting any positive value will cause the getConnection() call to time-out and break with an SQLException after the specified number of milliseconds.	Optional
testConnectionOnCheckout	If true, an operation will be performed at every connection checkout to verify that the connection is valid.	false	Be sure to set an efficient **preferredTestQuery** or **automaticTestTable** if you set this to true. **Performing the (expensive) default**	Optional

			Connection test on every client checkout will harm client performance. Testing Connections in checkout is the simplest and most reliable form of Connection testing, **but for better performance, consider verifying connections periodically using idleConnectionTestPeriod.**	
preferredTestQuery	Defines the query that will be executed for all connection tests.		Defining a preferredTestQuery that will execute quickly in your database may dramatically speed up Connection tests. (If no preferredTestQuery is set, the default ConnectionTester executes a getTables() call on the Connection's DatabaseMetaData. Depending on your database, this may execute more slowly than a "normal" database query.) NOTE: The table against which your preferredTestQuery will be run must exist in the database schema prior to your initialization of your DataSource. If your application defines its own schema, try automaticTestTable instead.	Optional
automaticTestTable	If provided, c3p0 will create an empty table of the specified name, and use queries against that table		If automaticTestTable is provided, c3p0 will generate its own test	Optional

	to test the Connection.		query, therefore any preferredTestQuery set will be ignored. You should not work with the named table after c3p0 creates it; it should be strictly for c3p0's use in testing your Connection.	
cache_entries_time_to_live_in_seconds	TTL of cached entries in seconds	0	Zero means that the 2nd level cache of Callback is disabled.	Optional
cache_max_entries	Max Number of cached entries in the local heap	1000		Optional
application_name	Specifies the name of the application that is using the connection. This allows a database administrator to see what applications are connected to the server and what resources they are using through views like pgstatactivity.	dd-apps		Optional
socketTimeout	The timeout value used for socket read operations. If reading from the server takes longer than this value, the connection is closed.	60000	This can be used as both a brute force global query timeout and a method of detecting network problems.The timeout is specified in seconds and a value of zero means that it is disabled.	Optional
tcpKeepAlive	Enable or disable TCP keep-alive probe. The default is false.	false		Optional
connection_timeout	The timeout value used for socket connect operations. If connecting to the server takes longer than this value, the connection is broken.	0	The timeout is specified in seconds and a value of zero means that it is disabled	Optional

Admin

File Location: <CBA_INSTALLATION_LOCATION>/apache-tomcat-adminapp/lib/callbackDatabaseConfiguration.properties

Setting	Meaning	Default value	Remarks	Mandatory /Optional
min_connections	Minimum number of Connections a pool will maintain at any given time.	2	Please ensure that min_connections <= max_connections	Mandatory
max_connections	Maximum number of Connections a pool will maintain at any given time	50	Please ensure that min_connections <= max_connections	Mandatory
connection_idle_test_period	If this is a number greater than 0, c3p0 will test all idle, pooled but unchecked-out connections, every this number of seconds	600		Mandatory
acquire_increment	Determines how many connections at a time c3p0 will try to acquire when the pool is exhausted	2		Mandatory
max_statements	The size of c3p0's global PreparedStatement cache.	0	If both maxStatements and maxStatementsPerConnection are zero, statement caching will not be enabled.	Mandatory
max_idle_time	Seconds a Connection can remain pooled but unused before being discarded.	0	Zero means idle connections never expire	Optional
acquire_retry_attempts	Defines how many times c3p0 will try to acquire a new Connection from the database before giving up.	2	If this value is less than or equal to zero, c3p0 will keep trying to fetch a Connection indefinitely.	Optional
acquire_retry_delay	Milliseconds, time c3p0 will wait between acquire attempts	2000		Optional
checkoutTimeout	The number of milliseconds a client calling getConnection() will wait for a Connection to be	3000	Zero means wait indefinitely. Setting any positive value will	Optional

	checked-in or acquired when the pool is exhausted.		cause the getConnection() call to time-out and break with an SQLException after the specified number of milliseconds.	
testConnectionOnCheckout	If true, an operation will be performed at every connection checkout to verify that the connection is valid.	false	Be sure to set an efficient **preferredTestQuery** or **automaticTestTable** if you set this to true. **Performing the (expensive) default Connection test on every client checkout will harm client performance**. Testing Connections in checkout is the simplest and most reliable form of Connection testing, **but for better performance, consider verifying connections periodically using idleConnectionTestPeriod.**	Optional
preferredTestQuery	Defines the query that will be executed for all connection tests.		Defining a preferredTestQuery that will execute quickly in your database may dramatically speed up Connection tests. (If no preferredTestQuery is set, the default ConnectionTester executes a getTables() call on the Connection's DatabaseMetaData. Depending on your database, this may execute more slowly than a "normal"	Optional

			database query.) NOTE: The table against which your preferredTestQuery will be run must exist in the database schema prior to your initialization of your DataSource. If your application defines its own schema, try automaticTestTable instead.	
automaticTestTable	If provided, c3p0 will create an empty table of the specified name, and use queries against that table to test the Connection.		If automaticTestTable is provided, c3p0 will generate its own test query, therefore any preferredTestQuery set will be ignored. You should not work with the named table after c3p0 creates it; it should be strictly for c3p0's use in testing your Connection.	Optional
cache_entries_time_to_live_in_seconds	TTL of cached entries in seconds	0	Zero means that the 2nd level cache of Callback is disabled.	Optional
cache_max_entries	Max Number of cached entries in the local heap	1000		Optional
application_name	Specifies the name of the application that is using the connection. This allows a database administrator to see what applications are connected to the server and what resources they are using through views like pgstatactivity.	admin		Optional
socketTimeout	The timeout value used for socket read operations. If reading from the server takes longer than this value, the connection is closed.	60000	This can be used as both a brute force global query timeout and a method of detecting network problems.The timeout is specified in seconds	Optional

			and a value of zero means that it is disabled.	
tcpKeepAlive	Enable or disable TCP keep-alive probe. The default is false.	false		Optional
connection_timeout	The timeout value used for socket connect operations. If connecting to the server takes longer than this value, the connection is broken.	0	The timeout is specified in seconds and a value of zero means that it is disabled	Optional

Warning:

It is not recommended to enable cache for the Admin application. The cache implementation of Callback Assist does not update the cache on write operations but it updates only on read operations. When cache is enabled in the admin UI and if a property already exists in the cache, updates of the property through admin UI will be saved in the database but reflected on the admin UI only when the cache expiration time of the property is reached. Until then the old value is displayed on the Admin UI. This can cause confusion to the users.

ICR Admin

File Location: <CBA_INSTALLATION_LOCATION>/apache-tomcat-adminapp/lib/icrConfiguration.properties

Setting	Meaning	Default value	Remarks	Mandatory /Optional
min_connections	Minimum number of Connections a pool will maintain at any given time.	2	Please ensure that min_connections <= max_connections	Mandatory
max_connections	Maximum number of Connections a pool will maintain at any given time	50	Please ensure that min_connections <= max_connections	Mandatory
connection_idle_test_period	If this is a number greater than 0, c3p0 will test all idle, pooled but unchecked-out connections, every this number of seconds	600		Mandatory
acquire_increment	Determines how many connections at a time c3p0 will try to acquire when the pool is exhausted	2		Mandatory

max_statements	The size of c3p0's global PreparedStatement cache.	0	If both maxStatements and maxStatementsPerConnection are zero, statement caching will not be enabled.	Mandatory
max_idle_time	Seconds a Connection can remain pooled but unused before being discarded.	0	Zero means idle connections never expire	Optional
acquire_retry_attempts	Defines how many times c3p0 will try to acquire a new Connection from the database before giving up.	2	If this value is less than or equal to zero, c3p0 will keep trying to fetch a Connection indefinitely.	Optional
acquire_retry_delay	Milliseconds, time c3p0 will wait between acquire attempts	2000		Optional
checkoutTimeout	The number of milliseconds a client calling getConnection() will wait for a Connection to be checked-in or acquired when the pool is exhausted.	3000	Zero means wait indefinitely. Setting any positive value will cause the getConnection() call to time-out and break with an SQLException after the specified number of milliseconds.	Optional
testConnectionOnCheckout	If true, an operation will be performed at every connection checkout to verify that the connection is valid.	false	Be sure to set an efficient **preferredTestQuery** or **automaticTestTable** if you set this to true. **Performing the (expensive) default Connection test on every client checkout will harm client performance**. Testing Connections in checkout is the simplest and most reliable form of Connection testing, **but for better performance, consider verifying connections periodically using idleConnectionTestPeriod.**	Optional

preferredTestQuery	Defines the query that will be executed for all connection tests.		Defining a preferredTestQuery that will execute quickly in your database may dramatically speed up Connection tests. (If no preferredTestQuery is set, the default ConnectionTester executes a getTables() call on the Connection's DatabaseMetaData. Depending on your database, this may execute more slowly than a "normal" database query.) NOTE: The table against which your preferredTestQuery will be run must exist in the database schema prior to your initialization of your DataSource. If your application defines its own schema, try automaticTestTable instead.	Optional
automaticTestTable	If provided, c3p0 will create an empty table of the specified name, and use queries against that table to test the Connection.		If automaticTestTable is provided, c3p0 will generate its own test query, therefore any preferredTestQuery set will be ignored. You should not work with the named table after c3p0 creates it; it should be strictly for c3p0's use in testing your Connection.	Optional
application_name	Specifies the name of the application that is using the connection. This allows a database administrator to see what applications are connected to the server and what resources they are using through views like pgstatactivity.	icr-admin		Optional

socketTimeout	The timeout value used for socket read operations. If reading from the server takes longer than this value, the connection is closed.	60000	This can be used as both a brute force global query timeout and a method of detecting network problems.The timeout is specified in seconds and a value of zero means that it is disabled.	Optional
tcpKeepAlive	Enable or disable TCP keep-alive probe. The default is false.	false		Optional
connection_timeout	The timeout value used for socket connect operations. If connecting to the server takes longer than this value, the connection is broken.	0	The timeout is specified in seconds and a value of zero means that it is disabled	Optional

Maintenance

File Location: <CBA_INSTALLATION_LOCATION>/maintenance/lib/callbackDatabaseConfiguration.properties

Setting	Meaning	Default value	Remarks	Mandatory /Optional
min_connections	Minimum number of Connections a pool will maintain at any given time.	2	Please ensure that min_connections <= max_connections	Mandatory
max_connections	Maximum number of Connections a pool will maintain at any given time	50	Please ensure that min_connections <= max_connections	Mandatory
connection_idle_test_period	If this is a number greater than 0, c3p0 will test all idle, pooled but unchecked-out connections, every this number of seconds	600		Mandatory
acquire_increment	Determines how many connections at a time c3p0 will try to acquire when the pool is exhausted	2		Mandatory
max_statements	The size of c3p0's global	0	If both maxStatements and	Mandatory

	PreparedStatement cache.		maxStatementsPerConnection are zero, statement caching will not be enabled.	
max_idle_time	Seconds a Connection can remain pooled but unused before being discarded.	0	Zero means idle connections never expire	Optional
acquire_retry_attempts	Defines how many times c3p0 will try to acquire a new Connection from the database before giving up.	2	If this value is less than or equal to zero, c3p0 will keep trying to fetch a Connection indefinitely.	Optional
acquire_retry_delay	Milliseconds, time c3p0 will wait between acquire attempts	2000		Optional
checkoutTimeout	The number of milliseconds a client calling getConnection() will wait for a Connection to be checked-in or acquired when the pool is exhausted.	3000	Zero means wait indefinitely. Setting any positive value will cause the getConnection() call to time-out and break with an SQLException after the specified number of milliseconds.	Optional
testConnectionOnCheckout	If true, an operation will be performed at every connection checkout to verify that the connection is valid.	false	Be sure to set an efficient **preferredTestQuery** or **automaticTestTable** if you set this to true. **Performing the (expensive) default Connection test on every client checkout will harm client performance**. Testing Connections in checkout is the simplest and most reliable form of Connection testing, **but for better performance, consider verifying connections periodically using**	Optional

			idleConnectionTestPeriod.	
preferredTestQuery	Defines the query that will be executed for all connection tests.		Defining a preferredTestQuery that will execute quickly in your database may dramatically speed up Connection tests. (If no preferredTestQuery is set, the default ConnectionTester executes a getTables() call on the Connection's DatabaseMetaData. Depending on your database, this may execute more slowly than a "normal" database query.) NOTE: The table against which your preferredTestQuery will be run must exist in the database schema prior to your initialization of your DataSource. If your application defines its own schema, try automaticTestTable instead.	Optional
automaticTestTable	If provided, c3p0 will create an empty table of the specified name, and use queries against that table to test the Connection.		If automaticTestTable is provided, c3p0 will generate its own test query, therefore any preferredTestQuery set will be ignored. You should not work with the named table after c3p0 creates it; it should be strictly for c3p0's use in testing your Connection.	Optional
cache_entries_time_to_live_in_seconds	TTL of cached entries in seconds	0	Zero means that the 2nd level cache of	Optional

			Callback is disabled.	
cache_max_entries	Max Number of cached entries in the local heap	1000		Optional
application_name	Specifies the name of the application that is using the connection. This allows a database administrator to see what applications are connected to the server and what resources they are using through views like pgstatactivity.	maintenance		Optional
socketTimeout	The timeout value used for socket read operations. If reading from the server takes longer than this value, the connection is closed.	60000	This can be used as both a brute force global query timeout and a method of detecting network problems.The timeout is specified in seconds and a value of zero means that it is disabled.	Optional
tcpKeepAlive	Enable or disable TCP keep-alive probe. The default is false.	false		Optional
connection_timeout	The timeout value used for socket connect operations. If connecting to the server takes longer than this value, the **connection** is broken.	0	The timeout is specified in seconds and a value of zero means that it is disabled	Optional

CBA Alarms List

CBA sends alarms to SYSLOG. By default it will send UDP datagrams to port 514 on localhost. To receive them, (r)syslogd must be configured to receive remote datagrams. Sending them to a different server can be achieved in one of 2 ways:

1) Changing the IP address of the syslog daemon in log4j configuration of each CBA Component(s):
   - <BSR_SERVICE_HOME>/webapps/axis/WEB-INF/classes/log4j.xml
   - <ENGINE_SERVICE_HOME>/lib/log4j.properties
   - <MAINTENANCE_SERVICE_HOME>/lib/log4j.properties

2) Configuring syslog to relay messages received locally to another server such as IBM Tivoli. Configuring syslog is outside the scope of this document.

Alarms are sent in 3 levels of priority: INFO, WARN and ERROR.

Main alarms generated by CBA Server are listed below:

Component	Priority	Alarm	Remarks	Since
BSR Server	WARN	requested service <X> is not configured. Please check Call Center Applications configuration.	An incorrect Call Center Application is being referred by a CBA Configuration. Make sure the Call Center Application exists or the CBA Configuration is correct.	CBA 4.1.8
BSR Server	ERROR	missing or invalid configuration attribute <X>	A service impacting configuration attribute is missing from BSR configuration. This is more likely to happen when the server is being started.	CBA 4.1.8
BSR Server	ERROR	service [X] - selection strategy has not returned winner, default VDN will be selected. Please check server logs for more information	For some reason the strategy is not able to select a winner. This typically happens if a Call Center Application is not being to poll any of its associated VDNs, due to incorrect configuration, timeout, Session Manager or Communication Manager unavailability. More information can be found at bsr.log file. Please notice this is a critical condition, as calls will be routed to a default VDN/SKILL instead of the properly segmented call center group.	CBA 4.1.8
BSR Server	WARN	polling vdn X - No bsr data returned from polling	Happens when a polling VDN does not exist, is misconfigured (incorrect Vector steps) or there is a connectivity issue between CBA and SM/CM.	CBA 4.1.8
BSR Server	WARN	BSR Service is started/stopped	This marker is sent whenever the BSR Service is started/stopped. Can be used to make sure SYSLOG and centralized logging configuration is configured propertly.	CBA 4.1.8
BSR Server	INFO	Polling Service is started/stopped	This marker is sent whenever the Polling Service is started/stopped. Can be used to	CBA 4.1.8

			make sure SYSLOG and centralized logging configuration is configured properly.	
BSR Server	ERROR	proxy<X>(<ip>:<port>) is	There's watchdog process that will start and monitor whenever a SIP proxy goes down for any reason. It will log an alarm at every 10 seconds	CBA 4.1.8
BSR Server	INFO	proxy<X>(<ip>:<port>) is alive	The watchdog process will log the alarm whenever the SIP Proxy that was previously down comes up again.	CBA 4.1.8
BSR Server	WARN	The active Sip Proxy has been changed from X to Y	Every time that the active SIP proxy being used changes for any reason an alarm is triggered.	CBA 4.1.8
BSR Server	ERROR	No Active Sip Proxy available	In case all SIP proxy goes down this alarm will be triggered for every polling attempt	CBA 4.1.8
BSR Server	INFO	The Sip Proxy load balancing feature is enabled	If the load balancing feature is enable an alarm will be triggered at startup time	CBA 4.1.8
Engine	WARN	Free Ports < 10%. Available Free Ports: <x>	The free CBA ports are less than 10% of the total CBA ports. <x> is the count of free ports at the time of Alarm.	CBA 4.1.8.1
Engine	ERROR	Free Ports = 0	There are no free CBA ports available. Callback requests won't be delivered until free ports are available.	CBA 4.1.8.1
Engine / Maintenance	WARN	Master node (<MASTER_NODE_IP>) dead , setting Local node (<LOCAL_NODE_IP>) as master	In HA environment, if the master node is dead, the local node becomes master.	CBA 4.3
Engine	ERROR	License Mode changing to RESTRICTED Mode.	Due to some license error, the License mode getting changed.	CBA 4.3
Engine	WARN	License Mode changing to ERROR Mode (Grace Period).	Due to some license error, the License mode getting changed.	CBA 4.3
Engine	INFO	License Mode changing back to NORMAL Mode	Due to some license error, the License mode getting changed.	CBA 4.3
Engine	INFO	Request with ID <REQUEST_ID> cannot be launched as there is no CBA port available	When there is no CBA port available to process the callback request.	CBA 4.3
Engine	ERROR	Cannot process callback requests	The callback request(s) may not be updated to INITIATING status or the task cannot be submitted into the CCXMLThread.	CBA 4.3
Engine	INFO	No phantom stations/outbound channels available.	When no VP ports are available to process the callback requests.	CBA 4.3
Engine	ERROR	Unexpected Error occurred during Callback Engine execution.	When any exception happened during callback engine execution.	CBA 4.3

Below is a snapshot of how the alarms will appear in SYSLOG.

```
May 14 20:26:57 2015-05-15 00: 26:57,584 cba-bsrserver: INFO [ip: 135.122.99.205]
- BSR service is started
May 14 20:26:57 2015-05-15 00: 26:57,590 cba-bsrserver: INFO [ip: 135.122.99.205]
- Cache service started
May 14 20:26:57 2015-05-15 00: 26:57,788 cba-bsrserver: INFO [ip: 135.122.99.205]
- The Sip Proxy load balancing feature is enabled
May 14 20:26:58 2015-05-15 00: 26:58,278 cba-bsrserver: INFO [ip: 135.122.99.205]
- Polling service started
May 14 23:21:25 2015-05-15 03: 21:25,997 cba-bsrserver: ERROR [ip: 135.122.99.205]
- proxy1[135.122.61.172:5060] is unavailable
May 14 23:21:34 2015-05-15 03: 21:34,216 cba-bsrserver: ERROR [ip: 135.122.99.205]
- proxy3[135.122.61.171:5060] is unavailable
May 14 23:21:41 2015-05-15 03: 21:41,580 cba-bsrserver: WARN [ip: 135.122.99.205]
- polling vdn 5007005 - service[328] - No bsr data returned from polling
May 14 23:21:44 2015-05-15 03: 21:44,387 cba-bsrserver: WARN [ip: 135.122.99.205]
- proxy3[135.122.61.171:5060] is alive again. Answered in: 120 (ms)
May 14 23:21:56 2015-05-15 03: 21:56,152 cba-bsrserver: WARN [ip: 135.122.99.205]
- proxy1[135.122.61.172:5060] is alive again. Answered in: 69 (ms)
```

Result Size of getWebCallbackConfigurations WS API

Maximum number of callback configurations returned from **getWebCallbackConfigurations** and **getWebCallbackConfigurationsByAgentQueue** WS APIs for each request can be configured to a preferred number now, for better performance. By default this result size is 50.

This setting is kept in database table; to update this setting to new value, execute below commands where PostgreSQL is installed.

Please note that providing higher number for this configuration may degrade WS response time.

```
psql -p {PORT} -d {DB_NAME} -U {USER_NAME}

UPDATE CONFIGURATION SET VALUE = {NEW_VALUE} WHERE ID = 100;

Example:

psql -p 6198 -d callback -U callback

UPDATE CONFIGURATION SET VALUE = 100 WHERE ID = 100;

\q
```

Web service request need to send following two input parameters to limit the number of results.

Parameter	Description
offset	First row to retrieve in each call; this cannot be negative number
limit	Number of results to return in each WS call; this cannot be negative number; if this value is greater than the one configured in database, that value will be used instead of this one

To refer sample SOAP request with these input parameters, see the *Avaya Callback Assist Web services API guide*.