

Oracle® Enterprise Communications Broker User's Guide



Release P-CZ3.2.0

F29515-01

January 2020

The Oracle logo, consisting of a solid red square with the word "ORACLE" in white, uppercase, sans-serif font centered within it.

ORACLE®

Copyright © 2014, 2020, Oracle and/or its affiliates.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or "commercial computer software documentation" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

About This Guide

My Oracle Support xiii

1 Oracle Enterprise Communications Broker Overview

Packet Processing by the Oracle Enterprise Communications Broker	1-2
Ingress INVITE Processing	1-4
Identifying Source Context	1-4
Dial Plan Processing	1-5
Route Engine Processing	1-5
Egress Processing	1-5
Call Handling Example	1-5

2 SIP Signaling Management

The Oracle Enterprise Communications Broker Dial Plan	2-1
Oracle Enterprise Communications Broker Contexts	2-1
Context Hierarchy	2-2
Geographic Contexts	2-2
Corporate Contexts	2-3
Oracle Enterprise Communications Broker Agents	2-4
Why You Need Agents	2-4
How to Use Agents	2-5
Agent Groups	2-5
Oracle Enterprise Communications Broker Routing	2-6
Recursive Routing	2-7
Identifying Contacts and Specifying Routes	2-8
Route Selection	2-9
Forking	2-9
Fork Groups	2-10
Fork Group Assignment	2-11
Additional Targets	2-12
Fork Groups Configuration Considerations	2-12

Routing and ENUM	2-14
Route Types and Precedence	2-15
Active Directory and Oracle ECB Routing	2-15
LDAP and Oracle ECB Routing	2-16
LDAP Messages	2-20
LDAP Failure Events	2-20
Oracle ECB Limitations using LDAP	2-21
Configuring LDAP for Routing	2-21

3 Registrar and Authentication

Register Refresh	3-2
Proxy Registrations	3-3
Message Authentication for SIP Requests	3-3
Authentication	3-4
SIP Authentication Challenge	3-4
Authentication Header Elements	3-5
SIP Authentication Response	3-5
Authentication Check	3-5
Retrieving Information from Active Directory	3-5
LDAP and Oracle ECB Authentication	3-6
Configuring LDAP for Authentication	3-6

4 Getting Started

User and Administrator Access	4-1
Simultaneous Logons	4-1
RADIUS Server Roles and Access Privileges	4-1
Log On to the Web GUI	4-2
Log Off from the Web GUI	4-3
Service Provisioning	4-3
Service Provisioning Configuration Objects	4-3
Web GUI Tools	4-4
The User Menu	4-4
Help	4-5
Help Topics	4-5
About This Software	4-6
The Search Tool	4-6
Customize the Page Display	4-8
Tool Tips	4-9
Configuration Tools and Behavior	4-9

Configuration Wizards	4-11
Set Boot Parameters Wizard	4-12
Set Entitlements Wizard	4-13
Set Initial Configuration Wizard	4-14
Set License Wizard	4-16
Set Login Banner Wizard	4-16
Set Time Zone Wizard	4-16
Upgrade Software Wizard	4-17
Edit, Copy, and Delete Configurations	4-17
Configuration States and Behavior	4-18
Configuration Error Messages	4-19
Uploading and Downloading Configuration File Elements	4-20
Using Tag Fields	4-20
Dashboard Tab Operations	4-21
Add a Dashboard Widget	4-22
Configure Data Sampling Settings for a Dashboard Widget	4-23
View A Widget That is Not on the Dashboard	4-23
Widgets Page	4-24
Widgets and Descriptions	4-25
Command Line Interface Widgets	4-31
License Widget	4-32
Display and Clear Alarms	4-33
Display Users	4-33
Accessibility Features	4-33

5 Agent Configuration

Alphanumeric User Database and Call Routing Entries	5-1
About Traffic Constraints	5-4
Configure a Session Agent	5-5
Configure a Session Agent Group	5-10
Configure ENUM Servers	5-10
Enable ENUM Session Agent Group Matching	5-12
Multi-Hop Agent Ping	5-12

6 Dial Plan Configuration

Dial Pattern Configuration	6-1
Dial Pattern Encoding Characters	6-2
Configure a Dial Plan	6-3

7	User Configuration	
	Configure a User Entry	7-1
	Resolving to the Longest Match in the User Database	7-2
8	Using Policy to Refine Routing	
	The Redirect Action	8-2
	Configuring CNAM Replacement	8-3
	Using Policy to Normalize SIP Headers	8-3
	ANI Masking	8-4
	ANI Masking Configurations	8-4
	Define a Policy	8-5
	Applying a Policy to a Route	8-10
	Runtime Routing with Policies in the User Table	8-10
	Applying a Policy to the Registrar	8-11
	Configurations Using Policy	8-11
	Priority Call Handling	8-12
	Priority Call Configurations	8-12
	Transcoding and the Oracle Enterprise Communications Broker	8-13
	Transcoding Configurations	8-13
	Multiple Outbound Translations	8-13
	Outbound Translation Configurations	8-14
	Routing Action Configurations	8-15
	Deny Route Policy Configurations	8-15
	Stop Recurse Route Policy Configurations	8-16
	Stop Recursion by SIP Response Code	8-17
	Skip Route Policy Configurations	8-17
9	Routing Configuration	
	Routing Fields	9-1
	Route Policy	9-2
	Deny Patterns in Route Parameter Syntax	9-3
	Loop Sensing for PSTN Calls	9-4
	Configure Loop Sensing for PSTN Calls	9-5
10	Registrar Configuration	
	Registrar Configuration Fields	10-1
	Local Subscriber Table	10-2
	LST Configuration	10-2

Configure the Registrar with an LST	10-2
Edit an LST File	10-3
LST Runtime Execution	10-4
LST Redundancy for HA Systems	10-4
LST File Compression	10-5
LST File Format	10-5
localSubscriberTable	10-5
subscriber	10-5
LST Subscriber Hash and Encryption	10-6

11 LDAP Client Configuration

LDAP Configuration Options	11-1
Making Settings	11-2
Configure LDAP Server Access	11-2
LDAP Groups	11-5
Matching Criteria in LDAP Groups	11-5
Configure LDAP Groups	11-6
Configure a Routing Query	11-7
Address of Record Configuration Fields	11-9
SIP Authentication Query Configuration Fields	11-10
Replacing the Calling Number in the FROM Header	11-11

12 ACL Configuration

Configure an ACL	12-1
------------------	------

13 ECB Sync

Synchronizing the Registration Cache	13-3
Enable ECB Sync Operations	13-3
Add an ECB Sync Agent	13-4
ECB Sync Monitoring	13-4

14 HMR Configuration

SIP Header Manipulation	14-1
Multi-Hop SIP Header Manipulation Rules	14-1
SIP Header Manipulation Configuration Dialogs	14-2
SIP Header Manipulation Rules Attributes and Values Reference	14-4
SIP Header Manipulation Configuration	14-10
Configure SIP Manipulation	14-10

Configure a SIP Manipulation Header Rule	14-11
Configure a MIME Rule	14-13
Configure a MIME ISUP Rule	14-15
Configure a MIME SDP Rule	14-18

15 Monitoring Tab Operations

Monitor and Trace	15-1
SIP Notable Events Summary	15-2
Display a Notable Events Report	15-3
SIP Registrations Summary	15-3
Display a Registrations Report	15-4
SIP Sessions Summary	15-4
Display a Sessions Report	15-5
SIP Subscriptions Summary	15-6
Display a Subscriptions Report	15-8
Ladder Diagrams and Information Display Controls	15-8
Display a Ladder Diagram	15-10
Session Summary	15-10
QoS Statistics	15-12
SIP Monitor and Trace Filter Configuration	15-14
Search for a Record	15-15
Search for a Report Record	15-16
Specify Additional Identifiers	15-19
Specify Additional Search Options	15-19
Exporting Information to a Text File	15-20
Export Report Information to a Text File	15-21
Widgets	15-21
Agent Status Widget	15-29
Broker Lookup Widget	15-29
License Widget	15-31
Registration Cache Dashboard Widget	15-32

16 Troubleshooting and Maintenance

Audit Logs	16-1
Secure FTP Push Configuration	16-4
Configure Secure FTP Push with Public Key Authentication	16-4
Generate an RSA Public Key	16-5
Generate a DSA Public Key	16-5
Import a DSA Public Key	16-6

Copy the RSA Public Key to the SFTP Server	16-6
Configure Audit Logging	16-7
System File Management	16-9
Upload a File	16-11
Download a File	16-12
Delete a File	16-12
Back Up a Configuration File	16-13
Restore a Configuration File	16-13
CSV Configuration File Creation	16-14
Create a CSV Configuration File	16-16
Automatically Upload Updated CSV Configuration Files	16-16
Enable Automatic CSV Configuration File Uploads	16-19
Upgrade Software	16-19
System Reboot	16-20
Display Log Files	16-21
Display System Health	16-21
Obtain Support Information	16-21

17 Active Directory Modifications

18 Configuration Examples

Configuration Sequence	18-2
Initial Agent Configuration	18-2
Dial Plan Strategies	18-3
Route Strategies	18-4
Small Enterprise Model	18-5
Large Enterprise Model - v2	18-6
Emergency Dial Configurations	18-8
Alternate Translation Modes	18-9
ENUM Example Configuration	18-10

19 Format of Exported Text Files

Exporting Files	19-1
Session Summary Exported Text File	19-2
Session Details Exported Text File	19-3
Ladder Diagram Exported HTML File	19-9

20 SIP HMR (Header Manipulation Rules)

Guidelines for Header and Element Rules	20-2
Splitting and Joining Headers	20-2
Precedence	20-3
Duplicate Header Names	20-3
Performing HMR on a Specific Header	20-3
Multiple SIP HMR Sets	20-4
MIME Support	20-4
Manipulating MIME Attachments	20-5
Escaped Characters	20-6
New Reserved Word	20-6
About the MIME Value Type	20-6
Back Reference Syntax	20-7
Notes on the Regular Expression Library	20-8
SIP Message-Body Separator Normalization	20-8
SIP Header Pre-Processing HMR	20-9
Best Practices	20-9
About Regular Expressions	20-10
Expression Building Using Parentheses	20-11
Configuration Examples	20-11
Example 1 Removing Headers	20-12
Example 2 Manipulating the Request URI	20-13
Example 3 Manipulating a Header	20-14
Example 4 Storing and Using URI Parameters	20-15
Example 5 Manipulating Display Names	20-16
Example 6 Manipulating Element Parameters	20-18
Example 7 Accessing Data from Multiple Headers of the Same Type	20-21
Example 8 Using Header Rule Special Characters	20-22
Example 9 Status-Line Manipulation	20-25
Example 10 Use of SIP HMR Sets	20-26
Example 11 Use of Remote and Local Port Information	20-27
Example 12 Response Status Processing	20-28
Example 13 Remove a Line from SDP	20-30
Example 14 Back Reference Syntax	20-31
Example 15 Change and Remove Lines from SDP	20-32
Example 16 Change and Add New Lines to the SDP	20-33
Dialog-Matching Header Manipulation	20-35
About Dialog-Matching Header Manipulations	20-35
Inbound HMR Challenge	20-35
Outbound HMR Challenge	20-36

Built-In SIP Manipulations	20-37
Unique HMR Regex Patterns and Other Changes	20-37
Manipulation Pattern Per Remote Entity	20-37
Reject Action	20-38
SNMP Support	20-39
Log Action	20-40
Name Restrictions for Manipulation Rules	20-40
New Value Restrictions	20-41
Header Manipulation Rules for SDP	20-41
SDP Manipulation	20-41
sdp-session-rule	20-42
sdp-media-rule	20-43
sdp-line-rule	20-45
Regular Expression Interpolation	20-46
Regular Expressions as Boolean Expressions	20-47
Moving Manipulation Rules	20-49
Rule Nesting and Management	20-50
ACLI Configuration Examples	20-50
Remove SDP	20-51
Remove Video from SDP	20-51
Add SDP	20-51
Manipulate Contacts	20-52
Remove a Codec	20-53
Change Codec	20-53
Remove Last Codec and Change Port	20-54
Remove Codec with Dynamic Payload	20-55
HMR Import-Export	20-56
Exporting	20-56
Importing	20-57
Using SFTP to Move Files	20-57

About This Guide

The *Oracle® Enterprise Communications Broker User's Guide* provides the following information about the Oracle Enterprise Communications Broker (OECB) hardware and software.

- Configuration examples
- Configuring SIP signaling management
- Configuring dial plans, agents, users, policies, registrars, LDAP, ECB sync, Header Manipulation Rules, and routing
- Maintenance and troubleshooting

Oracle Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Documentation Set

The following table describes the documentation set for the OECB.

Document Name	Document Description
Administrator's Guide	Describes how to deploy the system.
Embedded Help system	Contains task-oriented topics for configuring, administering, maintaining, and troubleshooting the ECB hardware and software.
Release Notes	Contains information about the current release, including specifications, requirements, new features, enhancements, inherited features, known issues, caveats, and limitations.
SBC Family Security Guide	Provides information about security considerations and best practices from a network and application security perspective for the Enterprise family of products.
User's Guide	Describes how to configure SIP signaling management and how to tailor the system to specific needs.

Related Documentation

The following table describes related documentation for the OECB.

Document Name	Document Description
ACLI Reference Guide	Contains explanations of how to use the ACLI, as well alphabetical listings and descriptions of all ACLI commands and configuration parameters.
Administrative Security Essentials Guide	Contains conceptual and procedural information for supporting the Admin Security and Admin Security with ACP feature sets.

Revision History

The following table lists changes to this document and the corresponding dates of publication.

Date	Description
November 2019	<ul style="list-style-type: none"> Initial Release
January 2020	<ul style="list-style-type: none"> Adds a note to "Routing Fields" regarding the use of wild cards.

My Oracle Support

My Oracle Support (<https://support.oracle.com>) is your initial point of contact for all product support and training needs. A representative at Customer Access Support (CAS) can assist you with My Oracle Support registration.

Call the CAS main number at 1-800-223-1711 (toll-free in the US), or call the Oracle Support hotline for your local country from the list at <http://www.oracle.com/us/support/contact/index.html>. When calling, make the selections in the sequence shown below on the Support telephone menu:

1. Select 2 for New Service Request.
2. Select 3 for Hardware, Networking, and Solaris Operating System Support.
3. Select one of the following options:
 - For technical issues such as creating a new Service Request (SR), select 1.
 - For non-technical issues such as registration or assistance with My Oracle Support, select 2.

You are connected to a live agent who can assist you with My Oracle Support registration and opening a support ticket.

My Oracle Support is available 24 hours a day, 7 days a week, 365 days a year.

Emergency Response

In the event of a critical service situation, emergency response is offered by the Customer Access Support (CAS) main number at 1-800-223-1711 (toll-free in the US), or call the Oracle Support hotline for your local country from the list at <http://www.oracle.com/us/support/contact/index.html>. The emergency response provides immediate coverage, automatic escalation, and other features to ensure that the critical situation is resolved as rapidly as possible.

A critical situation is defined as a problem with the installed equipment that severely affects service, traffic, or maintenance capabilities, and requires immediate corrective action. Critical situations affect service and/or system operation resulting in one or several of these situations:

- A total system failure that results in loss of all transaction processing capability
- Significant reduction in system capacity or traffic handling capability
- Loss of the system's ability to perform automatic system reconfiguration
- Inability to restart a processor or the system
- Corruption of system databases that requires service affecting corrective actions
- Loss of access for maintenance or recovery operations
- Loss of the system ability to provide any required critical or major trouble notification

Any other problem severely affecting service, capacity/traffic, billing, and maintenance capabilities may be defined as critical by prior discussion and agreement with Oracle.

Locate Product Documentation on the Oracle Help Center Site

Oracle Communications customer documentation is available on the web at the Oracle Help Center (OHC) site, <http://docs.oracle.com>. You do not have to register to access these documents. Viewing these files requires Adobe Acrobat Reader, which can be downloaded at <http://www.adobe.com>.

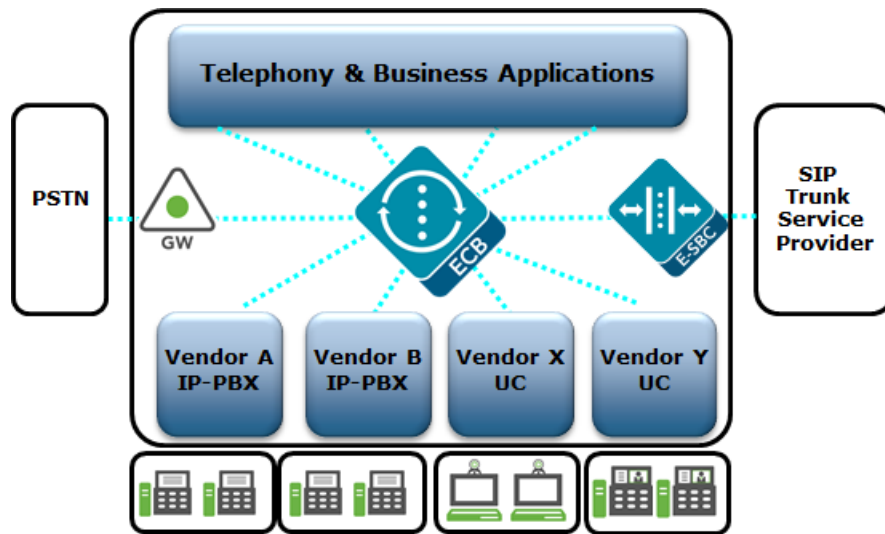
1. Access the Oracle Help Center site at <http://docs.oracle.com>.
2. Click **Industries**.
3. Under the Oracle Communications sub-header, click the **Oracle Communications documentation** link.
The Communications Documentation page appears. Most products covered by these documentation sets appear under the headings "Network Session Delivery and Control Infrastructure" or "Platforms."
4. Click on your Product and then Release Number.
A list of the entire documentation set for the selected product and release appears.
5. To download a file to your location, right-click the **PDF** link, select **Save target as** (or similar command based on your browser), and save to a local folder.

1

Oracle Enterprise Communications Broker Overview

The Oracle Enterprise Communications Broker (OECB) is an enterprise-class, core signaling component designed to simplify communications networks. It combines innovative approaches toward dial plan management and SIP topology-aware routing with a purpose-built, Graphical User Interface. While at its best in signaling environments comprised of products and solutions from multiple vendors, it is useful for consolidating policy enforcement decisions, integrating third-party applications, and managing a network-wide routing topology even in homogenous architectures.

The OECB is typically deployed in the core of a multi-vendor communications network where multiple UC, PBX, and service provider trunk interfaces must be interconnected. It normalizes communications between disparate premise-based systems and connects them to service provider networks and hosted applications through Enterprise SBCs.



Key benefits include:

- Increases scalability and simplicity
- Protects and extends investments in legacy communications infrastructure
- Reduces operations expenses
- Improves network availability
- Services and Applications

OECB operational functionality focuses around the following:

- SIP Signaling Management—The functional components of the OECB's software architecture for SIP signaling management focus around its dial plan, and its routing engine. These two components represent the foundation of the OECB's core SIP processing engine, and were specifically crafted to address, in a generic

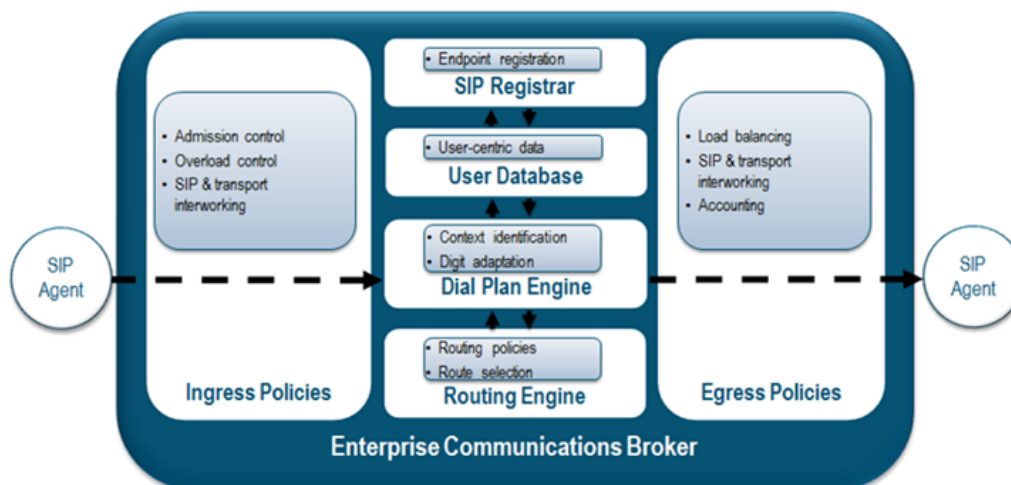
sense, the problems arising from the organic evolution of SIP-based enterprise communications networks.

- SIP Registrar—Provides a centrally-deployed location service for the enterprise.
- User Authentication—Provides for operation with an internal or external authentication resource, such as Active Directory, for authorization and authentication of users registering at the OECB.
- Header Manipulation—Provides telephony engineering with a means of assembling signaling header information specifically for the enterprise's operations, conformance and interoperability needs.

The *Oracle® Enterprise Communications Broker User Guide* provides operational explanations and configuration instructions for each of these.

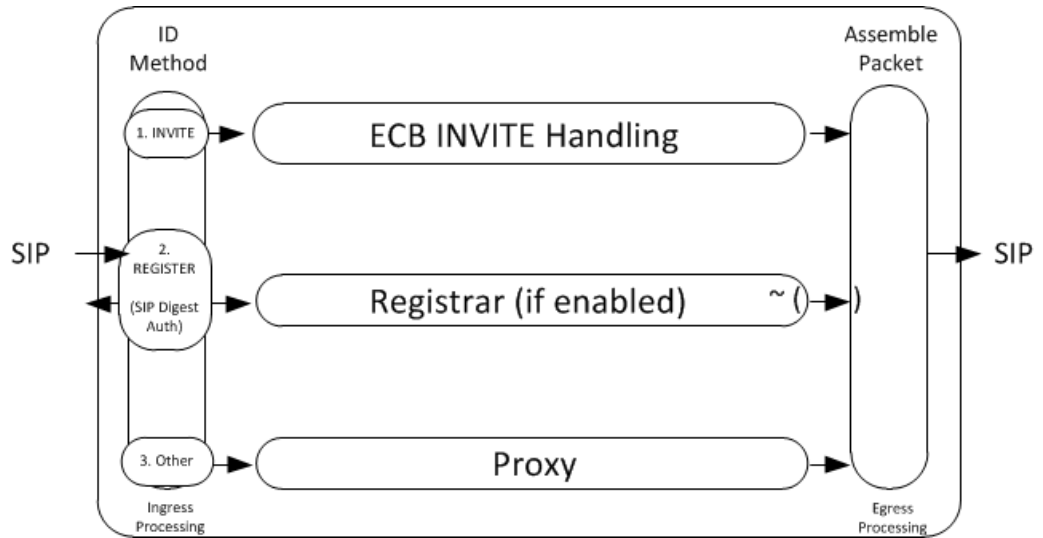
Packet Processing by the Oracle Enterprise Communications Broker

The following diagrams describe, at a high level, the processing performed by the elements of the Oracle Enterprise Communications Broker (OECB) to all traffic that it handles. Understanding this processing provides insight into configuration and troubleshooting tasks. The following diagram provides an example of the interactions of elements in a call flow between SIP agents.

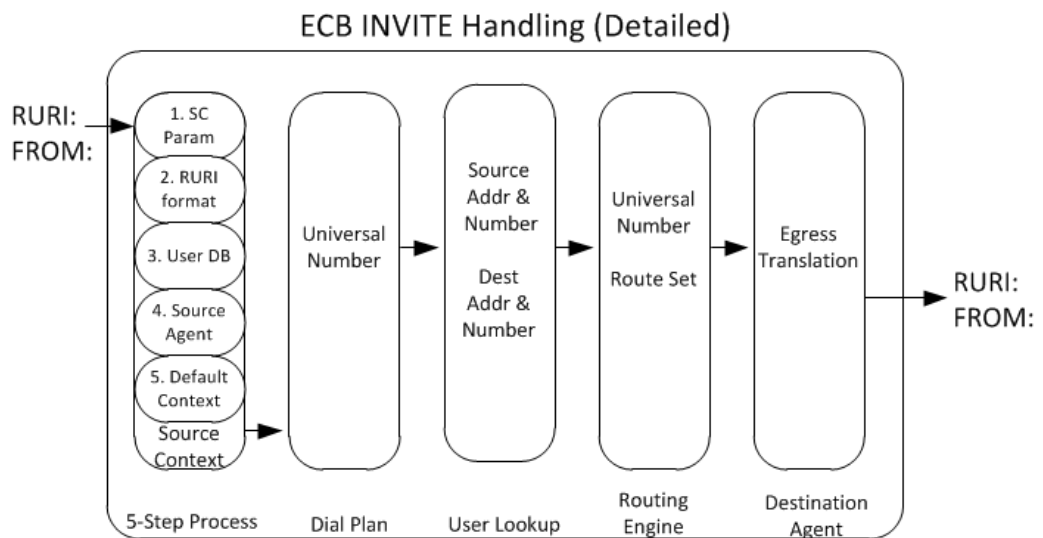


The following image displays OECB processing of different signaling messages, including:

- INVITE—Pass through the INVITE handling processes, which includes number normalization, route optimization, and multi-contact support.
- REGISTER—When configured as a SIP Registrar, registration traffic passes into the registrar for authorization, authentication, and caching. There are multiple means of performing authorization and authentication.
- Other—All other signaling traffic is proxied, based on RFC 3261 standards, including the insertion of VIA and Route Record headers to keep the OECB in the path of each applicable dialog's traffic.



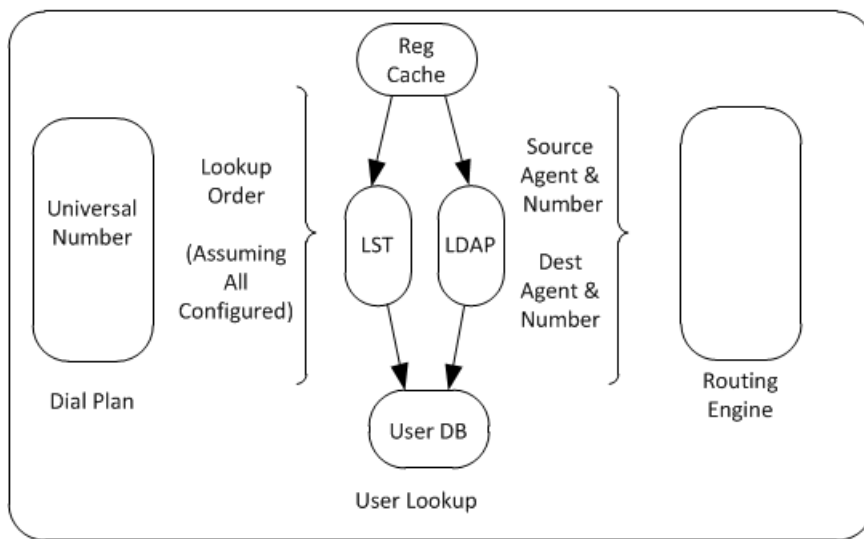
The following diagram displays the key processing elements handling an INVITE, including number normalization, based on context, end station lookup, and recursive route set creation.



The following diagram details the elements the system examines to perform user lookup. The OECB queries each of the objects shown in the diagram to identify the destination agent. After identifying the applicable agent, the user lookup hands everything the routing engine needs to recursively specify hop-by-hop routing through agents to reach the target.

Note that utilization of LST versus LDAP resources are independent and exclusive of each other. Either the LST or the LDAP resources perform the functions needed after registration cache procedures. The OECB allows you to configure either LST or LDAP resources.

User Lookup (Detailed)



Ingress INVITE Processing

When an packet arrives at a Oracle Enterprise Communications Broker ingress interface, standard link and network layer processing occurs to prepare the data for processing within the device. Subsequently, the Oracle Enterprise Communications Broker performs admission and overload control procedures to ensure it is both appropriate and possible to proceed with further processing. As discussed, ensuing processing is based on traffic type, of which INVITE processing is key to the overall purpose of the Oracle Enterprise Communications Broker. The sections below describe further INVITE processing.

Identifying Source Context

When receiving an inbound SIP message, the Oracle Enterprise Communications Broker first determines the *source context* of the calling party. This allows the Oracle Enterprise Communications Broker to interpret the dialed digits appropriately.

For example, a user dialing 911 in the United States has different expectations than a user dialing extension 911 in a European office.

The system performs four steps sequentially to identify the source context. If a step identifies a source context, the system skips the next steps and provides the information to the dial plan engine for subsequent processing. These steps include:

1. The system searches the FROM address in the signaling for a source context (SC) parameter. This parameter, if present, identifies the UA's source context.
2. If the number presented in the RURI begins with a "+" sign, assume the RURI is an e.164 number and bypass the source context identification.
3. The system treats the digits received in the userinfo portion of the From header as a universal address and checks to see if the calling party is in its User database.
 - a. If there is a match and the user has a source context configured, the system uses that as the call's source context.

- b. If the user has no source context configured, the system check the user's home agent for a source context and, if configured, uses that as the call's source context.
4. The system looks for a Source Context value in the configuration for the Agent from which the message was received.
5. If the above fail, the system uses the default Source Context, as configured in the SIP Interface settings.

Dial Plan Processing

The dial plan receives the dialed digits and the source context of that signaling message, and uses the rules associated with the identified context to prepare the *universal address* from the digits that were dialed. As described in the section on the dial plan engine, this may involve stripping routing digits out of the dial sequence, adding addressing digits into the sequence, or both.

The result of the dial plan processing yields the universal address that the system passes into the routing engine.

Route Engine Processing

The route engine receives the information from the dial plan lookup and builds a search key based on the calling number, called number, source agent, and destination agent for that call. As described in the section on Oracle Enterprise Communications Broker Routing, it recursively processes each route lookup result to construct full route sets.

Egress Processing

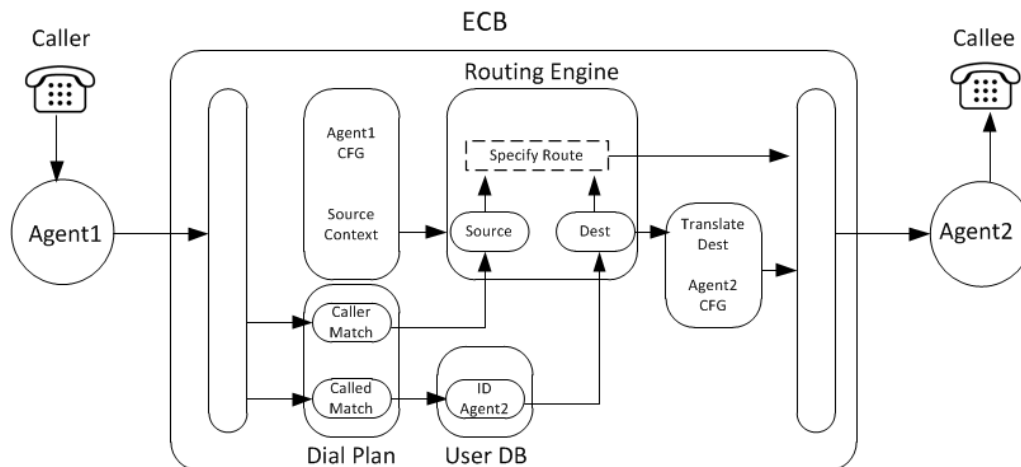
Now that the Oracle Enterprise Communications Broker has a fully qualified universal address, a route or set of routes to use for processing that call, it will prepare the universal address to suit the formatting requirements of the destination. It does this by looking for the Number Translation Mode of the *destination agent* (not any intermediate agents) and applying the transformation identified within that agent's configuration.

Lastly, the message is sent on its way based on the most preferred route. If that route fails, the Oracle Enterprise Communications Broker will try all subsequent route sets that it learned via the routing engine, in order from least cost to highest cost. This may also involve re-writing the universal address to suit the new "last hop".

Call Handling Example

The following illustration and call flow steps describe call handling through the Oracle Enterprise Communications Broker (OECB). This example assumes no registration cache and no LDAP configuration.

The following diagram shows a simple, intra-organization call passing through the OECB.



A user within the organization calls another within the organization residing on a different PBX. The call proceeds through the components of the OECB using the following steps:

1. Call Received by Ingress Processing.
2. Ingress Processing hands FROM and Request-URI to Dial Plan.
3. System runs 5-step process to ID source context.
4. Dial Plan normalizes Source and Destination Numbers.
5. FROM handed to Routing Engine as Source.
6. Dial Plan hands Request-URI to User DB to ID Home Agent.
7. User DB hands Request-URI to Routing Engine.
8. Routing Engine uses FROM and Agent1 CFG to create new Source.
9. Routing Engine builds Route.
10. Routing Engine hands Request-URI to Agent 2 configuration.
11. Agent 2 configuration translates Request-URI into format compatible with Agent 2.
12. Agent 2 configuration hands Request-URI to Egress processing.
13. Egress processing builds INVITE.
14. Egress processing sends new INVITE to Agent2.

2

SIP Signaling Management

Oracle Enterprise Communications Broker SIP signaling management requires review of the following topics:

- Dial Plan—Normalizes dialing numbers.
- Contexts—Provides rules for dialing number normalization.
- Agents—Establish hop locations for routes.
- Routing—Builds hop-by-hop path to the end-station's target agent.

The Oracle Enterprise Communications Broker Dial Plan

The Oracle Enterprise Communications Broker's dial plan engine was designed from the ground up to simplify the administration of common, real-world dialing behaviors. Conceptually, the dial plan engine allows administrators to define the rules by which dialed digit strings are built up, or broken down into "universal addresses". A universal address may be thought of as an E.164 number, although this is not strictly required. Universal numbers are required to be globally unique, not E.164-compliant.

These rules are then grouped into a foundation data structure in the Oracle Enterprise Communications Broker, the *context*. The concept of a context is fundamental to the operation of the Oracle Enterprise Communications Broker's dial plan configuration, and is discussed below.

The dial plan engine serves two purposes. First, it constructs universal addresses from input received. Second, it prepares egress translation from universal addresses into contextually-appropriate addresses based upon a message's destination. An example of the latter is the system creating a URI for a remote phone that needs to be addressed with four digits rather than a fully-qualified E.164 number.

Oracle Enterprise Communications Broker Contexts

A context is a collection of rules used to manipulate strings of dialed digits. In most use cases, contexts are associated with a PBX or branch office where the users associated with a given PBX are all subject to its rules for making telephone calls, such as:

- Each user on the PBX dials the same digit for seizing an outside line
- All users may be able to reach other extensions on that PBX by dialing short dial strings
- All users in that environment have access to the same 'tie lines'.

The rules that govern how to interpret the series of digits do not differ from user to user within that PBX.

Note that this does not take user-based entitlements into consideration. For example, users within the same context all dial the same videoconferencing terminal in the

corporate boardroom using the same series of digits, even though all of the users are not authorized to use that equipment.

Determination of a SIP message's "source context" is critically important. This is covered in more detail in "Ingress Processing". Phone numbers within a SIP message may have vastly different interpretations when, for example, a user dials "0" from two different branch offices within the same enterprise. The *context* of the dialing user differentiates the dialed pattern for the Oracle Enterprise Communications Broker (OECB).

The following list describes the terminology used to define contexts applicable to the OECB.

- **Geographic context**—A collection of rules that define the dialing patterns applicable to a geography, usually a country. These rules are outside of an enterprise's control and are pre-configured for you on the OECB by Oracle. You can extend or modify these rules.
- **Corporate context**—Rules defined by the enterprise that specify routing, policy, access code, and extension range dialing patterns. Rules may vary based on applicable PBX or branch office. Context hierarchy manages such variations.
- **Source Context**—The context used when an Agent provides context detail for a given call. This is also the context within which a given user resides by way of configuration, to be understood as a user's default location.
- **Source Context Param**—"sc", meaning source context, is the syntax for a parameter on the FROM header presented by equipment external to the OECB that specifies the context from which the call originated. When presented, the rules of this contexts are always applied.

Refer to "Dial Plan Configuration" for more information about the related fields.

Context Hierarchy

Contexts within the Oracle Enterprise Communications Broker may be defined *hierarchically*, to offer a parent/child inheritance relationship. This is done to avoid data duplication and redundant configuration.

For example, a large enterprise may have a corporate dial plan (common phone numbers for the IT help desk, employee benefits group, travel desk, etc.) that is consistent among all branch offices, and unique extension ranges per branch location. By defining common data in a "parent" context, each child context will inherit these common dial plan values and avoid the need for configuring each of them over and over for each branch office turn-up.

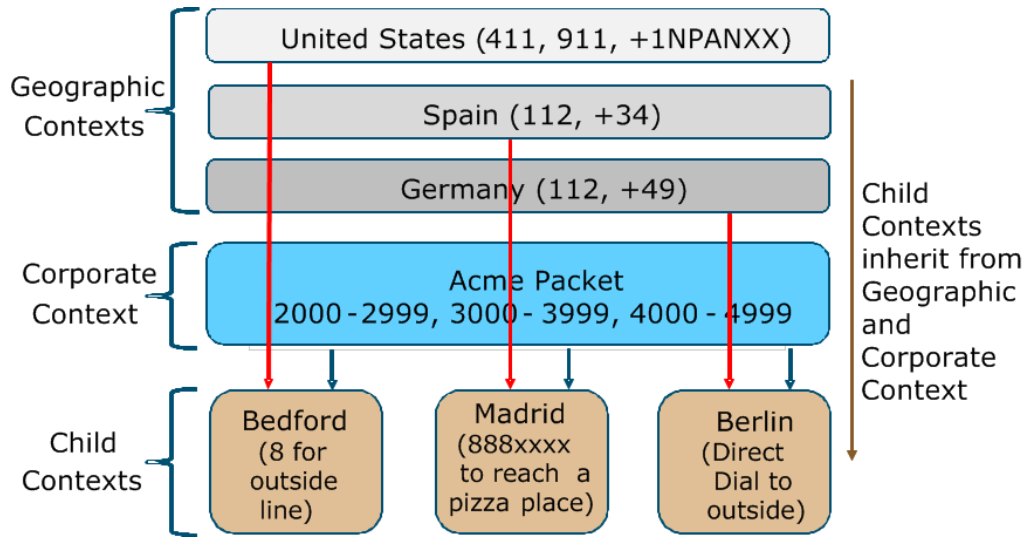
Each dialing context may have one corporate parent (for inheriting dialing rules that are unique for that enterprise) and one geographic parent (for inheriting common dialing rules pertaining to that branch office's physical location). Geographic and corporate contexts are described in the following sections.

Geographic Contexts

A geographic context is the set of rules for dialing within a given geography. It does not matter if you live in New York City or in Los Angeles, you'll still dial 011 for an international long distance call and 911 for emergency services because those are both part of the dial plan for the United States called the North American Numbering Plan (NANP). The Oracle Enterprise Communications Broker (OECB) ships with

geographic dial plans for the fifty most populous countries. You can override the default data or refresh it with future data, for example, to account for changes in the ITU dial plans.

Each context that you define on the OECB may have a geographic parent, configured as a geographic location. By configuring a geographic location, the child context inherits the dialing patterns for that geography. You do not need to configure the child contexts with rules for 011+, 911, 411, and so forth. They inherit these rules because they participate in that geography's parentage relationship.

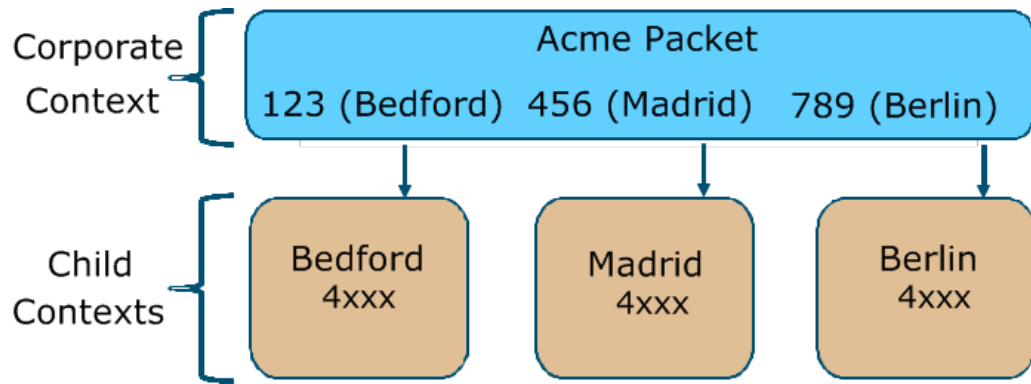


 **Note:**

The digit ranges within the child contexts do not overlap, presenting a simple means of identifying context. This represents a Small Enterprise OECB Configuration Model. The corporate dialing patterns are configured on the corporate context once. In the preceding diagram, when a caller dials 3xxx from any child context, the system always sends the call to Madrid.

Corporate Contexts

In contrast to geographic contexts, which are common for all telephone calls throughout the world and may be supplied with the Oracle Enterprise Communications Broker (OECB) software, corporate contexts are company-specific and define the dialing rules for the enterprise. A corporate context may include all branch offices, remote offices, PBXs, and so forth.



 **Note:**

In contrast to the example shown in "Geographic Contexts," the digit ranges in the child contexts overlap. The preceding diagram represents a Large Enterprise OECB Configuration Model. In this deployment, the system uses the dialed prefix to identify the child context. Each child context inherits the dial patterns of the parent to direct a call. Each child sends calls with the prefix 123 to a Bedford tie line, 456 to Madrid, and 789 to Berlin. You need to configure each pattern only once, on the parent (Acme Packet) context.

Oracle Enterprise Communications Broker Agents

An agent defines a signaling endpoint. It is a next hop signaling entity that applies traffic shaping attributes to flows. Agents provide important properties for Oracle Enterprise Communications Broker operation, including:

- Transit and termination points for Oracle Enterprise Communications Broker routes; and
- Context identification for use by the Oracle Enterprise Communications Broker dial plan.

Agents can include the following types of devices:

- Softswitches
- SIP proxies
- Application servers
- SIP gateways
- Indirect Agents

For each agent, concurrent session capacity and rate attributes can be defined. The Oracle Enterprise Communications Broker can provide load balancing across the defined agents.

Why You Need Agents

You can use agents to define hops the Oracle Enterprise Communications Broker can use in a signaling path. You can also use them to define and identify preferred carriers. This set of carriers is matched against the local policy for requests coming

from the agent. You can also set traffic constraints against specific hops via agent configuration.

In addition to functioning as a logical next hop for a signaling message, agents can provide information regarding next hops or previous hops for SIP packets, including providing a list of equivalent next hops.

How to Use Agents

Consider agents as next-hops within routing paths. Before configuring an agent, map out your session network and identify all potential agents. Each agent should be seen as a best hop based on its location, adjacencies and path costs. Redundant paths are also configurable using agents, allowing manual cost configurations for what may otherwise be equal cost paths.

In addition, consider the users for which each agent is a first hop. Agent configuration provides a method of defining routing and policy configuration for groups of users. Agents also provide a mechanism for defining source context for groups of users.

In some cases, specific addressing is not available or needed to access signaling endpoints. It may be that routing to a target domain is preferable to routing to a specific agent. In these cases, you can configure an agent using, for example, only the target domain name rather than a specific endpoint. When doing this, you assume that the domain itself is able to route to any further hops needed to reach the UA and that the same policies must be utilized from all traffic from that domain.

Agent Groups

Agent groups contain multiple agents. Members of an agent group are logically equivalent (although they might vary in their individual constraints) and can be used interchangeably as transit targets for SIP traffic. For one reason or another, a given agent may not be able to service traffic. You configure agent groups to establish multiple transit destinations for purposes such as redundancy.

Examples of agent groups include the following:

- Application Server cluster
- Media Gateway cluster
- Softswitch redundant pair
- SIP Proxy redundant pair
- Gatekeeper redundant pair

Agent group members do not need to reside in the same domain, network, or realm. The Oracle Enterprise Communications Broker (OECB) can allocate traffic among member agents regardless of their location. The OECB uses the allocation strategies configured for an agent group to allocate traffic across the group members.

You configure agent groups from the Agent configuration dialog on the GUI. The configuration consists of naming the group, selecting the allocation strategy, selecting recursion preference, and adding the agent group members.

After you configure the group, you configure agent group names as:

- A `Dest agent` in a routing table entry
- A `Route` in a routing table entry

- A user's `Home agent` in the user database
- A `Default home agent` within an LDAP query

The syntax for these entries appears as the word 'group' followed by a colon (:) and the group name. For example,

```
group:MyGroupName
```

When configuring the group, you select between the following allocation strategies to define the method of selecting the next member of the group for a connection attempt, if the previous connection attempt is unsuccessful:

Hunt	The Oracle Enterprise Communications Broker selects the agents in the order in which they are configured in the agent group. If the first agent is in service, and has not exceeded any defined constraints, all traffic is sent to the first agent. If the first agent is out of service, or is in violation of constraints, all traffic is sent to the second agent. And so on for all agents in the agent group. When the first agent returns to service, the new traffic is routed back to it.
Round robin	The Oracle Enterprise Communications Broker selects each agent in the order in which it is configured, routing a session to each agent in turn.

To summarize, agent group operation requires the following configuration:

- Two or more agents
- An agent group containing those agents
- A route, user configuration or LDAP query that directs traffic to that group

Recursion

Agent groups use a recursive process to communicate with agent group members. Recursion behavior is specified by the allocation strategy. You can optionally configure the Oracle Enterprise Communications Broker to attempt communications with only one member of the agent group by leaving the **Try All** control deselected, which means disabled.

The agent group performs its agent selection rotation process independently of the recursion setting. Each allocation strategy rotates agent selection as a means of selecting the first agent to try. This ensures that the system continues to use each agent in the group as a message target.

Routing paths may traverse multiple agent groups. The system, performs recursion only on the last agent group in the path to help prevent an inordinate number of connection attempts.

Oracle Enterprise Communications Broker Routing

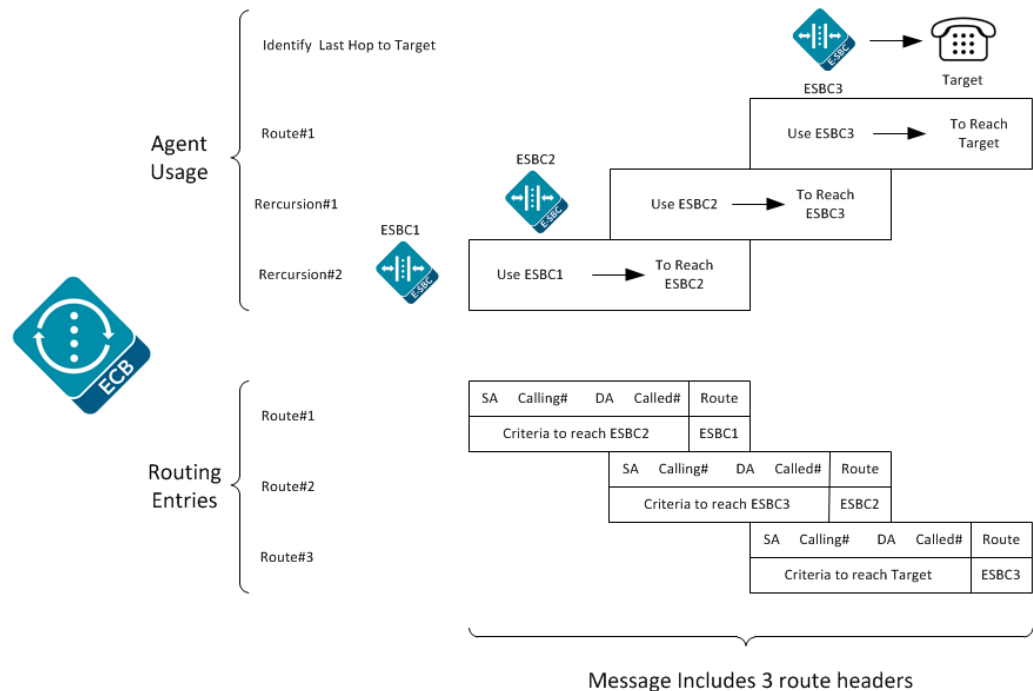
The Oracle Enterprise Communications Broker (OECB) employs a purpose-built SIP routing engine for packet processing. Unlike traditional SIP proxies, application servers, or Session Border Controllers, the OECB may be provisioned with a complete network topology map of all signaling entities, and use this provisioned data to make fully-informed routing decisions on how signaling flows should travel through a SIP

network. Beyond choosing a next hop and pushing the signaling message on its way, the OECB will look at the entire path from origin to destination to find the path with the least cost, fewest active sessions, least number of hops, and so forth. The OECB pre-populates the egress signaling message with a specific route set to inform each receiving device on the next element in sequence.

Recursive Routing

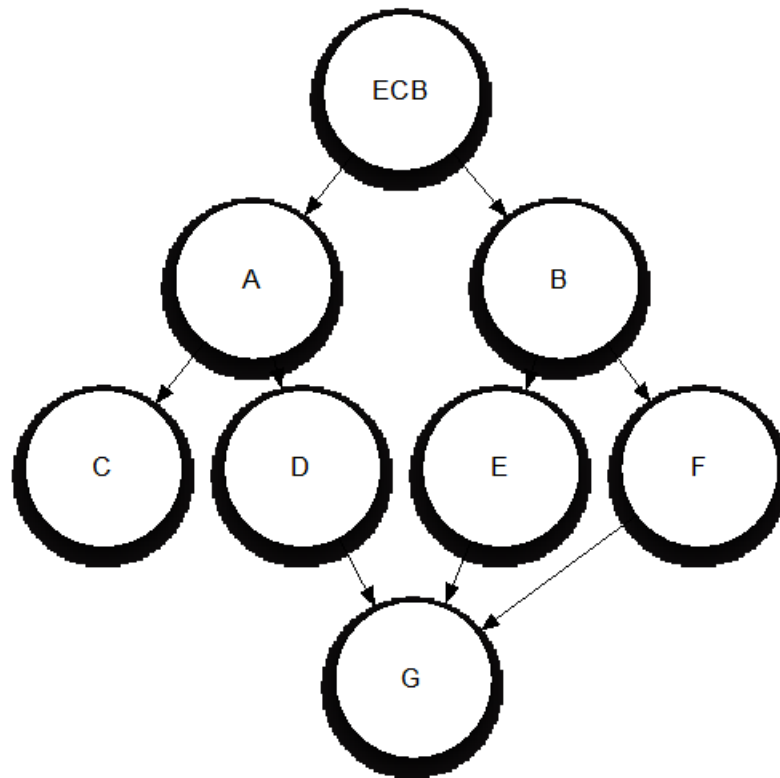
Conceptually, the routing engine in the Oracle Enterprise Communications Broker (OECB) is similar to the recursive routing engine in a layer three router. The system provides the route engine with input criteria, including calling number, called number, source agent, and destination agent. The routing engine returns a set of results based on the lookup. The OECB processes each result recursively until complete, with each loop building another hop on the route.

The process of using recursion to create routes consists of identifying individual hops for an end-to-end path beginning with the last hop before the destination. The routing engine selects subsequent elements (agents) to identify the hop that is next after the previously identified hop until it has a full path between itself and the UE. The OECB routing engine performs this process for all possible paths to each agent, creating multiple choices for the system to use as a route set for every individual call.



The following diagram is an example of a rudimentary network topology to illustrate how the OECB uses recursion to identify a route path.

1. The OECB discovers that Agent G is the last stop on the signaling message path.
2. The OECB resolves each way that it can reach Agent G. It finds that it can reach G by way of agents D, E, and F.
3. The OECB looks up how to reach D, E, and F, and the route table yields Agents A and B. Because Agents A and B are directly connected to the OECB, route path identification is complete.



Identifying Contacts and Specifying Routes

Having determined the target for a call, the Oracle Enterprise Communications Broker creates route sets to the target. In addition, the Oracle Enterprise Communications Broker also finds all of a target's contacts and builds route sets for each of them. Depending on deployment and configuration, these contacts are available from multiple sources, both internal and external to the Oracle Enterprise Communications Broker. These sources provide the Oracle Enterprise Communications Broker with the agent of each contact, from which it can build routes to the contacts.

The Oracle Enterprise Communications Broker uses called, calling number and source agent from the source of the call, and the called number and destination agent to create a route. It goes about collecting this detail for target and all contacts using the following procedure:

1. Find the AoR and all associated contacts in the registration cache. Store agent(s) for route creation.
2. Find caller and callee's source context, as presented earlier in this document.
3. Convert user portion of request-URI and From URI to universal number via the dial plan and source context. These become called and calling number.
4. Lookup called and calling number in the user database. Retrieve each number's agent for route creation.
5. If the user database lookup does not produce source and destination agent, request this information via the LDAP database. If necessary, the system converts the universal number so the lookup format is compatible with the LDAP database.

6. If the LDAP lookup does not return source and/or destination agent, the Oracle Enterprise Communications Broker uses the host portions of the Request-URI and FROM URI or inbound agent (agent on which the call was received) as agents.
7. For any AoR returned from LDAP, the Oracle Enterprise Communications Broker performs another lookup in the registration cache and creates routes for the AoR.
8. If the LDAP lookup identifies other contacts, the system passes those contacts through the registration cache to identify its agent and build the route set.
9. All routes are built and, depending on forking configuration, placed in order.

If any of these sources are not configured or operational, the Oracle Enterprise Communications Broker proceeds to the next source. Note that, if agents are found in the user database, the system does not perform an LDAP lookup.

 **Note:**

The procedures associated with an LDAP resource are equivalent to those with an LST resource. These procedures are also exclusive; LST and LDAP resources cannot be used simultaneously for routing.

The system forwards the request based on the route list and the forking configuration. By default, the system performs serial forking to all contacts using route cost to establish the order. The system can also perform parallel forking, if desired.

Route Selection

After constructing routes to use for a call, the Oracle Enterprise Communications Broker (OECB) often can choose from multiple routes. Cost calculations for each route identify the route that the system uses.

In the network example in "Recursive Routing," the OECB determined the three following potential routes for an inbound message sent to Agent G:

- A to D to G
- B to E to G
- B to F to G

Each route between agents in the OECB routing table may be assigned a cost that represents the desirability of that route. The OECB sums up the total cost for each route and orders them from lowest to highest cost. It then selects the lowest cost route and forwards the message.

Forking

Forking is a routing option that the Oracle Enterprise Communications Broker (OECB) uses to direct an INVITE to multiple targets. Several types of forking control the operations of the function, such as timing and target lists. The OECB performs serial forking to all targets by default. You can enable the OECB to perform parallel forking, which directs the INVITE to all targets for an Address of Record (AOR) simultaneously. When any target responds, the OECB issues a CANCEL to the other targets and ignores any responses from them. Should the OECB receive error messages from all contacts, it provides the lowest number message back to the caller.

The OEGB detects a user's multiple contacts from:

- A configured LDAP server
- The local registration cache
- The User Database

The system can fork to multiple contacts that come from the following locations:

- Registration cache (the minimum requirement)
- Registration cache + LDAP
- Registration cache + User Database

When the system finds the contact used in the RURI in the User Database, it uses that contact and then LDAP queries only on the user in the "From:" header.

When the User Database does not contain the number in the RURI, LDAP performs the lookup on the RURI and the system forks the call based on the result.

You cannot configure forking to:

- LDAP directly
- User Database directly
- Registration cache + LDAP + User Database

When the OEGB detects multiple routes to a contact, the system uses cost configuration to determine the preferred route. The system uses backup routes only when the primary routes do not respond.

You can enable parallel forking in the **SIP Interface** configuration.

Fork Groups

Fork-groups on the Oracle Enterprise Communications Broker are sets of one or more contacts that the system attempts to reach simultaneously. The system uses fork group order to specify when it tries to reach each fork group's contacts. This results in a hybrid of serial and parallel forking operation. The user can configure fork groups on agents, the registration cache and within the LDAP database. The user can also configure a global fork group timer with a value from 0 to 32 seconds on the sip-interface. If the system does not receive a response from any contact within that time, it tries the next fork group. Parallel forking must be enabled.

By default, the Oracle Enterprise Communications Broker assigns all contacts to fork group 1 and attempts to contact them serially, using the order in which it learns them. If desired, the user can enable parallel forking. By itself, parallel forking causes the system to attempt to reach all contacts simultaneously. Fork groups refine parallel forking, allowing the system to try all contacts in a group, and then move on to the next group.

The user names fork groups using decimal numbers between 1 and 100. This naming defines fork group order, with the system using fork group 1 first. The user configures objects with fork group numbers, based on a forking plan they devise.

The user can also configure a lookup query to LDAP databases to retrieve individual contacts' fork groups. The user must have previously modified the LDAP database to include a custom fork group field in contact records.

A use case for this feature could include the system attempting to reach a user's BYOD and desk phone simultaneously, then forwarding to an enterprise-preferred voicemail server if neither answers. For this to work, the BYOD and desk phone would be in the same fork group. The voicemail server would be a member of a higher numbered fork group. To ensure this order, the system assigns lower numbered fork groups with a higher precedence.

After establishing a session, other contacts may respond to try and start the session themselves. The Oracle Enterprise Communications Broker replies to these messages with a CANCEL.

Fork group operation does not exclude the use of primary and backup routes. The Oracle Enterprise Communications Broker still creates route sets for all contacts. If a contact fails via a primary route, the system attempts to reach the contact using all backup routes, based on cost.

If the Oracle Enterprise Communications Broker receives a redirect from an endpoint, the system adds the redirect target to the current fork-group and tries to contact it before attempting the next fork-group. If the global fork group timer expires before the system receives a redirect, however, the system proceeds to the next fork group.

The flexibility inherent in fork group operation requires the user to carefully plan forking prior to configuration. For each call, the system creates an ordered contact list, based on fork group configuration. Because the fork group assignment may affect multiple contacts, such as agent configuration, the user must be careful not to configure a sequence that would adversely affect calls to different end stations behind that agent.

Fork Group Assignment

You configure fork groups to specify call attempt order for a given call. The Oracle Enterprise Communications Broker (OECB) creates call attempt lists based on each contact's fork group assignment.

Upon configuration, the system assigns fork groups to target endpoints, as follows:

- User database—Each user database entry is assigned to the home-agent's fork-group.
- Registration cache—Each registration cache entry is assigned to the SIP registrar's fork-group.
- LDAP server—Each contact retrieved from an LDAP server is assigned to the a fork-group specified in the server's user record. If no fork-group is configured for the user in the Active Directory, the system assigns the target endpoint to the fork-group of the user's home-agent, as configured on the LDAP server.

The OECB uses the following contact source order:

1. Registration cache contacts
2. User database contact
3. LDAP contacts
4. LDAP AORs generating subsequent contact dips for additional registration cache contacts

By default, the OECB collects contacts from these sources and creates a contact list that follows the order in which the system learns them. This behavior is in accordance with default fork group operation, where all contacts are in fork group 1 and the mode

is to fork serially only. As soon as the system finds differentiation between contact fork groups, it arranges contact lists using the fork group order.

Additional Targets

You may want to include forking targets to stations that are not resolved as original call targets. Examples of these scenarios include directing calls to a preferred enterprise voice mail server, if they are not picked up. The Oracle Enterprise Communications Broker (OECB) provides for this using Additional target configurations. You manually configure these devices within **Additional target groups**, which includes one or more end stations. Agent and registrar configuration allows you to select these groups as additional forking targets for all calls that use that agent or registrar's entries.

An additional target group is a list of agents or end stations that the OECB uses as candidates for either parallel or serial forking. You configure these groups with fork group numbers, which the system then uses to define fork group order. The system adds additional target contacts to the forking sequence the same way it adds contacts for other objects with fork group configurations.

Fork Groups Configuration Considerations

Fork Group configuration requires that you establish a clear plan prior to any configuration. Configurations established by this planning may include:

- Identify or create new agents as fork group targets.
- Identify usage and precedence policy for forking via the Registrar.
- Adjust fork group identification and precedence based on preferred LDAP lookup scenarios.

Coordinating the use of these sources and configuring the applicable objects establishes and refines fork group configuration. Applicable configuration objects include:

- **Agent**—Create new agents specifically for use in a fork group, or uses existing agents. Configures an agent with a single fork group number, which the system applies to every call using that agent as a route.
- **Additional targets**—Create sets of targets to manually establish forking targets.
- **Registrar**—Set the registrar to a single fork group, which the system applies to every contact in the registrar.
- **LDAP**—Define a lookup query that pulls the pre-configured fork group assignment defined for the queried contact. The query must pull this fork group assignment from a custom attribute established on the LDAP database.

Configure Fork Groups on Agents

1. Access the Agent configuration object.
Configuration, Service Provisioning, Agents, Session Agent.
2. Click either Add to create a new agent or **Edit** to add the agent to a fork group.
3. On the **Agents** page, do the following:

Additional Target Group	Select a target group from the drop down list.
Fork Group	Enter a digit to specify this agent's fork group priority in a target list. Range: 0-100.

4. Click **OK**.
5. Save the configuration.

Configure Additional Target Groups

Additional targets are agents or end stations that are not contacts already targeted by a given call. You assign additional target groups on a per-agent and a per-registrar basis.

Configure target groups.

To configure additional target groups:

1. Access the Agent configuration object.
Configuration, Service Provisioning, Agents, Additional Target Group.
2. On the Additional Target page, click **Add**.
3. On the Add Additional Target page, enter a name for this target. Use this name to assign this group to an agent or the registrar.
4. Under Additional Target, click **Add**.

The system displays the Additional Target Group / Additional Target page.

5. On the Additional Target Group / Additional Target page, do the following:

Additional Session Agent	Select an existing target group from the drop down list, or enter an IP address of a target station.
Fork Group	Enter a digit to specify this agent's fork group priority. The lower the digit, the higher the priority. Default: 1. Range: 1-100.

6. Click **OK**.
7. Click **OK**.
8. Save the configuration.

Assign the Additional Target Groups to the appropriate agent or the registrar.

Configure Fork Groups on a Registrar

1. Access the SIP Registrar configuration object.
Configuration, System Administration, SIP Registrar.
2. On the Settings page, do the following:

Additional target Group	Select an existing target group from the drop down list.
Fork Group	Enter a digit to specify the fork group for every contact in the SIP Registrar end points. Default: 1. Range: 0-100.

3. Click **OK**.
4. Save the configuration.

Configure LDAP for Fork Groups

- Configure the LDAP server that you want to use for fork groups.
 - Define and populate the custom LDAP attribute for specifying a fork group.
1. Access the LDAP Configuration object.
Configuration, System Administration, LDAP, LDAP Config.
 2. Open an existing LDAP configuration for editing.
 3. Scroll to **Routing, Lookup Queries**, and click **Add** .
The system displays the Add LDAP Config / Routing / Lookup Query dialog.
 4. In the **Fork Group Attribute** field, enter the name of the custom attribute in the LDAP database that includes fork group assignments.
 5. Click **OK**.
 6. Click **OK**.
 7. Save the configuration.

Configure the Global Fork Group Timer

The global fork group timer specifies the amount of time the system waits for responses from a fork group before it tries contacting the next fork group.

After this timeout, the system drops responses received from contacts in the expired fork group.

1. Access the SIP Interface configuration object.
Configuration, System Administration, SIP Interface, SIP Config.
2. On the SIP Config page, for the **Fork Group Timeout** parameter, enter a time in seconds. Default: 0. When set to the default, the system waits for the standard SIP INVITE transaction timeout to expire before proceeding with the next group.
Range: 0-32.
3. Click **OK**.
4. Save the configuration.

Routing and ENUM

The ENUM functionality lets the Oracle Enterprise Communications Broker make an ENUM query for a SIP request. The ENUM lookup capability lets the Oracle Enterprise Communications Broker transform E.164 numbers to URIs during the process of routing (or redirecting) a call. During the routing of a SIP call, the Oracle Enterprise Communications Broker determines if an ENUM query is required and if so which ENUM server(s) need to be queried. A successful ENUM query results in a URI that is used to continue routing or redirecting the call.

Refer to the chapters on Agent and Route configuration for instructions on the related fields.

Route Types and Precedence

There are three types of routes used by the Oracle Enterprise Communications Broker. These include configured, default and implicit. The Oracle Enterprise Communications Broker uses these types, in conjunction with route cost, to determine route order. You create both configured and default routes in your route table. A default route is simply a route configured with wildcards for called number, calling number, source agent and destination agents. The system installs implicit routes dynamically when there are no explicitly configured routes to an agent. The system assumes the agent is to be a directly connected next-hop, and subsequently relies on the network infrastructure to reach that agent when needed.

The system orders routes by cost first, with the lowest cost being preferred. If costs are equal, the system orders by type, with the preference given to configured, then implicit and then default. If route cost and type are all equal, the system orders routes according to hop count, with the lowest number of hops being preferred.

Refer to the chapter on Route configuration for instructions on the related fields.

Active Directory and Oracle ECB Routing

A large percentage of Enterprises use call servers with Active Directory (Domain Controller) such as Media Servers, Exchange Servers, Lync Servers, and so on. For Enterprises that integrate these servers in parallel to their existing communications infrastructure, or transition from their legacy Private Branch Exchange (PBX) to these types of servers, Active Directory becomes a more efficient and cost-effective way of routing the incoming calls within the core Enterprise network.

Clients using Microsoft servers such as a Lync Server deploy their own URI. Therefore, a user in a network with both a desk phone and a Lync client may have an IP PBX extension/URI for the desk phone, and a different URI for the Lync client. Currently, all PSTN traffic is sent by default to a legacy PBX in the core network. If the PBX does not recognize the extension/URI, the PBX forwards it to the Lync client. Sending traffic to the PBX first and then to the Lync Server can be costly in terms of computing resources and licensing fees. Routing all incoming sessions from a SIP trunk to the Lync Server first and then to a PBX can be costly.

As a solution, the Oracle Enterprise Communications Broker (OECB) initiates a query to the Active Directory to determine how to route the call. The data fetched is the agent of the targets pre-configured in the database.

The OECB stores the data used to facilitate the routing decision of the call (performed by Lightweight Directory Access Protocol (LDAP) and routes the call the first time to the applicable destination (PBX or Lync Server).

In scenarios where a user has multiple contacts such as both a Lync phone and a legacy PBX phone, calls destined for the Lync phone number can be routed to the PBX phone number, or calls destined for the PBX phone number can be routed to the Lync phone number. The destination is dependant on the current OECB configuration. The OECB uses the information stored in the Enterprise's Active Directory, compares it to the OECB configuration and then determines which phone number to utilize.

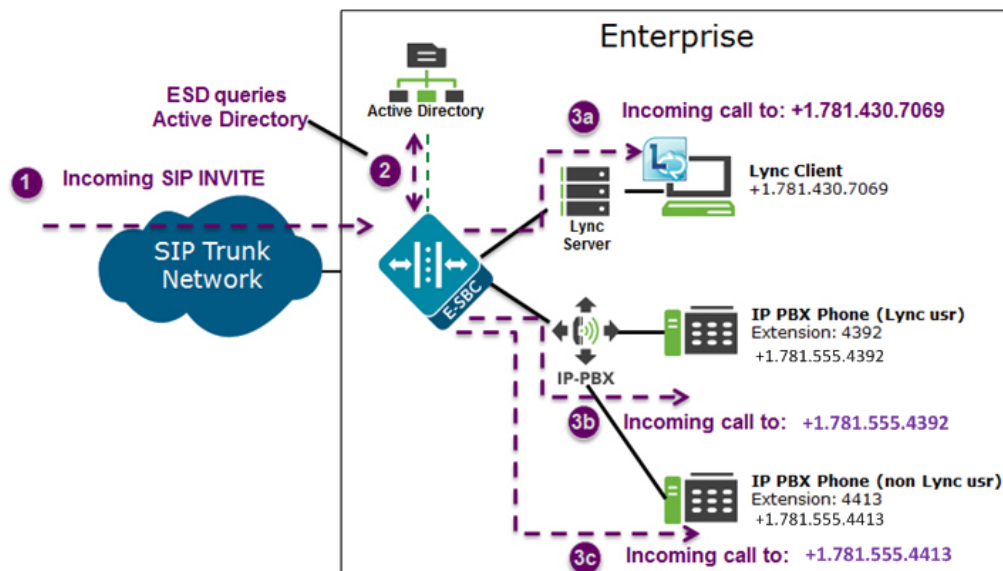
Note:

The Active Directory-based call routing feature supports confidential and secure LDAP traffic support by using SSL TLS (LDAPS).

Active Directory-based call routing is a feature of the OECB that facilitates the routing of incoming calls to the appropriate destinations within the Enterprise core network. The OECB LDAP query to the Active Directory yields the agent of the phone number.

The following diagram shows incoming call flow with Active Directory and the OECB in the network. The IP PBX extension (4392) is the primary telephone number (+1.781.555.4392). The secondary transition number (+1.781.430.7069) is assigned to Lync.

1. When the OECB receives an inbound SIP INVITE over a SIP Trunk, it checks the current LDAP configuration in the OECB.
2. Depending on the configuration, the OECB accesses the Enterprise's Active Directory to search for the applicable number being called by way of an LDAP query.
3. When the LDAP query finds the agent of the called number, the OECB builds a route set for the call and routes the call, per the routing engine, directly to the call server client (3a) or to the IP PBX phone (3b) and (3c).



The Enterprise is responsible for migrating phone numbers from the legacy PBX to the call server by making the necessary updates in their Active Directory so the OECB can route the call properly. In the illustration above,

LDAP and Oracle ECB Routing

The Oracle Enterprise Communications Broker (OECB) uses Lightweight Directory Access Protocol (LDAP) to perform queries to the Enterprise's Active Directory (AD) to determine where to route incoming calls to the call server or the IP PBX in the Enterprise network. Session requests and responses are sent and received based on

the OECB's LDAP routing configuration. LDAP determines the destination, to a call server user or a non-call server user, and forwards the call accordingly.

The OECB, using LDAP, performs the following on an inbound call:

- Creates an LDAP search filter based on the dialed number and the configured LDAP attributes
- Sends an LDAP search query to the configured LDAP Server
- Creates a route list based on the query responses received from the LDAP Server and the applicable attributes it already has (caller number, callee number, caller agent)
- Routes calls based on the route list and routing order. The routing order is dependent on the LDAP attribute configuration and whether there was an exact match for the dialed phone number in the Enterprise's Active Directory.
- If configured, searches for additional AoR matches in Active Directory so that it can create additional routes to target users that have contacts stored in separate records.

To use AD as a source for home agent names, you create lookup queries from the LDAP routing configuration dialogs. The OECB uses LDAP to retrieve that information and create routes. If the system cannot derive a home agent name from the query results, it routes the call to the configured default home agent.

 **Note:**

You must ensure that phone numbers in the LDAP database are unique. If the OECB encounters multiple records with the same number, it cannot process the lookup.

The OECB keeps a permanent LDAP session open to all configured call servers. It sends an LDAP bind request on all established connections to those servers. The first call server is considered the primary LDAP Server, and all others are secondary LDAP servers. If a query request sent to the primary server is unsuccessful, the OECB sends the request to the next configured LDAP Server until the request successfully in gets a response. If the OECB receives no response and cannot find another route successfully, it sends a busy to the caller.

LDAP performs call routing based on LDAP attributes configured on the OECB. The **route-mode** attribute setting determines how LDAP handles the called number when accessing the Enterprise's Active Directory. You can set routing modes to any of the following:

- Match-only (default)
- Attribute-order
- Match-first

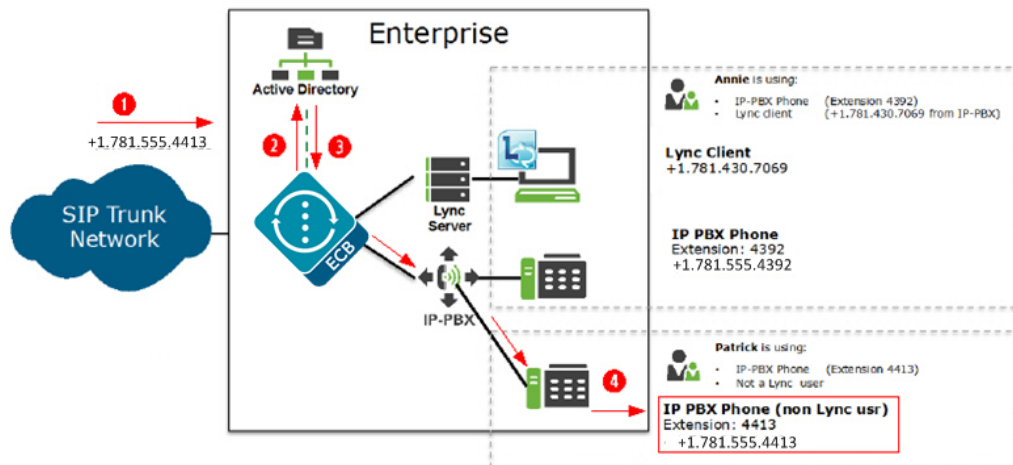
Match-only

When you set the LDAP **route-mode** attribute to match-only, the OECB performs as follows.

When the OECB receives an incoming call to the Enterprise network, LDAP queries the Active Directory to find the number that matches the incoming number exactly. If

the number is found, the OECB retrieves the entry's agent and builds a route list for the call. The following steps describe the match-only call flow in the following diagram.

1. A call comes into the Enterprise network (+1.781.555.4413).
2. Using the configured route-mode of match-only, LDAP queries the exact matching number in the Enterprise's Active Directory.
3. The Active Directory finds the matching number and includes that number's agent in the response to the LDAP query.
4. The OECB creates a route set for the call and forwards the call towards the destination phone number (same number as the number that initially called into the Enterprise in Step 1 (+1.781.555.4413)).



Attribute-order

When you set the LDAP **route-mode** attribute-order, the OECB performs as follows.

The order in which you configure the LDAP attributes on the OECB determines the priority of each route. If an incoming call is destined for the IP-PBX, but the attribute name for a Lync client is configured first, the OECB uses the corresponding agent (Lync Server) to create the first route in the route list.

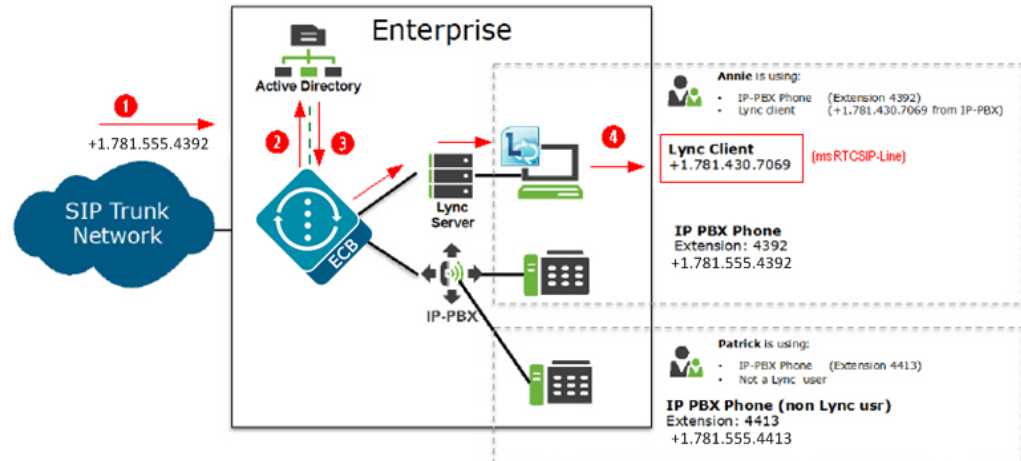
An entry in an LDAP search response must have at least one attribute that it matches in the Active Directory.

For example, suppose the incoming phone number is +1.781.555.4392 (which matches the IP-PBX phone number), and the attribute name msRTCSIP-Line (Lync attribute) in the response is +1.781.430.7069 (Lync phone number). The OECB creates a route for the Lync phone number, even though the incoming phone number matches the IP-PBX phone number because the msRTCSIP-Line attribute was configured first. The OECB forwards the call to the Lync destination.

If an Enterprise uses the same phone number for both Lync and IP-PBX phones, and the attribute-name msRTCSIP-Line is configured first (a Lync attribute), the OECB uses the corresponding agent (Lync Server) to create the first route in the route list. The following steps describe the attribute-order call flow in the following diagram.

1. A call comes into the Enterprise network (+1.781.555.4392).
2. Using the configured route-mode of attribute-order, LDAP queries the Active Directory for the agent of the matching number.

3. The Active Directory responds with the agent associated with the first configured LDAP attribute (+1.781.430.7069). In the diagram, the number is associated with a Lync Client (msRTCSIP-Line) that was configured first in the LDAP configuration.
4. The OECB forwards the call to the applicable destination phone number's agent from the Active Directory response. (+1.781.430.7069).



When you configure the attribute name msRTCSIP-Line first, the Oracle Enterprise Communications Broker uses the corresponding next hop (Lync Server) to create the second highest priority route in the route list. For example, the dialed telephone number could be +1.781.555.4392 (IP-PBX phone number), and the attribute-name msRTCSIP-Line in the response could be +1.781.430.7069 (Lync phone number). A route is created for the Lync phone number, even though the dialed telephone number is the PBX phone number.

Match-first

When you set the LDAP **route-mode** attribute to match-first, the OECB performs as follows.

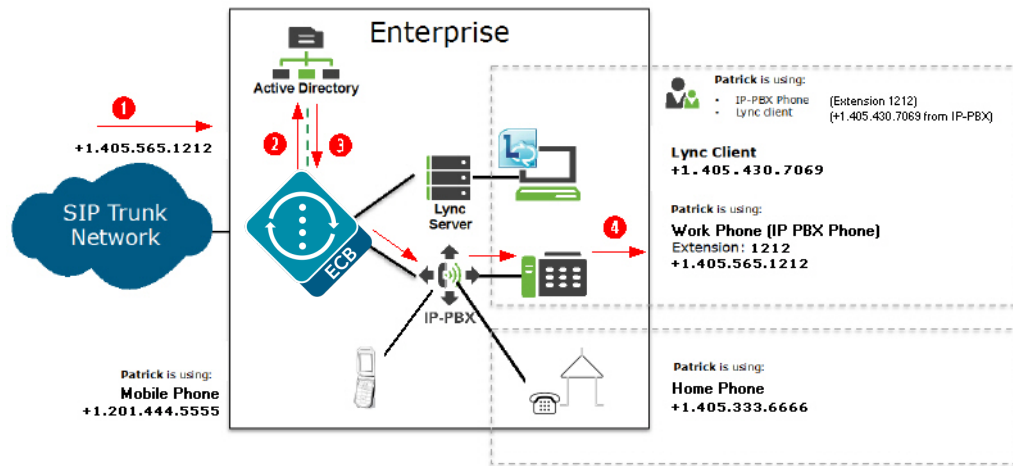
When the LDAP query is sent to the Active Directory, the first exact match of the incoming phone number that the LDAP query finds in the Directory is the number whose corresponding route gets the highest priority in the route list. For all other routes configured on the OECB, the ordering of LDAP attributes in the LDAP configuration determines the priority for each route.

For example, if the incoming number is +1.405.565.1212, and the Active Directory includes a configured mobile number first (+1.201.444.5555), a home number second (+1.405.333.6666), and a work number third (+1.405.565.1212), the LDAP query searches the mobile number first, then the home number, then finds the exact match on the work phone number. The Active Directory responds with the agent information for the work phone number and the OECB creates a route list with this exact phone number and forwards the call accordingly. The following steps describe the match-first call flow in the following diagram.

1. A call comes into the Enterprise network (+1.405.565.1212).
2. Using the configured route-mode of match-first, LDAP queries the Active Directory for the agent of the matching number.
3. The LDAP query searches throughout the Active Directory until it finds the first exact match on the number. Active Directory responds with the exact phone

number associated with the incoming number (+1.405.565.1212). In the diagram, the number was associated with the work phone.

4. The OECB forwards the call to the agent of the applicable destination phone number from the Active Directory response. (+1.405.565.1212).



LDAP Messages

If LDAP message logging is enabled in the Active Directory, the Oracle Enterprise Communications Broker sends LDAP messages to a message log called sipddap.log. This log records all received and sent LDAP messages. Messages are in ASCII encoded binary format.

Additionally, when LDAP is invoked for routing, the LDAP messages display in the GUI under the Monitor and Trace tab.

Note:

The Oracle Enterprise Communications Broker also supports transmitting LDAP messages using the IPFIX Protocol for the Palladion Mediation Engine.

LDAP Failure Events

If an incoming registration to a primary phone number in Lync fails, the phone number is routed to the IP PBX. If failures occur during LDAP queries for all LDAP Servers, the Oracle Enterprise Communications Broker logs the failure to the sipddap.log, and proceeds with normal configured routing policies, if available.

Note:

The Oracle Enterprise Communications Broker always establishes the TCP/TLS connection towards the configured LDAP server(s). If a TCP connection fails, the Oracle Enterprise Communications Broker continues to attempt to re-establish the connection.

An LDAP connection failure can be due to any one of the following events:

- Oracle Enterprise Communications Broker receives a CANCEL message (LDAP connection termination). The Oracle Enterprise Communications Broker detects this if it receives or issues an 'unbind' operation. The session is then closed down at TCP/TLS.
- Oracle Enterprise Communications Broker receives a call failure message from Lync (TCP/TLS socket termination). If either side receives a finish message (FIN) or reset message (RST), the TCP socket closes per standard behavior, which triggers the LDAP layer to detect connection failure. The Oracle Enterprise Communications Broker fails over to a secondary LDAP Server, if configured; otherwise it periodically attempts to reconnect to the Primary LDAP Server.
- Oracle Enterprise Communications Broker is unreachable and SIP session towards Lync times out. User is enabled for Lync but the Lync Server is unreachable by the Oracle Enterprise Communications Broker so a timeout occurs. When consecutive LDAP queries timeout, the Oracle Enterprise Communications Broker concludes that the LDAP session has failed, and then proceeds to terminate the TCP/TLS connection.

The number of consecutive queries that timeout before a connection is considered failed, and the number of successive query timeouts for each LDAP Server can be set via configuration.

Oracle ECB Limitations using LDAP

The Oracle Enterprise Communications Broker uses LDAP in the Active Directory when determining the destination of incoming calls. However, the Oracle Enterprise Communications Broker has the following limitations when using LDAP:

- Supports LDAP sessions over the Oracle Enterprise Communications Broker media interfaces only (i.e., not on wancom0).
- Supports LDAPv3 only.
- Establishes a session over the following connections only:
 - LDAP over TCP - default
 - LDAP over TLS (LDAPS)

Configuring LDAP for Routing

LDAP is the Protocol that the Active Directory uses for general interaction between and LDAP client and an LDAP server. You can configure the LDAP Server(s) in your network, and set the filters and the local policy that the LDAP Server uses when handling inbound Lync and PBX calls in the Enterprise core network.

You can use the following objects in the Web GUI to configure LDAP:

- LDAP Config—Configures the LDAP functionality on the Oracle Enterprise Communications Broker (i.e., name, state, LDAP servers, realm, authentication mode, username, password, LDAP search filters, timeout limits, request timeouts, TCP keepalive, LDAP security type, LDAP TLS profile, and LDAP transactions).
- Routing—Configures Active Directory attribute names, attribute format and regex extractions for routing SIP requests to the target's home agent. You configure this object for LDAP search queries in the Active Directory.

- Address of Record—Configures Active Directory attribute names, attribute format and regex extractions for identifying other addresses of record for the request URI and from. The Oracle Enterprise Communications Broker uses any AoR information provided by these queries to generate additional routes for the session, using the same process it used for the original request URI and from.

3

Registrar and Authentication

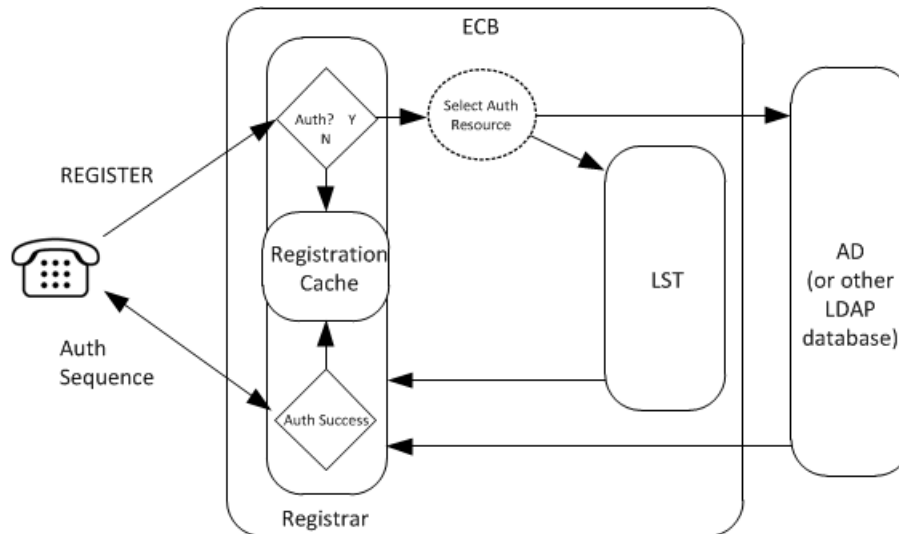
By providing a location service from within it, the Oracle Enterprise Communications Broker (OECB) offloads related infrastructure from providing that information for every session. The OECB can use SIP digest authentication to confirm service authorization and verify user registrations by way of internal or external mechanisms. If using an external mechanism for this purpose, some adaptation of that mechanism is required. You enable the registrar, configure the applicable domains (including serviced and digest domains) and, if required, define the authentication to use for all registrations by way of configuration on the GUI.

You enable the OECB single registry service globally. When registration functionality is enabled, the OECB registers endpoints rather than only caching and forwarding registrations to another device.

On receiving a REGISTER message, the OECB checks if it is responsible for the domain contained in the Request-URI, as configured in the domains list. The OECB begins registrar functions for all requests that match a configured domain.

When there is no authentication configured, the system adds every user that attempts to register to the registration cache. When you configure authentication, the system can authorize and verify the caller by way of the LST or an external LDAP resource. In these use cases, the system uses SIP digest to authenticate the caller, based on authentication information from the LST or LDAP. The following diagram and steps explain a call flow with authentication and interaction with LDAP resources, especially Active Directory.

1. A UA is fully registered after the system installs it in the registration cache.
2. The OECB sends a 200 OK message back to the registering UA.
3. When a user registers with the registrar, the system looks for the To header AoR in the LST. If the LST contains a subscriber with the AoR (or username if no AoR specified) that matches, the system adds the universal number of the subscriber as an alias to the registration cache.



Register Refresh

When a UA sends a register refresh, the Oracle Enterprise Communications Broker first confirms that the authentication exists for that UE's registration cache entry, and then is valid for the REGISTER refresh. (If a valid hash does not exist for that AoR, then the Oracle Enterprise Communications Broker sends a request to its source database (LST or LDAP) to retrieve authentication data once again).

Next, the Oracle Enterprise Communications Broker determines it can perform a local REGISTER refresh or if the source database needs to be updated. If any of the following 3 conditions exists for the re-registering UA, the Oracle Enterprise Communications Broker updates the database:

- The location update interval timer has expired—This value, configured in the sip registrar configuration element ensures that source database always has the correct Oracle Enterprise Communications Broker address by periodically sending request messages for each registered contact.
- The message's call-id changes while the **forward-reg-callid-change** option in the sip config configuration element is set. This covers the case where the UA changes the Oracle Enterprise Communications Brokers through which it attaches to the network.
- The REGISTER message's Cseq has skipped a number. This covers the case in which a user registered with Oracle Enterprise Communications Broker1, moves to Oracle Enterprise Communications Broker2, and then returns to Oracle Enterprise Communications Broker1.

If the Oracle Enterprise Communications Broker updates the source database because of matching one of the above conditions, the access side expiration timer per contact is reset to the REGISTER message's Expires: header value, and returned in the 200 OK. This happens even in the case when the reREGISTER was received in the first half of the previous Expires period. In addition, the core-side location update interval timer are refreshed on both active and standby.

When the above three conditions are not met, the registration expiration proceeds normally.

If the timer has not exceeded half of its lifetime, a 200 OK is returned to the UA. If the timer has exceeded half of its lifetime, the Oracle Enterprise Communications Broker just refreshes the access-side expiration timer; the registration cache expiration timer for that AoR begins its count again.

Proxy Registrations

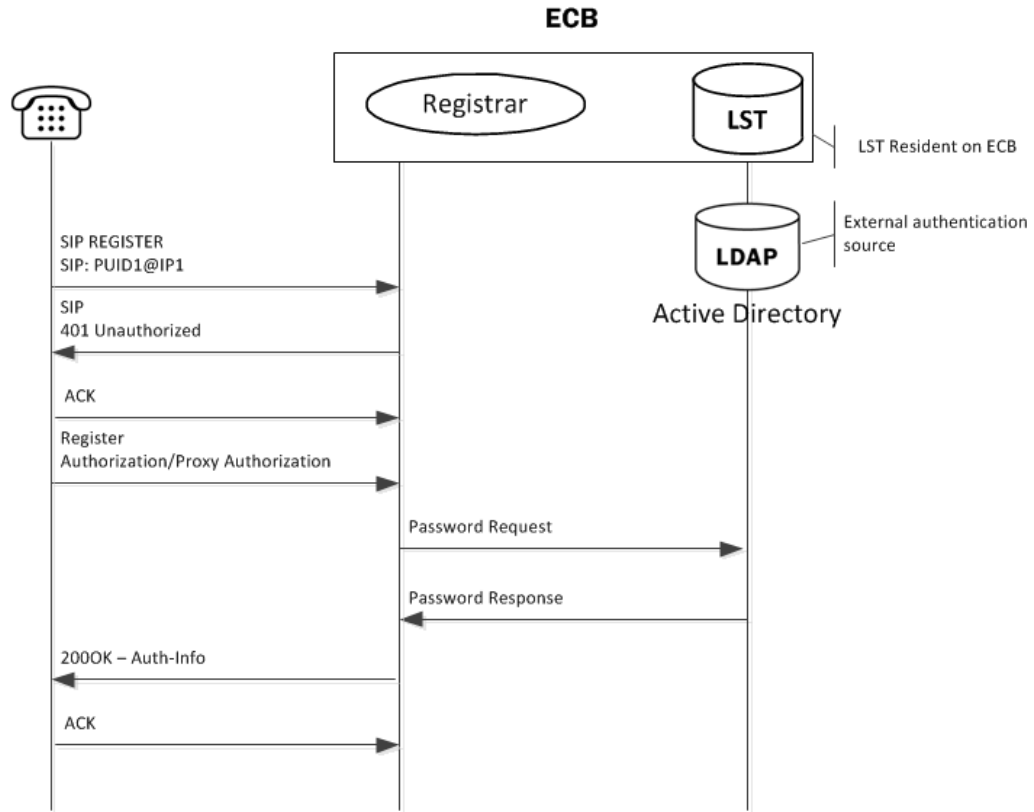
By default, the Oracle Enterprise Communications Broker (OECB) rejects a REGISTER request from a domain for which it is not the registrar. You can enable the OECB to proxy such registration requests by way of the **Proxy Registrations** control in the **SIP Interface** configuration.

In the **SIP Interface** configuration, select **Proxy Registrations** to tell the OECB to proxy the registration towards the intended registrar. When you deselect **Proxy Registrations**, the OECB responds with a **403: Unauthorized** message.

Message Authentication for SIP Requests

You can configure the Oracle Enterprise Communications Broker (OECB) to authenticate REGISTER requests. The OECB offers a single Registrar for location services on user-specified listed domains. Registration may or may not include user authentication. If includes authentication, you can select a local, text-based resource called the Local Subscriber Table (LST) as an authentication source. You can also configure the OECB as an LDAP client, allowing it to perform LDAP-compliant processes and retrieve authentication information from an external resource, usually Active Directory. The OECB populates the registration cache with contacts for AORs upon successful authorization/authentication.

The OECB uses SIP digest authentication as a means of challenging an end point for applicable registration attempts. The following diagram shows the overall authentication and authorization sequence, including the OECB confirming the registration by way of an LST or an external LDAP server.



Authentication

To authenticate the registering user, the Oracle Enterprise Communications Broker needs the hash of the end station's password. It requests these from the local LST of an LDAP server by sending it an LDAP query for the configured field.

The hash consists of an MD5 hash made up of the following components:

```
MD5(username:digest-realm:password)
```

The transaction is conducted with the server defined in the Registrar configuration's credential retrieval method parameter. This parameter is populated with the name of the LDAP sever.

SIP Authentication Challenge

When the Oracle Enterprise Communications Broker receives a response from the HSS including the hash value for the user, it sends a SIP authentication challenge to the endpoint, if the endpoint did not provide any authentication headers in its initial contact the with Oracle Enterprise Communications Broker. If the endpoint is registering, the Oracle Enterprise Communications Broker replies with a 401 Unauthorized message with the following WWW-Authenticate header:

```
WWW-Authenticate: Digest realm="atlanta.com", domain="sip:boxesbybob.com",
qop="auth", nonce="f84f1cec41e6cbe5aea9c8e88d359", opaque="", stale=FALSE,
algorithm=MD5
```

Authentication Header Elements

- Domain—A quoted, space-separated list of URIs that defines the protection space. This is an optional parameter for the "WWW-Authenticate" header.
- Nonce—A unique string generated each time a 401/407 response is sent.
- Qop—A mandatory parameter that is populated with a value of "auth" indicating authentication.
- Opaque—A string of data, specified by the Oracle Enterprise Communications Broker which should be returned by the client unchanged in the Authorization header of subsequent requests with URIs in the same protection space.
- Stale—A flag indicating that the previous request from the client was rejected because the nonce value was stale. This is set to true by the SD when it receives an invalid nonce but a valid digest for that nonce.
- Algorithm—The Oracle Enterprise Communications Broker always sends a value of "MD5"

SIP Authentication Response

After receiving the 401/407 message from the Oracle Enterprise Communications Broker, the UA resubmits its original request with an Authorization: header including its own internally generated MD5 hash.

Authentication Check

At this point, the Oracle Enterprise Communications Broker has received an MD5 hash from the HSS and an MD5 hash from the UA. The Oracle Enterprise Communications Broker compares the two values and if they are identical, the endpoint is successfully authenticated. Failure to match the two hash values results in a 403 or 503 sent to the authenticating endpoint.

Retrieving Information from Active Directory

The Oracle Enterprise Communications Broker performs SIP Digest authentication against users attempting to register. It can use pre-configured information from Active Directory to perform such authentication. Access to Active Directory uses standard LDAP processes to retrieve the information needed and to offload the processing from other resources to the Oracle Enterprise Communications Broker.

The Oracle Enterprise Communications Broker can obtain registration authentication information directly from Active Directory when you modify the Active Directory schema to include the Oracle-specific attributes and object classes that the Oracle Enterprise Communications Broker needs to authenticate users..

The Oracle Enterprise Communications Broker operates by issuing LDAP requests from Active Directory for data from "password" attributes, using Active Directory's standard **sAMAccountName** to match the Request URI username to create new attributes in Active Directory. One of these attributes must be populated with the digest realm. A Dynamic Link Library (DLL) installed on Active Directory intercepts the password change hashes and writes them to another attribute. The DLL then creates a hash of the username, digest realm, and password hash to be returned to the Oracle

Enterprise Communications Broker within the LDAP response. The Oracle Enterprise Communications Broker extracts the password hash, compares it to the hash provided by way of SIP digest, authenticates, and registers the user when there is a match.

- `orclDigestRealmAttribute`—Populated with digest realm.
- `orclDigestPwdAttribute`—Populated with hash of Active Directory password during each password change.
- `orclAgentNameAttribute`—Populated with user's agent for the purpose of routing. See Active Directory and Oracle ECB Routing in this document to understand how the Oracle Enterprise Communications Broker uses this attribute.

Oracle can provide the `oidpwdcn.dll`, scripts to create the needed attributes, scripts to populate the digest realm attribute, and a `README.TXT` with instructions on how to perform all procedures. Appendix C provides instruction on getting this methodology operational.

LDAP and Oracle ECB Authentication

Lightweight Directory Access Protocol (LDAP) is the Protocol that the Oracle Enterprise Communications Broker uses to perform queries to the Enterprise's Active Directory to validate registration attempts in the Enterprise network. Requests and responses are sent/received based on the Oracle Enterprise Communications Broker's LDAP configuration. The Oracle Enterprise Communications Broker's LDAP client queries an LDAP server, usually Active Directory for password information for a user attempting to register. This request and response process verifies that the user can get registration servers (authorization) and verifies that the user is who they say they are (authentication). Once both these stages complete successfully, the Oracle Enterprise Communications Broker registers the user.

The Oracle Enterprise Communications Broker, using LDAP, performs the following on a registration attempt:

- Creates an LDAP search filter based on the dialed number and the configured LDAP attributes.
- Sends an LDAP search query to the configured LDAP server.

You configure LDAP servers and filters, on the Oracle Enterprise Communications Broker.

The Oracle Enterprise Communications Broker keeps a permanent LDAP session open to all configured call servers. It sends an LDAP bind request on all established connections, to those servers. The first call server is considered the primary LDAP server, and all others are secondary LDAP servers. If a query request sent to the primary server fails, the Oracle Enterprise Communications Broker sends the request to the next configured LDAP server, until the request is successful in getting a response. If no response is received by the Oracle Enterprise Communications Broker, it replies to the registering endpoint with a (401? authentication failure?).

Configuring LDAP for Authentication

LDAP is the protocol that the Active Directory uses for general interaction between and LDAP client and an LDAP server. You can configure the LDAP server(s) in your network, and set the filters and the local policy that the LDAP server uses when handling inbound Lync and PBX calls in the Enterprise core network.

You can use the following objects in the Web GUI to configure LDAP:

- LDAP Config—Configures the LDAP functionality on the Oracle Enterprise Communications Broker (i.e., name, state, LDAP servers, realm, authentication mode, username, password, LDAP search filters, timeout limits, request timeouts, TCP keepalive, LDAP security type, LDAP TLS profile, and LDAP transactions).
- SIP Authentication—Configures the Active Directory attribute names for the Oracle Enterprise Communications Broker's query-digest-username-attribute and digest-hash-attribute fields. These fields specify where the Oracle Enterprise Communications Broker verifies authentication attempts.

See the section on Active Directory and Oracle ECM Routing for important information about:

- LDAP messages
- LDAP failure events
- Oracle ECB limitations using LDAP

That information applies equally to the authentication functionality explained here.

4

Getting Started

Oracle® recommends that you review the topics in "Getting Started" before working with the system to ensure success with the tools and functions provided.

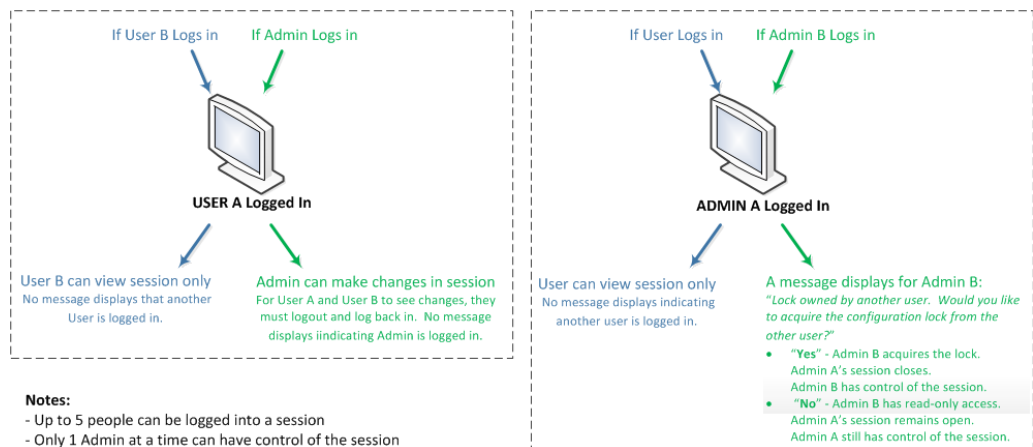
User and Administrator Access

You can log on to the Web GUI using your Web browser. There are two types of user logons:

- **User** - Allows viewing (read-only) access to the Web GUI.
- **Administrator** - Allows Superuser access to the Web GUI.

Simultaneous Logons

The Web GUI allows simultaneous logons for both the User and Administrator. Session availability to the User and Admin depends on which type of user is logged on to the session. The following illustrations depict and explain the system behavior when a User and an Administrator log on to a Web GUI session.



Up to five users can log onto the same session at the same IP address at the same time. Only one Administrator at a time can have full control of a simultaneous session. If more than five users attempt to log on, the system displays the following error message:

User limit reached. Please try again later.

RADIUS Server Roles and Access Privileges

The Web GUI supports RADIUS authentication functionality similar to a user logging on by way of Secure Shell (SSH) and SSH File Transfer Protocol (SFTP).

Available functions depend on the role that you assign to the "userclass" on the RADIUS server.

- When you configure the RADIUS server as userclass=admin, the system allows the Administrator full access to all features and functions after logging onto the GUI.
- When you configure the RADIUS server as userclass=user, the system limits User access to the following features and functions after logging onto the GUI.
 - Full access to all System features and functions
 - Can download the following files in System File Management:
 - * Backup Configuration
 - * Configuration CSV
 - * Fraud Protection Table
 - * Log
 - * Audit Log
 - * Playback Media
 - * Software image
 - * SPL Plug-in (SPL)

 **Note:**

The "User" account cannot upload files in System File Management.

Log On to the Web GUI

You can log on to the Oracle Enterprise Communications Broker (OECB) as a User or as an Administrator, depending on your permissions.

You need your User name, Password, and optionally your Passcode to log in. The system requires your passcode when two factor authentication is enabled. If your system Administrator configured the optional log on page message, the system displays the message after you enter your log on credentials. After reading the message, click **Close**, and the system displays the GUI.

1. On a PC, open a supported Internet browser. See "Browser Support."
2. Start the GUI with either the HTTP or HTTPS log on.

```
http://<Server IP address>  
https://<Server IP address>
```

 **Note:**

Whether you log on using HTTP or HTTPS depends on the settings for your deployment.

3. In the Welcome dialog, enter your Web GUI username and password.

Welcome to Enterprise Session Border Controller

* Username

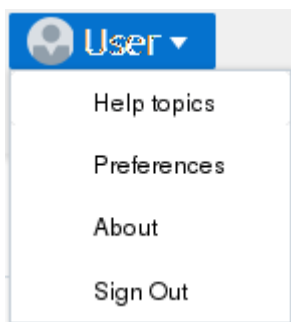
* Password

Sign In

4. If your OECB requires two-factor authentication, do the following:
 - a. In the **Login** dialog, enter your passcode.
 - b. In the **Confirm** dialog, acknowledge the Last Login information. Yes—allows you to login. No—ends the session.
5. Click Login.

Log Off from the Web GUI

To log off from the Web GUI, click **Sign Out** from the user menu in the upper right corner of the Web GUI.



The system logs you off and re-displays the log on page.

Service Provisioning

After the Oracle Enterprise Communications Broker is operational, network architects and communications service provisioning technicians specify call services using the controls available from the Service Provisioning icons available from the Configuration tab.

System Administration controls, also available from the Configuration tab, specify how to manage the system and are documented in the *Oracle Enterprise Communications Broker Administrator's Guide*.

Service Provisioning Configuration Objects

The following information describes the Oracle Enterprise Communications Broker (OECB) Service Provisioning objects.

Configuration Object	Description
Agents	<p>Add agents.</p> <p>An agent is usually a SIP-aware device that serves as a transit target and/or source for signaling managed by the OECB. Agents are often specified as next-hops for the purposes of routing.</p> <p>Indirect agents, OECB route termination points that require further routing to reach an end station are also configured here.</p> <p>In addition, configuration used to access ENUM servers is performed here.</p>
Dial Plan	<p>Add multiple dialing-contexts and dial-patterns.</p> <p>Dialing-contexts define the system behavior for calls placed to and from either a corporate or geographic focus.</p> <p>Dialing-contexts include multiple dial-patterns, which define the normalization required to most effectively manage diverse signaling structures.</p>
Policy Entries	<p>Add policies and define the conditions of their use.</p>
Routing Table	<p>Add service routes.</p> <p>Route-entries specify paths for signaling traffic, allowing you to specify policy and cost for traffic based on source and/or destination.</p>
User Entries	<p>Users - Add user and other key phone numbers associated with the enterprise. The user database serves as a directory for phone numbers that need communications services.</p> <p>This database can specify each entry's source context, which can provide a starting point for processing the logic behind a user's call treatment. It also can specify each user's agent, providing a physical location for routing user's calls.</p>

Web GUI Tools

The Web GUI provides various tools to conduct operations. Some tools apply globally, while others apply only to a specific function on the tab. For example, the "Tree View" and "List View" controls on the Preferences menu apply to all pages and "Add a Widget" applies only to the Home page.

The User Menu

Oracle Enterprise Communications Broker displays the User menu in the upper right-hand corner of the branding bar, and is labeled with the currently logged in user's name.

The user menu provides the following links.

- Help Topics—Links to the embedded Help system.
- Preferences—Provides the choices for how you want to see the configuration objects displayed.
- About—Displays information about the software, for example, the release and build numbers.
- Sign Out—Logs you off of the Web GUI.

Help

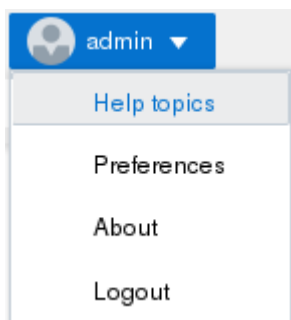
The logged on user button on the Web GUI displays the following information:

- **Screen Help.** Short descriptions of elements on the page.
- **Help Topics.** Online Help system containing topics about the tasks that you can perform on the Web GUI.
- **About.** Oracle notices and disclaimers, Oracle terms and restrictions, and third-party notices.

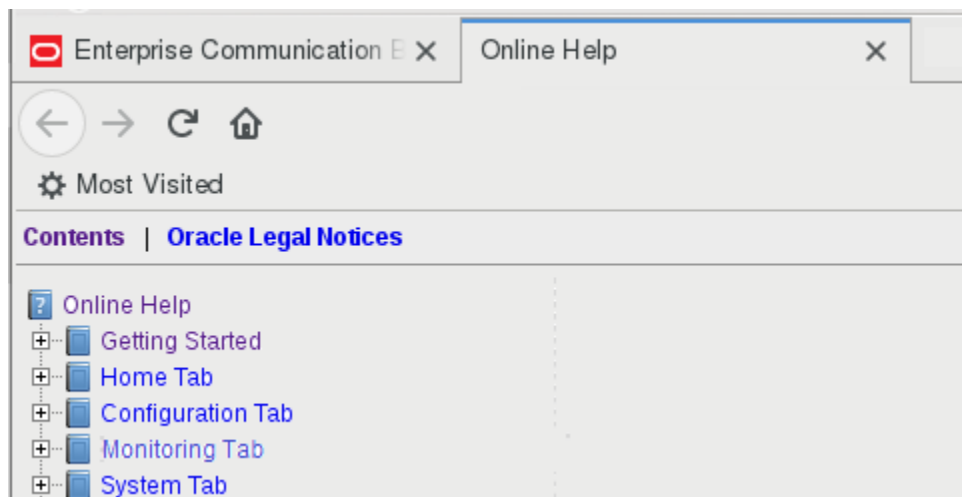
Help Topics

The Oracle Enterprise Communications Broker (OECB) includes an online Help system that provides the concepts and procedures that you need to know for working with the system.

From any page in the GUI, click **admin, Help topics** in the upper right corner of the page.



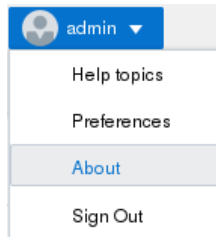
After you click **Help topics**, the system displays the Online Help Table of Contents.



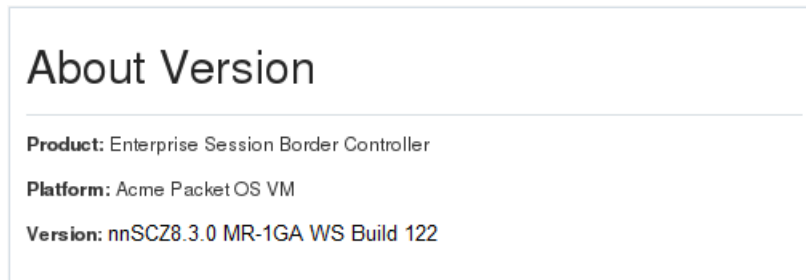
About This Software

You can display information about the software currently installed on the Oracle Enterprise Communications Broker (OECB) that you are logged on to by clicking **About** on the **User** menu located in the upper right corner of the Web GUI.

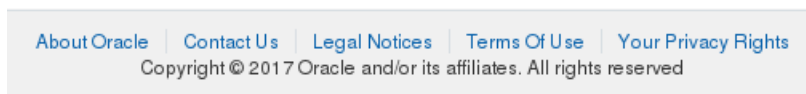
The following screen capture shows the location of **About** on the User menu.



The following screen capture shows an example of the information that the About link displays.



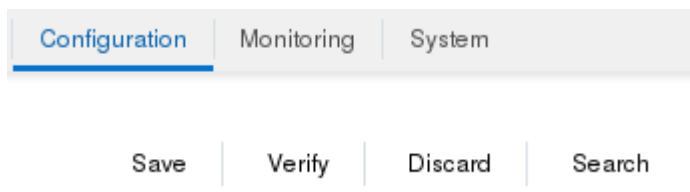
The About page also displays the following links to more information about Oracle and the OECB.




The Search Tool

The Oracle Enterprise Communications Broker provides search operations for both system-wide and a configuration-specific objects.

The global Search button on the branding bar opens the Search text box where you enter the text to search for, such as an object name, an attribute, or value.




Search

Search 

The resulting display includes links to the configuration object, for example:

Search

Search 

Search results for "snmp-address"

[esm trap > trap_1.1.1.1](#)
Type: object
Attribute: IP address
Attribute value: 1.1.1.1

[esm trap > trap_5.5.5.5](#)
Type: object
Attribute: IP address
Attribute value: 5.5.5.5

[esm trap > trap_6.6.6.6](#)
Type: object
Attribute: IP address
Attribute value: 6.6.6.6

Multi-instance configuration objects display a table of configurations and the table includes a text box for local search within the table.

User Entries

Search Criteria: All

Add Delete All Upload Download Search

AoR	Number Or Pattern	Description	Dialing Context	Tags	Learned From
agnes@service...					local
andy@IPservice...					local
bill@oracle.com			nnnn		local
dana@servicep...					local
deb@servicepr...					local
fonzi@servicepr...					local
fred@IPservice...					local
hawkeye@servi...					local

The result of the search is displayed in the table. For example, searching for "henry" in the User Entries list above results in the display of henry's record, which you can edit, copy, or delete.

User Entries

Search Criteria: Search = henry; Sync = Both

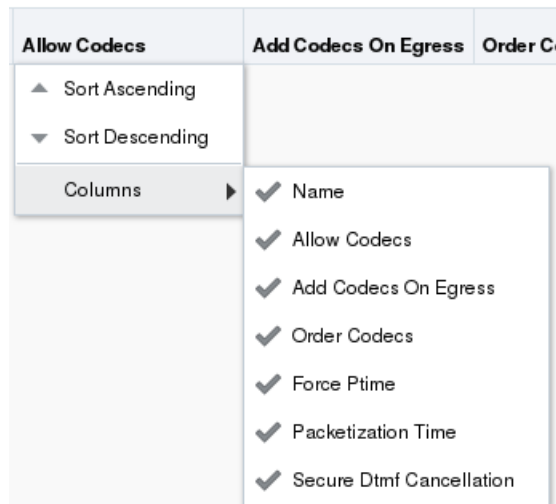
Add Delete All Upload Download Search

AoR	Number Or Pattern	Description	Dialing Context	Tags	Learned From
⚙️ henry@service...					local

Customize the Page Display

You can customize the display of the data on Web GUI pages by selecting which columns display, the information type, and the sort order. You can also re-order the column headings.

1. Place the cursor on a column heading.
The system displays a down arrow in the column heading.
2. Click the down arrow to display the customization menus. For example,



3. Select or deselect items, as needed.
4. Re-order the column headings by placing the cursor on a column heading and sliding it left or right into the location you want.

Tool Tips

A tool tip is a brief description of a parameter on a configuration screen in the Web GUI, that also includes the default setting and valid values for the corresponding parameter.

To view a tool tip, place the cursor in the parameter field.

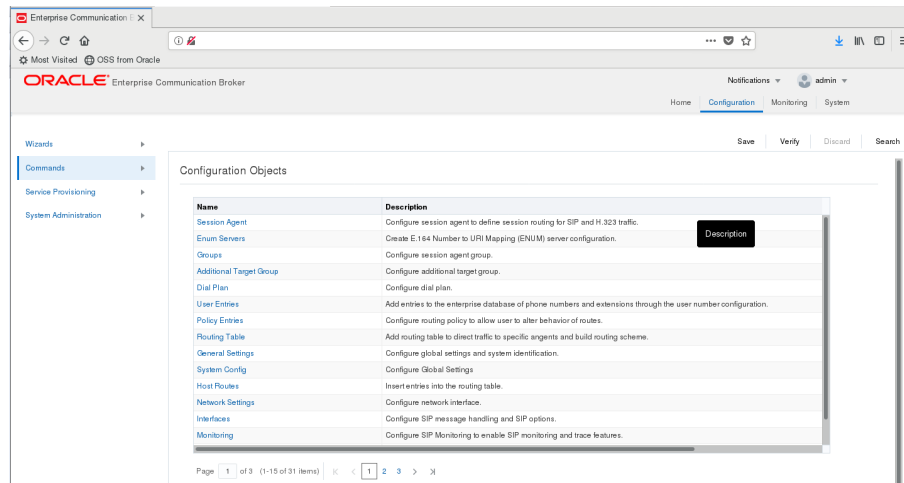
To close the tool tip, click away from the parameter.

Configuration Tools and Behavior

The Oracle Enterprise Communications Broker (OECB) Web GUI provides the following tools for working with configurations. Some tools are located in the navigation pane and others are located at the top of the center pane.

The Configuration Tab Display

The following screen capture shows the locations of all of the Configuration tab controls.



Controls in the Navigation Pane

The navigation pane displays links to Configuration Wizards and Show Commands. (Descriptions are listed in the center pane.) The Wizards help you specify certain parts of the configuration and the Commands show you information about the configuration on the system.

Wizards Objects

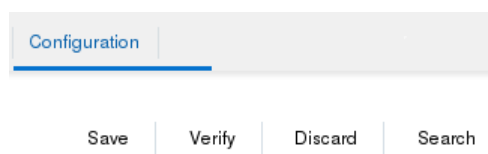
Name	Description
Set Boot Parameters	Set boot parameters for the system.
Set Login Banner	Set login banner for user.
Set Time Zone	Set time zone.
Upgrade Software	Software upgrade.
Set Initial Configuration	Set initial configuration for the system.
Set License	Set License.
Set Entitlements	Set advance entitlements.

Commands Objects

Name	Description
Show Editing Configuration Short	Editing configuration short
Show Running Configuration Short	Show running config short
Show Configuration Version	Configuration version number table

Controls in the Center Pane

The controls located at the top of the center pane on a configuration page help you manage configuration objects.



Save	Use to save the current configuration session. Upon Save, the system displays a prompt giving you a choice of whether or not to activate the configuration. If you do not activate the configuration, you can continue to make changes and Save again. When finished, you can save and activate all of the configuration changes.
Verify	Use to confirm that the configuration is valid before you save it.
Discard	Use to undo all configuration changes made in the current session. The system can only discard the changes that you saved. It cannot discard any changes that you activated.
Search	Use to find and display the current settings for the configuration you are editing. Enter the name of the configuration object.

Controls for Multi-Instance Objects

The controls located at the top of a list of multi-instance objects help you manage the objects on the list. The following screen capture shows the controls.



Add	Use to add another instance of the configuration object with one or more parameters set differently.
Delete All	Use to delete all instances.
	<div style="border-left: 2px solid orange; padding-left: 10px; background-color: #fff9c4;"> <p>⚠ Caution:</p> <p>You cannot select a sub-set of instances and delete only those. The system deletes all instances, regardless of how many you select.</p> </div>
Upload	Use to upload a configuration file in CSV format.
Download	Use to download a configuration file in CSV format.
Search	Use to search the list of multi-instances for a specific instance. When you enter the name of an instance, the Web GUI displays it at the top of the list.

Configuration Wizards

The Wizards control in the navigation pane displays a list of Wizards, for performing selected configuration procedures for the Oracle Enterprise Communications Broker (OECB).

The Wizards help you perform the following tasks:

Set Boot Parameters	Specify the boot file and the boot parameters.
---------------------	--

Set Entitlements	Set the number of sessions that a license entitles you to, and enable advanced features.
Set Initial Configuration	Configure a new OECB or reconfigure an existing one. Includes configuring High Availability.
Set License	Enter the license number for a feature that requires a license.
Set Logon Banner	Customize the text on the Web GUI log on banner.
Set Time Zone	Select the time zone for the deployment.
Upgrade Software	Upload a newer version of the software.

Set Boot Parameters Wizard

The Oracle Enterprise Communications Broker (OECB) requires you to enter the necessary parameters to boot the system in your deployment.

You can set the OECB boot parameters from the Set Boot Parameters Wizard on the Web GUI in either Basic mode or Expert mode.

1. Access the Set Boot Parameters Wizard: **Configuration, Wizards, Set Boot Parameters**.
2. In the Set Boot Parameters dialog, enter the following information:

Boot File	Name of the image file.
IP Address	Enter the IP address of the OECB.
VLAN	Range: 0-4095
Net Mask	Enter the net mask IP address in dot decimal format. For example, 255.255.0.0.
Gateway	Internet address of the boot host. Leave blank if the host is on the same network.
IPv6 Address	Enter the IPv6 address that you want to use.
IPv6 Gateway	Enter the IPv6 gateway that you want to use.
FTP Host IP	Enter the IP address of the FTP host.
FTP Username	Enter the FTP username for the FTP user on the boot host.
FTP Password	Enter the FTP password for the FTP user on the boot host.
Flags	Hexadecimal. Always starts with 0x. See "Configurable Boot Loader Flags."
Target Name	Name of the OECB, as displayed at the system prompt.
Console Device	Enter the type of console device. For example, VGA.
Console Baud Rate	Select a console baud rate from the drop-down list.
Other	For miscellaneous and deployment-specific boot settings.

3. Click **Complete**.

The system displays a success message.

4. Click **OK**.

Configurable Boot Loader Flags

You may configure the following boot flags in the boot loader:

- 0x04 - disables autoboot timeout (ap3820 and ap4500 only)
- 0x08 - extend autoboot countdown timer to 15 seconds
- 0x40 - use DHCP for wancom0 (VM Edition only)
- 0x80 - network boot using TFTP instead of FTP

Set Entitlements Wizard

Use the Set Entitlements Wizard to enter the maximum number of sessions that your license allows.

- Note the session limit number from your license.

You can launch the Set Entitlements Wizard on the Web GUI in either Basic mode or Expert mode.

1. Access the Set Entitlements Wizard: **Configuration, Wizards, Set Entitlements**.
2. In the Set Entitlements dialog, do the following:

Advanced	Select to add the Advanced license.
Admin Security	Select to enable your Admin Security license.
Data Integrity (FIPS 140 2)	Select to enable FIPS 140 2 data integrity.
Transcode Codec AMR	Select to enable AMR transcoding.
Transcode Codec AMRWB	Select to enable AMRWB transcoding.
Transcode Codec EVS	Select to enable EVS transcoding.
Session Capacity	Set the session limit number from the license. Valid values: 0-512000
Transcode Codec AMR Capacity	Set the transcoding capacity for AMR. Valid values: 0-102375.
Transcode Codec AMRWB Capacity	Set the transcoding capacity for AMRWB. Valid values: 0-102375.
Transcode Codec EVS Capacity	Set the transcoding capacity for EVS. Valid values: 0-102375.
Transcode Codec Opus Capacity	Set the transcoding capacity for Opus. Valid values: 0-102375.
Transcode Codec SILK Capacity	Set the transcoding capacity for SILK. Valid values: 0-102375.

3. Click **Complete**.

The system displays a success message.

4. Click **OK**.

Set Initial Configuration Wizard

Use the Set Initial Configuration wizard to perform the initial configuration on an unconfigured system and to change the configuration on a configured system. During the configuration, you select the scope of configuration that you want to perform, define the boot parameters, opt to set a VLAN, and configure features such as High Availability (HA) and access to the Oracle Communications Session Delivery Manager (OC SDM). A valid license is required to run the Set Initial Configuration wizard.

Launch the Set Initial Configuration Wizard

- Unconfigured system. The system launches the Web GUI Set Initial Configuration wizard upon the first logon. When the initial configuration is complete, the system saves the configuration, activates the configuration, and reboots. The system does not backup the initial configuration of an unconfigured system.
- Configured system. From the Configuration tab on the Web GUI, click the Wizards button and click Set Initial Configuration. When the re-configuration is complete, the system saves a backup of the existing configuration, saves the new configuration, activates the new configuration, and reboots. The backup is stored in /code/bkups.

Before you can configure the OECB, the wizard requires you to make the following selections that determine which configuration parameters the wizard displays.

Enable Web GUI: Yes or No	If you select No , you may continue using the Wizard to set the initial configuration until you reboot. After you reboot, the system no longer displays the Web GUI. If you want to enable the Web GUI in the future, configure the Web Server Config object from the ACLI.
Choose Web GUI Mode: Basic or Expert	<p>When selecting Basic mode or Expert mode, the decision is about how much control you want in the configuration process and whether or not you want to use one of more of the advanced features and settings provided in Expert mode.</p> <ul style="list-style-type: none"> • In Basic Mode, the system displays the minimum number of settings that you need to successfully deploy and operate the OECB. While you cannot configure the advanced settings and features in Basic mode, you can switch to Expert mode to do so. Note that when you switch to Expert Mode and perform Save, you cannot switch back to Basic Mode. The Web GUI will display the Expert Mode from then on, including after a new log on. • In Expert Mode, you can use the advanced settings and options to control the configuration with more granularity. The system does not require you to configure all advanced settings and features. You can choose what you need for your deployment.
OECB Mode: Standalone or High Availability	<ul style="list-style-type: none"> • If you select Standalone, you can begin configuring the parameters displayed. • If you select High Availability, the GUI adds OECB Role: Primary/Secondary to the display.

OECB Role: Primary or Secondary	If you selected High Availability for OECB Mode: <ol style="list-style-type: none">1. Select Primary, and configure the displayed parameters.2. Select Secondary, and select Yes or No for Acquire Configuration from Primary. If you select No, the GUI adds a field where you enter the Peer Target Name.
---------------------------------------	---

 **Note:**

Unlike other OECBs, which provide 2 management interfaces and 2 media interfaces, the Acme Packet 1100 provides 1 management interface and 2 media interfaces. When configuring HA, the configuration dialogs for the Acme Packet 1100 differ from the other OECBs because you must create a second, virtual management interface. For creating the second management interface, the HA dialogs on the Acme Packet 1100 contain more attributes than the dialogs for the other OECBs. Regardless of the OECB model, the path through the Set Initial Configuration wizard to the HA dialogs is the same as described in this topic.

Configure the System

The system requires an initial configuration of attributes, such as modes and IP addresses, before it can function in the network.

Use the Set Initial Configuration Wizard to define the attributes for the system. The system displays the Set Initial Configuration Wizard upon the first logon.

1. Logon to the Oracle Enterprise Communications Broker.
The system displays the Set initial configuration wizard.
2. Run the Set Initial Configuration Wizard, and click **Complete**.
The system saves the configuration, activates the configuration, and re-boots.
 - Configure the system objects.

Reconfigure the System

You can reconfigure the system from the Web GUI.

Use the Set Initial Configuration Wizard to change the initial configuration on a configured system, for example, change attributes such as IP addresses and modes.

1. Log on to the system.
2. Access the Set Initial Configuration Wizard: **Configuration, Wizards, Set Initial Configuration**.
3. Run the Set Initial Configuration Wizard and change the attributes, as needed.
4. Click **Complete**.

The system saves a backup of the existing configuration, saves the new configuration, activates the new configuration, and automatically re-boots.

- (Optional) Reconfigure the system objects.

Set License Wizard

Use the Set License Wizard to enter the serial number for your license. You can use the Set License Wizard in Basic Mode and Expert Mode.

- Obtain the license, which includes the serial number, for the feature that you want to add to the deployment. See "Obtain a License" in the *ACLI Configuration Guide*.

You need the license number for the following procedure.

1. Access the Set License Wizard: **Configuration, Wizards, Set License**.
2. In the Set License dialog, enter the license serial number in the Add license field.
3. Click **Complete**.
The system displays a success message.
4. Click **OK**.

Set Login Banner Wizard

Use the Set Login Banner Wizard to add customized text to the log on page. You can use the Set Login Banner Wizard in Basic mode and Expert Mode.

You can customize the log on page by adding text to help the user. For example, Welcome to <company name> <business unit> <location> session border controller <device name>.

1. Access the Set Login Banner Wizard: **Configuration, Wizards, Set Login Banner**.
2. In the Set Login Banner dialog, enter the text that you want to display on the log on page.
3. Click **Complete**.
The system displays a success message.
4. Click **OK**.

Set Time Zone Wizard

The system requires a setting for time zone.

You can set the system time from the Set Time Zone Wizard on the Web GUI. You can select a time zone or Coordinated Universal Time (UTC). You can use the Set Time Zone Wizard in Basic Mode and Expert Mode.

1. Access the Set Time Zone Wizard: **Configuration, Wizards, Set Time Zone**.
2. From the drop down list, select one of the following:
 - Time zone by locale
 - UTC
3. Click **Complete**.
The system displays a success message.
4. Click **OK**.

Upgrade Software Wizard

You can upgrade the system software with the Upgrade Software Wizard on the Web GUI. You can use the Upgrade Software Wizard in Basic Mode and Expert Mode.

Use the Upgrade Software Wizard to perform the following tasks:

- Check the system health before the upgrade
- Download new software
- Change boot parameters
- Reboot the system

The system requires a reboot after the upgrade for the changes to take effect.

1. Access the Upgrade Software Wizard: **Wizards, Upgrade Software** .
2. (Optional) In the Upgrade Software dialog, click **Verification**, and do the following:
 - Click **View Synchronization Health**, and confirm that the system components are synchronized.
 - Click **View Configuration Version**, and note the Current Version and Running Version.
 - Click **View Disk Usage**, and confirm that the system has enough free space.
3. In the Upgrade Software dialog, do the following:

Upload method	Select an upload method from the drop-down list.
Software file to upload	Browse to the file to upload.
Reboot after upload	Select to reboot the system after the upgrade.

4. Click **Complete**.
 - If you did not select **Reboot After Upload**, the system displays a message stating that a reboot is required for the changes to take effect.
 - If you selected **Reboot After Upload**, the system displays a message stating that it is about to reboot.
5. Click **OK**.

The system performs the file transfer and any boot parameter changes. If you selected **Reboot After Update**, the system reboots.

Edit, Copy, and Delete Configurations


You can edit, copy, and delete multi-instance configurations by way of the controls that the Web GUI displays on each multi-instance configuration page. The edit and copy functions act only on a single instance of a configuration. The delete function can act on either a single instance or all instances.

To edit, copy, or delete a single multi-instance configuration, select the configuration and right-click. The Web GUI displays the edit, copy and delete menu.

Codec Policy

Search Criteria: All

Add Delete All Upload Download

Name	Allow Codecs
Codec Policy 1	
Codec Policy 2	
 Codec Policy 3	
<div style="border: 1px solid #ccc; padding: 2px;"> Edit Copy Delete </div>	

When you click Delete, the system displays a confirmation dialog before performing the operation. When you click either Copy or Edit, the GUI displays the configuration dialog.

To delete all configurations at the same time, use Delete All.

Caution:

You cannot select several instances and delete only those. The system deletes them all, regardless of how many you select. For example, if you select two of three configurations and click Delete All, the system deletes all three.

Configuration States and Behavior

After you finish creating or modifying a configuration, you must save and activate the configuration before the Oracle Enterprise Communications Broker (OECB) saves the changes to the running configuration.

At any time, the following three versions of the configuration can exist on the OECB.

- **Editing.** The editing configuration is the version that you are making changes to from the Web GUI. The editing version is stored in the OECB volatile memory. The editing version cannot survive a system reboot.
- **Saved.** The saved configuration is the version of the editing configuration that the system copies into the non-volatile memory when you click **Save** on the Web GUI. Until you activate the saved configuration, the changes do not take effect on the OECB. The system does not load the saved, but not activated, configuration as the running configuration on reboot.
- **Running configuration.** The running configuration is the configuration that the system is using. When you activate the saved configuration it becomes the running configuration. Most configuration changes can take effect upon activation. Some configuration changes require a system reboot. On reboot, the system loads the running configuration.

The process for saving and activating a configuration, includes the following steps.

1. **OK.** All configuration dialogs display an **OK** button that saves changes to the editing memory. If you reboot before the next step, the OECB does not save the changes.
2. **Save.** The **Save** button on the Web GUI toolbar verifies the configuration, displays errors, saves the current configuration to the last-saved configuration, and stores it on the **OECB**. The system displays any errors at the bottom of the Configuration page.
3. **Activate.** After you finish making one or more configuration changes, **OK** and **Save** from the last configuration dialog that you need to edit at this time. The system displays the Confirmation dialog containing the **Activate** button. When you click **Activate**, the OECB activates all of the saved configuration changes and saves the new configuration to the running configuration. If you cancel the activation function, the OECB saves the configuration in a file and does not change the running configuration. You can continue to make changes to the configuration.

Configuration Error Messages

If you save a configuration that contains errors, the system displays the following error message: *There were errors! Are you sure you want to activate the configuration?*

The system displays a list of errors at the bottom the page. Click an error to go to the location in the configuration where the error occurred and edit the configuration as needed.

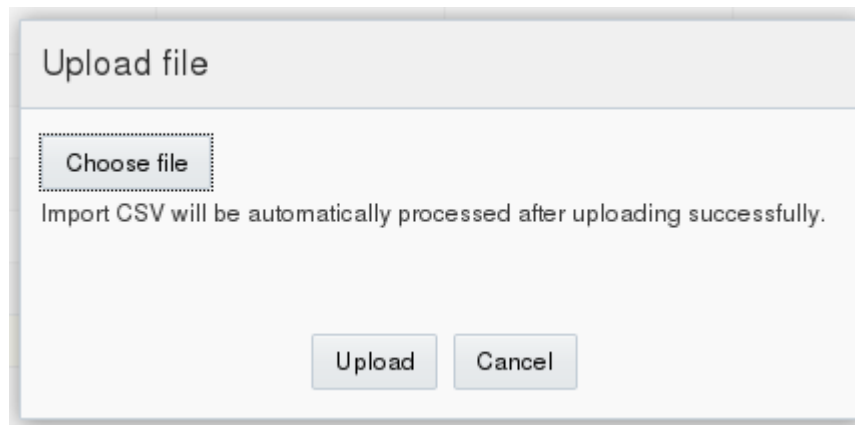
Severity	Identifies the level of severity that the Oracle Enterprise Session Border Controller assigns to the error. Valid values are: <ul style="list-style-type: none"> • ERROR—Means that the issue identified in the Message column is not correctly configured or it does not exist. You can still verify, save, and activate the configuration if this severity exists. • WARNING—Means that the configuration contains invalid information for the element field identified in the Message column. You can still verify, save, and activate the configuration if this severity exists. • CRITICAL—Means that a critical error occurred in the configuration and you cannot verify, save, or activate until the error is corrected. The Message column indicates the element field where the error has occurred.
Message	Identifies the element field where the error, warning, or critical error occurred, and the reason for the error.
Object	Identifies the element and the field for that element where the error occurred.
Attribute Name	Identifies the attribute within the element where the error occurred.
Other	Identifies any other pertinent information relating to the error.

Uploading and Downloading Configuration File Elements

The Oracle Enterprise Communications Broker allows you to upload and download files containing certain key elements of your configuration that are more easily managed separately from the overall system configuration. Such elements include:

- Dial Plans
- Dial Patterns
- User Database Entries
- Route Database Entries
- SIP Manipulation Rules

The corresponding dialogs for the items in the preceding list include controls for uploading and downloading files, depending on which action you choose. When you click the Upload button, for example, the system displays the Upload file dialog, which includes a Browse button. When you click Browse, the system opens a browse dialog, from which you can select the file that you want to upload. Download behavior is similar. The system performs the file transfer to the correct system directories in the correct format without requiring any intervention. The following screen capture shows an example of the Upload File dialog.



Using Tag Fields

The Oracle Enterprise Communications Broker provides a configuration element data field referred to as a tag. You enter information into the tag field for descriptive and grouping purposes. You can establish your own criteria for labeling configuration elements with these tags. Tag fields have no operational effect on signaling services.

The following configuration objects display the Tags text field:

- Agents
- Users
- Routes

You can enter any text that you want into the field and you can apply as many tags to a configuration object as needed. You can filter the element list searches using tags as a means of organizing these objects. Applicable element list search fields include a down arrow that exposes a tag drop-down list, from which you select the tag on which

to filter the list. Tags have no operational function other than supporting this kind of filtering.

Dashboard Tab Operations

The Oracle Enterprise Communications Broker (OECB) provides a web-based Dashboard that can display SIP data statistics to help you monitor and manage the system. The OECB collects only SIP data for the dashboard widgets, including the default CPU and Memory widgets. For this reason, you must set up a valid SIP configuration before the OECB can display any data on a dashboard widget.

The Dashboard supports up to 18 widgets. Each widget can display up to 100 data samples in intervals of 1 hour, 1 minute, or 1 second. You can select a chart, graph, table, web form, or text for the display, depending on the widget. You can customize the dashboard by adding, deleting, and moving the widgets. You can refresh the statistics displayed on the dashboard and you can reset the dashboard to its default display. The default display includes:

- Highest CPU Usage
- Current Memory Usage
- Historical Memory Usage
- Alarms

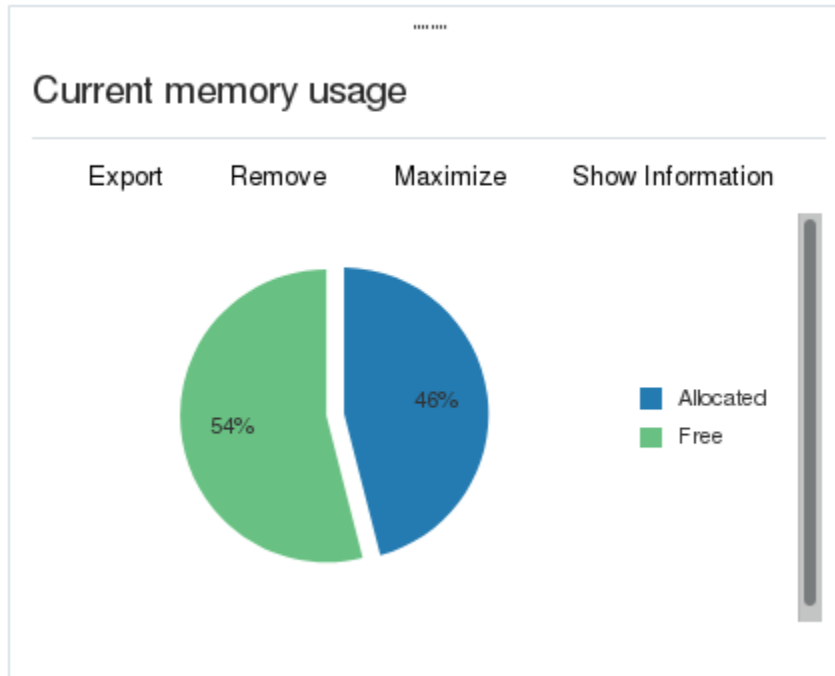
The Dashboard page provides the following controls to manage the widgets:

Refresh—Updates all of the widgets on the Dashboard.

Add Widget—Displays a list of widgets that you can add to the Dashboard.

Reset—Resets the Dashboard to display the default widgets and removes all other widgets from the Dashboard.

Each widget contains the particular controls that you need to manage the widget. The controls display as text until you hover over one. Upon hover, the text becomes a button that you click to perform a task. The following screen capture shows Refresh as a button upon hover, while Settings, Export, Removed, and Maximize remain static text.



The controls that can display in a Widget include:

- **Export**—Download the data displayed in the Widget.
- **Remove**—Remove the Widget from the dashboard. When you remove all Widgets from the dashboard, the GUI displays the following message: " Your dashboard is empty, please add a widget or reset to restore the original dashboard."
- **Settings**—Set the data sampling parameters for the Widget.
- **Maximize**—Display the Widget in full-screen size.
- **Show Information**—Display data in a graph or table.

Note that the operation of widgets, such as those that require the SIP Session module, may affect system performance. The system displays a warning when you add a widget that may affect performance. Oracle recommends adding such widgets at a time when the performance impact will not degrade service.

Add a Dashboard Widget

You can add up to 18 widgets to the Web GUI Dashboard to display SIP and System statistics to help you monitor and manage the system.

The system does not require a re-boot after performing the following procedure.

1. On the Home page, click **Add Widget**.
2. On the **Add Widget** page, in the left pane, expand **Widgets**, and click the name of the Widget to add.

The Web GUI displays the widget name in the center pane.

3. In the center pane, under **Command**, click **Add**.

The system displays a success message.

 **Note:**

If the system displays a warning that adding this widget requires enabling the SIP.Message module, the system enables the module when you add the widget.

4. Click **OK**.
5. Click **Close**.

The system displays the newly added widget on the Dashboard right away.

See "Configure Data Sampling Settings for a Dashboard Widget."

Configure Data Sampling Settings for a Dashboard Widget

- Confirm that the widget that you want to configure is on the Dashboard. See *Add a Widget*.

To display SIP and System statistics on a Dashboard widget, the system requires a setting for how often to refresh the display. You can use the default interval or select one from the Auto-refresh interval drop-down list on the widget. Some widgets also display the Table Name drop-down list, where you can set the data sampling frequency. For example, you might configure the widget to refresh the display every 40 seconds and to display the data samples in one minute increments.

1. Click the **Home** tab.
2. On the Widget, click **Maximize**, and click **Settings**.
3. Select a widget display refresh frequency from the **Auto-Refresh Interval (seconds)** drop down list.
4. If the widget displays the **Table Name** drop-down list, select a data sampling increment for the widget display.
5. Click **OK**.
6. Click **Close**.

View A Widget That is Not on the Dashboard

The Dashboard is limited to displaying a maximum of 18 widgets. To see a widget that is not on the Dashboard, use the Widgets link on the Monitoring page to see a list of all of the Widgets.

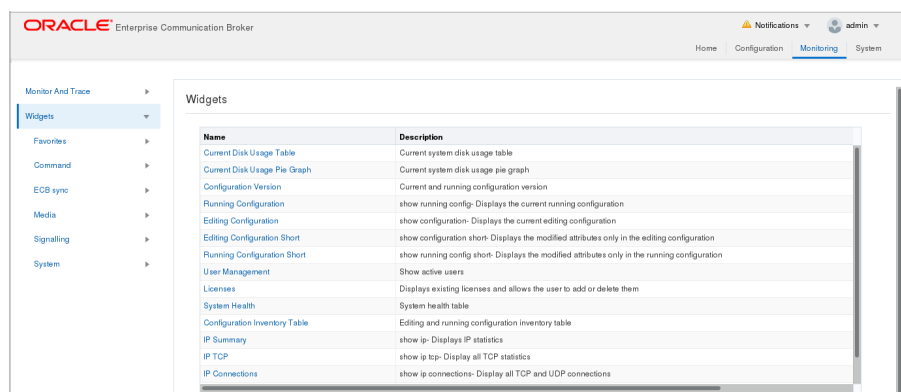
Use the following procedure to view widgets that are not displayed on the Home page as Dashboard widgets.

1. Access Widgets: **Monitoring, Widgets**.
2. In the navigation pane, click a Dashboard Widget name.
The system displays the Widget.
3. Optional—Add the widget to the Dashboard, if the Dashboard is not full.

Widgets Page

The Widgets section of the Monitoring page provides a place where you can select and view graphical displays of data and statistics about the Oracle Enterprise Communications Broker (OECB) in the form of a widget. Most widgets correspond to an equivalent ACLI `show` command. A widget can display a list, a table, a graph, or text depending on the type of data and the purpose of the display.

The Widgets page displays a categorical list of widgets in the left-hand navigation pane and provides their descriptions in the center pane along with the corresponding ACLI `show` command, when one exists. You can populate the Favorite Widgets list with the widgets that you use the most often. The following screen capture shows the Widgets page display.



Some widgets display static information, while others display actionable information. For example, the IP Connections widget displays a static list of internet connections, and the Licenses widget displays a list of licenses along with controls to Add and Delete licenses. Some widgets immediately display any available information when you click the name of the widget, for example, Configuration Inventory. Other widgets require you specify a few settings before displaying information, for example, Agent Individual. Most widgets allow you to set the Auto Refresh Interval.

Each Widget displays the particular controls used for its purpose. The following table lists all of the possible Widget controls and describes their purpose.

Tool	Description
Refresh	Update the data in the Widget.
Settings	Set the display settings per widget. Settings vary per widget, and can include: <ul style="list-style-type: none"> Parameters Auto refresh interval Table name Exclude processes
Add to Dashboard	Add the Widget to the dashboard.
Add to Favorites	Add the widget to the Favorite Widgets list.
Show Information	Displays a short description of the current widget.

Widgets and Descriptions

For each **show** command that you can use from the CLI to see ECB Sync, Media, Signaling, and System data about the Oracle Enterprise Communications Broker (OECB), the system provides a corresponding Widget on the Web GUI. Widgets display the data in a graphical or textual format.

A Widget can display a table, text, a pie graph, or a line graph depending on the type of data and the purpose of the display. For example, the SIP Realms All widget displays an actionable table and the Recording widget displays static text. You can access the list of widgets from either the navigation pane on the Monitoring tab or from the Home page by clicking Add Widget.

Most of the Widgets automatically display any available data when you click the name of the Widget, but some Widgets require further input. Such Widgets include a Settings button in their display that launches a parameters dialog where you specify the data to display. For example, the Realm Specifics Widget requires you to set the name of the realm and the auto refresh interval.

Note:

You must set up a valid SIP configuration before the OECB can display any SIP data in a Widget, including the default Widgets.

The following tables list all of the widgets in the left column. The middle column displays the corresponding CLI **show** command, if one exists. The right column describes the data that the Widget displays. Note that a few widgets do not correspond to a **show** command.

Note:

The following tables list the Widgets in the same order as displayed on the GUI, which is not alphabetical.

ECB Sync

Widget	Show Command	Description
ECB Sync Peer Status Table	No show command	Displays the sync state and uptime for both the transmitter and the receiver.

Media

Widget	Show Command	Description
NAT By Index	show nat by-index	Displays the specified range of entries in the NAT table, with a maximum of 5024 entries. The default range is 1-200. The range corresponds to the line numbers in the table, not to the number of the entry.
NAT in Tabular	show nat in-tabular	Displays a specified range of entries in the NAT table.

Signaling

Widget	Show Command	Description
DNS	show dns	Displays statistics for the DNS configuration.
ENUM	show enum	Displays ENUM statistics.
LDAP	show ldap	Displays LDAP statistics.
LRT	show lrt	Displays the Local Routing Table statistics.
Registration by Realm	show registration sipd by realm	Displays calls that registered through a specified ingress realm for which you want to view cache information.
SIP ECB Registration Cache	show registration sipd by-realm <realm id>	Displays user and contact information.
ECB Sync Remote Registration	spl show sip ecb-sync registrations	Displays the address of record for the sync agent.
Registration SIP	show registration SIP	Displays SIP registrations.
Registration Statistics	show registration statistics	Displays a table of counters showing the total and periodic number of registrations by protocol.
Sessions	show sessions	Displays the session capacity for license and session use.
Agent Details	show sipd agents	Displays statistics related to defined DIP session agents.
Agent Groups	show sipd groups	Displays cumulative information for all session agent groups.
Agent Individual	show sipd agents <agent name>	Displays statistics related to the specified SIP session agent.
Agent Status Table	show sipd agents-status	Displays the state of the agent and active inbound and outbound calls.
Client Trans	show sipd client	Displays statistics for SIP client events when the SBC is acting as a SIP client in the B2BUA role.

Widget	Show Command	Description
SIP Errors	show sipd errors	Displays statistics for SIP media event errors.
Interface Individual	show sipd interface	Displays SIP interface statistics for the specified realm.
Interface Summary	show sipd interface	Displays all SIP interface statistics.
Method Ack	show sipd ack	Displays all SIP ACK method statistics.
Method Bye	show sipd bye	Displays the SIP BYE method statistics.
Method Cancel	show sipd cancel	Displays all SIP CANCEL method statistics.
Method Info	show sipd info	Displays the SIP INFO method statistics.
Method Invite	show sipd invite	Displays the SIP INVITE method statistics.
Method Message	show sipd message	Displays the SIP MESSAGE statistics.
Method Notify	show sipd notify	Displays the SIP NOTIFY statistics.
Method Options	show sipd options	Displays the SIP OPTIONS statistics.
Method Prack	show sipd prack	Displays the SIP PRACK method statistics.
Method Publish	show sipd publish	Displays the SIP PUBLISH method statistics.
Method Refer	show sipd refer	Displays the SIP REFER method statistics.
Method Register	show sipd register	Displays the SIP REGISTER method statistics.
Method Subscribe	show sipd SUBSCRIBE	Displays the SIP SUBSCRIBE method statistics.
Method Update	show sipd update	Displays the SIP UPDATE method statistics.
SIP Realms All	show sipd realms	Displays realm statistics related to SIP processing.
SIP Realms Individual	show sipd realms <realm name>	Displays realm statistics related to SIP processing for the specified realm.
SIP Redundancy	show sipd redundancy	Displays information about SIP redundancy.
Server Trans	show sipd server	Displays statistics for SIP server events when the SBC acts as a SIP server in the B2BUA role.
SIP Sessions All	show sipd sessions all	Displays all SIP sessions currently on the system.
Forked Sessions	show sipd forked	Displays forked sessions statistics.

Widget	Show Command	Description
SIP Sessions	show sipd sessions	Displays the number of sessions and dialogs in various states for the Period and Lifetime spans.
SIP Status	show sipd status	Displays information about SIP transactions.

System

Widget	Show Command	Description
Accounting	show accounting	Displays a summary of statistics for configured external accounting servers.
ACL	show acl all	Displays cumulative and per-interface statistics on ACL traffic and drops, displaying Recent, Total, and PerMax counts. The display also separates traffic from trusted from untrusted sites.
Alarms	show alarms	Displays existing alarms and allows you to clear them.
Authentication RADIUS	show radius all	Displays the status of established RADIUS accounting connections.
Authentication TACACS	show tacacs stats	Displays statistics related to communications between the OECB and configured TACACS servers .
Communications Monitor Errors	show comm-monitor errors	Displays Communications Monitor aggregate error statistics information.
Communications Monitor Internal	show comm-monitor internal	Displays Communications Monitor aggregate internal statistics information.
Communications Monitor Stats	show comm-monitor stats	Displays statistics related to connections between the OECB Communications Monitor probe and any configured Communications Monitor servers.
Editing Configuration	show configuration	Displays the current editing configuration.
Editing Configuration Short	show configuration short	Displays only the parameters that you modified in the editing configuration.
Configuration Inventory	show configuration inventory	Displays the editing and running configuration inventory of all configured elements.
Running Configuration	show running-config	Displays the current running configuration.

Widget	Show Command	Description
Running Configuration Short	show running-config short	Displays only the parameters that you modified in the running configuration.
Configuration Version	show version	Displays the configuration version number table.
Highest Task CPU Usage Line Graph	No show command	Displays a line graph with 5-10 tasks with the highest CPU usage in percent, during a specific period of time.
Highest Task CPU Usage Table	No show command	Displays a table with 5-10 tasks with the highest CPU usage in percent, during a specific period of time.
Current Disk Usage Pie Graph	No show command	Displays the current disk usage in a pie graph.
Current Disk Usage Table	No show command	Displays the current disk usage in a table.
Features	show features	Displays the features that are currently enabled, based on added licenses.
Interfaces	show interfaces	Displays all of the information concerning the rear interfaces.
Interfaces Brief	show interfaces brief	Displays key running statistics about the rear interfaces in one graphic.
Interface Mapping	show interface mapping	Displays the configured physical interfaces with their MAC addresses and label.
Virtual Interfaces	show virtual interfaces	Displays the virtual interfaces for signaling services.
Wancom	show Wancom	Displays negotiated duplex mode and speed for all system control interfaces.
ARP Info	show arp info	Displays the current Internet-to-Ethernet address mappings in the ARP table.
ARP Statistics	show arp statistics	Displays ARP statistics.
ARP Summary	show arp	Displays the current Internet-to-Ethernet address mappings in the ARP table.
IP Connections	show ip connections	Displays all TCP and UDP connections.
Neighbor Table	show neighbor table	Displays the IPv6 neighbor table and validates that an entry for the link local address exists and that the gateway uses that MAC address.
Routes	show routes	Displays the current system routing table.
IP Summary	show ip	Displays IP statistics.
IP TCP	show ip tcp	Displays all TCP statistics.
IP UDP	show ip udp	Displays all UDP statistics.

Widget	Show Command	Description
Licenses	license	Displays existing licenses and allows you to add or delete them.
Current Memory Usage Pie Graph	No show command	Displays the current percentage of free and allocated memory in a pie graph.
Current Memory Usage Table	No show command	Displays the current percentage of free and allocated memory in a table.
Historical Memory Usage Line Graph	No show command	Displays a line graph of the kilobytes of free and allocated memory over a period of time.
Historical Memory Usage Table	No show command	Displays a table of the kilobytes of free and allocated memory over a period of time.
Memory Summary	show memory	Displays statistic related to the memory.
Platform All	show platform all	Displays full platform information.
Platform CPU load	show platform cpu-load	Displays current CPU load.
Platform Errors	show platform errors	Displays service pipe write errors.
Platform Limits	show platform limits	Displays platform related limits.
Processes	show processes	Displays statistics for all active processes.
SNMP Community Table	show snmp-community-table	Displays all information for configured SNMP communities including requests and responses for each community.
Trap Receiver	show trap-receiver	Displays trap receiver information for each configured SNMP community.
SPL Memory	show spl memory	Displays SPL memory for each task SPL engine.
SPL Options	show spl-options	Displays information on all SPL options.
SPL Statistics	show spl statistics	Displays statistics for all tasks.
SPL Version	show spl	Displays the version of the SPL engine.
System Health	show health	Displays the system health table for HA pairs.
Clock	show clock	Displays the current date and time.
NTP Server	show ntp server	Displays information about the quality of the time used for offset and the delay measurement maximum error bounds.

Widget	Show Command	Description
NTP Status	show ntp status	Displays information about configuration status, NTP daemon synchronization, NTP synchronization in process, and if NTP is not responding.
Time Zone	show timezone	Displays the time zone.
Clock UTC	show clock utc	Displays the current date and time in Coordinated Universal Time (UTC).
Uptime	show uptime	Displays information about the length of time the system has been running in days, hours, and seconds. Also displays the current date and time information.
User Management	show users	Displays a table that lists all users currently logged on to the system.
Version Boot	show version boot	Displays the boot version.
Version CPU	show version cpu	Displays the CPU version.
Version Hardware	show version hardware	Displays the hardware version.
Version Image	show version image	Displays the image version.
Version Summary	show version	Displays the Operating System version information, including the OS version number, the manufacturing date of the current version, and other details.

Command Line Interface Widgets

Like many devices, the Oracle Enterprise Communications Broker (OECB) includes an underlying management interface called the Command Line Interface (CLI). Support technicians use the CLI to display detailed information about the system in text format. Oracle makes this information available from the graphical user interface (GUI) with CLI Widgets, available from the OECB Widget page. The CLI Widgets can provide useful troubleshooting information, as well as insight into system operations.

To reach the CLI portal go to **Monitoring, Widgets, Command, Command Line Portal**. When you click the CLI portal link, the OECB displays the **Settings** page.

CLI portal settings include:

Command	A drop-down list where you choose the CLI command.
Parameter	A text box where you enter additional parameter text to refine the command output with a command argument.
Auto-refresh Interval	A drop-down list where you specify how often the system refreshes the widget's data. Default: Never. Valid values: Never 30 40 50 60.

The portal includes an **OK** button and a **Cancel** button. When you click **OK**, the system displays the output of the command.

The system produces two types of CLI widgets, depending on the command invoked:

- Text Display—The system displays the output of the command in an all-text format.
- Tabular Display—The system displays the output of these commands in a table.

License Widget

The License widget on the Web GUI provides a workspace where you can view, add, and delete Oracle Enterprise Communications Broker (OECB) licenses.

From the Widgets tab on the Web GUI, the system displays the Licenses page when you click **Widgets, System, Licences**.

The Licenses page displays a list of your OECB licenses with the following information.

Licenses	The name of the license.
Session Count	The number of session entitlements for the license.
Install Date	The date when the license is added to the system.
Begin Date	The date when the license begins service.
Expire Date	The date when the license ends service.

If you want to see the details of a particular license, click the show-hide toggle by the license name to expand the view to show all of the details. The following illustration shows an example of license details.

License	Session count	Install date	Begin date	Expire date
Acme Developer License 1	8000	12:24:18 Apr 24 2015		00:59:59 Apr 24 2015 expired!
Acme Developer License 2	8000	12:31:15 Apr 24 2015		00:59:59 May 11 2015 expired!

Acme Developer License: 8000 sessions, SIP, MGCP, H323, IMF, QoS, ACP, Routing, Load Balancing, Accounting, High Availability, PAC, LI, External SIP Mgmt, TLS, Software TLS, External CLF Mgmt, External Policy Services, ENUM, H248, H248 SCF, H248/BGF, NSRP, RPH, LI Debug, Session Duplication for Recording, Transcode Codec AMR (uncompressed AMR transcoding sessions), Transcode Codec EVRC (uncompressed EVRC transcoding sessions), DMS, IPSEC, IP4-v6 Interworking, RTP, EIC, Transcode Codec EVRCB (uncompressed EVRCB transcoding sessions), Software PCOM, Security Gateway, SIP Authorization/Authentication, Database Registrar (0 contacts), IOS Advanced, SLB (20000 endpoints), Software SRTP, Allow Unsigned SPL files, Session Recording, Diameter, Diameter, Transcode Codec AMR-WB (uncompressed AMRWB transcoding sessions), CK, Transcode Codec G729 (uncompressed G729 transcoding sessions), Transcode Codec Opus (uncompressed OPUS transcoding sessions), Transcode Codec SILK (uncompressed SILK transcoding sessions)

The Licenses widget provides the controls to Add and Delete licenses.

When you click **Add**, the system displays the Set license dialog.

When you select a license from the Licenses list and click **Delete**, the system displays the delete Confirmation dialog.

The License widget includes the Refresh, Download, Add to Dashboard, Pin to Favorites, and Help icons, familiar from other widgets, in the top, right-hand corner. Note that the License widget does not include the Settings icon and the Auto-refresh function because these operations do not apply to licenses.

The Set License wizard is linked to the License widget, so that you can view your licences from the wizard. After launching the Set License wizard, use the "View current license information" link in the Set License dialog to see a view-only list of your OECB licenses.



The only operations allowed in view mode are Refresh and Download.

Display and Clear Alarms

The Oracle Enterprise Communications Broker provides a Widget that allows you to see all current alarms that the system triggered.

1. Access the Widgets link.
Monitoring, Widgets.
2. Click **System, Alarms.**
3. (Optional)—To clear alarms, click **Clear All.**

Display Users

The Oracle Enterprise Communications Broker provides a widget that allows you to see a list of other users currently logged on to the system.

1. Access the Widgets link.
Monitoring, Widgets.
2. Click **System, User Management.**
The Oracle Enterprise Communications Broker displays the User Management Page.
3. (Optional)—Right-click a user, and click **Disconnect.**

Accessibility Features

The following accessibility features are available in this release.

Keyboard Navigation

You can access GUI functionality using only the keyboard. Use the following keys to navigate:

- Tab key: Move to the next control, such as a dynamic target menu, navigation tree, content pane, or tab in a page. Tab traverses the page left to right, top to bottom. Use Shift +Tab to move to the previous control.
- Up and Down Arrow keys: Move to the previous or next item in the navigation tree, menu, or table. Down Arrow also opens a menu.
- Left and Right Arrow keys: Collapse and expand an item in the navigation tree or a submenu.
- Spacebar: Activate a control. For example, in a check box, spacebar toggles the state, checking or unchecking the box.
- Enter: Activate a button.



Note:

Some functionality cannot be accessed through keyboard navigation.

Microsoft Screen Narration and Jaws

This product is compatible with common OS's screen reader programs.

Microsoft Magnifier

This product is compatible with common OS's screen magnifying features.

5

Agent Configuration

Agent configurations specify the hops by which the Oracle Enterprise Communications Broker (OECB) defines routes. Network architects select agents to identify and delineate between key logical and/or physical locations in the overall network topology. Mapping out agent topology can also identify logical or physical locations that would benefit from an agent. Agents are usually physical devices, such as proxies, SBCs, PBXs and gateways. But agents can also be logical entities, including domain names.

You can configure agents with:

- Location definition
- Context
- Translation mode
- Inbound and outbound manipulation rules
- Traffic constraints

The last two bullets are particularly impactful when the agent is the first hop in a route, but are applicable to agents multiple hops from the OECB.

Alphanumeric User Database and Call Routing Entries

The Oracle Enterprise Communications Broker (OECB) allows you to enter alphanumeric entries in the User database and in the routing table because it can modify the request URI to use either alphanumeric or numeric entries, depending on the setting you choose for **egress-URI-mode**.

Egress-URI-mode

When the OECB locates the home agent to which the call is routed, it uses one of the following configurable values from the **egress-URI-mode** parameter in the **Agent** configuration to determine whether or not to convert the request URI:

- No-conversion (Default)—The OECB adds only the IP address of the Home Agent in the outbound call.
- Convert-to-aor— When the incoming URI is a number, the OECB replaces it with the configured address of record in the outbound call.
- Convert-to-number—When the incoming URI is an Address of Record, the OECB replaces it with the configured number in the outbound call.

Example

Assume that the User database includes "aor1@oracle.com" for the AOR, "123@oracle.com" for the number, and the destination agent is named Agent123.

When a call comes in, the OECB checks the value in the **egress-URI-mode** parameter for the destination agent Agent123 and does the following:

- If the request URI in the incoming call is "aor1@oracle.com", and the egress-URI-mode for Agent123 is set to "convert-to-number", the OECB modifies the request URI to 123@oracle.com.
- If the request URI in the incoming call is "123@oracle.com", and the egress-URI-mode for Agent123 is set to "convert-to-aor", the OECB modifies the request URI to "aor1@oracle.com."
- If you set the egress-URI-mode for Agent123 to "no-conversion", the user part of the request URI remains unchanged. The OECB changes only the host part of the request URI to use the IP address of Agent123.

Alphanumeric User Database Entries

In the User Entries configuration, both the **Address of Record** parameter and the **Number or Pattern** parameter can accept alphanumeric entries. Note that you can leave either the **Address of Record** or the **Number or Pattern** parameter empty, but not both at the same time. The user database supports the following combinations of entries:

Address of Record	Number or Pattern	Home Agent	Dialing Context
username1@company.com	1234567890@company.com	Agent1	region.country
username2@company.com	1234567890	Agent2	region.country
username3@company.com		Agent3	region.country
username1@1.1.1.1	1234567890	Agent3	
	11234567890	Agent4	
	12324567890	Agent4	
	12345678[01-04]	Agent5	
	xxxxxxxx	Agent5	

The user database does not support wild card entries in the **Address of Record** parameter, but the **Number or Pattern** parameter supports the following wild card patterns:

- 12345678xx@company.com
- 12345678xx@country1.company.com
- 12345678xx@country2.com
- 123456[01-20]@country.company.com

Example of a valid configuration

The following example shows the **Address of Record** and the **Number or Pattern** entered in the User database in a valid configuration, where the entries are properly formatted.

Address of Record	Number or Pattern	Home Agent	Dialing Context
username@company.com	1234567890@company.com	Agent1	region.country

With "convert-to-number" selected, the OECB modifies the request URI to use the number 1234567890@company.com. With "convert-to-aor" selected, the OECB modifies the request URI to use the **Address of Record** username@company.com.

Example of an invalid configuration

The following example shows an invalid user record which includes two different address of record entries for one user (rows 1 and 2), and duplicate address of records for two different numbers (rows 3 and 4).

Address of Record	Number or Pattern	Home Agent	Dialing Context
username@company.com	1234567890@company.com	Agent1	region.country
user@1.1.1.1	1234567890@company.com	Agent1	region.country
auser@company.com	5551234567@company.com	Agent1	region.country
auser@company.com	0005551234@company.com	Agent1	region.country

Routing Table Entries

The routing table supports the following syntax for alphanumeric entries in the **Calling Number** and **Called Number** parameters:

- 1234567890
- 123[000-999]
- 123[x]xxx
- 1234567890@company.com
- user_name@company.com
- user.name@1.1.1.1

Dial Plan Lookup

When the OECB attempts a dial plan lookup, it first confirms that the user part of the request-uri is a valid phone number. For a valid number, the OECB performs the dial plan lookup. When user part of request-uri is an alphanumeric string, the OECB skips the dial plan lookup.

ENUM

When the OECB uses a configured route to an ENUM server, it does not convert the request-uri as specified in the destination agent configuration. If the destination agent (returned by the ENUM server) is not reachable, and OECB chooses a backup route from the user database, the OECB converts the request-uri of the outgoing INVITE based on the **egress-uri-mode** parameter set for the destination agent in the user database.

LDAP

When a user database entry and an LDAP entry both exist, the OECB gives priority to the entry in user database. Conversion of the request-uri is performed based on the configuration of the destination agent. If a user database entry does not exist, LDAP takes the precedence and the system does not convert the request-uri.

About Traffic Constraints

You use traffic constraint configuration to establish call admission control (CAC) thresholds and triggers that determine whether the OECB can accept and process network traffic, including calls. The OECB provides call CAC based on the following policies:

- Bandwidth
- Session capacity
- Session rate

Bandwidth-Based Admission Control

The OECB is a policy enforcement point for bandwidth-based call admission control. Sessions are admitted or rejected based on bandwidth policies, configured on the OECB for each agent.

As the OECB processes call requests to and from a particular agent, the bandwidth consumed by the traffic is decremented from its bandwidth pool.

Session Capacity- and Rate-based Admission Control

A session agent defines a signaling endpoint. It is a next hop signaling entity that can be configured to apply traffic shaping attributes. You can define concurrent session capacity and rate attributes for each session agent.

You can configure a set of attributes and constraints for each session agent to support session access control. And you can set up session admission control so that the OECB limits the number of concurrent inbound and outbound sessions for any known service element.

The OECB denies a call request to any destination that has exceeded its configured policies for session capacity and session rate. The OECB might reject the call request back to the originator. If multiple destinations are available, the Oracle Enterprise Communications Broker checks current capacity and rate for each destination and attempt to route the call only to destinations whose policy limits have not been reached.

CAC Utilization Statistics via SNMP

The OECB allows you to retrieve information on current session utilization and burst rate as a percentage of their configured maximums on per session-agent basis. The OECB uses the configured max-session and max-burst-rate settings in conjunction with a percentage formula to calculate this value. The system also uses a configuration setting to establish the threshold at which trap and trap clear messages are sent from the SNMP agent to the configured manager(s). You must configure traps and enable the **SNMP Monitor Traps** parameter for this functionality.

You must load the MIB version associated with this software version on all pertinent SNMP managers to query these CAC utilization (occupancy) values and interpret the traps. In addition, the user must configure the threshold at which the system generates the CAC utilization trap. Note that the corresponding clear trap uses the same threshold setting, sending the clear trap when utilization falls below 90% of the threshold.

The OECB can issue a trap when either the value of max-session or CAC burst rate exceeds a configured value. The system only sends one trap when the threshold is exceeded. When the value falls back under 90% of this threshold, the OECB sends a clear trap.

You configure the value that triggers these traps as a percentage of the max-session and max-burst-rate settings configured for the applicable session agent. The system uses the same setting to specify when to send both the sessions and burst rate traps. The name of this parameter is the **cac-trap-threshold**. You must express the value as a number less than 100. There is no default setting; the system does not generate a trap if you have not configured this setting.

The apSipCACUtilAlertTrap identifies the threshold exceeded on a per-element and per-value (session count or burst rate) for each trap, including:

- apSipSaCacSessionUtilLevel
- apSipSaCacBurstRateUtilLevel

Configure a Session Agent

You can enable and configure a variety of constraints that the Oracle Enterprise Communications Broker (OECB) applies to regulate session activity with the session agent.

Configure the following before you configure a session agent.

- Media profile
- Out Translation ID
- Local Response Maps
- Codec Policy
- Session Recording Server
- TLS profile
- SIP header manipulation IDs
- LDAP
- One or more target groups

The following procedure lists all of the possible parameters that you can set on a session agent. The parameters that the OECB allows you to set depends on your deployment configuration, the licenses that you own, whether or not you configured the prerequisite that a particular parameter requires, and whether or not you enabled a requisite function. Within those guidelines, you can choose which parameters that you want to set for your deployment.


1. On the **Agents** page, do the following:

Hostname	Enter the name of the host associated with the agent in host name, FQDN, or IP address format. This field is required and the name cannot include blank spaces. The value entered here must be unique to this agent because no two agents can use the same host name.
----------	---

	<ul style="list-style-type: none"> If you enter the host name as an IP address, you do not have to enter an IP address in the optional IP address parameter. If you enter the host name in FQDN format, and you want to specify an IP address, enter it in the optional IP address parameter.
IP address	(Optional) Enter the IP address for the host name that you entered in FQDN format if you want to specify the IP address. Otherwise, you can leave this parameter blank to allow a DNS query to resolve the host name.
Port	<p>Enter the number of the port associated with this agent.</p> <ul style="list-style-type: none"> 0. If you enter zero, the OECB cannot initiate communication with this agent (although it will accept calls). 1025-65535. The default value is 5060. <p>If the transport method value is TCP, the OECB will initiate communication on the TCP port of the agent.</p>
State	Select State to enable this agent.
RURI with Hostname	<ul style="list-style-type: none"> Select to resolve all outgoing requests to the Session Agent to the Session Agent name in the RURI. Deselect to resolve all outgoing requests to the Session Agent to the Session Agent IP address in the RURI. <p>Default: deselected.</p>
Transport method	<p>Select the transport protocol for communicating with this agent. Valid values:</p> <ul style="list-style-type: none"> UDP - Default UDP+TCP Dynamic TCP Static TCP Dynamic TLS Static TLS DTLS TLS+DTLS Static SCTP
TLS profile	Select a TLS profile from the drop-down list.
Realm ID	Enter the name of the realm in which this agent resides.
Description	Enter descriptive text to identify this agent.

Source context	Select the dialing context the OECB uses when receiving a call from this agent.
Egress URI mode	Select the method you want for sending the outbound call from this agent. Default: no-conversion. Valid values: <ul style="list-style-type: none"> no-conversion—The OECB adds only the IP address of the Home Agent in the outbound call. convert-to-aor—When the incoming URI is a number, the OECB replaces it with the configured address of record in the outbound call. convert-to-number—When the incoming URI is an Address of Record, the OECB replaces it with the configured number in the outbound call.
Egress number translation mode	Select a number translation code from the drop-down list. Default: E164. Valid values: <ul style="list-style-type: none"> E164—Example: +14445551234 E164 no plus—Example: 14445551234 no country code—Example: 4445551234 pattern only—Use only the portion matching the dial pattern. Example: 1234 n digit dialing—Use the specified number of digits. Example: 5551234
Number of digits for n digit dialog	Specify the number of digits to use for translation mode (n-digit dialing). Default: 4. Range: 0-25.
Prepend prefix on egress	Specify the characters to prepend to the outbound number after translation. Valid values: Telephony characters 0-9, start, hash sign, and A-D. Maximum: 25 characters.
Outbound translate from number	Select to apply outbound translation to the FROM number in addition to the request-URI. Default: Disabled.
Stop recurse	Enter one or more response codes that you want to cause this session agent to stop route recursion. Default: 401,407. Valid range: 300-599. You can enter individual response codes separated by a comma, such as 301,305 or a range such as 300-380.
Constraints	Select to enable the use of constraints on this agent.
Max sessions	Enter the maximum number of sessions allowed for this constraint. Default: 0. Range: 0-999999999.
Max inbound sessions	Enter the maximum number of inbound sessions allowed from this session agent. Default: 0. Range: 0-999999999.
Max outbound sessions	Enter the maximum number of outbound sessions allowed for this constraint. Default: 0. Range: 0-999999999.

Max burst rate	Enter the maximum number of invites allowed in a burst time period. Default: 0. Range: 0-999999999.
Max inbound burst rate	Enter the maximum inbound burst rate in INVITEs per second from this session agent. Default: 0. Range: 0-999999999.
Max outbound burst rate	Enter the maximum outbound burst rate in INVITEs per second from this session agent. Default: 0. Range: 0-999999999.
Max sustain rate	Enter the number of sustained INVITEs per second. Default: 0. Range: 0-999999999.
Max inbound sustain rate	Enter the maximum inbound sustain rate of session invitations allowed within the current time period for this constraint. Default: 0. Range: 0-999999999.
Max outbound sustain rate	Enter the maximum outbound sustain rate of session invitations allowed within the current time period for this constraint. Default: 0. Range: 0-999999999.
Session max life limit	Enter the maximum time you want a session to stay active. Default: 0, which means disabled. Range: 0-2073600
Time to resume	Set the delay, in seconds, after which the SIP proxy resumes sending session invitations to this session agent. Default: 0. Range: 0-999999999.
In service period	Set the amount of time, in seconds, that the session agent must be operational after communication is re-established before it is declared as being in service. Default: 0. Range: 0-999999999.
Burst rate window	Set the window period, in seconds, that the system uses to measure the burst rate. Default: 0. Range: 0-999999999.
Sustain rate window	Set the sustained window period, in seconds, that the system uses to measure the sustained rate. This value must be greater than the one set for "maximum sustain rate". Default: 0. Range: 0-999999999.
Ping interval	(SIP only) Set the time, in seconds, for how often to ping the session agent. Range: 0-4294967295. Default: 0.
SPL options	<p>You can add features or parameters here. For example:</p> <ul style="list-style-type: none"> feature=<value feature> You can include the original address in the SIP message from the ECB to the proxy in the Via header parameter by entering the following option: via-origin=<parameter-name> <p>The original parameter is included in the Via of the requests sent to the session agent. The via origin feature can take a value that is the parameter name to include in</p>

	<p>the Via. If the value is not specified for via origin, the parameter name is origin.</p> <div style="border: 1px solid #0070C0; background-color: #E6F2FF; padding: 5px;"> <p> Note:</p> <p>If the feature value is a comma separated list, enclose it in quotation marks.</p> </div>
Apply outbound manipulation on	Specify when to apply outbound header manipulation. Default: next-hop-only. Valid values: last-hop-only, next-and-last-hop.
In manipulation ID	Enter the name of the SIP header manipulation configuration that you want to apply to the traffic entering the OECS through this session agent. No default.
Out manipulation ID	Enter the name of the SIP header manipulation configuration that you want to apply to the traffic exiting the OECS through this session agent. No default.
Early media inhibit	Select to inhibit early media. Default: disabled. Valid values: Disabled, enabled.
Enable options ping	Select to enable the use of OPTIONS pings to determine the status of this agent.
LDAP	Select the name of the ldap-group or ldap-config-group that you want this agent to use.
Additional target group	Select an additional target group from the drop-down list.
Fork group	Enter the number for the fork group number, which determines this session agent's priority in a target list. The lower the number, the higher the priority. Default: 1.
Refer call transfer	Select to enable the REFER method for call transfer. Default: disabled. Valid values: Disabled, enabled.
Refer notify provisional	<p>Select when to send a NOTIFY for REFER provisional responses.</p> <ul style="list-style-type: none"> • None—The system sends no intermediate NOTIFY message. • Initial—The system sends an intermediate 100 Trying NOTIFY message. • All—The system sends an intermediate 100 Trying NOTIFY message, plus a NOTIFY for each non-100 provisional received by the OECS.
Monitoring filters	Click Add to add one or more monitoring filters.

2. Click **OK**.

3. Save the configuration.

Configure a Session Agent Group

Create a session agent group to define a signaling endpoint configured to apply traffic shaping attributes and information about next hops and previous hops. Session agent groups contain individual session agents in a logical grouping. You can configure each member of the group with individual constraints. The system does not require all members of a group to apply the same constraints.

Session agent group members do not need to reside in the same domain, network, or realm. The Oracle Enterprise Communications Broker (OECB) can allocate traffic among member session agents regardless of their location by using the allocation strategy that you select allocate traffic across the group members.

1. Access the Groups configuration object.

Configuration, Service Provisioning, Agents, Groups.

2. On the **Groups** page, click **Add** and do the following:

Group Name	Enter a name for this Session Agent Group.
Description	Enter a description of this Session Agent Group.
Strategy	Select a strategy for choosing this agent from the drop-down list. Default: Hunt. Valid values: Hunt Round Robin.
Dest	Enter one or more destinations for this group.
SAG Recursion	Select to enable Session Agent Group recursion. Default: Disabled.
Stop SAG Recurse	Enter one or more response codes that you want to cause this session agent group to stop route recursion. Default: 401,407. Valid response code values range from 300-599. You can enter individual response codes separated by a comma, such as 301,305 or a range such as 300-380.

3. Click **OK**.
4. Save the configuration.

Configure ENUM Servers

You can create E.164 Number to URI Mapping (ENUM) server configurations on the Oracle Enterprise Communications Broker (OECB) to resolve SIP URIs presented to the OECB in a call. An ENUM server configuration points to one or more ENUM servers from which the OECB can request resolutions. Configuring with multiple servers provides redundancy when a particular server cannot respond or provide a resolution.

- Configure a top-level domain
- Configure one or more ENUM servers to add to the ENUM settings configuration


Similar to an agent configuration, an ENUM server configuration includes number translation settings for the strings returned by the ENUM infrastructure. The ENUM

configuration also includes configuration values to support the interaction with the servers.

1. Access the ENUM Servers configuration object.

Configuration, Service Provisioning, Agents, ENUM Servers.

2. On the ENUM Servers page, click **Add**, and do the following:

Name	Enter a unique name for this ENUM configuration. You may need to use this name in other areas of the OECB configuration to refer to this ENUM configuration. For example, in route configuration.
Top Level Domain	Enter the domain extension to use when querying the ENUM servers for this configuration. For example, e164.arpa. The query name is a concatenation of the number and the domain.
Servers	Enter one or more ENUM servers (an ENUM server and corresponding redundant servers) to query. Separate each server address with a space and enclose list within parentheses. <div style="border: 1px solid #0070C0; background-color: #E6F2FF; padding: 10px; margin: 10px 0;"> <p> Note:</p> <p>The OECB media interface does not support management traffic for ENUM. When configuring connectivity to a media interface, do not configure these resources within a media interface subnet range.</p> </div>
Number Translation Mode	Select the translation mode required by this agent. The modes define how to format the ENUM request to accommodate the specific ENUM server. Valid values: <ul style="list-style-type: none"> • E164 - Default. The server can accept numbers in E164 format. • E164-no-plus. The server uses numbers in E164 format, with the exception of the plus sign. • no-country-code. The server cannot use a country code. • pattern-only. The server cannot use any string that varies from the configured pattern. • n-digit-dialing. The server requires the specified number of digits.
Number of Digits for n Digit Dialing	If you selected n-digit-dialing as the Number Translation Mode for this agent, specify the number of digits the OECB must send to this server.
Prepend Prefix	Specify a prefix the OECB must send to this server. For example, the digit 9, which may be required to allow outbound traffic.
Advanced Settings	Expand to display the following settings.

Query Method	Set the strategy the OECB uses to contact ENUM servers. Valid values are: hunt—Directs all ENUM queries toward the first configured ENUM server. When the first server is unreachable, the OECB directs all ENUM queries to the next configured ENUM server, and so on. round-robin—Cycles all ENUM queries sequentially among all configured in-service ENUM servers. The OECB directs query 1 to server 1, query 2 to server 2, query 3 to server 3, and so on.
Time Out	Set the number of seconds to elapse before a query sent to a server (and its retransmissions) times out. Range: 0-4294967295 seconds.
Lookup Length	Set the length of the ENUM query, starting from the most significant digit. Default: 0. Range: 1-255.
Max Response Size	Enter the maximum size in bytes for UDP datagrams in DNS NAPTR responses. Range: 512 (default)-65535. Oracle recommends configuring values that do not exceed 4096 bytes.
Health Query Number	Set a standard ENUM NAPTR query that will consistently return a positive response from the ENUM server. Blank = disabled.
Health Query Interval	Set the number of seconds to perpetually probe ENUM servers for health. Range: 0-65535 seconds.
ENUM Options	Add one or more ENUM options.

3. Click **OK**.
4. Save and activate the configuration.

Enable ENUM Session Agent Group Matching

When you want the Oracle Enterprise Communications Broker (OECB) to consider session agent groups as part of the response from the ENUM server, set the **Enum Sag Match** parameter in **SIP Config** to enable. With this parameter enabled, the OECB matches session agent group names received from the ENUM server with the hostname portion in the naming authority pointer. In this way, the OECB supports a group of session agents as a SIP designation in the OECB ENUM request.

1. Access the SIP Interface configuration object.
Configuration, System Administration, SIP Interface, SIP Config.
2. On the **SIP Config** page, enable **Enum Sag Match**.
3. Save the configuration.

Multi-Hop Agent Ping

The Oracle Enterprise Communications Broker (OECB) ping function can test connectivity to endpoints that are not directly adjacent to the source OECB. This multi-hop ping capability requires that you configure special routes dedicated to sending SIP options pings to these targets.

To enable ping tests to targets that are more than one hop from the OECB, configure routes that have the string "ping:" in the **Called number** field. These routes also have a first agent towards the target configured in the **Route** field, and the last agent toward the target configured in the **Destination Agent** field.

The system invokes these ping: routes for ping traffic only. In addition, the system prioritizes SIP signaling traffic over ping: route traffic.

The following table shows an example of a multi-hop ping route.

Source Agent	Calling Number	Destination Agent	Called Number	Route	Cost	Policy
*	*	Target_Agnt	ping:	Adja_Agnt	0	

Having configured this route, you can initiate a ping to a target, or configure agent pinging to Target_Agnt by way of Adja_Agnt.

You can also set up multi-hop ping recursion by creating multiple ping: routes that specify the complete path to the target. To create these paths, you can configure ping: routes using the Destination Agent and Route fields to define each hop in a "ping path".

Based on the two route entries in the following example, ping attempts to reach the device defined as Target_Agnt follow a two-hop path:

1. OECB to Adja_Agnt
2. Adja_Agnt to Interim_Agnt
3. Interim_Agnt to Target_Agnt

Source Agent	Calling Number	Destination Agent	Called Number	Route	Cost	Policy
*	*	Interim_Agnt	ping:	Adja_Agnt	0	
*	*	Target_Agnt	ping:	Interim_Agnt	0	

The OECB uses an agent status, determined by OPTIONS ping, to validate all routes using that agent. Specifying an agent's status, including in-service and out-of-service, is the same for agents using either single or multi-hop ping. The system does not use routes to out-of-service agents for any signaling traffic.

6

Dial Plan Configuration

Dial plans specify how you want the Oracle Enterprise Communications Broker (OECB) to process calls and route them according to contexts and patterns that you configure. Use the Dial Plan icon on the Configuration tab to access the Dialing Contexts configuration page, where you define how you want the system to handle dial patterns.

The Dialing Contexts configuration page displays the following permanent dialing context hierarchy parents:

- **Corporate**—The parent of child dialing contexts of groups of users that you add, who share dialing patterns that you specify. For example, the employees in a branch office. You can add thousands of child contexts to the Corporate parent and you can set thousands of dial patterns for each child.
- **Geographic**—The parent of sets of child dialing contexts pre-defined by the OECB for every geographic location in the world. You can add thousands of child contexts to the Geographic parent and you can set thousands of dial patterns for each child.

Plan and configure your contexts in hierarchical priority. The OECB always uses the most specific match it can find when performing dial pattern matches. When the OECB finds no match in a child context, it searches for a dial pattern match in the parent context.

You can configure a dial plan on the OECB by way of the Web GUI or you can upload a dial plan in a .csv file.

Dial Pattern Configuration

A dial pattern defines the prefix and pattern that the Oracle Enterprise Communications Broker (Oracle Enterprise Communications Broker) receives and defines the transformation that the system performs.

Only one transformation type is valid for each prefix and pattern match. The system displays an error when you try to configure multiple transformation types. Transformation types include replacement prefix, replacement URI, and Go to context. The following table shows examples of patterns, the transformation types, and the results of the transformation.

Prefix and pattern	Transformation type	Result
8 - xxxx	Replacement prefix	Replace with configured digits, which in this case are suitable for an outside line.
911	Replacement URI	Insert the configured service URN for emergency services.

Prefix and pattern	Transformation type	Result
*123	Go to context	Present the prefix/pattern to another context. Matching occurs based on the target context's configured dial patterns.

Overlapping dial pattern matches in the same context that result in finding the same target number produce configuration errors. Such errors produce an ambiguity that the system cannot resolve, so it does not forward the message. You must configure patterns, especially those that use encoding characters, very carefully to avoid ambiguities that the system cannot resolve.

When overlapping patterns result in the same target number, the system forwards to that number without error. When the system finds overlapping patterns in different contexts, the system chooses the most specific context as a match. Child contexts are more specific than parent contexts. When configuring dial patterns, Oracle recommends keeping them unique even across contexts.

Dial Pattern Encoding Characters

The Oracle Enterprise Communications Broker (OECB) allows multiple matches to a specified dial pattern, such as the ones you add to a dialing context. You can use selected characters to help you encode dial patterns to meet your needs.

In the following table, the Character column lists the encoding characters that a dial pattern allows and the Usage column explains what the characters mean in a pattern and how to use them.

Character	Usage
Brackets []	Use brackets to enclose digit ranges you need to express for a pattern. The OECB parses the pattern 8[1-20]9 as 8[01-20]9, adding an implied 0 before 1. The OECB considers both values to contain the same number of digits. The OECB strictly enforces the range and the number of characters in the preceding examples, as follows: <ul style="list-style-type: none"> • 8019 matches • 819 does not match • 8119 matches
The "x" character	Use the x character as a wildcard in dial pattern strings. Use the "x" character can only at the end of a string. When you configure a pattern that includes an "x" character followed by digits, the system displays an error.

Character	Usage
Parenthesis ()	<p>Use parenthesis to enclose wildcard characters and express a pattern. The OECB does not strictly enforce the range and the number of characters in patterns with "x" characters in parenthesis. Consider the pattern 8xx(xx):</p> <ul style="list-style-type: none"> • 812 matches • 8123 matches • 81234 matches

Note that the use of encoding characters can result in overlapping dial-pattern matches. Overlapping dial-pattern matches that result in multiple targets introduce ambiguity that the OECB cannot resolve. As a result, the system does not forward the signaling.

For example, the following two dial-patterns overlap:

- 4000
- 4xxx

Double check dial patterns made up of encoding characters to avoid overlaps.

Configure a Dial Plan

You can add one or more child dialing contexts to the Corporate and Geographic parent dialing contexts to specify how you want the Oracle Enterprise Communications Broker (OECB) to extrapolate and present universal strings to the routing engine upon ingress, as well as to create target URIs for egress based on these rules.

Configuration includes specifying the applicable:

- Prefixes, including:
 - Access codes
 - Tie line digit sequences
- Services, such as 411 and 911 services
- Dialing Ranges
- Dialing Range exceptions (gaps in dialing ranges)

1. Access the Dial Plan configuration object.

Click **Configuration, Service Provisioning, Dial Plan**.

2. On the Dialing Contexts page, click **Corporate** or **Geographic**, and click **Add**.

The attributes and instructions in the next step apply to both Corporate and Geographic.

3. On the Add Dialing Context page, do the following:

Name	Enter a name for this dial plan. Required.
------	--

Geographic Location	Select a geographic location from the drop-down list to use for pattern matching when the OECB cannot find a pattern match in this context's dial-patterns.
Description	(Optional) Enter a description of this dialing context.
Country Code	Enter the E.164 country code for which this dialing context exists. For example, 1 for North America.
Outside Line Prefix	Do one of the following: <ul style="list-style-type: none"> • Enter the characters required for PSTN access. Valid values: 0-9, *, #, and A-D. Maximum: 25 characters. • Leave blank to allow direct dialing.
Dial Pattern	<ol style="list-style-type: none"> 1. Click Add, and do the following: <ul style="list-style-type: none"> • Remove Prefix—Enter the prefix of the dial pattern entry that you want stripped for translation. Valid values: 0-9, *, #, and A-D. Maximum: 25 characters. Allow ranges in brackets []. For example, 555[2000-3999]. • Pattern—Enter telephony digits 0-9. Maximum: 25 characters. Allow ranges in brackets []. For example: 555[2000-3999]. Use x at the end of a pattern to mean 0-9. Use parens () around optional digits. For example: 555xx(xxxx) means 555 followed by 2 to 6 digits. • Description—Enter a description of this dial pattern. • Country Code—Enter the country code for this dial pattern or leave blank to inherit from the context. • Replacement Prefix—Enter the replacement prefix to add to the translated number. Valid values: 0-9. Maximum: 25 characters. • Replacement URI—Enter the URI to use without outbound translation. • Go to Context—Select the target context from the drop-down list. 2. Click OK. 3. (Optional) Add another dial pattern.

4. Click **OK**.
5. (Optional) Add another child dial plan.
6. Save the configuration.

7

User Configuration

User configuration is a required task. It is, in fact, the only way to assign an agent to a user. Complex VoIP deployments can derive additional benefits from the user database. Examples include contiguous dial strings being deployed across multiple PBXs, making it impossible to identify the target PBX by dial string alone.

When you need a user database, you configure entries for all source and destination numbers that you know require user database support. The user database effectively performs dial plan tasks on individual numbers. Such tasks include identifying a user's agent and source context. These entries provide the Oracle Enterprise Communications Broker with shortcuts for determining this information.

A user often represents an actual account that exists within an enterprise's dialing network. Examples of users include an extension, a subscriber or a phone number. Each user may have a source context, which translates into "Default" dialing rules for processing that user's calls using the appropriate contextual transformation rules and the agent at which they are located.

You can configure this database manually. Alternatively, you can upload user information in a format pre-configured to translate into a user database.

Configure a User Entry

To build a user database, you configure individual entries for users. Each entry defines the name or dialing pattern, dialing context, agent, and policy for a user. You can optionally add a description and tags to create and filter reports for users. A user can represent an account in your dialing network, such as an extension, a subscriber, or a phone number.

- Configure one or more agents.
- Configure one or more policies.
- Plan any tags that you want to use to create and filter reports for users.

In the following procedure you can add multiple policies and tags to this user record. Repeat the procedure for each user that you want to manually add to the user database.

1. Access the User Entries configuration object.
Click **Configuration, Service Provisioning, Users**.
2. On the Add User Entry page, do the following:

AoR	Enter the user's address of record. Default: Empty. Default: Empty. Valid values: Alphanumeric characters in the following formats: username@companyname.com username@ipaddress. See "Alphanumeric User Database Entries" for more information.
Name or pattern	Enter the E.164 number associated with this user using the same pattern rules as in dial plans. Do not include the + character.

Description	Enter a description of this user.
Dialing context	Select or enter the name of the context that defines the preferred dialing rules for this user.
Agent	Select or enter the name of the agent to which this user is connected. This agent is the closest agent to the user.
Policy	Add one or more policies to apply to this user.
Tags	Add one or more tags to this user entry.

3. Click **OK**.
4. Save the configuration.

Resolving to the Longest Match in the User Database

The Oracle Enterprise Communications Broker (OECB) user database supports resolution of overlapping numbers during lookups by selecting entries with the longest matches.

Flexibility in creating user database records allows for overlapping records, which can create ambiguity when the OECB looks for a match. To resolve such ambiguity, the OECB selects the entry that matches the most digits. As you build your user database, consider how you create user database entries to take advantage of flexible matching while preventing ambiguity.

The following table shows a set of overlapping user number entries in the Pattern column. The Match column shows the corresponding number of digits that must exactly match the pattern. Note that the match length does not include number ranges and wildcards.

Pattern	Match
17815551111	11
17815551[000-999]	8
17815551[111-999]	8
17815551xxx	8
1781555(x)xxx	7
xxxxxxxxxxx	0

When performing a user database lookup, the system uses the entry that matches the pattern with the longest match length. The following table explains how different dialed numbers match the numbers configured in the user database. The Dialed Number column lists examples of dialed numbers. The Matching Pattern column lists the configured pattern to match. The Reason column explains how the match does or does not work.

Dialed Number	Matching Pattern	Reason
17815551111	17815551111	This dialed number matches all patterns with an exact match and has the highest match length (11).

Dialed Number	Matching Pattern	Reason
17815551112	17815551[000-999] or 17815551[111-999] or 17815551xxx	This dialed number matches all but the first pattern, but the three patterns have the longest match length of 8. When multiple entries have the same match length, the selection between them is undefined and causes an unsuccessful lookup due to the ambiguity.
17815552111	1781555(x)xxx	This number matches only the last two patterns, but this pattern has a higher match length (7).
2222222222	xxxxxxxxxxx	This number matches only the last pattern.

8

Using Policy to Refine Routing

The Oracle Enterprise Communications Broker supports policy-based routing, allowing the user to select pre-defined policies or create their own policies and apply them to routes or the Registrar. In turn, these policies impact the behavior of the applicable routes when traffic matches user-defined conditions. The user configures new policies from the Oracle Enterprise Communications Broker's **Policy** icon (or uses pre-built policies) and applies them to routes and/or the Registrar. Routes support multiple policies.

Policy provides the Oracle Enterprise Communications Broker with a generic approach to configuring routing applications. Policy configuration assumes a desired behavior that has been identified by the user. The most common objectives include:

- Establish more specific routing decisions
- Apply additional services

The Oracle Enterprise Communications Broker abstracts policy behavior into three components including:

- Route
- Condition
- Action

The combination of route and condition define when the policy applies. The action refines the way in which the traffic uses the route. Note that possible actions may include not using the route.

This generic approach to route policy provides great flexibility in policy definition, but also imposes a level of complexity on the user, requiring them to:

- Identify the application they want to create.
- Determine how to identify the applicable traffic.
 - Use or create new routes specifically for the application.
 - Define the condition that causes the system to apply the policy to the route.
- Test traffic matching and application action.
- Ensure no overlapping configurations cause the system to use or not use the policy unexpectedly.

There are two components of a policy configuration:

- Conditions:
 - codec-condition—Tell the system to determine the presence of absence of specific codecs within an offer.
 - time-condition—Specify the day(s) of the week and time(s) of the day when the system uses the policy.
- Actions:

- routing-action—Tells the system to modify how the route is applied.
- redirect-action—Tells the system to direct traffic to the configured agent when the route and conditions match.
- outbound-translation-action—Tells the system to perform the configured outbound translation when the route and conditions match.

Policies allow for both multiple conditions and actions. All conditions must be met for the condition to be true. Alternatively, the user can configure no conditions, meaning the policy's condition is always true. When a policy includes multiple actions, it performs all of the actions in the configured order.

The user may also need to create route(s) specific to the policy. Whether an existing or newly configured route, the user configures the route to use one or more policies to complete the application configuration.

The Redirect Action

The redirect action causes the Oracle Enterprise Communications Broker (OECB) to redirect the incoming call through a particular agent by way of policy. You can use redirect to send a call to an external resource or service, such as a transcoding Session Border Controller or a call-recorder.

When applied, the policy engine performs an additional routing lookup for the call to the specified redirect agent. The system pre-pends additional hops from the redirect to the hops that were already calculated for the current route. The redirect action adjusts the routes to send the call to the specified redirect agent first, and then to the call destination.

Redirect action configuration includes the **Hairpin signaling** field. When you enable **Hairpin signaling**, the OECB routes the call to the redirect agent first, then routes it back to the OECB before sending it to the original destination. Hairpin signaling ensures that the OECB can route the call even when the redirect agent cannot reach the final destination. Note that keeping **Hairpin signaling** disabled eliminates the extra hops and extra session required when the destination is reachable by the redirect agent.

The OECB uses the same source agent, calling number, and called number parameters as the original call to reach the redirect agent. Only the dest-agent parameter gets replaced with the redirect agent specified in the redirect policy action. Take special care with default routes or routes that use a wildcard in the dest-agent field because such routes can become part of the path to the redirect agent.

Note the following details when evaluating and configuring redirect action:

- Hops incurred by the redirect action do not affect the route cost. The system determines the route-set and order-set before the redirection takes place.
- The system does not evaluate policies applied to the redirect routes, which prevents redirection loops and other undesirable behavior.
- The system uses only the first (lowest cost) redirection path, which prevents the exponential increase of backup paths.
- You can configure redirection to agent groups, which operate normally.
- The system applies the same routing parameters of the call (source agent/number and dest number) to the redirection route lookup as the original route.

- The system does not use default routes (* for all match patterns) for redirection. When the OECEB finds no valid routes for the redirect, the OECEB rejects the call.

Configuring CNAM Replacement

The Oracle Enterprise Communications Broker provides the user with the ability to specify the value of the caller name (CNAM) in the FROM header. A simple use case would consist of an enterprise inserting the name of their company into the FROM value, in place of the original caller name. The user configures the system to use other policy or routing configurations to determine when to replace a CNAM. The system applies this policy action on SIP requests immediately prior to egress.

To configure CNAM replacement, add the **cnam-masking-action** action to the desired policy. The **Modify Policy / cnam masking action** dialog includes two fields.

- **Name**—Assigns a name as an identifier to your action.
- **Display name**—Defines the text the Oracle Enterprise Communications Broker inserts as the CNAM value in the FROM header.

Note that, if no user name is in the original FROM, the Oracle Enterprise Communications Broker inserts text configured via the CNAM mask and encloses it in brackets per RFC 2822.

Using Policy to Normalize SIP Headers

The Oracle Enterprise Communications Broker supports policy-based SIP Header Normalization, allowing the user to copy and change information in headers when their user parts are Tel-URIs or SIP URIs composed of numbers. The system writes header changes caused by policy after any inbound and before any outbound manipulation performed by header manipulation rules. These policies work for registered users and targets derived from the user database or LDAP. The user configures header normalization policies from the Oracle Enterprise Communications Broker's **Policy** icon and applies them to routes and/or the Registrar. Routes support multiple policies.

The header normalization policy works with the existing outbound translation policy. Each policy consists of one or more rules by which the system changes headers to messages that match the routes or registrar to which the policy is configured. Multiple rules can be defined for the same SIP header. The system evaluates and executes rules in the configured order and changes the header values with each rule that has a valid **New value** field.

Header normalization configuration fields include:

- **SIP header name**—The name of the SIP header to be normalized. This field cannot be empty.
- **Dialing context**—Name of dialing context that defines the dialing rules to be applied to the phone number in this SIP header. An empty value in this field indicates that no dialing rules need to be applied to the corresponding SIP header in the rule.
- **Result name**—A temporary variable name to store the result of the dialing rules on the SIP header. This name has to be unique within this policy action. The value is saved between policy rule evaluation, but is not saved after the policy is fully evaluated.

- **New value**—The **Result name** whose value will be used as the new value of the SIP header. If this field is left empty, the system does not change the value of the SIP header.

If a rule specifies a SIP header name that is absent in the SIP request and the **New value** field has a valid **Result name**, the system adds the SIP header and set it to the value of **Result name**.

ANI Masking

The Oracle Enterprise Communications Broker provides a means for ensuring that the automatic number identification (ANI) presented to a service provider be recognized as a valid screened telephone number (STN) and, therefore, is not dropped by that service provider. The user configures this function as a Header Normalization policy that rewrites the applicable header with a recognizable STN. Applicable headers are DIVERSION, P-ASSERTED-IDENTIFY or FROM, depending on the service provider's requirements.

ANI Masking Configurations

The following example shows a header normalization policy that ensures that the system presents an Automatic Number Identification (ANI) as a valid Screened Telephone Number (STN). The example assumes that you created a dialing context from which the system can determine an STN.

Policy

Name	Description
ANI_Mask1	North American international dialing

Condition

The default condition, which requires no configuration, is to always apply the policy.

Actions

The following table shows an example configuration for the action named ANI_MaskforSP1, which is of the header-normalization-action element type with two actions. The first row in the table shows the SIP header name as "From," the dialing context, and the result name. The second row shows the SIP header name as "Diversion," the result name, and the new value.

SIP header name	Dialing context	Result Name	New value
From	find_Verizon_STN	STN	
Diversion		original_diversion	STN

The results of the actions in the preceding table follow.

1. Use the value of the "from" header and run it through a dialing context called "find_Verizon_STN" and store the result in a variable called "stn". Because the new value field is empty the value of the "from" header is unaltered.
2. Because the dialing context is empty, copy value of the "Diversion" header to the variable called "original_diversion". If the incoming SIP request did not have a

"Diversion" header "original_diversion" will be set to an empty string. Copy the value of the variable "STN" into the "Diversion" header.

Route and Registrars

Configure either routes or registrars as triggers by setting their **Policy** fields to the **ANI_Mask1** policy.

Results

The following table shows the results of this example configuration. The SIP Message Text column shows the SIP To and From addresses. The Mask Function column shows the dialing context used to create the new value. The Result SIP Message Text column shows the SIP To, From, and Diversion addresses.

SIP Message Text	Mask Function	Result SIP Message Text
To: sip: 781.630.1111@oracle.com	Use Dialing context titled "find_Verizon_STN" to create new value titled "STN"	To: sip: 781.630.1111@oracle.com
From: sip: 781.630.2222@oracle.com		From: sip: 781.630.2222@oracle.com
		Diversion: sip: 978.528.1234@oracle.com

Define a Policy

The Policy configuration object the Oracle Enterprise Communications Broker (OECB) provides access to the policy list and configuration dialogs, where you can define and manage policies.

- Analyze traffic patterns and message content before configuring a custom policy.
- Identify the message contents and metadata that allows a policy to uniquely target the traffic.
- Configure any agents the you want to use for redirected calls.

Use the following procedure to add a new policy or modify an existing one.

1. Access the Policy Entries configuration object. **Configuration, Service Provisioning, Policy Entries.**
2. On the Policy Entries page, click **Add**, and do the following:

Parameters	Instructions
Name	Enter a name for the policy.
Description	Enter descriptive text that explains the purpose of the policy.
Conditions	Set the conditions under which the system performs the actions that you specify.

Parameters	Instructions
	<p>For Codec Condition, do the following:</p> <ol style="list-style-type: none">a. Click Add.b. Click Codec Condition.c. On the Add Policy / Codec Condition page, do the following:<ol style="list-style-type: none">i. Name—Enter a name for the Codec Condition.ii. Do one or both of the following:<ol style="list-style-type: none">i. Contains Codecs—Click Add, select a codec from the drop-down list, and click either OK or Apply/Add another.ii. Missing Codecs—Click Add, select a codec from the drop-down list, and click either OK or Apply/Add another.d. Click OK. <p>For time-condition, do the following:</p> <ol style="list-style-type: none">a. Click Add.b. Click Time Condition.c. On the Add Policy / Time Condition page, do the following:<ol style="list-style-type: none">i. Name—Enter a name for the Time Condition.ii. Days—Click Add, select a day from the drop-down list, and click either OK or Apply/Add another.iii. Start time—Set the time when you want the condition to start on the specified day.iv. End time—Set the time when you want the condition to end on the specified day.d. Click OK.
Actions	Set the one or more actions that you want the system to perform for each specified condition.

Parameters	Instructions
	<p data-bbox="987 233 1365 254">For Routing Action, do the following:</p> <ul style="list-style-type: none"><li data-bbox="987 279 1143 300">a. Click Add.<li data-bbox="987 325 1252 346">b. Click Routing Action.<li data-bbox="987 371 1398 426">c. On the Add policy / Routing Action page, do the following:<ul style="list-style-type: none"><li data-bbox="1032 447 1455 468">i. Enter a unique name for the action.<li data-bbox="1032 493 1409 548">ii. Select a routing mode from the drop-down list.<li data-bbox="987 569 1133 590">d. Click OK. <p data-bbox="987 615 1377 636">For Redirect Action, do the following:</p> <ul style="list-style-type: none"><li data-bbox="987 661 1143 682">a. Click Add.<li data-bbox="987 707 1260 728">b. Click Redirect Action.<li data-bbox="987 753 1406 808">c. On the Add policy / Redirect Action page, do the following:<ul style="list-style-type: none"><li data-bbox="1032 829 1425 884">i. Name—Enter a unique name for the action.<li data-bbox="1032 905 1458 984">ii. Redirect to Agent—Select an agent to redirect the call to before sending the call to the destination.<li data-bbox="1032 1005 1458 1119">iii. Hairpin Signaling—Select to enable rerouting a call back to the OECB from a redirect agent before sending the call to the destination.<li data-bbox="987 1140 1133 1161">d. Click OK.

Parameters	Instructions
	<p>For Outbound Translation Action, do the following:</p> <ol style="list-style-type: none"><li data-bbox="906 306 1065 333">a. Click Add.<li data-bbox="906 352 1383 989">b. On the Add policy / Outbound Translation Action page, do the following:<ol style="list-style-type: none"><li data-bbox="954 457 1349 512">i. Name—Enter a unique name for the action.<li data-bbox="954 531 1360 611">ii. Egress Number Translation Mode—Select a mode from the drop-down list.<li data-bbox="954 630 1377 800">iii. Number of Digits for n Digit Dialing—If you selected n-digit-dialing as the number translation mode, enter the number of digits the OECB must send to the agent. Range: 0-25.<li data-bbox="954 819 1377 989">iv. Prepend Prefix on Egress—Specify the number of digits to prepend to the outbound number after translation. Valid entires include telephony digits 0-9, star, pound, and A-D. Max: 25 digits.<li data-bbox="906 1008 1057 1035">c. Click OK.
	<p>For Constraints Action, do the following:</p> <ol style="list-style-type: none"><li data-bbox="906 1104 1065 1131">a. Click Add.<li data-bbox="906 1150 1383 1383">b. On the Add policy / Constraints Action page, do the following:<ol style="list-style-type: none"><li data-bbox="954 1230 1349 1285">i. Name—Enter a unique name for the action.<li data-bbox="954 1304 1317 1383">ii. Ignore Constraints—Select to ignore call admission control restraints. Default: Disabled.<li data-bbox="906 1402 1057 1430">c. Click OK.

Parameters	Instructions
	<p data-bbox="984 226 1453 289">For Header Normalization Action, do the following:</p> <ol data-bbox="984 304 1453 1144" style="list-style-type: none"><li data-bbox="984 304 1453 331">a. Click Add.<li data-bbox="984 346 1453 430">b. On the Add Policy / Header Normalization action page, do the following:<ol data-bbox="1031 451 1453 1144" style="list-style-type: none"><li data-bbox="1031 451 1453 514">i. Name—Enter a unique name for the action.<li data-bbox="1031 525 1453 588">ii. Header Normalization rules—Click Add, and do the following:<ol data-bbox="1079 598 1453 1144" style="list-style-type: none"><li data-bbox="1079 598 1453 724">i. Header Name—Enter the name of the SIP header to normalize with dialing rules. Required.<li data-bbox="1079 735 1453 819">ii. Dialing Context—Select a dialing context from the drop-down list.<li data-bbox="1079 829 1453 955">iii. Result Store—Specify the temporary variable name to store the result of the dialing rules on the SIP header.<li data-bbox="1079 966 1453 1144">iv. New Value—Enter the result name to use as the new value in the SIP header. The system does not change the value of the SIP header unless you specify a value. <li data-bbox="984 1155 1453 1182">c. Click OK. <p data-bbox="984 1207 1453 1234">For CNAM Masking Action, do the following:</p> <ol data-bbox="984 1249 1453 1564" style="list-style-type: none"><li data-bbox="984 1249 1453 1276">a. Click Add.<li data-bbox="984 1291 1453 1375">b. On the Add policy / CNAM Masking Action page, do the following:<ol data-bbox="1031 1375 1453 1564" style="list-style-type: none"><li data-bbox="1031 1375 1453 1438">i. Name—Name—Enter a unique name for the action.<li data-bbox="1031 1449 1453 1564">ii. Display Name—Enter the caller name replacement value to display in the SIP From header. For example, your company name.<li data-bbox="984 1575 1453 1602">c. Click OK.

3. Click **OK**.
4. Save the configuration.

You must apply the policy to the Registrar and any route that you want by way of the OECB Registrar or route configuration dialogs.

Applying a Policy to a Route

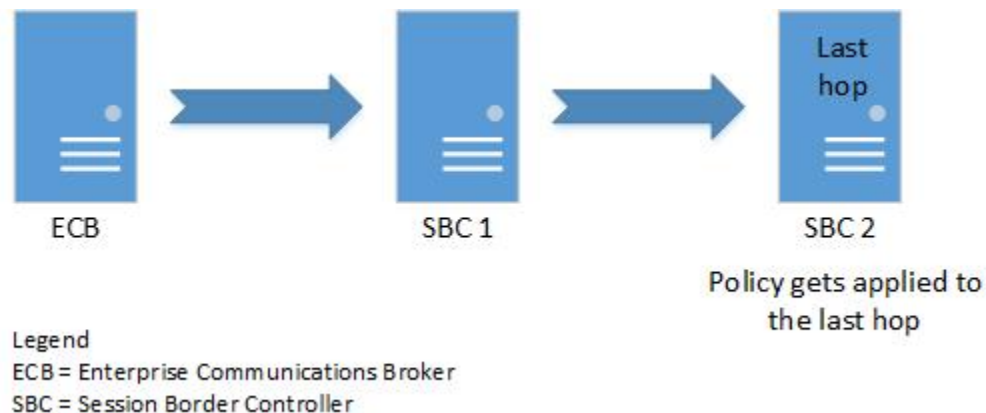
The user configures an Oracle Enterprise Communications Broker route's Policy from the **Modify Routing entry** dialog. The dialog presents a list box titled **Policy**, with controls that allow the user to **Add**, **Edit** or **Delete** policies to the route. When the user clicks **Add**, the system presents a drop-down selection box displaying the names of all currently configured Policies. This list includes user-configured policies, as well as the system's pre-configured policies, Deny, Emergency and Stop-Recurse. The user selects the desired policy and clicks **OK**, **Add/Apply another** or **Cancel** to complete the procedure.

Runtime Routing with Policies in the User Table

You can apply a policy to the Oracle Enterprise Communications Broker (OECB) user table for more granular control over where a policy gets applied to runtime routing decisions.

Applying a policy to the user table provides more granularity than the routing table can provide because user table entries allow individual phone numbers and phone number ranges.

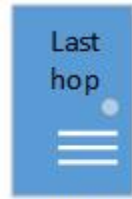
The OECB applies policies from the user table to the last hop on the route list specified in the routing table.



Note:

The system uses the policy applied to the called number from the user table.

You can use policies from the user table and the routing table together. When used together, the system obeys the policy in the user table first and the routing table second.



SBC 2

Policy priority

1. User table policy
2. Routing table policy

When you configure a policy in the User database for a particular entry, the OECB applies the policy to all agents for that entry including any Additional Target Groups among those agents. When the session agent is a Session Agent Group, the OECB applies the policy to the session agent of that Session Agent Group.

To apply a policy to the user table:

1. Add the policy to the OECB.
2. Add the policy to a user entry in the user table.

Applying a Policy to the Registrar

The user configures the Oracle Enterprise Communications Broker Registrar with a Policy from the **Modify Registrar settings** dialog. The dialog presents a list box titled **Policy**, with controls that allow the user to **Add**, **Edit** or **Delete** policies to the registrar. When the user clicks **Add**, the system presents a drop-down selection box displaying the names of all currently configured Policies. This list includes user-configured policies, as well as the system's pre-configured policies, Deny, Emergency and Stop-Recurse. The user selects the desired policy and clicks **OK**, **Add/Apply another** or **Cancel** to complete the procedure.

Configurations Using Policy

Application examples using policy as a primary construct are presented below. These application examples are valid for both the Small and Large Enterprise models as base configurations.

Configuration examples include:

- Priority Call Handling, Using Constraint Policy Action
- Transcoding, Using Redirect Policy Action
- Call Recording, Using Redirect Policy Action
- Domestic vs. International Call, Using Translation Action
- Deny Routes, Using Routing Action
- Stop Recurse Routes, Using Routing Action
- Skip Routes, Using Routing Action

Priority Call Handling

The Oracle Enterprise Communications Broker provides support for ensuring that high priority calls, such as 911 calls, are not constrained by system utilization thresholds. Non-priority calls are still subject to system constraint configuration. The user configures priority call behavior by creating a policy with the constraint-action, then applying that policy to the applicable routes. The Oracle Enterprise Communications Broker applies this policy to the configured route, as well as every subsequent backup route, even if the backup routes are not configured with the policy.

The system ignores constraints, such as session and rate/burst limits, for calls that match the priority route. The **constraints-action** provides a built-in action called **IgnoreConstraints**. This action causes the policy to mark a call to ignore constraints. All conditions configured on the policy must be satisfied for the action to be applied. The constraints-action is special in that, when enabled, it sets the call to ignore constraints for ALL routes. This is necessary because backup routes may be utilized that are not specific to priority/emergency calls.

For example, an emergency call might first be routed to a 911 service, but on failure it would be sent to the PSTN. The call cannot be limited by constraints on the PSTN route.

The built-in **Emergency** policy is provided for user convenience, and uses the **IgnoreConstraints constraints-action**. This built-in policy can be edited, if desired.

Note that this setting does not affect the "Deny" and "StopRecurse" policies on routes. It is simply not known whether these policies are for constraining purposes or because the network infrastructure cannot support calls over those routes.

Priority Call Configurations

This example shows a policy configuration that excludes 911 calls from system constraints, ensuring that the system does not encumber the call delivery. The following configuration documents the built-in policy titled "Emergency" applied to a route for 911 calls.

Name	Emergency
Description	Built-in policy to ignore constraints for emergency or priority calls.
Condition	The default condition, which requires no configuration, is to always apply the policy.
Action	<ul style="list-style-type: none"> Enter a name for this action. Set Ignore Constraints.

Example Route

In addition to the preceding configuration, the system needs the following route.

Source Agent	Calling #	Destination Agent	Called #	Route	Cost	Policy
*	*	*	911	EmrSvc	0	Emergency

Transcoding and the Oracle Enterprise Communications Broker

Transcoding is the conversion of media streams' encoding between endpoints that use different codecs. The Oracle Enterprise Communications Broker allows the user to configure routing policies that identify session agents deployed for transcoding media sessions and redirect applicable signaling and media streams to those agents.

Transcoding Configurations

This example shows a policy configuration that identifies an offer that does not include PCMU or PCMA and routes the call to a Transcoding device, in this scenario an SBC, to ensure that the media uses either PCMU or PCMA.

Name	XcodePcmuPcma
Description	Send calls that need transcoding to PCMU or PCMA to a transcoding SBC
Condition	<ul style="list-style-type: none"> Name—XcodePcmu Contains codecs—Intentionally blank PCMU
Action	<ul style="list-style-type: none"> Name—XcodeRedirect Redirect to Agent—XcodeSBC Hairpin signaling—Set to enabled

Example Route

In addition to the preceding configurations, the system includes the following route.

Source Agent	Calling #	Destination Agent	Called #	Route	Cost	Policy
*	*	PBX	*	PBX	0	XcodePcmu

This configuration results in the system sending traffic that contains the PCMU codec to the PBX, and traffic without the PCMU codec to the XcodeSBC, back to the Oracle Enterprise Communications Broker and then to the PBX.

Multiple Outbound Translations

The Oracle Enterprise Communications Broker provides a means for refining outbound translation configurations using routing policy. Fixed outbound translation configuration is available at the agent level. Routing policy, however, includes an outbound translation action that takes precedence over agent and sip interface configuration and applies translation based on individual route matches.

Outbound Translation Configurations

This example shows an outbound translation configuration that configures E164 numbers for domestic calls, and E164-no-plus with 011 prepended for international calls. The example calls for two policies and two routes.

Name	Policy 1—InternationalDialNA Policy 2—DomesticDialNA
Description	Policy 1—North American international dialing Policy 2—North American domestic dialing
Condition	The default condition, which requires no configuration, is to always apply the policy.
Actions	<p>Outbound Translation Action (InternationalDial)</p> <ul style="list-style-type: none"> • Name—InternationalDial • Egress number translation mode—E164-no-plus • Number of digits for n digit dialing—0 • Prepend prefix on egress—011 <p>Outbound Translation Action (DomesticDial)</p> <ul style="list-style-type: none"> • Name—DomesticDial • Egress number translation mode—E164-no-plus • Number of digits for n digit dialing—0 • Prepend prefix on egress—Intentionally blank

Routes

In addition to the dial patterns in this example, the system needs the following two new routes.

Route #	Source Agent	Calling #	Destination Agent	Called #	Route	Cost	Policy
1	*	*	PSTN-NA	11*	PSTN-NA	0	InternationalDialNA
2	*	*	PSTN-NA	1*	PSTN-NA	10	DomesticDialNA

Results

The configuration in this example provides the results shown in the following table.

From	Dial String	Transformation	Result
*	15551234	1555123415551234	Call is routed without dial string change
*	34555555	11345555550	Call is routed with dial string change

The system routes any call that includes a country code and does not begin with the digit "1" internationally, and with the applicable transformation.

Routing Action Configurations

The Oracle Enterprise Communications Broker includes three pre-configured **Policies** (based on the pre-configured **DenyAction**, **StopRecurseAction** and **IgnoreConstraints** routing actions) that the user can apply to routes. In addition, the pre-configured **Skip** routing action mode is available for easy Policy configuration, which the user can then apply to routes.

Modes for routing actions include:

- **Deny**—The Oracle Enterprise Communications Broker should not forward the call at all.
- **StopRecurse**—Stop trying backup routes if the current route fails.
- **Skip**—Do not try this route, based on condition; proceed to any backup options.

Deny Route Policy Configurations

This example shows a policy configuration that prevents the system from allowing the request's source to reach the destination. The configuration shown here documents the built-in policy titled "Deny" applied to an example route.

The **Deny** policy in any resultant SIP routes to a destination prevents the Oracle Enterprise Communications Broker (OECB) from forwarding a SIP request to the destination point in question. You can use the **Deny** policy to prevent sessions between two endpoints for policy or cost reasons.

The OECB includes a pre-configured deny route policy that you can use without modification.

Name	Deny
Description	Built-in policy to deny the incoming session
Condition	The default condition, which requires no configuration, is to always apply the policy.
Action	Name—DenyAction Routing Mode—deny

Example Route

In addition to the configurations in this example, consider the following policy applied against this route.

Source Agent	Calling Number	Destination Agent	Called Number	Route	Cost	Policy
Class_B	*	Protect_Agent	*	Transit_Agent	0	Deny

This route entry is designed to prevent calls passing through the This route entry may be installed as part of one or multiple routes that could potentially reach

`Protect_Agent`. The routing engine recognizes the presence of this route entry and rejects the call.

In addition, the system applies this Deny regardless of the backup route on which it is specified.

Stop Recurse Route Policy Configurations

This example shows a policy configuration that prevents the system from recursing through ensuing backup routes if the route configured with this policy stops responding. The configuration shown here documents the built-in policy titled "StopRecurse" applied to an example route.

A Stop-Recurse hop stops further attempts to forward a SIP request to a destination after the system tries the route with the Stop-Recurse hop. You can use the Stop-Recurse hop to prevent calls from being forwarded on certain routes that are cost-prohibitive or might cause loops in SIP call flows.

The Oracle Enterprise Communications Broker (OECB) includes a pre-configured stop recurse policy, with Routing Action applied.

Name	StopRecurse
Description	Built-in policy to prevent further backup route attempts.
Condition	The default condition, which requires no configuration, is to always apply the policy.
Action	Name—StopRecurse Routing Mode—stop-recurse

Example Route

To expand upon the example, consider the following route.

Source Agent	Calling Number	Destination Agent	Called Number	Route	Cost	Policy
Class_A	*	Protect_Agent	*	Transit_Agent	0	StopRecurse

This route allows traffic to reach `Protect_Agent` as long as it originates by way of the agent named `Class_A`.

For example, consider the scenario where an endpoint can be reached using two routes.

- Route 1: Agent_1 to Agent_2 (**StopRecurse**) to PBX
- Route 2: Agent_1 to PSTN

Route 1 uses a **StopRecurse** policy defined on the hop between Agent_2 and PBX. The OECB stops processing the call if Route 1 does not receive a successful response. This configuration can prevent calls initially targeted for Route 1 from using the PSTN.

Stop Recursion by SIP Response Code

The Stop Recurse parameter provides you with control over how the system allows or stops recursion on routing by allowing you to set the specific response codes that you want the system to act upon. You can control recursion by response code globally through the SIP Interface configuration or locally through an agent or agent group configuration. Valid response code values range from 300-599. You can enter individual response codes separated by a comma, such as 301,305 or a range such as 300-380. The system uses response codes 401 and 407 for the defaults.

Skip Route Policy Configurations

This example shows a policy configuration that prevents the system from using this route based on condition. This policy contrasts with the StopRecurse policy because the routing engine is free to recurse through other routing options after skipping.

The Oracle Enterprise Communications Broker includes a pre-configured Skip Action, which you can select to define a policy.

Name	MySkip
Description	Do not use this route, depending on condition.
Condition	The following Time Condition Fields activate the skip policy on weekdays from 9am to 5pm. Name—When to enforce MySkip Days—Monday, Tuesday, Wednesday, Thursday, Friday Start time—09:00:00 End time—16:59:59
Action	Name—SkipRoute Routing mode—skip

Example Route

In addition to the preceding configurations, this example applies the following policy against the target routes.

Source Agent	Calling Number	Destination Agent	Called Number	Route	Cost	Policy
Class_B	*	Protect_Agent	*	Transit_Agent	0	MySkip

The resulting configuration prevents the system from using this route to **Transit_Agent** as a means of reaching **Protect_Agent** on weekdays from 9 to 5.

9

Routing Configuration

The Oracle Enterprise Communications Broker performs session routing via its route configuration. Route configuration establishes hop-by-hop paths to signaling endpoints.

End stations may or may not be known by the Oracle Enterprise Communications Broker; endpoints configured within the user database are known, all others are not. Whether or not they are known by the system, the last hop (agent) leading towards that end station is often known. For this reason, the Oracle Enterprise Communications Broker builds its hop list by starting with the last agent it knows in the path and recursively adding hops (agents) needed to get to that hop. In the case where the last hop is not known, the system provides its last known hop with endpoint information and allows unknown hops to try to find the endpoint.

Oracle Enterprise Communications Broker routing configuration allows the user to specify a route's cost to specify route preference. Cost may or may not be based on monetary considerations. But the reach of an enterprise's network often does allow the user to configure routes that keep session traffic within the enterprise infrastructure rather than incurring cost associated with a service provider.

The Oracle Enterprise Communications Broker allows for a range of route preference criteria to differentiate between routing paths. Criteria includes source routing based on the agent or calling number. Target-oriented criteria is also available, allowing the enterprise to designate preferred paths for specific called numbers.

Routing Fields

The **Routing Table** configuration displays the route list.

To add routes, click **Add**. The Oracle Enterprise Communications Broker displays the **Routing Table** configuration.

 **Note:**

The routing engine finds the optimum route based on the Source and Destination agents. Oracle recommends avoiding wild cards in both Source and Destination Agents. When these values are configured as wild cards, the ECB must check the entire routing table for a match and this negatively impacts performance.

The following list describes the fields available in the **Add** dialog and the corresponding configuration instructions.

Source Agent	Select the agent from which the traffic must come to match the route. The default is the wildcard *, meaning to match traffic from any agent.
--------------	---

Calling Number	Type in the number from which the traffic must come to match the route. A valid entry includes numeric characters or an FQDN resolvable through ENUM. The default is the wildcard *, meaning to match traffic with any calling number.
Dest Agent	Select the agent to which the traffic must be targeted to match the route. The default is the wildcard *, meaning to match traffic to any agent.
Called Number	Type in the number to which the traffic is targeted to match the route. A valid entry includes numeric characters or an FQDN resolvable through ENUM. The default is the wildcard *, meaning to match traffic with any called number.
Route	Select the agent that is the next hop in the route path.
Cost	Enter a cost associated with this route to specify route use preference when there are multiple routes to the same destination. A valid entry is numeric ranging from 0 to 100, with the lowest number (cost) being the preferred route.
Description	Enter any descriptive text you find helpful in identifying this route.
Tags	Enter one or more tags.

Route Policy

The Oracle Enterprise Communications Broker (OECB) includes a route configuration control called **Policy** that allows you to alter the behavior of routes. As the OECB assembles hops together to create a complete route, it considers and acts on the policies configured on each route entry.

You configure a route **Policy** from the Modify Routing entry dialog. The parameter values include:

Policy [Deny | StopRecurse]

The default setting is empty, which imposes no policy on the route.

The **Deny** policy in any resultant SIP routes to a destination prevents the OECB from forwarding a SIP request to the destination point in question. You can use the **Deny** policy to prevent sessions between two endpoints for policy or cost reasons. Consider this route entry.

Source Agent	Calling Number	Destination Agent	Called Number	Route	Cost	Policy
Class_B	*	Protect_Agnt	*	Transit_Agnt	0	Deny

The preceding route entry is designed to prevent calls passing through the agent named `Class_B` from reaching any endpoint behind `Protect_Agnt`. This route entry

may be installed as part of one or multiple routes that could potentially reach `Protect_Agent`. The routing engine recognizes the presence of this entry and rejects the call.

To expand upon the example, consider this route.

Source Agent	Calling Number	Destination Agent	Called Number	Route	Cost	Policy
Class_A	*	Protect_Agent	*	Transit_Agent	0	StopRecurse

The preceding route configuration allows traffic to reach `Protect_Agent` as long as it originates through the agent named `Class_A`.

The **StopRecurse** policy stops further attempts to forward a SIP request to a given destination if that route stops responding. The **StopRecurse** policy can prevent the system from forwarding calls on routes that are cost-prohibitive or might cause loops in SIP call flows.

For example, consider the scenario where an endpoint can be reached using two routes.

- Route 1: Agent_1 to Agent_2 (**StopRecurse**) to PBX
- Route 2: Agent_3 to PSTN

Route 1 uses a **StopRecurse** policy defined on the hop between Agent_2 and PBX. The OECB stops processing the call if Route 1 does not receive a successful response. This configuration can prevent calls initially targeted for Route 1 from using the PSTN.

Deny Patterns in Route Parameter Syntax

The Oracle Enterprise Communications Broker (OECB) allows you to configure routes that refine the routing engine's behavior, based on the called and calling number. Such configuration prevents the routing engine from using that route entry as a member of the applicable route sets.

You configure this behavior by creating a route that prepends the called and calling numbers with an exclamation point (!). Configure the digits following the exclamation point as the first digits in the string, not the entire string. The following table shows an example of a route entry using a deny pattern.

Source Agent	Calling Number	Destination Agent	Called Number	Route	Cost	Policy
Agent_1	!123	*	*	Agent_2	0	

In the preceding route example, traffic that includes a calling number starting with the digits 123 does not match this route. The OECB does not use this route in any applicable route set.

You can also configure both calling and called numbers with the deny format to establish an "and" condition to the route.

Source Agent	Calling Number	Destination Agent	Called Number	Route	Cost	Policy
Agent_1	!123	*	!456	Agent_2	0	

In the preceding route example, traffic that includes a calling number starting with the digits 123 and a called number starting with the digits 456 does not match this route.

Conversely, the "and" condition allows the following traffic originating from `Agent_1` by way of `Agent_2` to match this route:

- From any calling number that does not start with the digits 123 to any called number.
- From any calling number that starts with the digits 123 to any called number other than one that starts with 456.
- To any called number that does not start with the digits 456.
- To any called number that starts with the digits 456 from any called number other than one that starts with 123.

Loop Sensing for PSTN Calls

When the Oracle Enterprise Communications Broker (OECB) routing table contains no explicit route to the agent, the OECB routing engine uses an implicit route that assumes the OECB is directly available to the agent. Depending on your routing needs, you might choose to configure a default route to handle calls not routed by the implicit routes. For some calls, the default route will result in a loop back to the previous hop. Such looping can cause undesirable traffic in the network and result in calls that never reach the end point. To prevent call looping, use the **next-hop-policy-condition** parameter to construct a policy for the default route that prevents the OECB from sending a call back to the previous hop.

Using the **next-hop-policy-condition** in a policy, in combination with a skip action, removes the default route as a choice for some calls and prevents the OECB from forwarding a call:

- back to the agent that sent the call.
- to a group of agents when the agent that sent the call is a member of the group.
- back to the agent when the IP address matches the host in the from-uri of the received message.
- to a group of agents when any agent in the group uses an IP address and port that matches the host in the from-uri of the received message.

The **next-hop-policy-condition** compares the next hop in the route to either the **previous hop** or the **from-uri-host**. When either **previous hop** or **from-uri-host** resolves to true, the OECB executes the **deny** action to stop call looping.

The process for configuring call loop prevention requires the following steps:

1. Set the **next-hop-policy-condition** in a policy, along with the comparison type and the deny action.
2. Add the policy to the target route.

Configure Loop Sensing for PSTN Calls

To prevent call looping, use the **next-hop-policy-condition** parameter to construct a policy for the default route that prevents the Oracle Enterprise Communications Broker (OECB) from sending a call back to the previous hop when the agent does not respond.

1. Access the Policy Entries configuration object. **Configuration, Service Provisioning, Policy Entries.**
2. On the Policy entries page, click **Add**, and do the following:

Name	Enter a name for this policy.
Description	(Optional) Enter a description of the policy.
Conditions	Click Add , select next-hop-compare-condition , and do the following: <ol style="list-style-type: none"> a. Enter a name for this condition. b. Select the next hop compare mode that you want from the drop-down list. c. Click OK.
Actions	Click Add , and do the following: <ol style="list-style-type: none"> a. Select routing-action from the drop down list. b. Enter a name for this routing action. c. Select deny from the drop-down list. d. Click OK.

3. Click **OK**.
4. Save the configuration.
 - Apply the policy to the target route.

10

Registrar Configuration

When enabled, the Oracle Enterprise Communications Broker's registrar provides location service and registration authentication functions. The user must decide whether to use authentication and, if so, which authentication resource to utilize. Should the user decide to use a local authentication resource, they configure it via Local Subscriber Tables (LST), available from the registrar configuration. Configure external authentication resource configuration via the LDAP configuration dialogs.

Registrar Configuration Fields

To configure the Oracle Enterprise Communications Broker to act as a SIP Registrar:

1. Access the SIP Registrar configuration object.

Configuration, System Administration, SIP Registrar.

2. On the Settings page, do the following:

State	Select to use this SIP registrar configuration element.
Domains	Enter one or more domains that this configuration element will invoke SIP registration for. Wildcards are valid for this parameter. Multiple entries can be entered in quotes, separated by commas.
Minimum Register Expiration	Enter the expire time in seconds to be used in the REGISTER. Default: 300 (5 minutes). Valid values: 0-999999999
Policy	Enter one or more policies that you want to use to alter the processing of routes to a destination with registered endpoints.
Additional Target Group	List additional targets for the SIP registrar.
Fork Group	Select a number from 1-100 for the fork group registrar endpoints.
LST File	Enter the file name of the Local Subscriber Lookup table to use for authentication and user lookups.
Authentication Method	Set the method for the registrar to use to authenticate incoming registrations.

3. Save the configuration.

Local Subscriber Table

A local subscriber table (LST) is an XML formatted file that contains one or more usernames associated with a hash as encrypted or plaintext. The LST is saved locally on the Oracle Enterprise Communications Broker's file system.

LSTs enable a standalone Oracle Enterprise Communications Broker node or high-availability (HA) pair to forego relying on an external user database. Thus the Oracle Enterprise Communications Broker does not need to communicate with a server to authenticate users. This can eliminate the operational complexity of deploying a highly available credential storage system.

LST Configuration

To configure the Oracle Enterprise Communications Broker to use LSTs for authentication, you need to create a local subscriber table configuration element that identifies that LST. The LST must include users with minimum configuration of user name and password. Alternatively, an LST entry can include an AOR and a universal number. If there is no AOR, the username is assumed to be the AOR. The universal number field assigns a universal number to all contacts registered to the AOR.

You have the option of setting the registrar to authenticate. When messages requiring authentication are received and processed by the sip registrar, the Oracle Enterprise Communications Broker uses the identified LST for authentication.

In a local subscriber table configuration, you must define an object **name**. If the filename is entered without a path, the Oracle Enterprise Communications Broker looks in the default LST directory, which is /code/lst. If the LST file is located elsewhere on the Oracle Enterprise Communications Broker, you must specify the filename and absolute path.

When the registrar configuration includes a reference to an LST, the registrar uses it as its user list. The configuration may or may not include digest authentication functionality, depending on user configuration. Additional registrar configuration includes setting the **digest realm** appropriately (this is required for authentication), and setting the hash secret. At this point you may save and activate your configuration.

Unencrypted passwords for each user in the table is computed with the MD5 hash function as follows:

```
MD5(username:digest-realm:password)
```

Configure the Registrar with an LST

Define the registrar for using the Local Subscriber Table (LST) for registration authentication as opposed to an external resource, or to accept registrations without authentication.

1. Access the SIP Registrar configuration object.
Configuration, System Administration, SIP Registrar.
2. On the Settings page, click **Manage LST**, and do the following:

LST File	Specify the LST file for this registrar. Choose an existing LST file from the drop-down list.
Manage LST (Use to create an LST file.)	Do the following: <ul style="list-style-type: none"> a. File Name—Enter a filename for your new LST XML file. If you do not specify a path is given, the Oracle Enterprise Communications Broker looks in the /code/lst directory. Provide the complete path, if the file is located elsewhere. b. Digest Realm—Enter the name (Realm ID) of the host realm initiating the authentication challenge. This value defines the protected space in which the digest authentication is performed. Valid value is an alpha-numeric character string. c. Encrypt File—Select o cause the system to encrypt the file. d. Encryption Secret—Click Set to display the Set Encryption secret dialog. Enter and then confirm the secret used in encryption and decryption of the passwords in the XML file. Once saved, the system does not echo this value back to the screen in plaintext format. e. Click OK—This creates your LST file and allows you to add subscriber entries. f. Click the Add button. The system displays the Add Local Subscriber Entry dialog. Enter Username, Password and AoR for this subscriber.
Authentication Method	Select LST from the drop-down list.
Digest Realm	Enter the name (realm ID) of the host realm initiating the authentication challenge. This value defines the protected space in which the digest authentication is performed. Valid value is an alpha-numeric character string.
LST Hash Secret	Click the Set button to display the Set LST hash secret dialog. Enter and then confirm the secret used in encryption and decryption of the LST.

3. Save the configuration.

Edit an LST File

In the SIP Registrar configuration, select the LST file from the drop-down list and click the Manage LST button to display the Edit Local subscriber table dialog, from which you can add, change, copy and delete users from the LST.

Additional dialog controls include:

- **Verify**— Parse the LST for format errors and report if the syntax/format is incorrect.
- **Save**—Dismiss the dialog and save all changes.
- **Compare**—Identify the changes made to this LST since the last save.
- **Change secret**—Change the hash secret used to encrypt username, digest realm and password.
- **Close**—Dismiss the dialog without saving changes.

Perform the following steps to add a user to the LST:

1. Access the SIP Registrar configuration object.
Configuration, System Administration, SIP Registrar.
2. Click the **Add** button. The system displays the Add Local Subscriber Entry dialog.
3. **username**— Enter a username for this subscriber. Optional configuration includes, password, universal number and AoR.

The value given in the username attribute must be the same as the username that will be sent in the Authorization Header in the Request message from the users. Refer to RFC 2617 Http Authentication for details.

4. **Password**—Enter the password associated with the username of the client. This is required for all LOGIN attempts. The password displays while typing but is not saved in clear-text (i.e., ****). Valid value is an alphanumeric character string.
5. **Aor**—The Address of Record attribute is optional to specify the address of record for the subscriber if it is different than the username.
6. **Universal number**—The user's number in a format compatible for use within the routing table.
7. Repeat the subscriber add process for as many subscribers as intended.
8. Save and activate your changes when finished.

LST Runtime Execution

The LST is loaded on boot up when the configuration is appropriately set. Incoming messages thereafter can then be authenticated based on the credentials in the LST. If the Oracle Enterprise Communications Broker can not load an LST file, three things occur:

1. The following log message is recorded at the NOTICE level:

```
LST [table-name] was not loaded - [filename] has error loading XML file
```
2. The message stated above is printed on the GUI.
3. A 503 Response is returned to the UA that sent the initial REGISTER message to the Oracle Enterprise Communications Broker.

LST Redundancy for HA Systems

The Oracle Enterprise Communications Broker synchronizes LSTs between redundant nodes to ensure that the standby node contains identical LST files. This process

occurs automatically when the user uploads an LST via the GUI and when the user saves a change to an LST file via the GUI.

LST File Compression

To save local disk flash space, you can compress the LST XML file using .gz compression. The resultant file must then have an .xml.gz extension.

LST File Format

The LST file format is as follows:

```
<?xml version='1.0' standalone='yes'?>
<LocalSubscriberTable>
  <realm>aaa</realm>
  <encryption>disabled</encryption>
  <secret>02:4B:20:99:60:D2:73:4A:7B:66:B0:62:AC:8D:B5:7D:67:5F:4B:5B:
47:F2:2E:50:B5</secret>
  <subscriber username="alice" aor="alice@company.com"
universalNum="1231231234" hash="d9bfe1cac8e7fe6b79da42d03b03b96b"/>
  <subscriber username="bob" aor="bob@company.com"
universalNum="1231232345" hash="af586127536d20f4c6e88a2921780b18"/>
  <subscriber username="carol" aor="carol@vendor.com"
universalNum="1231233456" hash="b695bc18bef48e2141555e7736bd88ec"/>
</LocalSubscriberTable>
```

The LST file's elements are explained below.

localSubscriberTable

This is the head element in the XML file. Each file can have only one head element. The following attribute is found in this element:

- **realm**—Specifies the name of digest realm.
- **encryption**—This indicates whether or not the hash in the XML file is encrypted (MD5). The key for this encryption will be a preshared key and is configurable in the local subscriber table configuration element with the secret parameter.
- **secret**—Included if encryption is used, this is the encrypted secret.

subscriber

This element has the subscriber information. And has the following 5 attributes:

- **username**—The value given in the username attribute must be same as the username that will be sent in the Authorization header in the request message from the users. Refer RFC 2617 Http Authentication for details.
- **aor**—The aor attribute is optional to specify the address of record for the subscriber if it is different than the username.
- **universalNum**—The universalNum attribute is optional to specify the universal number (E.164) for the subscriber.

- hash—The hash provided in the XML must be an MD5 hash of the username, digest-realm and the password of the user. This is same as the H(A1) described in RFC 2617.
hash = md5(username:digest-realm:password)

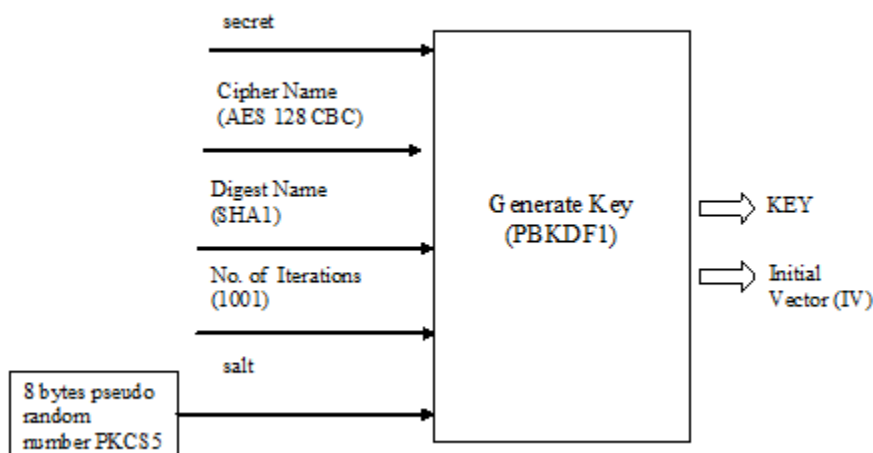
LST Subscriber Hash and Encryption

You can use AES-128 CBC to encrypt the hash in the subscriber element in the LST XML file. The PSK used for encryption is configured in the **secret** parameter and an 8-byte pseudo random number is used as the salt.

The LST file must set the encrypted attribute per subscriber element to true. To derive the final encrypted data you place in the XML file, three steps are performed according to the following blocks. The output of the last step, Formatting final Encrypted Data, is inserted into the LST files, subscriber element's hash value, when the encrypted attribute is set to true.

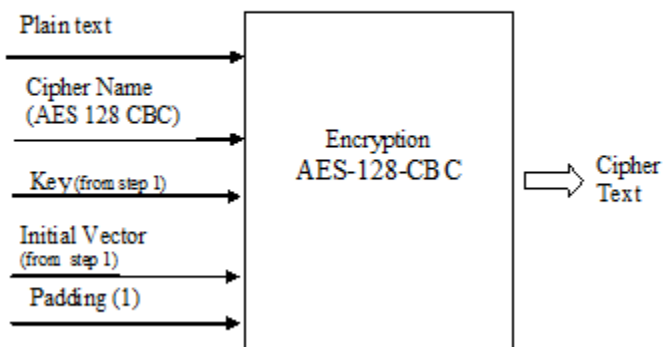
Key Initialization Vector

STEP 1: Key /IV Generation



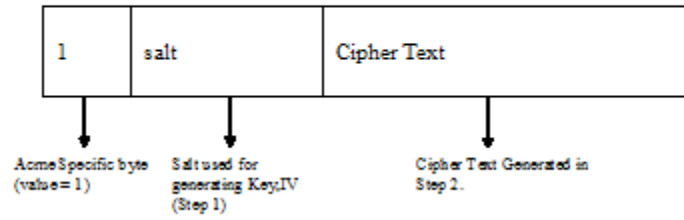
Encryption

STEP 2: Encryption



Formatting Final Encryption

STEP 3: Final Encrypted Data



LDAP Client Configuration

The Oracle Enterprise Communications Broker supports LDAP as a communications mechanism for interaction with an LDAP server. For many enterprises, this means utilizing Active Directory, a common LDAP-based service, to request information used in SIP session routing and authentication. The Oracle Enterprise Communications Broker's LDAP client requires configuration on the Oracle Enterprise Communications Broker and the LDAP server.

Configuration aspects of LDAP client configuration include:

- LDAP server access—The user specifies LDAP server location and access preferences.
- Routing queries—The user specifies the conditions wherein the Oracle Enterprise Communications Broker performs an LDAP dip to obtain location information (home agent) for FROM and REQUEST-URIs.
- AoR queries—Optionally searches for additional AoR matches in Active Directory so that it can create additional routes to target users that have contacts stored in separate records.
- SIP Authentication queries—As an optional registration authentication mechanism, LDAP client configuration can utilize domain authentication or customized authentication server configuration on the LDAP server, as follows:
 - The use of domain authentication requires an application be installed on the domain controller.
 - Customized authentication requires the specification of compatible authentication fields on both client and server.

 **Note:**

The user must ensure that phone numbers in the LDAP database are unique. If the Oracle Enterprise Communications Broker encounters multiple records with the same number, the lookup fails.

LDAP Configuration Options

The Oracle Enterprise Communications Broker provides options for LDAP configuration. These options provide the user with the flexibility to implement lookup precedence and preference on a per-agent and global basis.

An overall LDAP configuration on the Oracle Enterprise Communications Broker can include the following types of configurations.

- LDAP Config—A configuration that reaches one or multiple LDAP servers that refer to the same search base, use the same credentials, and typically service the same domain. The user can apply these configuration to agents.

- **Global Config**—A special LDAP configuration that the system uses as a default LDAP Config. When enabled, the system uses this LDAP Config for agents that are not configured with an LDAP Config or LDAP Group. The user cannot rename or delete the Global Config.
- **LDAP Group**—Multiple LDAP configs grouped together that allow the user to refine LDAP lookups across disparate LDAP domains.

The Oracle Enterprise Communications Broker's operational precedence for using your LDAP configuration on a per call basis is as follows:

1. If the applicable agent has an LDAP group applied, calls from that agent use the group configuration.
2. If the applicable agent does not have an LDAP group configuration, but does have an LDAP Config applied, calls from that agent use the LDAP Config.
3. If the call arrives on an agent without any LDAP configuration, calls from that agent use the Global LDAP configuration.

The Global LDAP configuration is disabled by default. This, in conjunction with the absence of any LDAP Configs, excludes the use of LDAP.

Making Settings

The Oracle Enterprise Communications Broker provides access to configuration fields via lists on the GUI.

When the user clicks the LDAP icon, the system displays a navigation pane on the left side of the screen, which includes the following links:

- LDAP
- Groups

When the LDAP link is selected, the system displays the list of currently configured LDAP Configs, including the Global Config. See the *Oracle® Enterprise Communications Broker User's Guide*, Release P-CZ2.0.0 for instructions on configuring LDAP Config fields.

Clicking the Groups link displays the LDAP Groups list, from which the user configures LDAP Groups.

Configure LDAP Server Access

Use the following procedure to configure the Oracle Enterprise Communications Broker (OECB) to access one or more LDAP servers.

1. Access the LDAP Configuration object.
Configuration, System Administration, LDAP, LDAP Config.
2. On the LDAP config page, do the following:

Name	
State	Select to enable the LDAP configuration.

LDAP Servers	Enter one or more IP addresses and optionally the port numbers for each LDAP Server that you want to add to the LDAP configuration. The first server listed is considered the primary LDAP Server, and the remaining servers are considered the secondary LDAP Servers. The HUNT strategy is used to determine the active LDAP Server. Default ports: 389 (for LDAP over TCP) and 636 (LDAP over TLS). IP Address must be entered in dotted decimal format (0.0.0.0). Default is blank.
Realm	Enter the name of the realm to receive requests on. Default: ecb.
Username	Enter the username that the LDAP bind request uses for authentication before access is granted to the LDAP Server. Valid values are alpha-numeric characters. Default: blank.
Password	Enter the password to pair with the username attribute, that the LDAP bind request uses for authentication before access is granted to the LDAP Server. Valid values are alpha-numeric characters. Default: blank.
LDAP SearchBbase	Enter the base Directory Number you can use for LDAP search requests. Valid values are alpha-numeric characters. Default is blank.
Timeout Limit	Enter the maximum amount of time, in seconds, for which the OECB waits for LDAP requests from the LDAP server before timing out. When an LDAP response is not received from the LDAP server within the time specified, the request is retried again based on the max-request-timeouts parameter value. Default: 15. Valid values:1 to 300 seconds.
Max Request Timeouts	Enter the maximum number of times that the LDAP Server is sent LDAP requests before the OECB determines that the server is unreachable and terminates the TCP/TLS connection. When an LDAP response is not received within the time specified for the timeout-limit parameter value, the request is retried the number of times specified for this max-request-timeouts value. Default: 3. Valid values: 0-10.
TCP Keepalive	Specify whether or not the OECB keeps the TCP connection to the LPAD Server alive. Default: Disabled. Valid values: Enabled Disabled.

LDAP Sec Type	<p>Select the LDAP security type to use when the OECB accesses the LDAP server. This parameter enables the use of LDAP over TLS (LDAPS). If you set a value for this parameter, you must also specify an LDA TLS Profile value. Default: none Valid values: none (No LDAP security type specified.) LDAPS (Method of securing LDAP communication using an SSL tunnel. This is denoted in LDAP URLs. The default port for LDAP over SSL is 636.)</p>
Routing	<ul style="list-style-type: none"> • State—Select to enable routing. • Route Mode—Select how you want the OECB to order routes. Valid values: match-only match-first attribute-order. • From Header Replacement—Enter any text you want replaced in the from header. • Lookup Queries—Click Add, set the values for lookup, and click OK. •
Address of Record	<ul style="list-style-type: none"> • Lookup Number Attribute—Enter the name of the attribute to query. Default: sAMAccountName. • Lookup Number Format Type—Select a type of translation to apply to the number before the query. Default: None: Valid values: None-use the called number as-is. E164-+14445551234 E164-No-Plus-14445551234 No Country code-4445551234 Pattern Only-use a portion of a matching dial plan Regular Expression-apply a regular expression. • Lookup Number regex pattern—Enter an expression. • Lookup Number regex result—Enter the format to create a telephone number or query from values captured in a regular expression. • AoR Attribute—Enter the name of an address of record attribute to return from the directory. • AoR Extraction Regex—Enter a regular expression to parse the address of record returned from the directory. • AoR Value Format—Enter the format to create the address of record from values captured in a regular expression.
SIP Authentication	<ul style="list-style-type: none"> • Username Attribute—Set the name of the attribute to query. Default: sAMAccountName. • Digest Has Attribute—Enter the name of the hash attribute to return from the directory. Default: orclDigestPwdAttribute.

TLS Profile

Select the name of the Transport Layer Security (TLS) profile that the OECB uses when connecting to the LDAP Server. The ldap-sec-type must be set to **LDAPS** for this profile to apply. Valid values are alpha-numeric characters. Default is blank. See the *Oracle ECB Administrator's Guide* for instructions on how to create a TLS profile.

3. Click **Back**.
4. Save and activate the configuration.

LDAP Groups

LDAP groups on the Oracle Enterprise Communications Broker group **LDAP configs** together, allowing the user to refine lookups to multiple LDAP servers. The user configures **LDAP groups**, defines the matching criteria by which the system selects servers to query, and applies LDAP groups as profiles to agents.

When the system determines that it may find information it needs in the LDAP database, it checks to see if there is an **LDAP group** configured on the applicable agent. If there is no group, the system uses the LDAP configuration to control its lookups. This configuration can include a single LDAP Config configured on the agent or a Global LDAP config. If there is a group, the system:

1. Checks the matching criteria in the group to identify relevant LDAP servers, and
2. Performs lookups to relevant servers using the order that the administrator has configured in the group.

If there is no match with any of the group's servers, the system does not perform an LDAP lookup and proceeds with the process sequence it uses to find information.

Matching Criteria in LDAP Groups

The Oracle Enterprise Communications Broker (OECB) uses configurable **Matching Criteria** to determine whether or not to perform a lookup to each server in an **LDAP Group**. The OECB supports regex expressions within matching criteria configuration.

The system evaluates matching criteria for all LDAP servers listed in the group. If there are no matches, the system proceeds without querying LDAP. If there are any matches, the system initiates lookups to servers in the order listed in the group. If there is no match, the system skips those servers.

Consider the example where the system performs a lookup for a phone number in LDAPGroup1. The **LDAP Group** selection is based on the agent configuration of the applicable end station. In this example, the LDAP1 agent configuration specifies LDAPGroup1 as its **LDAP group**.

In this example, the Administrator configured the first lookups within LDAPGroup1 to be directed to LDAP1. If the number matches a criteria entry, the system adds that server to the lookup list. If not, the system skips to the next servers in the group, evaluating their **Matching Criteria**.

Matching criteria for LDAPGroup1 could include the following entries.

Matching Criteria	LDAP Server
+1*	LDAP1
*@Div1.com	LDAP1
.*	LDAP2
+44*	LDAP3
+34555*	LDAP3

For LdapGroup1, the system queries LDAP1 only if the number starts with a +1 or has a host of Div1.com. The system queries LDAP2 in all cases based on its wildcard criteria. The system queries LDAP3 if the called number has a UK area code or has a Spain area code, then starts with 555.

Configure LDAP Groups

Configure LDAP before configuring LDAP groups.

Use the following procedure to create a new LDAP group:

1. Access the LDAP Configuration object.

Configuration, System Administration, LDAP, LDAP Groups.

2. On the Add LDAP Group page, click **Add**, and do the following:

Group Name	Enter a name for this group. The name must be between 1 and 128 alpha-numeric characters without spaces. The name can include the underscore, comma, period and dash characters, but not as the first characters in the name.
Description	(Optional) Enter a description of this LDAP group.
State	Select to enable this LDAP group.
LDAP Agents	Click Add and do the following: <ol style="list-style-type: none"> a. Matching Criteria—Define the criteria the system must use to determine whether it should perform a lookup in the associated server for end station information. Regex is supported as a means of configuring matching strings. b. LDAP Config—Select the LDAP configuration that you want to use.

3. Click **OK**.

The system displays the LDAP group configuration page, where you can optionally add more agents.

4. After you add the necessary agents, click **OK**.
5. Save the configuration.

Apply your LDAP group to the applicable Agents from the **LDAP** drop-down list under the Agent controls.

Configure a Routing Query

To configure the Oracle Enterprise Communications Broker (OECB) to query an LDAP database for the purpose of obtaining a call's routing information, per the OECB processing sequence:

1. Access the LDAP Configuration object.

Configuration, System Administration, LDAP, LDAP Config.

2. On the LDAP Config page, expand **Routing** to expose the Routing parameters, and do the following:

State	Select to enable the use of routing queries for your configured LDAP servers.
Route Mode	<p>Specify the route priority that the OECB uses in the route list that determines which routes to create, and the priority of the routes within the route list. Default: Match-only. Valid values: Match-Only Attribute Order Match First.</p> <ul style="list-style-type: none"> • Match-Only—When an exact match between the dialed telephone number and an LDAP attribute value in the search response entry occurs, a route is created corresponding to that LDAP attribute. When an exact match on multiple attributes occurs, the ordering of LDAP attributes in the LDAP configuration determines the priority for each route. For example, in an enterprise that uses the same phone number for both Lync and IPPBX phones, when the msRTCSIP-Line attribute is configured first, the corresponding next hop (Lync Server) is used to create the first route in the route list. • Attribute Order— The ordering of LDAP attributes in the LDAP configuration determines the priority for each route. When the msRTCSIP-Line attribute is configured first, the corresponding next hop (Lync Server) is used to create the first route in the route list. If there is a valid value present in the search response entry for a LDAP attribute, a route is created corresponding to that LDAP attribute. • Match First— If there is an exact match between the dialed telephone number and an LDAP attribute value in the search response entry, the corresponding route gets the highest priority in the route list. For the rest of the routes, the ordering of LDAP attributes in the LDAP configuration determines the priority for each route. So if the msRTCSIP-Line attribute is configured first, the corresponding next hop (Lync Server) would be used to create the second highest priority route in the route list. If there is a valid value present in the search response entry for an LDAP attribute, a route is created corresponding to that LDAP attribute.

 **Note:**

The LDAP attribute must have a valid value in the response; a match is not necessary for that attribute. If an entry is returned in the search response, there must be a match on at least one other attribute. For example, the dialed telephone number could be +17813284392 (IP-PBX Phone#), and the msRTCSIP-Line in the response could be +17814307069 (Lync phone#). A route is created for the Lync phone#, even though the dialed telephone number is the PBX Phone#.

From Header Replacement	Enter the name of the LDAP lookup query to use as the From Header replacement.
-------------------------	--

3. Click **Add** in the Lookup Query dialog, and do the following:

Each element identifies a lookup number attribute and dialed pattern with which the OECB finds matches in the LDAP database and identifies contacts to which it builds routes. Multiple matches result in multiple targets to which the OECB creates routes for call forking.

Lookup Number Attribute	Enter the Active Directory attribute name. Default: Telephone Number. Valid values: Alpha-numeric characters. Some examples of Active Directory attribute names include: <ul style="list-style-type: none"> • ipPhone and msRTCSIP-Line for Lync phone number • telephoneNumber for IP PBX phone number • mobile for Mobile phone number
Lookup Number Format Type	Select the expected attribute format from the drop down list. Default: None. Valid values: E164 E614 No Plus No Country Code None Pattern Only Regular Expression.
Lookup Number Regex Pattern	Enter the regular expression pattern used to break down the string of digits in the phone number extracted from the request URI and the FROM of the SIP request. The variables extracted from the phone number can be used in the attribute-value-format parameter. Default regex: "\+?1?(\d{2})(\d{3})(\d{4})\$" Valid values: Alpha-numeric characters. (The default value assumes that the phone number is a North American phone number specified in the E.164 format.) It extracts the following variables from the phone number: \$1 is the area code. \$2 and \$3 are the next 3 and 4 digits in the phone number. Note that the system queries only for the home agent of the FROM if it has not already found it. The setting applies only when you set Lookup number format type to regular-expression .
Lookup Number Regex Result	Lookup number regex result —Enter the format for the attribute value. These format values are extracted from the phone number using the extraction-regex parameter. Default: "tel:+1\$1\$2\$3". This value assumes that the phone number is a North American phone

	number specified in the E.164 format, and it recreates the phone number in E.164 format. Valid values: Alpha-numeric characters. In addition to the E.164 format, the OECB uses other formats as well to store the phone numbers. You can customize the value specified for this parameter to enable successful queries for phone numbers in other formats. The setting applies only when you set Lookup number format type to regular-expression .
Home Agent Attribute	Enter the Active Directory attribute name for the agent field. Default: Blank. Valid values: Alpha-numeric characters. If created with the Oracle tools described in this document, the name would be orclAgentNameAttribute.
Home Agent Regex Pattern	Enter the regular expression pattern used to break down the agent name. Default: Blank.
Home Agent Regex Result	Enter the format of the regex result. Default: Blank.
Default Home Agent	Enter the name of the home agent to use for routing if the query does not return one.

4. Click **OK** to save the routing query.
5. Save the configuration.

Address of Record Configuration Fields

You can configure the Oracle Enterprise Communications Broker (OECB) to query an LDAP database for the purpose of identifying additional Addresses of Record (AoR) that may apply to a call. The system identifies additional contacts from the AoRs and creates additional routes with which it can fork the calls.

1. Access the LDAP Configuration object.
Configuration, System Administration, LDAP, LDAP Config.
2. On the LDAP Config page, do the following:

Lookup Number Attribute	Enter the Active Directory attribute name. Default: sAMAccountName, which is the standard Active Directory username attribute. Valid values: Alpha-numeric characters.
Lookup Number Format	Select the expected phone number format from the drop down list. Default: None. Options include: E164 E164 No Plus No Country Code None Pattern Only Regular Expression.
Lookup Number Regex Pattern	Enter the regular expression pattern used to break down the string of digits in the phone number extracted from the request URI of the SIP request. The variables extracted from the phone number can be used in the attribute-value-format parameter. Valid values:

	Alpha-numeric characters. Default: " <code>^\+?1?(\\d{2})(\\d{3})(\\d{4})\$</code> ". (This value assumes that the phone number is a North American phone number specified in the E.164 format.) It extracts the following variables from the phone number: \$1 is the area code \$2 and \$3 are the next 3 and 4 digits in the phone number. Note that this setting applies only when you set Lookup number format type to regular-expression .
Lookup Number Regex Result	Enter the format for the attribute value. These format values are extracted from the phone number using the extraction-regex parameter. Default: " <code>tel:+1\$1\$2\$3</code> ". This value assumes that the phone number is a North American phone number specified in the E.164 format, and it recreates the phone number in E.164 format. Valid values: Alpha-numeric characters. In addition to the E.164 format, the OECB uses other formats as well to store the phone numbers. You can customize the value specified for this parameter to enable successful queries for phone numbers in other formats. The setting applies only when you set Lookup number format type to regular-expression . In addition to the E.164 format, the OECB uses other formats as well to store the phone numbers. You can customize the value specified for this parameter to enable successful queries for phone numbers in other formats.
AoR Attribute	Enter the Active Directory attribute name established to contain the AoR. Default: Blank. Valid values: Alpha-numeric characters.
AoR Extraction Regex	Enter the regular expression pattern used to break down the AoR. Valid values: Alpha-numeric characters.
AoR Value Format	Enter the format of the regex result. Default: Blank.

3. Click **OK** to save the routing query.
4. Save configuration.

SIP Authentication Query Configuration Fields

To configure the Oracle Enterprise Communications Broker to specify an alternate authentication field in a remote database for the purpose of authenticating registration attempts:

1. Access the LDAP Configuration object.
Configuration, System Administration, LDAP, LDAP Config.
2. On the LDAP Config page, expand **SIP Authentication** and do the following:

Username Attribute	Enter the name of the attribute where the digest username is stored in your LDAP database. Default: <code>sAMAccountName</code> . (The standard Active Directory username attribute.)
Digest Hash Attribute	Enter the name of the attribute where the digest authentication hash is stored in your LDAP database. Default:

orclDigestPwdAttribute. (A custom field populated by the oidpwdcn password filter.)

3. Click **OK** to save the configuration.

Replacing the Calling Number in the FROM Header

The Oracle Enterprise Communications Broker provides for replacement of the calling number in SIP messages' FROM headers. Applicable messages include INVITEs that match the query, and all messages sent by the Oracle Enterprise Communications Broker to those calls' callees. An example application is allowing recipient UEs to display a caller ID that would be recognized by the recipient, even during an enterprise's transition to new dialing schemes.

This calling number replacement function refers to LDAP resources as the source of the replacement calling number. The user configures a lookup query from the **Modify LDAP config** dialog to specify this source. Configured lookup queries become available in the **FROM header replacement** drop-down list, from which the user selects their query. This selection specifies and enables the replacement.

This feature piggybacks normal LDAP lookup procedures by collecting an additional value within the LDAP query request/response sequence. The Oracle Enterprise Communications Broker replaces the FROM header of the outgoing message with this value.

While processing this LDAP response for calling number, the Oracle Enterprise Communications Broker stores the result of the query and uses it to create the FROM header user parts for applicable outgoing messages. For traffic in which there is no match to the calling number, the Oracle Enterprise Communications Broker simply uses the original calling number.

The user can disable this replacement function by clearing the lookup query attribute name from the **FROM header replacement** field.

12

ACL Configuration

The Oracle Enterprise Communications Broker (OECB) provides for realm-based access control lists (ACLs) so you can explicitly permit or deny access to the realm on a per-packet basis. By default, the OECB allows all packets to access the realm. You configure ACLs to filter packets by IPv4 address, application protocol (SIP) and transport protocol. You can also specify ranges of that IPv4 addressing for which you want to filter.

The OECB allows ACL configuration via the GUI's System Administration controls. You set the OECB to control access to the realm based on available parameters. You can also configure a free text description of each ACL to clarify their purpose.

In the absence of any ACLs, the OECB allows access to all packets. In addition, ACLs support real-time configuration, meaning they do not require system reboot to become functional.

Note:


The OECB supports static ACLs only.

Configure an ACL

You can add to specify how you want the OECB to enforce realm access by configuring access control entries.

1. Access the Access Control configuration object
Click **Configuration, System Administration, DoS**
2. On the Access Control page, do the following:

Realm ID	Enter the Name of the ingress realm to which this ACL applies.
Description	Enter a text description of the ACL for identification purposes.
Source Address	Enter the source IPv4 address and port number for the host in the following format: <IP address>[/number of address bits][[:<port>]][/<port bits>] For example: 10.0.0.1/24:5000/14 10.0.0.1/16

	<pre>10.0.0.1/24:5000 10.0.0.1:5000</pre>
Destination Address	<p>(This is ignored if you configure an application protocol.) Enter the destination IPv4 address and port for the destination in the following format:</p> <pre><IP address>[/number of address bits][:<port>[/<port bits>]]</pre> <p>You do not need to specify the number of address bits if you want all 32 bits of the address to be matched. You also do not need to specify the port bits if you want the exact port number matched. If you do not set the port mask value or if you set it to 0, the exact port number will be used for matching. The default value is 0.0.0.0.</p>
Application Protocol	<p>Enter the application protocol type for this ACL entry. The valid values are:</p> <ul style="list-style-type: none"> • SIP • None <div style="background-color: #e6f2ff; padding: 10px; margin-top: 10px;"> <p> Note:</p> <p>If application-protocol is set to none, the destination-address and port will be used. Ensure that your destination-address is set to a non-default value (0.0.0.0.)</p> </div>
Transport Protocol	<p>Select the transport-layer protocol configured for this ACL entry. The default value is ALL. The only valid value is:</p> <ul style="list-style-type: none"> • ALL
Access	<p>Enter the access control type or trusted list based on the trust-level parameter configuration for this host. The default value is permit. The valid values are:</p> <ul style="list-style-type: none"> • permit—Puts the entry into the untrusted list. • deny—Puts the entry in the deny list.

3. Click **OK**.
4. (Optional) Add another Access Control list.
5. Save the configuration.

13

ECB Sync

The Oracle Enterprise Communications Broker (OECB) allows you to configure multiple OECBs to interact with each other and share operational information. This functionality, called ECB Synchronization, works like layer-three routers dynamically exchanging routing information. This results in extensible OECB deployments where any OECB can use information from a peer OECB to make a routing decision, including simply forwarding to that peer so that it can perform the routing.

ECB sync uses the Cluster Network Protocol (CNP). OECBs configured for synchronization share the following information:

- User database
- Routes
- Dial plan
- Agents

ECB sync shares information between OECBs configured as pairs, where a transmitter sends to a receiver. The receiver updates its configuration with data from the transmitter. Categories of information exchanged include:

- Routing information - how to reach a destination.
- Context information - how to handle a given endpoint.

ECB sync configurations begin with establishing peers. Peers can operate as:

- Transmitter—Provides its peers with its information.
- Receiver—Uses the information received from a transmitter to extend its operational scope.
- Both Transmitter and Receiver.

Any OECB can peer with up to 10 other OECBs. A transmitter can provide its information to no more than 10 receivers. A receiver can obtain information from no more than 10 transmitters.

ECB Sync uses SIP SUBSCRIBE to establish relationships and exchange data. The system never displays the data in SIP SUBSCRIBE traffic in clear text, providing a layer of obfuscation. The system also uses SIP Digest authentication for authentication and authorization of peers. Optionally, you can configure TLS to secure ECB Sync traffic.

As soon as you add agents to the ECB Sync agent list and the configuration activated, the OECB begins to send SUBSCRIBEs to its Sync agents. Subscriptions, subscription refreshes, and subscription termination all follow standard SIP procedures. Transmitters use NOTIFYs to send information to receivers. The process can use multiple, concurrent NOTIFY messages if the amount of information in the NOTIFY exceeds maximum payload. The transmitter also compresses information. The receiver un-compresses information.

Detail on transmitter/receiver interaction includes:

- The transmitter sends only its own configuration data, not data learned from another OECB.
- The transmitter updates all of its receivers with full configuration information upon an activate if any applicable configuration information has changed. Changes, for example, to a network interface would not trigger an ECB Sync update.
- The receiver's subscription refresh interval is 1/2 the expires time (30 seconds). This timing is not configurable. If the transmitter does not receive any refresh within the expires time, it terminates the subscription.
- The transmitter does not send subsequent NOTIFY messages without positive confirmation of the previous NOTIFY by the receiver.
- The receiver re-sends unanswered and rejected SUBSCRIBE messages in 60 second intervals until it receives a response.
- The transmitter cancels all NOTIFY procedures upon receipt of an error message from the receiver. When this happens, the receiver restarts the NOTIFY process from the beginning. The transmitter makes this additional attempt to resend the configuration/user data only once if it receives such an error.
- When the receiver encounters an unrecoverable error, it discards all information previously learned from the transmitter, cancels the subscription, and initiates a new subscription.
- The receiver un-subscribes from a transmitter gracefully upon activation of a configuration change that removes that transmitter from the receiver's list.

The OECB secures SUBSCRIBE and NOTIFY transactions using SIP Digest procedures. The transmitter challenges the receiver upon receipt of a subscription. Using SIP Digest, the receiver replies to the challenge with standard SIP Digest user, realm, and password hash for the transmitter to verify. ECB sync configuration includes specifying the secret to use for the password. You must configure all receivers and all transmitters to use the same secret for this purpose.

The receiver tracks ECB sync information, differentiating between its own configuration and that of each transmitter. Grouping sync information by its source allows the receiver to discard the correct information if a transmitter becomes unreliable or invalid. The receiver also uses this grouping to prioritize overlapping configuration information. Overlapping configuration objects include those using the identical key. The rules include:

- A receiver uses its own object if it is in the running configuration.
- If multiple transmitters provide information on the same object, the receiver uses transmitter name alphabetical order to determine which information to use.

Receivers keep all objects from all sources in memory should the source in use become invalid.

Note that all transmitters and receivers must use the same CNP version to operate properly. If a receiver gets CNP data from a higher version, it returns a 489 "Bad Event" error and drops the payload.



Note:

After you configure ECB Sync, you must reboot the system.

ECB Sync and High Availability

A receiver updates its standby OECB with ECB sync data and status. If an active OECB stops responding, the standby becomes available for immediate use as a receiver. Conversely, if a transmitter transitions from active to standby, the receiver assumes its subscription is still valid until it refreshes and receives a 481 error message. Upon receiving the 481 message, the receiver terminates the existing subscription and starts a new one with the new active. Note that the receiver retains learned configuration information despite the subscription change.

Synchronizing the Registration Cache

ECB Sync data includes the Oracle Enterprise Communications Broker's registration cache. The user can enable registration cache sync via a checkbox within the Sync config settings under the ECB Sync icon.

When enabled, the Oracle Enterprise Communications Broker presents its registration cache to all ECB Sync agents every nine minutes. Each ECB Sync agent uses this data to create a separate, ECB Sync-only registration cache table that includes contacts and the ECB from which it learned the cache entry.

When a call comes for a contact found in the ECB Sync-only registration cache, the Oracle Enterprise Communications Broker receiving the call adds a URI parameter to the request URI of the TO header and forwards the message to the Oracle Enterprise Communications Broker in the table. This URI parameter informs the target Oracle Enterprise Communications Broker that it must only use its registration cache for routing this call. This parameter appears as follows.

```
TO sip:user2@server2.com;orcl-regonly=true
```

In these cases, both Oracle Enterprise Communications Brokers forward the call. The Oracle Enterprise Communications Broker receiving the call uses all other routing sources to route the call, including LDAP, LST and UserDB. The Sync agent Oracle Enterprise Communications Broker routes the call using its registration cache, and skips all other routing sources, including LDAP, LST and UserDB.

Enable ECB Sync Operations

Enable ECB sync when you want one Oracle Enterprise Communications Broker (OECB) to share information with another OECB. For example, to share information about users, routes, dial plans, and agents. To use ECB Sync, you must enable the service and set the authentication secret. You can optionally enable the system to sync the configuration and the registration cache from one OECB to another.

The secret that you enter in the ECB Sync configuration must match the secret used by the ECB sync agents.

1. Access the Sync Config Settings configuration object.
Configuration, System Administration, Sync, Sync Config Settings.
2. On the Modify Configuration Settings page, do the following:

Enable sync	Select to enable ECB sync operations.
-------------	---------------------------------------

Secret	<ol style="list-style-type: none"> a. Click Set. b. Secret—Enter the secret that you use for authenticating with OECB peers. c. Confirm secret—Re-enter the secret. d. Click OK.
Configuration	(Optional)—Select to enable the system to sync the configuration from one OECB to another.
Registration	(Optional)—Select to enable the system to sync the registration cache from one OECB to another.

3. Click **OK**.

Add an ECB Sync Agent

Add an ECB Sync Agent

ECB Sync requires you to specify at least one sync agent for the Oracle Enterprise Communications Broker (OECB) to peer with before the system can perform sync operations. Each OECB can peer with up to ten ECB Sync agents and each ECB Sync agent can peer with up to 10 OECBs.

- Configure the agents that you want to add as ECB Sync peers.
- Enable ECB Sync.

Use the optional step in the following procedure to add more agents to ECB Sync.

1. Access the Sync Agents configuration object.
Configuration, System Administration, Sync, Sync Agents.
2. On the Sync Agents page, click **Add**.
3. On the Add Sync Agent Settings page, select an agent from the drop-down list.
4. Click **OK**.
5. (Optional)—Repeat step 3 to add another sync agent.
6. Click **Back**.
7. Save the configuration.

ECB Sync Monitoring

As described, the Oracle Enterprise Communications Broker keeps track of ECB sync peers' status and data.

This information is visible via the GUI, as follows:

- Peer Status—The ECB sync peer status widget displays all peer relationships that apply to the current Oracle Enterprise Communications Broker. Information displayed includes:
 - Status—Indicating whether the peer relationship is In Service or Out of Service. This status indicates reachability.

- Transmitter and Receiver Sync State—Verifying subscription status for both peers.
- Transmitter and Receiver Uptime—Displaying the uptime of the subscription.
- Peer Data—The routing table and user table include a column titled "Learned From" that displays the source of the listed element, including any ECB Sync peer.

HMR Configuration

Header manipulation rules (HMRs) use Oracle Enterprise Communications Broker-specific controls and/or REGEX to identify information in signaling messages the user wants to change. These rules get applied to Oracle Enterprise Communications Broker agents, and operate globally on all applicable signaling traffic that reaches that agent. This section provides instructions on configuring HMR on the Oracle Enterprise Communications Broker GUI interface.

See the Header Manipulation Appendix in this guide for full explanation of how HMRs work. This appendix is provided for those with no prior experience with HMRs. HMR configuration errors can adversely impact all of an agent's traffic. Be fully confident about the intent of an HMR, and review your HMR configurations carefully before activating them.

SIP Header Manipulation

SIP header manipulation allows you to add, delete, or modify SIP message attributes on the Oracle Enterprise Communications Broker (OECB). For example, SIP headers and SIP header elements.

The most common reason for manipulating SIP headers and SIP header elements is to fix an incompatibility problem between two SIP endpoints. For example, Softswitch - PSTN incompatibility or a SIP messaging problem between two different IP PBX platforms in a multi-site deployment where calls between the platforms are unsuccessful due to problems in the SIP messaging.

To enable the SIP header manipulation, create rule sets in which you specify header manipulation rules and, optionally, header element manipulation rules. SIP header elements are the sub-parts of the header, such as the header value, the header parameter, the URI parameter, and so on, excluding the header name. You can specify the actions that you want the system to perform for each header element.

After creating the header manipulation rule set, apply it to a session agent or SIP interface as "inbound" or "outbound."

Multi-Hop SIP Header Manipulation Rules

Oracle Enterprise Communications Broker (OECB) Header Manipulation Rule (HMR) support allows you to specify a manipulation type depending on an agent's location (hop) in a route. Applicable hops include the next and last hop of a route. Applying an HMR when an agent is the last hop in a route is referred to as "multi-hop" HMR. You do not need to make any changes to the HMR configuration for it operate as a multi-hop HMR.

The default setting makes the system apply the outbound HMR only when the agent is the next hop in the route's path.

If there are multiple HMRs that the OECB must apply for the route, it applies the HMR for the last hop first. If the same agent is both next and last hop for any given traffic,

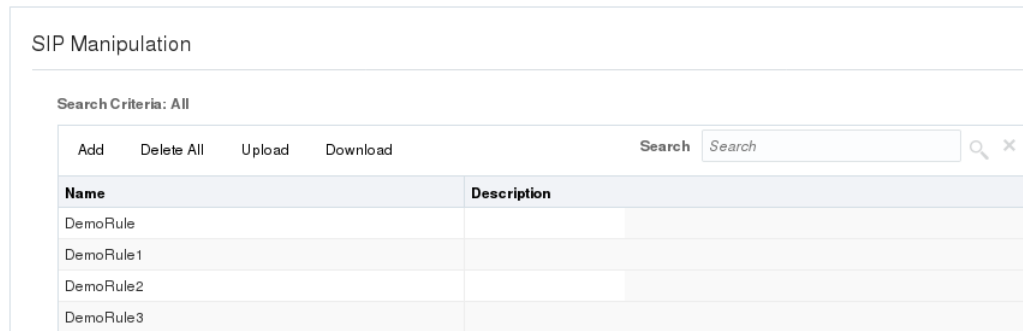
the OECB applies the HMR only once regardless of the `Apply Outbound Manipulation On` setting.

Set the `Apply Outbound Manipulation On` parameter on the agent to specify when the system applies the agent's outbound HMR. Default: Next Hop Only. Valid values: Next Hop Only | Last Hop Only | Next and Last Hop.

SIP Header Manipulation Configuration Dialogs

You configure Header Manipulation Rules (HMR) on the Oracle Enterprise Communications Broker (OECB) by way of the SIP Manipulation page. In the configuration, you use the HMR dialogs to assign HMRs to agents, which defines where the system uses the rule.

The first HMR page, located at **Configuration, System Administration, SIP Manipulation**, is the SIP Manipulation list. This list shows all manually configured HMRs available on the system. The dialog includes controls to add new HMRs, to edit, copy, and delete existing rules, and to upload and download pre-configured HMRs in to the system. The following screen capture shows the SIP manipulation page which displays any existing manually configured HMRs.



The screenshot shows the 'SIP Manipulation' page with a search bar and a table of HMRs. The search criteria are set to 'All'. The table has columns for 'Name' and 'Description'. The table contains four rows: 'DemoRule', 'DemoRule1', 'DemoRule2', and 'DemoRule3'. Above the table are buttons for 'Add', 'Delete All', 'Upload', and 'Download', and a search input field with a 'Search' button and a clear icon.

Name	Description
DemoRule	
DemoRule1	
DemoRule2	
DemoRule3	

When you click the **Add**, the system displays the Add SIP Manipulation dialog where you create a new HMR. The following screen capture shows the Add SIP Manipulation page.

Add SIP Manipulation

Name

Description

Split Headers

Join Headers

CfgRules

Add ▾ Upload Download

Name	Element Type
• No Data to Display	

OK Back

After naming and describing the HMR, scroll to the **CfgRules** section of the page, which displays the list of header or MIME rules that you can apply to this HMR. Note that header rule order of execution is critical when an HMR contains multiple rules. You can manage the order of HMR execution with the **Move** control in the menu that displays when you right-click a row in the table. The following screen capture shows the **CfgRules** drop-down list.

Modify SIP Manipulation Show Configuration

Name

Description

Split Headers

Join Headers

CfgRules

Add ▾ Upload Download

- header-rule
- mime-rule
- mime-isup-rule
- mime-sdp-rule

Name	Element Type
• No Data to Display	

OK Back

When you click a Configuration Rule, the system displays the configuration the particular configuration page for the rule type. The following illustration shows the **Add SIP Manipulation / Header Rule/ Element Rule** page as an example of a

configuration for a Configuration Rule.

Add Sip manipulation / header rule

Name	<input type="text"/>
Header Name	<input type="text"/>
Action	<input type="text" value="none"/> ▼
Comparison Type	<input type="text" value="case-sensitive"/> ▼
Msg Type	<input type="text" value="any"/> ▼
Methods	<input type="text"/>
Match Value	<input type="text"/>
New Value	<input type="text"/>

CfgRules

Add ▼ Upload Download

Name	Element Type
• No Data to Display	

SIP Header Manipulation Rules Attributes and Values Reference

Refer to the following table for information about the attributes that you can configure for SIP header manipulation rules.

Attributes	Values and Descriptions
Action	<ul style="list-style-type: none"> • add—Adds a new header, if that header does not exist. • delete—Deletes the header, if it exists. • find-replace-all—Finds all matching headers and replaces with the header you specified for "Split" and "Join." • log—Logs the header. • manipulate—Manipulates the elements of this header to the element rules configured. • monitor—Monitors the header. • store—Stores the header. • none—(default) No action is taken. • reject—Rejects the header. • sip-manip—Manipulates the SIP elements of this header to the element rules configured. <p>Default: None.</p>

Attributes	Values and Descriptions
Comparison type	<ul style="list-style-type: none"> • boolean—Header is compared to header rule and must match exactly or it is rejected. • case-insensitive—Header is compared to header rule regardless of the case of the header. • case-sensitive—(default) Header is compared to the header rule and case must be exactly the same or it is rejected. • pattern-rule—Header is compared to the header rule and the pattern must be exactly the same or it is rejected. • refer-case-insensitive—Header is compared to the header rule regardless of the case in a REFER message. • refer-case-sensitive—Header is compared to the header rule and the case must be exactly the same as in the REFER message or it is rejected. <p>Default: Case-sensitive.</p>
Format	<ul style="list-style-type: none"> • ascii-string - A character-encoding scheme that represents text (128 ASCII codes, 7 bits). • binary-ascii - An encoding scheme where each byte of an ASCII character is used. Can use up to 256 bit patterns . • hex-ascii - An encoding scheme that uses a string of numbers (no spaces) to represent each ASCII character.
Header name	<p>The name of the header to which the rule applies. Case-sensitive.</p>
Match value	<p>The value that you want to match against the element value for an action to be performed.</p>
Match val type	<p>The type of value to match to the match-field entry for the action to be performed.</p> <ul style="list-style-type: none"> • any—(default) Element value in the SIP message is compared with the match-value field entry. If the match-value field is empty, all values are considered a match. • fqdn—Element value in the SIP message must be a valid FQDN to be compared to the match-value field entry. If the match-value field is empty, any valid FQDN is considered a match. If the element value is not a valid FQDN, it is not considered a match. • ip—Element value in the SIP message must be a valid IP address to be compared to the match-value field entry. If the match-value field is empty, any valid IP address is considered a match. If the element value is not a valid IP address, it is not considered a match.

Attributes	Values and Descriptions
Media type (SDP descriptor for SDP media rule)	<ul style="list-style-type: none">• m—Media name and transport address• i—Media title• c—Connection information (optional when configured at the session level)• b—Zero or more bandwidth information lines• k—Encryption key• a—Zero or more media attribute lines• t—The session time is active• r—Zero or more repeat times
Methods	SIP method names to which you want to apply the header rule. For example, INVITE, ACK, BYE. When this field is empty, the system applies the MIME rule to all methods. Default: Blank.
Mime header	The parameter name to which the rule applies. The parameter name depends on the element name you entered. For uri-param, uri-user-param, and header-param it is the parameter name to be added, replaced, or deleted. For all other types, it serves to identify the element rule and any name can be used. Alpha-numeric characters. Default: blank.
Msg type	<ul style="list-style-type: none">• any—(default) Requests, replies, and out-of-dialog messages• out-of-dialog—Out of dialog messages only.• reply—Reply messages only• request—Request messages only Default: Any.
Name	The name you want to use for the rule. Default: Blank.

Attributes	Values and Descriptions
New value	<p>The value for a new element or replacement value for an existing element. You can enter an expression that includes a combination of absolute values, pre-defined parameters, and operators.</p> <ul style="list-style-type: none"> • Absolute values—Use double quotes for clarity. You must escape all double quotes and back slashes that are part of an absolute value, and enclose the absolute value in double quotes. • Pre-defined values.—Pre-defined parameters always start with a \$. For valid values, see the Pre-defined Parameters table. • Operators parameters—For valid values, see the Operators table. <p>The following table describes the pre-defined parameters.</p>

Pre-defined Parameter	Description
\$ORIGINAL	Original value of the element is used.
\$LOCAL_IP	IP address of the SIP interface on which the message was received for inbound manipulation; or sent on for outbound manipulation.
\$REMOTE_IP	IP address the message was received from for inbound manipulation; or being sent to for outbound manipulation.
\$REMOTE_VIA_HOST	Host from the top Via header of the message is used.
\$TRUNK_GROUP	Trunk group is used.
\$TRUNK_GROUP_CONTEXT	Trunk group context is used.

The following table describes the Operators.

Operator	Description
+	Append the value to the end. For example: acme"+"packet generates acmepacket
+^	Prepends the value. For example: acme"+"^"packet generates packetacme
-	Subtract at the end. For example: 112311"-11

Attributes	Values and Descriptions	
	Operator	Description
		generates 1123
	-^	Subtract at the beginning. For example: 112311"-^"11 generates 2311
Parameter name	The parameter name to which the rule applies. The parameter name depends on the element name you entered. For uri-param, uri-user-param, and header-param it is the parameter name to be added, replaced, or deleted. For all other types, it serves to identify the element rule and any name can be used. Alpha-numeric characters. Default: Blank.	

Attributes	Values and Descriptions
Type	<p data-bbox="959 275 1458 327">The type of element on which to perform the action. Default: Blank.</p> <ul data-bbox="959 331 1458 1509" style="list-style-type: none"><li data-bbox="959 331 1458 384">• header-param—Perform the action on the parameter portion of the header.<li data-bbox="959 388 1458 441">• header-param-name—Perform the action on the header parameter name.<li data-bbox="959 445 1458 497">• header-value—Perform the action on the header value.<li data-bbox="959 501 1458 596">• mime—Perform the action on Multipurpose Internet Mail Extensions (MIME).<li data-bbox="959 600 1458 653">• reason-phrase—Perform the action on reason phrases.<li data-bbox="959 657 1458 709">• status-code—Perform the action on status codes.<li data-bbox="959 714 1458 808">• teluri-param—Perform the action on a SIP telephone Uniform Resource Identifier (URI).<li data-bbox="959 812 1458 865">• uri-display—Perform the action on the display of the SIP URI.<li data-bbox="959 869 1458 963">• uri-header—Perform the action on a header included in a request constructed from the URI.<li data-bbox="959 968 1458 1020">• uri-header-name—Perform the action on a SIP URI header name.<li data-bbox="959 1024 1458 1077">• uri-host—Perform the action on a Host portion of the SIP URI.<li data-bbox="959 1081 1458 1134">• uri-param—Perform the action on the parameter included in the SIP URI.<li data-bbox="959 1138 1458 1190">• uri-param-name—Perform the action on the name parameter of the SIP URI.<li data-bbox="959 1194 1458 1247">• uri-phone-number-only—Perform the action on a SIP URI phone number only.<li data-bbox="959 1251 1458 1304">• uri-port—Perform the action on the port number portion of the SIP URI.<li data-bbox="959 1308 1458 1360">• uri-user—Perform the action on the user portion of the SIP URI.<li data-bbox="959 1365 1458 1417">• uri-user-only—Perform the action on the user portion only of the SIP URI.<li data-bbox="959 1421 1458 1474">• uri-user-param—Perform the action on the user parameter of the SIP URI.

Attributes	Values and Descriptions
Type (SDP descriptor for SDP line rule)	<ul style="list-style-type: none"> • v—Protocol version • o—Originator and session identifier • s—Session name • i—Session information • u—URI of description • e—Email address • p—Phone number • c—Connection information (not required when included in all media) • b—Zero or more bandwidth information lines or one or more time descriptions("t=" and "r=" lines) • z—Time zone adjustments • k—Encryption key • a—Zero or more session attribute lines or zero or more media descriptions • t—Time the session is active • r—Zero or more repeat times

SIP Header Manipulation Configuration

Configuring SIP manipulations from the Web GUI is a multi-faceted process performed through a series of nested dialogs that differ depending on the particular header and header element that you want to manipulate. It is not practical to document the entire SIP manipulations configuration process in one procedure. The documentation begins with the "Configure SIP Manipulation", topic where you can set the global parameters, if that is all you need. The documentation continues with procedures for each particular header and header element that you can manipulate. Each of those topics includes the global settings, so you can set or modify them there, as well.

header and header element that you can manipulate include the following:

- Configure MIME Rule—includes the mime-header-rule element.
- Configure MIME ISUP Rule—includes the mime-header-rule and isup-param-rule elements.
- Configure MIME SDP Rule—includes the mime-header-rule, sdp-session-rule, and the sdp-media-rule.

When you finish configuring SIP manipulations, apply the rules to a session agent or SIP interface as "inbound" or "outbound."

Configure SIP Manipulation

Descriptions of the applicable Oracle Enterprise Communications Broker configuration fields are provided below:

1. Access the SIP Manipulation configuration object.
Configuration, System Administration, SIP Manipulation.
2. On the SIP Manipulation page, do the following:

Name	Enter the unique identifier for this SIP Manipulation. Use this name when applying this SIP manipulation. Default: Blank.
Description	Enter a text description for this SIP Manipulation.
Split Headers	Enter a comma separated list of headers that the system separates prior to executing the manipulation. Example values: Allow,P-Asserted-Identity Diversion,Allow.
Join Headers	Enter a comma separated list of headers that the system joins together after the manipulation execution is complete. Example values: Allow,P-Asserted-Identity Diversion,Allow.

3. (Optional) Configure header rules and element rules. See "Configure a Header Rule" and "Configure an Element Rule."

Configure a SIP Manipulation Header Rule

You can configure SIP header rules and element rules on the Oracle Enterprise Communications Broker (OECB) from the "CfgRules" section of the "SIP Manipulations" page.

In the following procedure, you set the SIP Header Manipulation, Header Rule, and Element Rule parameters.

1. Access the SIP Manipulation configuration object.
Configuration, System Administration, SIP Manipulation.
2. On the SIP Manipulation configuration page, do one of the following:
 - a. Select an existing SIP manipulation configuration from the table, right-click, and click **Edit**. (Subsequent SIP manipulation pages use "Modify" in the title.)
 - b. Click **Add**. (Subsequent SIP manipulation pages use "Add" in the title.)
3. On the Add or Modify SIP Manipulation page, do one of the following.
 - a. If you chose **Add**, you must enter a name for this SIP Manipulation. (You can optionally complete the Description, Split Headers, and Join Headers parameters, at this time. See "Configure SIP Manipulation.") Proceed to the next step.
 - b. If you chose to edit an existing configuration, proceed to the next step.
4. On the Add or Modify SIP Manipulation page under Cgf Rules, click **Add** and click **header-rule**.
5. On the Add SIP Manipulation / Header Rule page, do the following.

Name	Enter a unique name for this rule set. Valid values: Alpha-numeric.
Header Name	Enter the name of the header on which you want the OECB to use this HMR. Set this parameter to @status-line, where the at-sign (@)—not allowed in SIP header names—to prevent undesired matches with header having the name status-code. Default: Blank.
Action	Select an action from the drop-down list for the header rule. Default: None. Valid values: Add Delete Find Replace All Log Manipulate Monitor None Reject SIP Manip Store.

Comparison Type	Specify how the OECB processes the match rules against the SIP header. Default: Refer Case Sensitive. Valid values: Boolean Case Insensitive Case Sensitive Pattern Rule Refer Case Insensitive Refer Case Sensitive.
Msg Type	Specify the message type this rule applies to. Default: Any. Valid Values: Any Out of Dialog Reply Request Out of Dialog.
Methods	Enter the method type to use when this SIP HMR is used. Default: Blank. Valid values" ACK CANCEL INVITE. When you do not set the method, the OECB applies the rule to all SIP methods.
Match Value	Enter the value to match against the header value in SIP packets; the OECB matches these against the entire SIP header value. This is where you can enter values to match using regular expression values. Your entries can contain Boolean operators. When you configure HMR (using SIP manipulation rules and elements rules, you can use escape characters in the match-value parameter to support escaping Boolean and string manipulation operators. You can also escape the escape character itself, so that it is used as a literal string. For example, the OECB treats the string \+1234 as +1234. The following are escape characters: +, -, +^, -^, &, , \, (,), ., \$, ^, and ". You can also use the variables, \$REMOTE_PORT and \$LOCAL_PORT, which resolve respectively to the far-end and remote UDP or TCP port value.
New Value	When you set the action parameter Add or to Manipulate, enter the new value that you want to substitute for the entire header value. This is where you can set stored regular expression values for the OECB to use when it adds or manipulates SIP headers. When you configure HMR (using SIP manipulation rules and elements rules, you can use escape characters in the match-value parameter to support escaping Boolean and string manipulation operators. You can also escape the escape character itself, so that it is used as a literal string. For example, the OECB treats the string \+1234 as +1234. The following are escape characters: +, -, +^, -^, &, , \, (,), ., \$, ^, and ". You can also use the variables, \$REMOTE_PORT and \$LOCAL_PORT, which resolve respectively to the far-end and remote UDP or TCP port value.
Cfg Rules	(Optional) Click Add, element-rule , and do the following. <ul style="list-style-type: none"> • Name—Enter a unique name for this header element rule. You can enter up to 128 alphanumeric characters with no spaces. The name can include the _, ., or - characters, cannot begin with either the . or the - characters. • Parameter name—Enter the parameter name to apply to the rule. • Type—Select the element type to which to apply this rule. • Action—Select an action from the drop-down list to apply to the element rule. Default: None. • Match Val Type—Select a match value type that this rule applies to from the drop-down list. Default: Any.

- Comparison Type—Select an element type from the drop-down list to which to apply the rule. Default: Case Sensitive.
- Match Value—Enter the match value to compare against the current object. (To clear the value, enter and empty string.)
- New Value—Enter a new value for the object. Quoted display named must be escaped within quotes. For example, \"MyName\". (To clear the value, enter and empty string.)
- Click **OK**. The system displays the SIP Manipulation / Header Rule page.

Do one of the following:

- Add another Element Rule.
- Finish the Header Rule configuration by completing the following steps.

6. Click **Back**.

The system displays the Modify SIP Manipulation page.

7. Click **Back**.

The system displays the SIP Manipulation page.

8. Save the configuration.

Configure a MIME Rule

You can configure Multi-Purpose Internet Mail Extensions (MIME) header rules and element rules on the Oracle Enterprise Communications Broker (OECB) from the "CfgRules" section of the "SIP Manipulations" page.

In the following procedure, you set the SIP Header Manipulation, MIME Rule, and MIME Header Rule parameters.

1. Access the SIP Manipulation configuration object.

Configuration, System Administration, SIP Manipulation.

2. On the SIP Manipulation configuration page, do one of the following:

- a. Select an existing SIP manipulation configuration from the table, right-click, and click **Edit**. (Subsequent SIP manipulation pages use "Modify" in the title.)
- b. Click **Add**. (Subsequent SIP manipulation pages use "Add" in the title.)

3. On the Add or Modify SIP Manipulation page, do one of the following.

- a. If you chose **Add**, you must enter a name for this SIP Manipulation. (You can optionally complete the Description, Split Headers, and Join Headers parameters, at this time. See "Configure SIP Manipulation.") Proceed to the next step.
- b. If you chose to edit an existing configuration, proceed to the next step.

4. On the Add or Modify SIP Manipulation page under Cfg Rules, click **Add** and click **mime-rule**.

5. On the Add or Modify SIP Manipulation / Mime Rule page, do the following.

Name	Enter a unique name for this rule set. Valid values: Alpha-numeric.
Content Type	Enter the name of the header on which you want the OECB to use this HMR. Set this parameter to @status-line, where the at-sign (@)—not allowed in SIP header names—to prevent undesired matches with header having the name status-code. Default: Blank.
Msg Type	Specify the message type this rule applies to. Default: Any. Valid Values: Any Out of Dialog Reply Request Out of Dialog.
Methods	Enter the method type to use when this SIP HMR is used. Default: Blank. Valid values" ACK CANCEL INVITE. When you do not set the method, the OECB applies the rule to all SIP methods.
Format	Select the encode - decode format from the drop-down list for the MIME content.
Action	Select an action from the drop-down list for the header rule. Default: None. Valid values: Add Delete Find Replace All Log Manipulate Monitor None Reject SIP Manip Store.
Comparison Type	Specify how the OECB processes the match rules against the SIP header. Default: Refer Case Sensitive. Valid values: Boolean Case Insensitive Case Sensitive Pattern Rule Refer Case Insensitive Refer Case Sensitive.
Match Value	Enter the value to match against the header value in SIP packets; the OECB matches these against the entire SIP header value. This is where you can enter values to match using regular expression values. Your entries can contain Boolean operators. When you configure HMR (using SIP manipulation rules and elements rules, you can use escape characters in the match-value parameter to support escaping Boolean and string manipulation operators. You can also escape the escape character itself, so that it is used as a literal string. For example, the OECB treats the string \+1234 as +1234. The following are escape characters: +, -, +^, -^, &, , \, (,), ., \$, ^, and ". You can also use the variables, \$REMOTE_PORT and \$LOCAL_PORT, which resolve respectively to the far-end and remote UDP or TCP port value.
New Value	When you set the action parameter Add or to Manipulate, enter the new value that you want to substitute for the entire header value. This is where you can set stored regular expression values for the OECB to use when it adds or manipulates SIP headers. When you configure HMR (using SIP manipulation rules and elements rules, you can use escape characters in the match-value parameter to support escaping Boolean and string manipulation operators. You can also escape the escape character itself, so that it is used as a literal string. For example, the OECB treats the string \+1234 as +1234. The following are escape characters: +, -, +^, -^, &, , \, (,), ., \$, ^, and ". You can also use the variables, \$REMOTE_PORT and \$LOCAL_PORT, which resolve respectively to the far-end and remote UDP or TCP port value.
Cfg Rules	Click Add, mime-header-rule , and do the following.

- Name—Enter a unique name for this header element rule. You can enter up to 128 alphanumeric characters with no spaces. The name can include the `_`, `.`, or `-` characters, but cannot begin with either the `.` or the `-` characters.
- Mime Header Name—Enter header name within the MIME part to which to apply the rule. Use `headername@peramble` to change the preamble of a SIP body. Use `headername@epilogue` to change the epilog of a SIP body.
- Action—Select an action from the drop-down list to apply to the element rule. Default: None.
- Comparison Type—Select the type of comparison from the drop-down list to use for the match value. Default: Match Value. (To clear the value, enter and empty string.)
- Match Value—Enter the match value to compare against the current object. (To clear the value, enter and empty string.)
- New Value—Enter a new value for the object. Quoted display named must be escaped within quotes. For example, `\\"MyName\"`. (To clear the value, enter and empty string.)
- Click **OK**. The system displays the SIP Manipulation / Mime Rule dialog.

Do one of the following:

- Add another mime-header-rule.
- Finish the MIME Rule configuration by completing the following steps.

6. Click **Back**.

The system displays the Add or Modify SIP Manipulation page.

7. Click **Back**.

The system displays the SIP Manipulation page.

8. Save the configuration.

Configure a MIME ISUP Rule

You can configure Multi-Purpose Internet Mail Extensions (MIME) header rules and element rules on the Oracle Enterprise Communications Broker (OECB) from the "CfgRules" section of the "SIP Manipulations" page.

In the following procedure, you set the SIP Header Manipulation, MIME ISUP Rule, MIME Header Rule, and ISUP Param Rule parameters.

1. Access the SIP Manipulation configuration object.

Configuration, System Administration, SIP Manipulation.

2. On the SIP Manipulation configuration page, do one of the following:

- a. Select an existing SIP manipulation configuration from the table, right-click, and click **Edit**. (Subsequent SIP manipulation pages use "Modify" in the title.)
- b. Click **Add**. (Subsequent SIP manipulation pages use "Add" in the title.)

3. On the Add or Modify SIP Manipulation page, do one of the following.
 - a. If you chose **Add**, you must enter a name for this SIP Manipulation. (You can optionally complete the Description, Split Headers, and Join Headers parameters, at this time. See "Configure SIP Manipulation.") Proceed to the next step.
 - b. If you chose to edit an existing configuration, proceed to the next step.
4. On the Add or Modify SIP Manipulation page under Cfg Rules, click **Add** and click **mime-header-rule**.
5. On the Add or Modify SIP Manipulation / Mime ISUP Rule page, do the following.

Name	Enter a unique name for this rule set. Valid values: Alpha-numeric.
Content Type	Enter the name of the header on which you want the OECB to use this HMR. Set this parameter to @status-line, where the at-sign (@)—not allowed in SIP header names—to prevent undesired matches with header having the name status-code. Default: Blank.
Msg Type	Specify the message type this rule applies to. Default: Any. Valid Values: Any Out of Dialog Reply Request Out of Dialog.
Methods	Enter the method type to use when this SIP HMR is used. Default: Blank. Valid values" ACK CANCEL INVITE. When you do not set the method, the OECB applies the rule to all SIP methods.
Format	Select the encode - decode format from the drop-down list for the MIME content.
Action	Select an action from the drop-down list for the header rule. Default: None. Valid values: Add Delete Find Replace All Log Manipulate Monitor None Reject SIP Manip Store.
Comparison Type	Specify how the OECB processes the match rules against the SIP header. Default: Refer Case Sensitive. Valid values: Boolean Case Insensitive Case Sensitive Pattern Rule Refer Case Insensitive Refer Case Sensitive.
Match Value	Enter the value to match against the header value in SIP packets; the OECB matches these against the entire SIP header value. This is where you can enter values to match using regular expression values. Your entries can contain Boolean operators. When you configure HMR (using SIP manipulation rules and elements rules, you can use escape characters in the match-value parameter to support escaping Boolean and string manipulation operators. You can also escape the escape character itself, so that it is used as a literal string. For example, the OECB treats the string \+1234 as +1234. The following are escape characters: +, -, +^, -^, &, , \, (,), ., \$, ^, and ". You can also use the variables, \$REMOTE_PORT and \$LOCAL_PORT, which resolve respectively to the far-end and remote UDP or TCP port value.
New Value	When you set the action parameter Add or to Manipulate, enter the new value that you want to substitute for the entire header value. This is where you can set stored regular expression values for the OECB to use when it adds or manipulates SIP headers.

When you configure HMR (using SIP manipulation rules and elements rules, you can use escape characters in the **match-value** parameter to support escaping Boolean and string manipulation operators. You can also escape the escape character itself, so that it is used as a literal string. For example, the OECB treats the string \+1234 as +1234. The following are escape characters: +, -, +^, -^, &, |, \, (,), ., \$, ^, and ". You can also use the variables, \$REMOTE_PORT and \$LOCAL_PORT, which resolve respectively to the far-end and remote UDP or TCP port value.

Cfg Rules
(instructions
for configuring
MIME
HeaderRule)

Click **Add, MIME Header Rule**, and do the following.

- Name—Enter a unique name for this header element rule.
- Header Name—Enter header name within the MIME part to which to apply the rule.
- Action—Select an action from the drop-down list to apply to the element rule.
- Comparison Type—Select the type of comparison from the drop-down list to use for the match value. (To clear the value, enter and empty string.)
- Match Value—Enter the match value to compare against the current object. (To clear the value, enter and empty string.)
- New Value—Enter a new value for the object. Quoted display named must be escaped within quotes. For example, \"MyName\" .
- Click **OK**. The system displays the SIP manipulation / Mime isup rule dialog.

Do one of the following:

- Add another MIME Header Rule.
- Add an ISUP Param Rule, using the steps in the following table cell.
- Finish the MIME ISUP rule configuration by completing steps 3-6.

Cfg Rules
(instructions
for configuring
ISUP Param
Rule)

Click **Add, isup-param-rule**, and do the following.

- Name—Enter a unique name for this header element rule.
- Type—Enter the parameter type that specifies the part of the isup body to manipulate.
- Format—Select a format from the drop down list for the encode - decode mode of the binary body form string form-ascii.
- Action—Select an action from the drop-down list to apply to the element rule.
- Comparison Type—Select the type of comparison from the drop-down list to use for the match value. (To clear the value, enter and empty string.)

- Match Value—Enter the match value to compare against the current object. (To clear the value, enter an empty string.)
- New Value—Enter a new value for the object. Quoted display names must be escaped within quotes. For example, "\"MyName\" " .
- Click **OK**. The system displays the SIP manipulation / Mime isup rule dialog.

Do one of the following:

- Add another ISUP Param Rule.
- Finish the MIME ISUP Rule configuration by completing the following steps.

6. Click **Back**.

The system displays the Add or Modify SIP Manipulation page.

7. Click **Back**.

The system displays the SIP Manipulation page.

8. Save the configuration.

Configure a MIME SDP Rule

You can configure Multi-Purpose Internet Mail Extensions (MIME) header rules and element rules on the Oracle Enterprise Communications Broker (OECB) from the "CfgRules" section of the "SIP Manipulations" page.

In the following procedure, you set the SIP Header Manipulation, MIME SDP Rule, MIME Header Rule, SDP Session Rule, and SDP Media Rule parameters.

1. Access the SIP Manipulation configuration object.

Configuration, System Administration, SIP Manipulation.

2. On the SIP Manipulation configuration page, do one of the following:

- a. Select an existing SIP manipulation configuration from the table, right-click, and click **Edit**. (Subsequent SIP manipulation pages use "Modify" in the title.)
- b. Click **Add**. (Subsequent SIP manipulation pages use "Add" in the title.)

3. On the Add or Modify SIP Manipulation page, do one of the following.

- a. If you chose **Add**, you must enter a name for this SIP Manipulation. (You can optionally complete the Description, Split Headers, and Join Headers parameters, at this time. See "Configure SIP Manipulation.") Proceed to the next step.
- b. If you chose to edit an existing configuration, proceed to the next step.

4. On the Add or Modify SIP Manipulation page under Cfg Rules, click **Add** and click **mime-sdp-rule**.

5. In the Add or Modify SIP Manipulation / MIME SDP Rule page, do the following.

Name	Enter a unique name for this rule set. Valid values: Alpha-numeric.
------	---

Content Type	Enter the name of the header on which you want the OECB to use this HMR. Set this parameter to @status-line, where the at-sign (@)—not allowed in SIP header names—to prevent undesired matches with header having the name status-code. Default: Blank.
Msg Type	Specify the message type this rule applies to. Default: Any. Valid Values: Any Out of Dialog Reply Request Out of Dialog.
Methods	Enter the method type to use when this SIP HMR is used. Default: Blank. Valid values" ACK CANCEL INVITE. When you do not set the method, the OECB applies the rule to all SIP methods.
Format	Select the encode - decode format from the drop-down list for the MIME content.
Action	Select an action from the drop-down list for the header rule. Default: None. Valid values: Add Delete Find Replace All Log Manipulate Monitor None Reject SIP Manip Store.
Comparison Type	Specify how the OECB processes the match rules against the SIP header. Default: Refer Case Sensitive. Valid values: Boolean Case Insensitive Case Sensitive Pattern Rule Refer Case Insensitive Refer Case Sensitive.
Match Value	Enter the value to match against the header value in SIP packets; the OECB matches these against the entire SIP header value. This is where you can enter values to match using regular expression values. Your entries can contain Boolean operators. When you configure HMR (using SIP manipulation rules and elements rules, you can use escape characters in the match-value parameter to support escaping Boolean and string manipulation operators. You can also escape the escape character itself, so that it is used as a literal string. For example, the OECB treats the string \+1234 as +1234. The following are escape characters: +, -, +^, -^, &, , \, (,), ., \$, ^, and ". You can also use the variables, \$REMOTE_PORT and \$LOCAL_PORT, which resolve respectively to the far-end and remote UDP or TCP port value.
New Value	When you set the action parameter Add or to Manipulate, enter the new value that you want to substitute for the entire header value. This is where you can set stored regular expression values for the OECB to use when it adds or manipulates SIP headers. When you configure HMR (using SIP manipulation rules and elements rules, you can use escape characters in the match-value parameter to support escaping Boolean and string manipulation operators. You can also escape the escape character itself, so that it is used as a literal string. For example, the OECB treats the string \+1234 as +1234. The following are escape characters: +, -, +^, -^, &, , \, (,), ., \$, ^, and ". You can also use the variables, \$REMOTE_PORT and \$LOCAL_PORT, which resolve respectively to the far-end and remote UDP or TCP port value.
Cfg Rules (instructions)	(Optional) Click Add, mime-header-rule , and do the following.

<p>for configuring mime-header-rule)</p>	<ul style="list-style-type: none"> • Name—Enter a unique name for this header element rule. • Mime Header Name—Enter header name within the MIME part to which to apply the rule. • Action—Select an action from the drop-down list to apply to the element rule. • Comparison Type—Select the type of comparison from the drop-down list to use for the match value. • Match Value—Enter the match value to compare against the current object. • New Value—Enter a new value for the object. Quoted display named must be escaped within quotes. For example, \"MyName\" . • Click OK. The system displays the SIP manipulation / Mime sdp rule dialog. <p>Do one of the following:</p> <ul style="list-style-type: none"> • Add another mime-header-rule. • Configure the sdp-session-rule and sdp-media-rule options, using the steps in the following table cells. • Finish the MIME SDP rule configuration by completing steps 3-6.
<p>Cfg Rules (instructions for configuring sdp-session-rule)</p>	<p>(Optional) Click Add, sdp-session-rule , and do the following.</p> <ul style="list-style-type: none"> • Name—Enter a unique name for this header element rule. • Action—Select an action from the drop-down list to apply to the this rule. • Comparison Type—Select the type of comparison from the drop-down list to use for the match value. • Match Value—Enter the match value to compare against the current object. • New Value—Enter a new value for the object. Quoted display named must be escaped within quotes. For example, \"MyName\". • CfgRules—(Optional) Click Add, sdp-line-rule. • Name—Enter a unique name for this rule. • Type—Enter a descriptor type to specify the SDP line to manipulate. • Action—Select an action from the drop-down list to apply to this rule. • Comparison Type—Select the type of comparison from the drop-down list to use for the match value. • Match Value—Enter the match value to compare against the current object.

- New Value—Enter a new value for the object. Quoted display named must be escaped within quotes. For example, \"MyName\".
- Click **OK**. The system displays the SIP manipulation / Mime sdp rule / Sdp session rule dialog.
- (Optional) Add another sdp-line-rule.
- Click **OK**. The system displays the SIP manipulation / Mime sdp rule dialog.

Do one of the following:

- Add another sdp-session-rule.
- Configure the mime-header-rule and sdp-media-rule options, using the steps in the corresponding table cells in this procedure.
- Finish the MIME SDP rule configuration by completing steps 3-6.

Cfg Rules
(instructions
for
configuring
sdp-media-
rule)

(Optional) Click **Add, sdp-media-rule**.

- Name—Enter a unique name for this header element rule.
- Media Type—Enter the media type to manipulate. For example, audio or video.
- Action—Select an action from the drop-down list to apply to the element rule.
- Comparison Type—Select the type of comparison from the drop-down list to use for the match value.
- Match Value—Enter the match value to compare against the current object.
- New Value—Enter a new value for the object. Quoted display named must be escaped within quotes. For example, \"MyName\" .
- Click **OK**.
- CfgRules—(Optional) Click **Add, sdp-line-rule**.
- Name—Enter a unique name for this rule.
- Type—Enter a descriptor type to specify the SDP line to manipulate.
- Action—Select an action from the drop-down list to apply to this rule.
- Comparison type—Select the type of comparison from the drop-down list to use for the match value. (To clear the value, enter and empty string.)
- Match Value—Enter the match value to compare against the current object. (To clear the value, enter and empty string.)

- New Value—Enter a new value for the object. Quoted display named must be escaped within quotes. For example, "MyName\."
- Click **OK**. The system displays the SIP manipulation / Mime sdp rule / Sdp media rule dialog.
- (Optional) Add another sdp-line-rule.
- Click **OK**. The system displays the SIP manipulation / Mime sdp rule dialog.

Do one of the following:

- Add another sdp-media-rule.
- Finish the MIME SDP rule configuration by completing the following steps.

6. Click **Back**.

The system displays the Add or Modify SIP Manipulation page.

7. Click **Back**.

The system displays the SIP Manipulation page.

8. Save the configuration.

Monitoring Tab Operations

The Monitoring tab displays tools to help you see the results of system and session data from the Oracle Enterprise Communications Broker. The Monitor and Trace link provides summary reports that include session data, ladder diagrams of call and media flows, and Quality of Service statistics. The Widgets link displays Widgets that you can use to monitor the system in a graphical or textual format.

Monitor and Trace

Monitor and Trace displays the results of filtered SIP session data from the Oracle Enterprise Communications Broker. Each summary page displays the results in a common log format for local viewing, which you can export as HTML and text files.

Monitor and Trace supports the following summary reports.

- Notable Events Summary
- Registrations Summary
- Sessions Summary
- Subscriptions Summary

Each summary provides sorting, searching, paging, and exporting functionality, as well as a link to a ladder diagram view where you can see a session summary, session details, and QoS statistics. You can choose which of the available columns that you want each summary to display.

The Monitor and Trace function can store messages per session and it can store cumulative sessions across all report types. When the sessions maximum is reached, the system removes the oldest call and adds the newest call.

- On systems with less than 4GB of RAM, the system can store:
 - 50 messages
 - 2,000 sessions
- On systems with more than 4GB of RAM, the system can store:
 - 50 messages
 - 4,000 sessions

The call database is not persistent across re-boots

The system can perform live paging from Monitor and Trace tables.

 **Note:**

Only one user at a time can view Monitor and Trace information. Monitor and Trace does not support multiple, simultaneous viewers.

SIP Notable Events Summary

The SIP Notable Events Summary contains all logged sessions that have a notable event on the Oracle Enterprise Communications Broker (OECB) associated with the session. The columns that display on the Notable Events Summary page depend on the columns that you selected in the "Customizing the Page Display" procedure.

The following table describes the columns that Notable Events Summary page can display.

Start Time	Timestamp of the first SIP message in the call session.
State	Status of the call or media event session. Valid values: INITIAL—Session for which an INVITE or SUBSCRIBE was forwarded. EARLY—Session received the first provisional response (1xx other than 100). ESTABLISHED—Session for which a success (2xx) response was received. TERMINATED—Session that has ended by receiving or sending a BYE for an "Established" session or forwarding an error response for an "Initial" or Early session. The session remains in the terminated state until all the resources for the session are freed up. FAILED Session that has failed due to a 4xx or 5xx error code.
Call ID	Identification of the call source. Includes the phone number and source IP address.
Request URI	Uniform Resource Identifier (URI) formatted string that identifies a resource through a protocol, name, location, and any other applicable characteristic, and is sent by the OECB in REQUEST headers.
From URI	URI formatted string that identifies the call source information.
To URI	URI formatted string that identifies the call destination information.
Ingress Realm	Name of the inbound realm.
Egress Realm	Name of the outbound realm.
Notable Event	Indicates if a notable event has occurred on the call session. Valid values: short session—Sessions that don't meet a minimum configurable duration threshold; Session dialogue, captured media information and termination signalling; Any event flagged as a short session interesting event. local rejection—Sessions locally rejected at the OECB for any reason (for example, Session Agent (SA) unavailable, no route found, SIP signalling error, etc.); Session dialogue, capture media information and termination signaling; Any event flagged as a local rejection interesting event.

Session ID	Identification assigned to the call session.
Ingress Src Addr	Source IP address of the incoming call or media event.
Ingress Src Port	Source port of the incoming call or media event.
Egress Dest Addr	Destination IP address of the outgoing call or media event.
Object ID	ID number of the object in a row. Use to aid troubleshooting.

Display a Notable Events Report

1. Access Notable Events: **Monitoring, Monitor and Trace, Notable Events.**
2. Use the controls on the page to view information about the records in this report.

SIP Registrations Summary

The SIP Registrations Summary displays a summary of all logged SIP registrations sessions on the Oracle Enterprise Communications Broker (OECB). The columns that display on the Registrations Summary page depend on the columns you selected in the "Customizing the Page Display" procedure.

The following table describes the columns available on the Registrations Summary page.

Start Time	Timestamp of the first SIP message in the call session.
Call ID	Identification of the call source. Includes the phone number and source IP address.
From URI	URI formatted string that identifies the call source information.
To URI	URI formatted string that identifies the call destination information.
Local Expires	The current setting for the expiration of a registration request sent from the Integrated Media Gateway (IMG) to a Remote SIP User Agent. Default: 3600 seconds.
Remote Expires	The current setting for the expiration of a registration request sent from the Remote SIP User Agent to the Integrated Media Gateway (IMG). Default: 3600 seconds.
Ingress Realm	Incoming realm name.
Egress Realm	Outgoing realm name.
Notable Event	Indicates a notable event that occurred on the call session. Valid value: local rejection - Sessions locally rejected at the E-SBC for any reason (for example, Session Agent (SA) unavailable, no route found, SIP signalling error, etc.); Session dialogue, capture media

	information and termination signalling; Any event flagged as a local rejection interesting event
Session ID	Identification assigned to the call session.
Ingress Src Addr	Source IP address of the incoming call or media event.
Egress Dest Addr	Destination IP address of the outgoing call or media event.
Request URI	Uniform Resource Identifier (URI) formatted string that identifies a resource with the protocol, name, location, and any other applicable characteristic. The SBC only sends the URI in REQUEST headers.
Object ID	ID number of the object in a row. Use to aid troubleshooting.

Display a Registrations Report

1. Access Registrations: **Monitoring, Monitor and Trace, Registrations.**
2. Use the controls on the page to view information about the records in this report.

SIP Sessions Summary

The SIP Sessions Summary is a SIP session summary of all logged call sessions on the Oracle Enterprise Communications Broker (OECB). When you enable Lightweight Directory Access Protocol (LDAP) on the Active Directory, LDAP session messages may also display. The columns that display on the Sessions Summary page depend on the columns that you specified in the "Customizing the Page Display" procedure.

The following table describes the columns on the SIP Session Summary page.

Start Time	Timestamp of the first SIP message in the call session.
State	Status of the call or media session. Valid values are: INITIAL—Session for which an INVITE or SUBSCRIBE was forwarded. EARLY—Session that received the first provisional response (1xx other than 100). ESTABLISHED—Session for which a success (2xx) response was received. TERMINATED—Session that ended by receiving or sending a BYE for an "Established" session or forwarding an error response for an "Initial" or "Early" session. The session remains in the terminated state until all the resources for the session are freed up. FAILED—Session that failed due to a 4xx or 5xx error code.
Call ID	Identification of the call source. Includes the phone number and source IP address.
Request URI	Uniform Resource Identifier (URI) formatted string that identifies a resource by way of a protocol, name, location, and any other

	applicable characteristic that is sent by the OECB in REQUEST headers.
From URI	URI formatted string that identifies the call source information.
To URI	URI formatted string that identifies the call destination information.
Ingress Realm	Name of the inbound realm.
Egress Realm	Name of the outbound realm.
Duration	Amount of time, in seconds, that the call or media event was active.
Notable Event	Indicates if a notable event has occurred on the call session. Valid values are: Short Session—Sessions that do not meet a minimum configurable duration threshold. Session dialogue, captured media information, and termination signalling. Any event flagged as a short session interesting event. Local Rejection—Sessions locally rejected at the OECB for any reason, for example, Session Agent (SA) unavailable, no route found, SIP signaling error, and so on. Session dialogue, capture media information, and termination signaling. Any event flagged as a local rejection interesting event.
Session ID	Identification assigned to the call session.
Ingress Src Addr	Source IP address of the incoming call or media event.
Egress Dest Addr	Destination IP address of the outgoing call or media event.
Calling Pkts	
Called Pkts	
Calling R	
Called R	
Calling MOS	
Called MOS	
Ingress Src Port	Source port of the incoming call or media event.
Object ID	ID number of the object in a row. Use to aid troubleshooting.

Display a Sessions Report

1. Access SIP Session Summary: **Monitoring, Monitor and Trace, Sessions**.
The system displays the SIP Sessions Summary page.
2. Use the controls on the page to manage the records in the report.

SIP Message Details

The SIP Message Detail window displays detailed information and data flow (ingress and egress) about the call or media event.

[+] Session Summary				
172.16.34.10	172.16.34.226	192.168.34.226	192.168.34.17	
2012-04-27 08:08:27.550	→ INVITE (1)	→		
2012-04-27 08:08:27.552	← Status:100 (1)	←		
MEDIA FLOW ADD, ID=65783, DIRECTION=CALLING				
MEDIA FLOW ADD, ID=65784, DIRECTION=CALLED				
2012-04-27 08:08:27.562	EGRESS ROUTE, TYPE=local-policy, NEXT HOP=sip.service@192.168.34.17:5060			
2012-04-27 08:08:27.562			→ INVITE (1)	→
2012-04-27 08:08:27.564			← Status:180 (1)	←
2012-04-27 08:08:27.568	← Status:180 (1)	←		
2012-04-27 08:08:27.569			← Status:200 (1)	←
MEDIA FLOW MODIFY, ID=65783, DIRECTION=CALLING				
2012-04-27 08:08:27.578	← Status:200 (1)	←		
2012-04-27 08:08:27.580	→ ACK (1)	→		
2012-04-27 08:08:27.584			→ ACK (1)	→
2012-04-27 08:09:14.732	→ BYE (2)	→		
2012-04-27 08:09:14.736			→ BYE (2)	→
SIP Message Details				
[+] QoS Stats				

When a session is routed using the a Lightweight Directory Access Protocol (LDAP) configuration (Active Directory) for the local policy, the LDAP information displays in the Session Summary window. The next hop value containing "enum:..." or "dns:..." displays. Similarly, the next hop value "ldap:..." displays for LDAP queries.

[+] Session Summary				
192.168.204.64	192.168.204.71	172.16.204.67	172.16.204.64	
2012-07-09 15:30:58.328	→ INVITE (1)	→		
2012-07-09 15:30:58.334	← Status:100 (1)	←		
MEDIA FLOW ADD, ID=65536, DIRECTION=CALLING				
MEDIA FLOW ADD, ID=65537, DIRECTION=CALLED				
2012-07-09 15:30:58.371	EGRESS ROUTE, TYPE=local-policy, NEXT HOP=ldap:lookup			
2012-07-09 15:30:58.371			→ INVITE (1)	→
2012-07-09 15:30:58.625			← Status:180 (1)	←
2012-07-09 15:30:58.633	← Status:180 (1)	←		
2012-07-09 15:30:58.729			← Status:200 (1)	←
MEDIA FLOW MODIFY, ID=65536, DIRECTION=CALLING				
2012-07-09 15:30:58.754	← Status:200 (1)	←		
2012-07-09 15:30:59.020	→ ACK (1)	→		
2012-07-09 15:30:59.028			→ ACK (1)	→
2012-07-09 15:31:01.754	→ BYE (2)	→		
2012-07-09 15:31:01.763			→ BYE (2)	→
2012-07-09 15:31:01.889			← Status:200 (2)	←
2012-07-09 15:31:01.900	← Status:200 (2)	←		
MEDIA FLOW DELETE, ID=65536, DIRECTION=CALLING				
MEDIA FLOW DELETE, ID=65537, DIRECTION=CALLED				
SIP Message Details				
[+] QoS Stats				

SIP Subscriptions Summary

The SIP Subscriptions Report displays a summary of all logged SIP subscription sessions on the Oracle Enterprise Communications Broker (OECB). The columns that

display on the Subscription Report page depend on the columns you selected in the "Customizing the Page Display" procedure.

The following table describes the columns on Subscriptions Report page.

Start Time	Timestamp of the first SIP message in the call session.
Call ID	Identification of the call source. Includes the phone number and source IP address.
From URI	URI formatted string that identifies the call source information.
To URI	URI formatted string that identifies the call destination information.
Events	<p>Specific subscribe event package that was sent from an endpoint to the destination endpoint. Applicable event packages can be:</p> <p>conference—Event package that allows users to subscribe to a conference Uniform Resource Identifier (URI).</p> <p>consent—pending additions - Event package used by SIP relays to inform user agents about the consent-related status of the entries to be added to a resource list.</p> <p>dialog—Event package that allows users to subscribe to another user, and receive notifications about the changes in the state of the INVITE-initiated dialogs in which the user is involved.</p> <p>message—summary - Event package that carries message-waiting status and message summaries from a messaging system to an interested User Agent (UA).</p> <p>presence—Event package that conveys the ability and willingness of a user to communicate across a set of devices. A presence protocol is a protocol for providing a presence service over the Internet or any IP network.</p> <p>reg—Event package that provides a way to monitor the status of *all* the registrations for a particular Address of Record (AoR).</p> <p>refer—Event package that provides a mechanism to allow the party sending the REFER to be notified of the outcome of a referenced request.</p> <p>winfo—Event package for watcher information. It tracks the state of subscriptions to a resource in another package.</p> <p>vq-rtcpX—Event package that collects and reports the metrics that measure quality for RTP sessions.</p>
Expires	The current setting for the expiration of a registration request. Default: 3600 sec.
Ingress Realm	Incoming realm name.
Egress Realm	Outgoing realm name.

Notable Event	Indicates if a notable event has occurred on the call session. Valid value is: local rejection - Sessions locally rejected at the OECB for any reason (for example, Session Agent (SA) unavailable, no route found, SIP signalling error, etc.); Session dialogue, capture media information and termination signaling; Any event flagged as a local rejection interesting event
Session ID	Identification assigned to the call session.
State	Status of the call or media session. Valid values are: INITIAL—Session for which an INVITE or SUBSCRIBE was forwarded. EARLY—Session that received the first provisional response (1xx other than 100). ESTABLISHED—Session for which a success (2xx) response was received. TERMINATED—Session that ended by receiving or sending a BYE for an “Established” session or forwarding an error response for an “Initial” or “Early” session. The session remains in the terminated state until all the resources for the session are freed up. FAILED—Session that failed due to a 4xx or 5xx error code.
Ingress Src Addr	Source IP address of the incoming call or media event.
Egress Dest Addr	Destination IP address of the outgoing call or media event.
Request URI	Uniform Resource Identifier (URI) formatted string that identifies a resource via a protocol, name, location, and any other applicable characteristic, and is sent by the OECB in REQUEST headers.
Object ID	ID number of the object in a row. Use to aid troubleshooting.

Display a Subscriptions Report

1. Access Subscriptions: **Monitoring, Monitor and Trace, Subscriptions.**
2. Use the controls on the page to view information about the records in this report.

Ladder Diagrams and Information Display Controls

A ladder diagram is a graphical representation of the flow of call and media packets on ingress and egress routes through the Oracle Enterprise Communications Broker (OECB), which can help you with troubleshooting efforts. The Web GUI can display a ladder diagram for each of the summary reports available through Monitor and Trace.

To display a ladder diagram for a specific record in a Summary Report, select a record in the summary table and right-click. Click **Ladder diagram** on the pop-up menu. Each ladder diagram contains the following sections.

- Session Summary—Displays session data. Use the [+] and [-] controls to toggle between show and hide.
- Ladder Diagram—Displays SIP message and call flow information. Hover over a line in the ladder diagram to see more information for the highlighted flow.
- QoS Statistics—Displays Quality of Service (QoS) information. Use the [+] and [-] controls to toggle between show and hide.

The following illustration shows an example of an OECB ladder diagram, and how you can make it display more information.

Click the + or the - to show or hide Session Summary Information. [+] [-] Session Summary

State: TERMINATED-300 Duration: 6

From URI: Anvil <cs:tripp@192.168.67.255:5060>-tag=8 To URI: sut <cs:service@192.168.67.193:5060>

Ingress Src IP:Port: :0 Egress Src IP:Port: 172.16.67.193:5060

Ingress Dest IP:Port: :0 Egress Dest IP:Port: 172.16.67.255:5060

Ingress Realm: access Egress Realm: com

Ingress Network Intf: m00 Egress Network Intf: m10

Ingress Transport: TCP Egress Transport: UDP

192.168.67.255 192.168.67.193 172.16.67.193 172.16.67.255

2019-06-11 11:48:50.390 INVITE (1)

2019-06-11 11:48:50.390 Status:100 (1)

2019-06-11 11:48:50.390 MEDIA FLOW ADD, ID=150994946, DIRECTION=CALLING

2019-06-11 11:48:50.390 MEDIA FLOW ADD, ID=150994946, DIRECTION=CALLED

2019-06-11 11:48:50.390 EGRESS ROUTE, TYPE=local-policy, NEXT_HOP=cs:cp:012@172.16.67.255:5060

2019-06-11 11:48:50.391 Status:180 (1)

2019-06-11 11:48:50.391 MEDIA FLOW MODIFY, ID=150994946, DIRECTION=CALLI

2019-06-11 11:48:50.391 Status:200 (1)

2019-06-11 11:48:50.430 ACK (1)

2019-06-11 11:48:50.430 BYE (2)

2019-06-11 11:48:56.433 Status:200 (2)

2019-06-11 11:48:56.434 MEDIA FLOW DELETE, ID=150994946, DIRECTION=CALLI

2019-06-11 11:48:56.434 MEDIA FLOW DELETE, ID=150994946, DIRECTION=CALLI

SIP Message Details

Click the + or the - to show or hide Quality of Service information. [+] [-] QoS Stats

Flow ID	Direction	Total Pkts Received	Total Octets Received	Pkt Lost	Avg Jitter	Max Jitter	RTCP			QoS		
							Avg Latency	Max Latency	Pkt Lost	Avg Jitter	Max Jitter	R-Factor
150994945	CALLING	0	0	0	0	0	0	0	0	0	0.00	0.00
150994946	CALLED	0	0	0	0	0	0	0	0	0	0.00	0.00

Additional controls: Refresh Export diagram Export session details Close

The following table describes the additional controls on a Ladder Diagram page.

Refresh	Refreshes the data on the page.
Export Diagram	Exports all of the information in the Ladder Diagram (Session Summary, SIP Message Details, and QoS Statistics) to a file in text format on the local machine.
Export Session Details	Exports detailed information about the SIP messages and media events associated with the session in focus to a file in text format on the local machine.
Close	Closes the Ladder Diagram page.



Note:

The OECB captures SIP messages, applies the Header Manipulation Rules (HMR) that you configured on the OECB, and applies the Session Plug-in Language (SPL) to that message. When the OECB sends the message, it applies the SPL, the HMR, and sends the captured SIP message. When viewing the session detail on a Ladder Diagram, the HMR and SPL information may be present.

Display a Ladder Diagram

You can display a ladder diagram of call and media flow from any of the Monitor and Trace summary pages in the Web GUI. Each summary page displays a table, where each row represents one session. You can view a ladder diagram for each session with the following procedure.

1. Access a Summary: **Monitoring, Monitor and Trace.**
2. Under **Monitor and Trace**, select one of the following summaries:
 - Notable Events
 - Registrations
 - Sessions
 - Subscriptions
3. On the Summary page, select a row in the table and right-click.
4. On the pop-up menu, click **Ladder Diagram**.
5. (Optional)—On the ladder diagram, click the [+] to expand the Session Summary and QoS Stats sections.

Session Summary

To see a Session Summary, open a ladder diagram from a record in a Summary report and click the [+] control that precedes "Session Summary" at the top of the page. Monitor and Trace displays the session data and statistics.

The following screen capture shows a sample Session Summary page generated from a selected item on a ladder diagram.

[-] Session Summary			
State	TERMINATED-200	Duration	47
From URI	sipp <sip:sipp@100.10.30.10:5060>;tag=24	To URI	sut <sip:service@100.10.30.226:5060>;tag=99
Ingress Src IP:Port	100.10.30.10:5060	Egress Src IP:Port	200.20.40.226:5060
Ingress Dest IP:Port	100.10.30.226:5060	Egress Dest IP:Port	200.20.40.17:5060
Ingress Realm	access	Egress Realm	backbone
Ingress Network Intf	access	Egress Network Intf	backbone
Ingress Transport	UDP	Egress Transport	UDP

The following table describes each field in the Session Summary report.

State	Status of the call or media session. Valid values are:
-------	--

	INITIAL—Session for which an INVITE or SUBSCRIBE was forwarded.
	EARLY—Session received the first provisional response (1xx other than 100).
	ESTABLISHED—Session for which a success (2xx) response was received.
	TERMINATED—Session that has ended by receiving or sending a BYE for an “Established” session or forwarding an error response for an “Initial” or Early session. The session remains in the terminated state until all the resources for the session are freed up.
	FAILED—Session that has failed due to a 4xx or 5xx error code.
Duration	Amount of time, in seconds, that the call or media session was active.
From URI	URI formatted string that identifies the call source information.
To URI	URI formatted string that identifies the call destination information.
Ingress Src IP:Port	Source IP address and port number of the incoming call or media session.
Egress Src IP:Port	Source IP address and port number of the outgoing call or media session.
Ingress Dest IP:Port	Destination IP address and port number of the incoming call or media session.
Egress Dest IP:Port	Destination IP address and port number of the outgoing call or media session.
Ingress Realm	Incoming realm name.
Egress Realm	Outgoing realm name.
Ingress Network Intf	Name of the incoming network interface on the Oracle Enterprise Communications Broker (OECB).
Egress Network Intf	Name of the outgoing network interface on the OECB.
Ingress Transport	Protocol type used on the incoming call or media session. Valid values are User Datagram Protocol (UDP) or Transport Control Protocol (TCP).
Egress Transport	Protocol type used on the outgoing call or media session. Valid values are User Datagram Protocol (UDP) or Transport Control Protocol (TCP).

Display a Session Summary

You can view data and statistics about a call or media session by displaying the Session Summary from a ladder diagram.

1. Access a Summary: **Monitoring, Monitor and Trace.**

2. Under **Monitor and Trace**, select one of the following summaries:
 - Notable Events
 - Registrations
 - Sessions
 - Subscriptions
3. On the Summary page, select a row in the table and right-click.
4. On the pop-up menu, click Ladder Diagram.
5. (Optional)—On the ladder diagram, click the [+] to expand the Session Summary section.

QoS Statistics

The Quality of Service (QoS) Stats section of the Session Summary displays information about the quality of the service for a selected call session or media event. To see the QoS statistics, open a ladder diagram from a record in a Summary report and click the [+] control that precedes "QoS Statistics" at the bottom of the page.

Expand QoS Stats section with the [+] control.

I-I QoS Stats														
Flow ID	Direction	Total Pkts		Total Octets		RTCP				RTP			CoE	
		Received	Received	Pkt Lost	Avg Jitter	Max Jitter	Avg Latency	Max Latency	Pkt Lost	Avg Jitter	Max Jitter	R-Factor	MOS	
100663297	CALLING	0	0	0	0	0	0	0	0	0	0	0	0.00	0.00
100663298	CALLED	0	0	0	0	0	0	0	0	0	0	0	0.00	0.00

The following table describes each column in the QoS Stats report.

Flow ID	ID number assigned to the call session or media event flow of data.
Direction	The direction of the call or media event flow. CALLING—egress direction CALLED—ingress direction
Total Pkts Received	Total number of data packets received on the interface during the active call session or media event.
Total Octets Received	Total number of octets received on the interface during the active call session or media event.
RTCP	Real-time Transport Control Protocol—used to send control packets to participants in a call.
Pkts Lost	Number of RTCP data packets lost on the interface during the active call session or media event.
Avg Jitter	Average measure of the variability, called jitter, over time of the RTCP packet latency across a network. A network with constant latency has no jitter. Jitter is referred to as Packet Delay Variation (PDV). It is the difference in the one-way end-to-end delay values for packets of a flow. Jitter is measured in terms of a time deviation from the nominal packet inter-arrival times for successive packets.

Max Jitter	Maximum measure of the variability, called jitter, over time of the RTCP packet latency across a network. A network with constant latency has no variation jitter.
Avg Latency	Average observed one-way signaling latency during the active window period. This is the average amount of time the signaling travels in one direction.
Max Latency	Maximum observed one-way signaling latency during the sliding window period. This is the maximum amount of time the signaling travels in one direction.
RTP	Real-Time Transport Protocol—a standard packet format for delivering audio and video over the internet.
Pkts Lost	Number of RTP data packets lost on the interface during the active call session or media event.
Avg Jitter	Average measure of the variability, called jitter, over time of the RTP packet latency across a network. A network with constant latency has no jitter. Jitter is referred to as Packet Delay Variation (PDV). It is the difference in the one-way end-to-end delay values for packets of a flow. Jitter is measured in terms of a time deviation from the nominal packet inter-arrival times for successive packets.
Max Jitter	Maximum measure of the variability, called jitter, over the time of the RTP packet latency across a network. A network with constant latency has no jitter.
QoE	Quality of Experience—measurement used to determine how well the network is satisfying the end user's requirements.
R-Factor	Rating Factor—An average Quality of Service (QoS) factor observed during the active window period. QoS shapes traffic to provide different priority and level of performance to different data flows. R-Factors are metrics in VoIP that use a formula to take into account both user perceptions and the cumulative effect of equipment impairments to arrive at a numeric expression of voice quality. This statistic defines the call or transmission quality, which is expressed as an R factor.
MOS	Mean Opinion Score (MOS) score—MOS is a measure of voice quality. MOS provides a numerical indication of the perceived quality of the media received after being transmitted and eventually compressed using Codecs.

Display QoS Statistics

When you want to view QoS statistics for a call or media flow session, you can do so from a ladder diagram in **Monitor and Trace**.

1. Access a Summary: **Monitoring, Monitor and Trace**.
2. Under **Monitor and Trace**, select one of the following summaries:
 - Notable Events
 - Registrations
 - Sessions

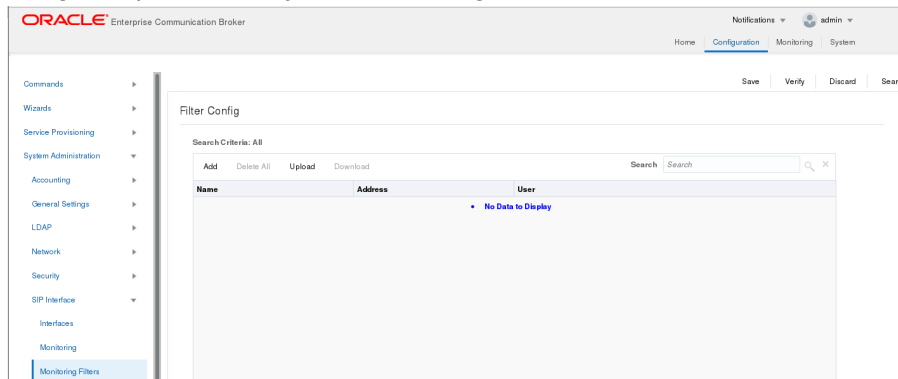
- Subscriptions
3. On the Summary page, select a row in the table and right-click.
 4. On the pop-up menu, click Ladder Diagram.
 5. On the ladder diagram, click the [+] to expand the **QoS Stats** section.

SIP Monitor and Trace Filter Configuration

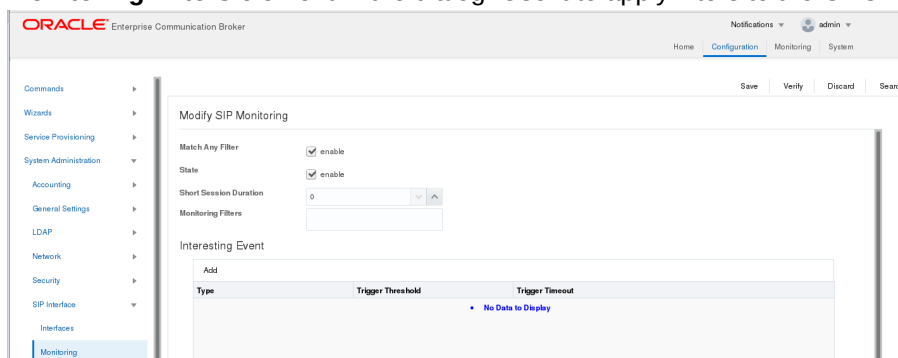
The SIP Monitor and Trace function allows you to monitor SIP sessions for notable events and display the results in the Oracle Enterprise Communications Broker (OECB) SIP Notable Events summary. Such information may help you perform troubleshooting. For more targeted monitoring, you can configure filters on particular users and addresses on the OECB, and on a specific agent.

The OECB Configuration page, located at **Configuration, System Administration, SIP Interface, Monitoring Filters**, includes the following objects for configuring SIP Monitoring filters:


- The SIP Interface configuration page displays the **Monitoring Filters** object in the navigation pane, which you use to configure individual filters.



- The **Monitoring** object on the SIP interface configuration page displays the **Monitoring Filters** element in the dialog. Use it to apply filters to the OECB.



- The Add Agents configuration page displays the **Monitoring Filters** configuration element to the Advanced section. Use it to apply filters to an agent.

-  **Note:**
 After the P-CZ2.0.0m4 release, the system does not support the former "Enable SIP Monitor and Trace" setting. You must re-configure SNMP event traps through the dialogs described in this topic.

Use the following filter configuration process for both new installations and upgrades.

1. Create one or more filters in the Monitoring Filters object. You may use an asterisk character as a filter, if you want to monitor all session data.
2. Add one or more filters to the Monitoring object.
3. (Optional) Add one or more monitoring filters to an agent that you want to monitor.

Search for a Record

The **Search** button at the top of the report page allows you to find a specific record within a Monitor and Trace report. It also allows you to specify criteria on which to perform the search.

After defining a search criteria in the Search Filter dialog box, clicking Search automatically populates the report page with the records that match the specified criteria specified. The search performs the filtering process of criteria dependent on the report page from which you are running the search.

For example, performing a search from the Sessions report page displays only the reports pertaining to call sessions. If you perform a search on the Registration report page, only the reports pertaining to call registrations displays on the report page. The search string containing the criteria on which you performed the search, displays in the top left corner of the page.

Note:

A SIP Monitor and Trace global search can find items in the SIP headers, as well.

The system saves the search criteria until you click **Reset** in the dialog box, or until you log out of the HTTP session.

Search for a Report Record

The **Search** button at the top of the report page allows you to find a specific record within a Monitor and Trace report. It also allows you to specify criteria on which to perform the search.

You can specify a value for any or all of the fields in the Search box. The search process searches for records with all of the values you specify and displays only the records with these values. If you perform a Global Search and specify values in other fields, the search process searches the other specified fields first and then filters on the Global Search field.

- If you specify a "*" in a search string, the search is performed on that exact string. For example, if you search for "123*45", the search shows results for all strings containing "123*45".
- You can use quotes (" ") to specify a search. For example, you can enter Smith and the search finds all of the records that match Smith, such as: John **Smith**field<sip:sipp@192.168.1.70:5070>;tag=12260SIPpTag001.
- If you enter a space before or after a quotation mark, (for example, "Smith "), the search returns no data.

1. On any reports page, click **Search** and do the following in the Search Filter dialog.

Global Search— Search all parameters in all records.	Enter the URI formatted string of the call source information you are searching. Valid values are alpha-numeric characters. For example, sipp<sip:sipp@172.16.34.10:5060;tag=24.
From URI— Search on the From-URI header.	Enter the URI formatted string that contains a protocol, name, location, or any other applicable characteristic, that is sent by the OECB in the FROM header. Valid values are alpha-numeric characters. For example, sip:service@172.16.34.226:5060.
Request URI— Search on the Request-URI header.	Enter the URI formatted string that contains a protocol, name, location, or any other applicable characteristic, that is sent by the OECB in the REQUEST header. Valid values are alpha-numeric characters. For example, sip:service@172.16.34.226:5060.
To URI—Search on the To-URI header.	Enter URI formatted string of the call destination information you are searching. Valid values are alpha-numeric characters. For example, sut<sip:service@172.16.34.226:5060;tag=99.
Start Date— Search from messages that start at the specified date and time.	<ul style="list-style-type: none"> • Enter a starting date to search on in the first text box in the format YYYY-MM-DD (where Y =year, M=month, and D=day). For example, 2012-04-15 would search for all records ending on April 15, 2012. Valid values are numeric characters only. • Enter a start time to search on in the last three text boxes in the format HH mm ss (where H=hour, m=minutes, and s=seconds). For example, 01 30 45 would search for all

	records starting at 1:30 and 45 seconds. Valid value are numeric characters only.
End Date— Search from messages that end at the specified date and time.	<ul style="list-style-type: none"> • Enter an end date to search on in the first text box in the format YYYY-MM-DD (where Y =year, M=month, and D=day). For example, 2012-04-15 would search for all records starting on April 15, 2012. Valid values are numeric characters only. • Enter an end time to search on in the last three text boxes in the format HH mm ss (where H=hour, m=minutes, and s=seconds). For example, 01 30 45 would search for all records ending at 1:30 and 45 seconds. Valid value are numeric characters only.
Session ID— Search on the monitored SIP session ID, as shown in the Summary table.	Enter the ID of the call session you want to search. Valid values are alpha-numeric characters. For example, 22-3412@172.16.34.1.
In Call ID— Search on the SIP call ID of the initial received request.	Enter the ID of the incoming call (phone number and source IP address). Valid values are alpha-numeric characters. For example, 25-3412@172.16.34.10.
Out Call ID— Search on the SIP call ID of the first routed request.	Enter the ID of the outgoing call (phone number and IP address). Valid values are alpha-numeric characters. For example, 14-3412@172.14.54.6.
State (with result code)—Search on the state of the call, as shown in the Summary table.	<p>Enter the status of the call session with the result code for which you want to search. Result codes can range from 1xx to 5xx. For example, terminated-200, or failed-400. Case-sensitive. Valid values include:</p> <ul style="list-style-type: none"> • INITIAL—<result code> • EARLY—<result code> • ESTABLISHED—<result code> • TERMINATED—<result code> • FAILED—<result code>
Notable Event— Search on a notable event type that you select from the drop-down list.	<p>Select the notable event for which you want to search. Valid values include:</p> <ul style="list-style-type: none"> • any event—search displays any notable event that was stored in memory. • hort session—search displays only records that indicate a short-session duration has occurred. • local rejection— search displays only records that indicate a local-rejection has occurred.

In Realm—Search the realm from which the initial request was received.	Enter the name of the realm for which the incoming call belongs. Valid values are alpha-numeric characters. For example, access.
Out realm—Search the realm to which the first routed request was sent.	Enter the name of the realm for which the outgoing call belongs. Valid values are alpha-numeric characters. For example, backbone.
Destination Agent—Search on the name of the session agent from which the initial request was received.	Enter the name of the session agent (SA) on the incoming call session. Valid values are alpha-numeric characters. For example, SA1.
Destination Session Agent—Search on the realm to which the first routed request was sent.	Enter the name of the session agent (SA) on the outgoing call session. Valid values are alpha-numeric characters. For example, SA2.
Ingress Destination Address—Search on the IP address from which the initial request was received.	Enter the source IP address of the SA that accepted the incoming call session. IP Address must be entered in dotted decimal format (0.0.0.0). For example, 172.45.6.7.
Egress Destination Address—Search on the IP address to which the initial request was sent.	Enter the destination IP address of the SA that accepted the outgoing call session. IP Address must be entered in dotted decimal format (0.0.0.0). For example, 172.64.56.7.
In Network Interface—Search on the IP address on which the initial request was received.	Enter the incoming core network interface that connects the Net-Net ECB to your network. IP Address must be entered in dotted decimal format (0.0.0.0). For example, 192.45.6.7.
Out Network Interface—Search on the IP address that was the source for the routed request.	Enter the outgoing network interface that connects your Net-Net ECB to the outside network. IP Address must be entered in dotted decimal format (0.0.0.0). For example, 192.45.6.8.

- Click **Search**.

Specify Additional Identifiers

To specify additional identifiers:

1. In the Session Id field, enter the ID of the call session you want to search. Valid values are alpha-numeric characters. For example, 22-3412@172.16.34.1.
2. In the In Call ID field, enter the ID of the incoming call (phone number and source IP address). Valid values are alpha-numeric characters. For example, 25-3412@172.16.34.10.
3. In the Out Call ID field, enter the ID of the outgoing call (phone number and IP address). Valid values are alpha-numeric characters. For example, 14-3412@172.14.54.6.
4. In the State (with result code) field, enter the status of the call session with the result code for which you want to search. Valid values are (case-sensitive):
 - INITIAL-<result code>
 - EARLY-<result code>
 - ESTABLISHED-<result code>
 - TERMINATED-<result code>
 - FAILED-<result code>

Result codes can range from 1xx to 5xx. For example, terminated-200, or failed-400.
5. In the Notable Event field, select the notable event for which you want to search. Valid values are:
 - any-event - search displays any notable event that was stored in memory.
 - short-session - search displays only records that indicate a short-session duration has occurred.
 - local-rejection - search displays only records that indicate a local-rejection has occurred.
6. To search on additional parameters, click on the Additional Search Options down arrow to expand the dialog box.

Specify Additional Search Options

To specify additional search options:

1. In the “**In Realm**” field, enter the name of the realm for which the incoming call belongs. Valid values are alpha-numeric characters. For example, access.
2. In the “**Out Realm**” field, enter the name of the realm for which the outgoing call belongs. Valid values are alpha-numeric characters. For example, backbone.
3. In the “**In SA**” field, enter the name of the session agent (SA) on the incoming call session. Valid values are alpha-numeric characters. For example, SA1.
4. In the “**Out SA**” field, enter the name of the session agent (SA) on the outgoing call session. Valid values are alpha-numeric characters. For example, SA2.

5. In the “**In Source Addr**” field, enter the source IP address of the SA that accepted the incoming call session. IP Address must be entered in dotted decimal format (0.0.0.0). For example, 172.45.6.7.
6. In the “**Out Dest Addr**” field, enter the destination IP address of the SA that accepted the outgoing call session. IP Address must be entered in dotted decimal format (0.0.0.0). For example, 172.64.56.7.
7. In the In Network Interface field, enter the incoming core network interface that connects the Net-Net ECB to your network. IP Address must be entered in dotted decimal format (0.0.0.0). For example, 192.45.6.7.
8. In the Out Network Interface field, enter the outgoing network interface that connects your Net-Net ECB to the outside network. IP Address must be entered in dotted decimal format (0.0.0.0). For example, 192.45.6.8.
9. Click <Search> to perform the search with the values you specified. A list of the records that the search process filtered, display in the window. The GUI saves the search specifications until you click <Reset> in the search dialog box, OR until you log out of the GUI.

Exporting Information to a Text File

Monitor and Trace allows you to export information to a text file from the Sessions, Registrations, Subscriptions and Notable Events Reports, as well as from a specific ladder diagram, or from a page containing the results of a search. The system exports data to a file that you can open and view as required.

You can export any of the following:

- All information from each report
- Information from a specific record only
- Information from a search result
- Information from a Ladder Diagram

The following list identifies the buttons to use to export specific information from Monitor and Trace. All the export buttons in the GUI export to text files.

Export Session Details	Exports the SIP messages and media events associated with the selected session, to a file in text format on the local machine.
Export Summary	Exports all logged session summary records to a file in text format on the local machine. Note: This button exports ALL call session summary records or the records that matched a search criteria to the file.
Export Diagram	Exports all of the information in the Ladder Diagram (Session Summary, SIP Message Details, and QoS statistics), to an HTML file format on the local machine.
Export Session Details	Exports detailed information about the SIP messages and media events in the Ladder Diagram associated with the selected session, to a file in text format on the local machine.

Export Report Information to a Text File

To export information from a Monitor and Trace report to a text file:

 **Note:**

The GUI exports Ladder Diagrams as HTML files.

1. Access Monitor and Trace: **Monitoring, Monitor and Trace**.
2. On the Monitor and Trace page, select a report type. For example, Subscriptions.
3. On the report Summary page, select a report from the list, right-click, and do one of the following:
 - Click **Export session details**.
 - Click **Export summary**.
4. In the SessionDetails.txt or SummaryExport.txt dialog, do one of the following:
 - Click **Open with**, and select the application with which to open the resulting text file.
 - Click **Save file** to save the text file to your local PC.
5. Click **OK** to export the report information.

Widgets

For each **show** command that you can use from the ACLI to see System, Media, and Signaling data about the Oracle Enterprise Communications Broker (OECB), the system provides a corresponding Widget on the Web GUI. Widgets display the data in a graphical or textual format.

A Widget can display a table, text, a pie graph, or a line graph depending on the type of data and the purpose of the display. For example, the SIP Realms All widget displays an actionable table and the Recording widget displays static text. You can access the list of widgets from either the navigation pane on the Monitoring tab or from the Home page by clicking Add Widget.

Most of the Widgets automatically display any available data when you click the name of the Widget, but some Widgets require further input. Such Widgets include a Settings button in their display that launches a parameters dialog where you specify the data to display. For example, the Realm Specifics Widget requires you to set the name of the realm and the auto refresh interval.

 **Note:**

You must set up a valid SIP configuration before the OECB can display any SIP data in a Widget, including the default Widgets.

The following tables list and describe the Widgets in the same order as they are displayed on the Web GUI.

System

The following table lists all of the System widgets in the left column. The middle column displays the corresponding ACLI **show** command, when there is one. The right column describes the data that the Widget displays.

Widget	Show Command	Description
Accounting	show accounting	Displays a summary of statistics for configured external accounting servers.
ACL	show acl all	Displays cumulative and per-interface statistics on ACL traffic and drops, displaying Recent, Total, and PerMax counts. The display also separates traffic from trusted from untrusted sites.
Alarms	show alarms	Displays existing alarms and allows you to clear them.
Authentication RADIUS	show radius all	Displays the status of established RADIUS accounting connections.
Authentication TACACS	show tacacs stats	Displays statistics related to communications between the OECB and configured TACACS servers .
Communications Monitor Errors	show comm-monitor errors	Displays Communications Monitor aggregate error statistics information.
Communications Monitor Internal	show comm-monitor internal	Displays Communications Monitor aggregate internal statistics information.
Communications Monitor Stats	show comm-monitor stats	Displays statistics related to connections between the OECB Communications Monitor probe and any configured Communications Monitor servers.
Editing Configuration	show configuration	Displays the current editing configuration.
Editing Configuration Short	show configuration short	Displays only the parameters that you modified in the editing configuration.
Configuration Inventory	show configuration inventory	Displays the editing and running configuration inventory of all configured elements.
Running Configuration	show running-config	Displays the current running configuration.
Running Configuration Short	show running-config short	Displays only the parameters that you modified in the running configuration.
Configuration Version	show version	Displays the configuration version number table.

Widget	Show Command	Description
Highest Task CPU Usage	No show command	Displays a line graph with 5-10 tasks with the highest CPU usage in percent, during a specific period of time.
Highest Task CPU Usage	No show command	Displays a table with 5-10 tasks with the highest CPU usage in percent, during a specific period of time.
Current Disk Usage Pie Graph	No show command	Displays the current disk usage in a pie graph.
Current Disk Usage Table	No show command	Displays the current disk usage in a table.
Features	show features	Displays the features that are currently enabled, based on added licenses.
Interfaces	show interfaces	Displays all of the information concerning the rear interfaces.
Interfaces Brief	show interfaces brief	Displays key running statistics about the rear interfaces in one graphic.
Interface Mapping	show interface mapping	Displays the configured physical interfaces with their MAC addresses and label.
Virtual Interfaces	show virtual interfaces	Displays the virtual interfaces for signaling services.
Wancom	show Wancom	Displays negotiated duplex mode and speed for all system control interfaces.
ARP Info	show arp info	Displays the current Internet-to-Ethernet address mappings in the ARP table.
ARP Statistics	show arp statistics	Displays ARP statistics.
ARP Summary	show arp	Displays the current Internet-to-Ethernet address mappings in the ARP table.
IP Connections	show ip connections	Displays all TCP and UDP connections.
Neighbor Table	show neighbor table	Displays the IPv6 neighbor table and validates that an entry for the link local address exists and that the gateway uses that MAC address.
Routes	show routes	Displays the current system routing table.
IP Summary	show ip	Displays IP statistics.
IP TCP	show ip tcp	Displays all TCP statistics.
IP UDP	show ip udp	Displays all UDP statistics.
Licenses	license	Displays existing licenses and allows you to add or delete them.

Widget	Show Command	Description
Current Memory Usage	No show command	Displays the current percentage of free and allocated memory in a pie graph.
Current Memory Usage	No show command	Displays the current percentage of free and allocated memory in a table.
Historical Memory Usage	No show command	Displays a line graph of the kilobytes of free and allocated memory over a period of time.
Historical Memory Usage	No show command	Displays a table of the kilobytes of free and allocated memory over a period of time.
Memory Summary	show memory	Displays statistic related to the memory.
Platform All	show platform all	Displays full platform information.
Platform CPU load	show platform cpu-load	Displays current CPU load.
Platform Errors	show platform errors	Displays service pipe write errors.
Platform Limits	show platform limits	Displays platform related limits.
PROM Info	show prom info all	Displays all available PROM information.
Temperature	show temperature	Displays the temperature in Celsius for all components with temperature sensors.
Processes	show processes	Displays statistics for all active processes.
SNMP Community Table	show snmp-community-table	Displays all information for configured SNMP communities including requests and responses for each community.
Trap Receiver	show trap-receiver	Displays trap receiver information for each configured SNMP community.
SPL Memory	show spl memory	Displays SPL memory for each task SPL engine.
SPL Options	show spl-options	Displays information on all SPL options.
SPL Statistics	show spl statistics	Displays statistics for all tasks.
SPL Version	show spl	Displays the version of the SPL engine.
System Health	show health	Displays the system health table for HA pairs.
TDM Channels	show tdm channels	Displays the TDM b and d channel configurations.
TDM Dialplan	show tdm dialplan	Displays the TDM dial plan configuration.
TDM Spans	show tdm spans	Displays the TDM spans configuration.

Widget	Show Command	Description
TDM Status	show tdm status	Displays the status of TDM operations.
Clock	show clock	Displays the current date and time.
NTP Server	show ntp server	Displays information about the quality of the time used for offset and the delay measurement maximum error bounds.
NTP Status	show ntp status	Displays information about configuration status, NTP daemon synchronization, NTP synchronization in process, and if NTP is not responding.
Time Zone	show timezone	Displays the time zone.
Clock UTC	show clock utc	Displays the current date and time in Coordinated Universal Time (UTC).
Uptime	show uptime	Displays information about the length of time the system has been running in days, hours, and seconds. Also displays the current date and time information.
User Management	show users	Displays a table that lists all users currently logged on to the system.
Version Boot	show version boot	Displays the boot version.
Version CPU	show version cpu	Displays the CPU version.
Version Hardware	show version hardware	Displays the hardware version.
Version Image	show version image	Displays the image version.
Version Summary	show version	Displays the Operating System version information, including the OS version number, the manufacturing date of the current version, and other details.

Media

The following table lists all of the Media widgets in the left column. The middle column displays the corresponding ACLI **show** command. The right column describes the data that the Widget displays.

Widget	Show Command	Description
Media Classify	show media classify	Displays network processor statistics.
Media Host Stats	show media host-stats	Displays statistics for the host processor, including the number and types of packets received at a specific port.

Widget	Show Command	Description
MBCD Acls	show mbcd acls	Displays Middlebox Control Daemon (MBCD) Access Control statistics.
MBCD All	show mbcd all	Displays related to many of the show MBCD sub-commands. The widget shows only those MBCD messages that show statistics.
MBCD Errors	show mbcd errors	Displays MBCD error statistics.
MBCD Realms	show mbcd realms	Displays statistics of all MBCD realms.
MBCD Statistics	show mbcd statistics	Displays information related to media flows established by the MBCD task.
NAT By Index	show nat by-index	Displays the specified range of entries in the NAT table, with a maximum of 5024 entries. The default range is 1-200. The range corresponds to the line numbers in the table, not to the number of the entry.
NAT in Tabular	show nat in-tabular	Displays a specified range of entries in the NAT table.
Realm Specifics	realm-specifics	Displays all realm-specific configuration based on the specified realm-id.
Realm Summary	show realm	Displays the realm summary statistics.
Xcode Codecs	show xcode codecs	Displays counts of codec pairs and ptime transrating in use.
Xcode Load	show xcode load	Displays the transcoding resources in use.
Xcode Xlist	show xcode xlist	Displays the XCode Xlist.

Signaling

The following table lists all of the Signaling widgets in the left column. The middle column displays the corresponding ACLI **show** command. The right column describes the data that the Widget displays.

Widget	Show Command	Description
DNS	show dns	Displays statistics for the DNS configuration.
ENUM	show enum	Displays ENUM statistics.
Fraud Protection All	show fraud-protection all	Displays all Fraud Protection entries.
Fraud Protection Black list	show fraud-protection blacklist	Displays the Fraud Protection blacklist entries.
Fraud Protection Rate limit	show fraud-protection rate limit	Displays the Fraud Protection rate limit list entries.

Widget	Show Command	Description
Fraud Protection Re-direct	show fraud-protection redirect	Displays the Fraud Protection redirect list entries.
Fraud Protection White List	show fraud-protection white list	Displays the Fraud Protection white list entries.
Fraud Protection All Matches	show fraud-protection all matches-only	Displays the Fraud Protection match-only entries.
Fraud Protection Blacklist Matches	show fraud-protection blacklist matches-only	Displays the Fraud Protection blacklist match-only entries.
Fraud Protection Rate Limit Matches	show fraud-protection rate limit matches-only	Displays the Fraud Protection rate limit match-only entries.
Fraud Protection Redirect Matches	show fraud-protection redirect matches-only	Displays the Fraud Protection redirect match-only entries.
Fraud Protection White List Matches	show fraud-protection white list matches-only	Displays the Fraud Protection white list match-only entries.
Fraud Protection Summary	show fraud-protection stats	Displays the Fraud Protection summary.
H323d	show h323d	Displays H323 statistics.
LDAP	show ldap	Displays LDAP statistics.
LRT	show lrt	Displays the Local Routing Table statistics.
Recording	show rec	Displays SIP REC statistics for the Recording Agent.
Registration by Realm	show registration sipd by realm	Displays calls that registered through a specified ingress realm for which you want to view cache information.
Registration H323 d	show registration h323d	Displays H323d registrations.
Registration SIP	show registration SIP	Displays SIP registrations.
Registration Statistics	show registration statistics	Displays a table of counters showing the total and periodic number of registrations by protocol.
Sessions	show sessions	Displays the session capacity for license and session use.
Agent Details	show sipd agents	Displays statistics related to defined DIP session agents.
Agent Groups	show sipd groups	Displays cumulative information for all session agent groups.
Agent Individual	show sipd agents <agent name>	Displays statistics related to the specified SIP session agent.
Client Trans	show sipd client	Displays statistics for SIP client events when the SBC is acting as a SIP client in the B2BUA role.
SIP Codecs	show sipd codecs	Displays codec usage for a realm.
SIP Errors	show sipd errors	Displays statistics for SIP media event errors.

Widget	Show Command	Description
Interface Individual	show sipd interface	Displays SIP interface statistics for the specified realm.
Interface Summary	show sipd interface	Displays all SIP interface statistics.
Method Ack	show sipd ack	Displays all SIP ACK method statistics.
Method Bye	show sipd bye	Displays the SIP BYE method statistics.
Method Cancel	show sipd cancel	Displays all SIP CANCEL method statistics.
Method Info	show sipd info	Displays the SIP INFO method statistics.
Method Invite	show sipd invite	Displays the SIP INVITE method statistics.
Method Message	show sipd message	Displays the SIP MESSAGE statistics.
Method Notify	show sipd notify	Displays the SIP NOTIFY statistics.
Method Options	show sipd options	Displays the SIP OPTIONS statistics.
Method Prack	show sipd prack	Displays the SIP PRACK method statistics.
Method Publish	show sipd publish	Displays the SIP PUBLISH method statistics.
Method Refer	show sipd refer	Displays the SIP REFER method statistics.
Method Register	show sipd register	Displays the SIP REGISTER method statistics.
Method Subscribe	show sipd SUBSCRIBE	Displays the SIP SUBSCRIBE method statistics.
Method Update	show sipd update	Displays the SIP UPDATE method statistics.
SIP Realms All	show sipd realms	Displays realm statistics related to SIP processing.
SIP Realms Individual	show sipd realms <realm name>	Displays realm statistics related to SIP processing for the specified realm.
SIP Redundancy	show sipd redundancy	Displays information about SIP redundancy.
Server Trans	show sipd server	Displays statistics for SIP server events when the SBC acts as a SIP server in the B2BUA role.
SIP Sessions All	show sipd sessions all	Displays all SIP sessions currently on the system.
SIP Sessions	show sipd sessions	Displays the number of sessions and dialogs in various states for the Period and Lifetime spans.
SIP Status	show sipd status	Displays information about SIP transactions.

Agent Status Widget

The GUI contains a widget for displaying the status of session-agents. The widget is available from the Statistics Portal.

The columns of each row display both configured and runtime information.

- Name
- IP Address
- State
- Active Calls - inbound
- Active Calls - outbound
- Round-Trip-Time

Because the number of session-agents might be very large, the use of paging, searching, and sorting is utilized.

User Management Interface

Grid widget

Name	IP Address	State	Active Calls (Inbound)	Active Calls (Outbound)	RTT (ms)
SBC 1	192.168.1.1	IN-SERVICE	1000	1000	10
SBC 2	192.168.1.2	OUT-OF-SERVICE	0	0	0
SBC 3	172.16.1.1	IN-SERVICE	30	30	123

Performance

This Widget does not impact signaling performance, but may occasionally become unresponsive.

Broker Lookup Widget

The GUI displays a widget for performing a broker lookup. The Broker Lookup widget tests how a call is processed, based on the current dial plan, agent, and routing configuration. You enter the FROM_URI, REQUEST_URI, and Source Agent for the call and click **Begin** to execute the test. The widget displays all possible ways that a call with those parameters can be processed, including all hops and their associated costs.

Requirements

The Broker Lookup widget uses the Settings Panel and the Results Panel. In the Settings Panel, you enter the From-URI and Request-URI information. You can also select Source Agent information. The GUI displays the results of the lookup in the results window.

Results Analysis

The following diagram displays examples of the Dial Plan Lookup and Agent Lookup results from the Broker Lookup widget.

Results

Dial Plan Lookup Results

Origin Context: AcmeBedford

Universal Number	Destination Context	Result Type
+17813284444	AcmeBedford	Best

Agent Lookup Results

	Number	Agent	Agent Source
Source	1	1.1.1.1	URI Host
Destination	17813284444	2.2.2.2	URI Host

Routing Lookup Results

Type	Cost	Hops
Implicit	0	— cost: 0 → 2.2.2.2
Default	0	— cost: 0 → testAgent192

The following lists describe the results shown in the preceding examples.

Dial Plan Lookup Results

- **Universal Number**—The complete universal (usually E.164) called number that the Dial Plan engine found for the information you entered.
- **Destination Context**—The Dialing Context associated with the universal number.
- **Result Type**—Indicates the certainty of the result. There are three possible values:
 - **Best**—The result is either the only result found, or is the best match. This is the universal number that will be called.
 - **Ambiguous**—This dial plan result conflicts with other dial plan results. The Oracle Enterprise Communications Broker (OECB) cannot route the call because the intended destination is unclear. If a result is ambiguous, there are other ambiguous results as well, and no result can be labeled as 'Best'.
 - **None**—This dial plan result does not conflict with others, but it is not the best result and will not be used. Another result is listed as 'Best'.

Agent Lookup Results

- **Unlabeled column**—Indicates whether the row refers to the source or destination agent.
- **Number**—The calling or called number, respectively, that was used to look up the agent.
- **Agent**—The agent that was found for this number.
- **Agent Source**—The source from which the agent was found. The OECB tries to find the agent associated with this number using the following methods in order:

- 1. User DB
- 2. LDAP
- 3. URI Host (taken from the From-URI)
- 4. Inbound Session-Agent (only applies to source agent)

Routing Lookup Results

- Each route can be one of the following types:
 - Implicit—Routing directly to the given Request-URI.
 - Configured—A route that you configured.
 - Default—A route that you configured that matches any incoming call. For example, a route with all fields configured as asterisks.
- This is the cost based on the routing configuration.
- This is a visual diagram of the route, in which each individual hop is displayed as bold text, and the cost incurred by routing the call to each given hop is displayed as a box with an arrow through it.

Performance

This Widget does not impact signaling performance, but may become unresponsive.

License Widget

The License widget on the Web GUI provides a workspace where you can view, add, and delete Oracle Enterprise Communications Broker (OECB) licenses.

From the Widgets tab on the Web GUI, the system displays the Licenses page when you click **Widgets, System, Licences**.

The Licenses page displays a list of your OECB licenses with the following information.

Licenses	The name of the license.
Session Count	The number of session entitlements for the license.
Install Date	The date when the license is added to the system.
Begin Date	The date when the license begins service.
Expire Date	The date when the license ends service.

If you want to see the details of a particular license, click the show-hide toggle by the license name to expand the view to show all of the details. The following illustration shows an example of license details.

Licenses					
Add	Delete				
License	Session count	Install date	Begin date	Expire date	
<input checked="" type="checkbox"/> Acme Developer License 1	8000	12/24/18 Apr 24 2015		00:59:59 Apr 28 2015 expired!	
<input checked="" type="checkbox"/> Acme Developer License 2	8000	12/31/15 Apr 24 2015		00:59:59 May 11 2015 expired!	

Acme Developer License: 8000 sessions, SIP, MGCP, H323, HNF, QoS, ACP, Routing, Load Balancing, Accounting, High Availability, PAC, LI, External SIP Mgmt, TLS, Software TLS, External CLF Mgmt, External Policy Services, ENUM, H248, H249 SCF, H248/BGF, NSEP, RPH, LI Debug, Session Replication for Recording, Transcode Codec AMR (un capped AMR transcoding sessions), Transcode Codec EVRC (un capped EVRC transcoding sessions), DaaS, IPSEC, IP4-v6 Interworking, RTP, E93, Transcode Codec EVRCB (un capped EVRCB transcoding sessions), Software PCOM, Security Gateway, SIP Authorization/Authentication, Database Registrar (0 contacts), IS5 Advanced, SLB (20000 endpoints), Software SRTT, Allow Unsigned SPL files, Session Recording, Diameter Director, Transcode Codec AMR-WB (un capped AMRWB transcoding sessions), CK, Transcode Codec G729 (un capped G729 transcoding sessions), Transcode Codec Opus (un capped OPUS transcoding sessions), Transcode Codec SILK (un capped SILK transcoding sessions)

The Licenses widget provides the controls to Add and Delete licenses.

When you click **Add**, the system displays the Set license dialog.

When you select a license from the Licenses list and click **Delete**, the system displays the delete Confirmation dialog.

The License widget includes the Refresh, Download, Add to Dashboard, Pin to Favorites, and Help icons, familiar from other widgets, in the top, right-hand corner. Note that the License widget does not include the Settings icon and the Auto-refresh function because these operations do not apply to licenses.

The Set License wizard is linked to the License widget, so that you can view your licences from the wizard. After launching the Set License wizard, use the "View current license information" link in the Set License dialog to see a view-only list of your OEGB licenses.

The only operations allowed in view mode are Refresh and Download.

Registration Cache Dashboard Widget

The dashboard displays the SIP Registrarions Cache widget for registration cache entries. The registration cache is implemented in the SIP tasks. The dashboard widgets back-end is implemented in web tasks. The SIP task will walk all SIP users, using the new users SPL API. The entire registration cache is displayed. This information will be packaged and sent back to the web task. The response will be parsed in the web task and returned to the user in the form of a widget. Multiple (AOR) contacts are displayed with the same user name, different contacts.

Widget Columns

The SIP registration cache table displays the following information.

User	Address of record
Universal number	ECB E.164 number mapping (+17811234567)
Contact endpoint	SIP contact
Contact expire endpoint	Expires (end point side)
Contact expire	Expires (real)

Performance

This widget will not impact signaling performance, but it may become unresponsive.

16

Troubleshooting and Maintenance

Whereas the Oracle Enterprise Communications Broker Administrator's Essentials Guide provides system-based troubleshooting information, this chapter presents parallel information related to the SIP services the Oracle Enterprise Communications Broker provides.

To a large extent, the Oracle Enterprise Communications Broker's widgets display status and quantitative information about SIP traffic. Many of these widgets provide information that is self-explanatory. This chapter provides descriptions and instructions on key widgets used to analyze service operations.

Maintenance consists of a variety of tasks, including managing system files. The Oracle Enterprise Communications Broker's System tab provides access to file management controls, and is described in this chapter.

Refer to the Oracle Enterprise Communications Broker Administrator's Essentials Guide for further troubleshooting and maintenance information related to system administration.

Audit Logs

The Oracle Enterprise Communications Broker (OECB) can record user actions in audit logs by way of the Web GUI. The audit logs record the creation, modification, and deletion of all user-accessible configuration elements, as well as attempted access to critical security data such as public keys. For each logged event, the system provides the associated user-id, date, time, event type, and success or failure data.

You can configure the system to record audit log information in either verbose mode or brief mode. Verbose mode captures the system configuration after every change, and displays both the previous settings and the new settings in addition to the event details. Brief mode displays only the event details. Although you can specify the recording mode, you cannot specify which actions the system records. The following list describes the actions that the system records.

Global	<ul style="list-style-type: none">• Log on and log off.• Save a template configuration.• Click Complete in a Wizard.
Home tab	<ul style="list-style-type: none">• Add, reset, and save.• Change Widget settings.
Configuration tab	<ul style="list-style-type: none">• Save and activate a configuration.• Discard a configuration.• Add, edit, delete, and copy configuration changes.

	<ul style="list-style-type: none"> • Run the generate and import certificate commands.
System tab	<ul style="list-style-type: none"> • Add audit entries to the system file management actions, such as upload, download, restore, backup, add, edit, and delete. • Force an HA switch over. • Run the Show Support Information command. • Run the Upgrade Software wizard. • Download and view an audit log.
Monitoring tab	<ul style="list-style-type: none"> • Export the summary. • Export the session detail. • Export from a Widget. • Add a Widget to favorites. • Clear, clear all on alarm, add, and delete license.

The system writes audit log events in Comma Separated Values (CSV) lists in the following format:

```
{TimeStamp,
src-user@address:port, Category, EventType, Result, Resource, Prev,
Detail}
```

The following list describes each value written to an audit log event.

TimeStamp	Shows the time when the system wrote the event to the audit log.
src-user@address:port	Identifies the system that wrote the audit log line.
Category	Classifies the event as: <ul style="list-style-type: none"> • Configuration • Security • System
EventType	Identifies the action that caused the event as: <ul style="list-style-type: none"> • Activate-config • Acquire-config • Create • Data-access • Delete • Halt • Login

	<ul style="list-style-type: none"> • Logout • Modify • Reboot • Save-config
Result	Identifies the outcome of the event as: <ul style="list-style-type: none"> • Failure • Success
Resource	Describes the action within the event. Some of the numerous actions that the system can log include: <ul style="list-style-type: none"> • Authentication • Banner (Means that someone edited the log on banner text.) • Download <filename> • Generate public key • Reboot • Upload <filename>
Prev—(verbose mode)	Displays the setting prior to this change.
Details—(verbose mode)	Displays additional information about the change, depending on the following event types: <ul style="list-style-type: none"> • Create—displays “New = element added.” • Data-access—displays “Element = accessed element.” • Delete—displays “Element = deleted element.” • Modify—displays “Previous = oldValue New = newValue.”

As the OECB records audit log data, users with admin privileges can read, copy, and download that information from the Web GUI. No one can delete or edit the original log. You can View, Refresh, and Download audit logs by way of the System tab. Go to Audit Log under File Management.

You can configure the system to transfer audit log files to an SFTP server by way of secure FTP push, when conditions satisfy one of the following specifications.

- The specified amount of time since the last transfer elapsed.
- The size of the audit log reached the specified threshold. (Measured in Megabytes)
- The size of the audit log reached the specified percentage of the allocated storage space.

The OECB transfers the audit logs to a designated directory on the target SFTP server. The audit log file is stored on the target SFTP server with a filename in the

following format: **audit<timestamp>**. The timestamp is a 12-digit string the YYYYMMDDHHMM format.

Use the following process to configure transferring audit logs to an SFTP server.

1. Configure secure FTP push. See "Secure FTP Push Configuration."
2. Configure audit logging. See "Configure Audit Logging."

Secure FTP Push Configuration

You can configure the Oracle Enterprise Communications Broker (OECB) to securely send audit log files to an SFTP push receiver for storage. Configure secure FTP push before you configure audit logging.

You can configure the Oracle Enterprise Communications Broker (OECB) to log on to a push receiver using one of the following authentication methods to create a secure connection.

Password

Configure a username and password, and leave the **public-key** parameter blank. Note that you must also import the host key from the SFTP server to the OECB for this type of authentication.

Public key

Set the **public-key** parameter to a configured public key record name including an account **username**, and configure the SFTP server with the public key pair from the OECB.

It is also common for the SFTP server to run the Linux operating system. For Linux, the command `ssh-keygen-e` creates the public key that you need to import to the OECB. The `ssh-keygen-e` command sequence requires you to specify the file export type, as follows.

```
[linux-vpn-1 ~]# ssh-keygen -e
Enter file in which the key is (/root/.ssh/id_rsa): /etc/ssh/
ssh_host_rsa_key.pub
```

If you cannot access the SFTP server directly, but you can access it from another Linux host, use the `ssh-keyscan` command to get the key. An example command line follows.

```
root@server:~$ssh-keyscan -t dsa sftp.server.com
```

Configure Secure FTP Push with Public Key Authentication

For increased security when sending files from the Oracle Enterprise Communications Broker (OECB) to an SFTP server, you can choose authentication by public key exchange rather than by password. To use a public key exchange, you must configure public key profiles on both devices and import the key from each device into the other.

The following list of tasks shows the process for configuring authentication by public key between the OECB and an SFTP server. For each step in the process, see the corresponding topic for detailed instructions.

1. Generate an RSA public key on the OECB. See "Generate an RSA Public Key."

2. Create a DSA public key on the SFTP server. See "Generate a DSA Public Key."
3. Import the DSA public key from the SFTP server into the OECB using the **known-host** option in the Import Key dialog. See "Import a DSA Public Key."
4. Add the RSA public key to the `authorized_keys` file in the `.ssh` directory on the SFTP server. See "Copy the RSA Public Key to the SFTP Server."

Generate an RSA Public Key

Add a public key profile on the Oracle Enterprise Communications Broker (OECB) and generate an RSA key. You will later import the RSA key into the SFTP server to enable authentication by way of public key exchange with the OECB.

1. Access the Public Key configuration object: **Configuration, Security, Public key**.
2. On the Public Key page, click **Add**.
3. In the Add Public Key dialog, do the following:

Name	Enter the name of this profile.
Type	Select RSA.
Size	Enter one of the following: <ul style="list-style-type: none"> • 512 • 1024 • 2048 • 4096

4. Click **OK** to create the public key profile.
The system displays the Public Key list box including the new profile.
5. Save and activate the configuration.
6. Select the newly created profile, and click **Generate key**.
The OECB displays the key in the Generate Key text box for you to copy to the SFTP server.
7. Save the configuration.
 - Generate a DSA public key.

Generate a DSA Public Key

Generate and save a DSA public key on the SFTP server. You will later import the DSA key into the Oracle Enterprise Communications Broker (OECB) to enable authentication by way of public key exchange with the SFTP server.

1. Run the following command on the SFTP server:
`ssh-keygen -e -f /etc/ssh/ssh_host_dsa_key.pub | tee sftp_host_dsa_key.pub`
2. Save the key to the `authorized_keys` file in the `.ssh` directory on the SFTP server.
 - Import the DSA key into the OECB.

Import a DSA Public Key

Import a DSA public key from the SFTP server into the Oracle Enterprise Communications Broker (OECB).

- Generate and save a DSA public key on the SFTP server.

Perform the following procedure on the OECB and select "known-host" for type.

1. Access the SSH file system on the SFTP server by way of a terminal emulation program.
2. On the SFTP server, copy the base64 encoded public file. Be sure to include the Begin and End markers, as specified by RFC 4716 *The Secure Shell (SSH) Public Key File Format*.

For OpenSSH implementations host files are generally found at `/etc/ssh/ssh_host_dsa_key.pub`, or `/etc/ssh/ssh_host_rsa.pub`. Other SSH implementations can differ.

3. On the OECB, click **Configuration, Security, Public Key**.
4. On the Public key page, click **Import key**, and do the following.

Type	Select known-host.
Name	Enter a name for your profile, which the OECB displays in public key drop-down lists.
SSH Public Key	Paste the DSA public key from the SFTP server into the text box. Ensure that the text of the key ends with a semi-colon.

5. Click **Import**.

The OECB imports the key and makes it available for configuration as the public key on an external device.

Copy the RSA public key to the SFTP server.

Copy the RSA Public Key to the SFTP Server

Copy the RSA public key from the from the Oracle Enterprise Communications Broker (OECB) to the `authorized_keys` file in the `.ssh` directory on the SFTP server.

- Confirm that the `.ssh` directory exists on the SFTP server.
- Confirm the following permissions: `Chmod 700` for `.ssh` and `Chmod 600` for `authorized_keys`.


When adding the RSA key to the `authorized_keys` file, ensure that no spaces occur inside the key. Insert one space between the `ssh-rsa` prefix and the key. Insert one space between the key and the suffix. For example, `ssh-rsa <key> root@1.1.1.1`.



1. Access the SSH file system on a configured SFTP server with a terminal emulation program.
2. Copy the RSA key to the SFTP server, using a text editor such as `vi` or `emacs`, and paste the RSA key to the end of the `authorized_keys` file.

Configure Audit Logging

The Oracle Enterprise Communications Broker (OECB) provides a means of tracking user actions through Audit Logs. You can specify how the system records audit log information, and where to send the logs for archiving. You can configure the system to record in either brief or verbose mode. Verbose mode captures the system configuration after every change, and displays both the previous and new settings in addition to the event details. Brief mode displays only the event details.

- Configure one or more push receivers to receive the audit logs. See the documentation for the receiver.
 - If you want to use public keys for authentication between the OECB and the push receiver, configure public key profiles on both devices before configuring audit logging. See "Configure Secure File Transfer with Public Keys."
1. Access the Audit Logging configuration object: **Configuration, Security, Security, Admin-Security, Audit Logging**.
 2. On the Audit Logging page, do the following:

State	Select to enable event recording in the audit log.
Detail Level	Select brief (default) or verbose output.
Audit Trail	Enables logging every command that is processed by the OECB. <ul style="list-style-type: none"> • enabled: Logs all commands that the OECB can process. • disabled: Logs only relevant information. • Default: disabled
Audit Record Output	Indicates how the OECB logs audit records. <ul style="list-style-type: none"> • syslog: The OECB logs audit records over syslog. • file: The OECB logs audit records to a file. • both: The OECB logs audit records over both syslog and to a file. • Default: file
File Transfer Time	Specify the amount of time, in hours, from the completion of the last transfer to the beginning of the next transfer. This determines when a file transfer occurs unless the Max storage space or Max file size triggers the transfer first. Default: 720. Range: 0-65535. <div style="border: 1px solid #0070C0; background-color: #E6F2FF; padding: 10px; margin-top: 10px;"> <p> Note:</p> <p>0 disables this parameter.</p> </div>
Max Storage Space	Specify the maximum amount of space that the audit log can consume on the OECB in MB. Default: 32. Range: 0-32.

Percentage Full	<p>Use in conjunction with Max storage space to specify the percent of the Max storage space that triggers file transfer. This determines when a file transfer occurs unless the File transfer time or Max file size triggers the transfer first. Default: 75. Range: 0-99.</p> <div data-bbox="630 415 760 457"> Note:</div> <p>0 disables this parameter.</p>
Max File Size	<p>Set the maximum size in Mega Bytes that the audit log can be before the system transfers the file. This determines when a file transfer occurs unless the Max storage space or Max file size triggers the transfer first. Default: 5. Range 0-10.</p> <div data-bbox="630 787 760 829"> Note:</div> <p>0 disables this parameter.</p>
Storage Path	<p>Specifies the directory that houses the audit log. Default: /code/audit .</p>
Push Receiver	<p>Add a push receiver and configure the following parameters for sending audit log files from the OECB to the receiver:</p> <ul style="list-style-type: none">• Server—Enter the IP address of the FTP/SFTP server to which you want the OECB to push audit log files. Default: 0.0.0.0.• Port—Enter the port number on the FTP/SFTP server to which the OECB will send audit log files. Default: 22 Range: 1-65535.• Remote Path—Enter the pathname to send the audit log files to the push receiver. Files are placed in this location on the FTP/SFTP server. Value: <string> remote pathname.• Filename Prefix—Enter the filename prefix to prepend to the audit log files that the OECB sends to the push receiver. The OECB does not rename local files. Values: <string> prefix for filenames.• Username—Enter the username the OECB uses to connect to this push receiver.• Auth Type—Select the authentication methodology. Password (default) or public key.• Do one of the following:

Password—When you set the Auth type to password, click **Set** to enter and confirm the password used to access this push receiver.

Public Key—When you set the Auth type to public key, select the public key profile that you want from the drop-down list.

3. Click **OK**.
4. Save the configuration.

System File Management

You can manage Oracle Enterprise Communications Broker (OECB) system and configuration files from the Web GUI on the **System tab** under **File Management**.

The following table lists and describes the file types that you can manage.

File Type	Format	Description
Backup Configuration	.gz	Contains a backup of the OECB software configuration. You can apply this file to restore a previous configuration.
Configuration CSV	.csv	Contains Comma Separated Value configuration files that you can upload.
Local Route Table (LRT)	.xml, .gz	Contains the Local Routing Table (LRT) file that you can apply to the OECB. The LRT is an in-memory table that contains IP addresses that the local router recognizes. It calculates the destinations of messages it is responsible for forwarding.
Fraud Protection Table	.gz, .gzip, .xml	Contains fraud protection files that you can upload, download, delete, or open to modify.
Log	Text	Contains Log files with information about the various aspects of the OECB. For example, information logged about the ACLI, SIP, or H323. Note: Only the Download and Delete functions are applicable to log files on the OECB.
Audit Log	Text	Contains a list of files that log system events.

File Type	Format	Description
Playback Media	Any media format valid in an RTP audio stream	Contains call progress playback files. The OECB can use these files in generated media streams. Note: The media files are raw binary files that contain data for the codec that you want played in the media stream. The OECB plays the data on the first audio flow in the Session Description Protocol (SDP).
Software Image	.gz, .bz	Contains bootable images.
SPL Plug-in	.lua	Contains a Session Plug-in Language (SPL) file that you can apply to the OECB to incorporate additional functionality. The SPL file contains a programming language capable of performing various tasks by utilizing APIs and callbacks in the OECB.

The following table lists and describes the file management controls that display in the file type dialog, according to the supported behavior for the file type.

Control	Description
Refresh	Updates the screen to display the latest data.
Add (Fraud Protection files, only.)	Adds a new Fraud Protection file.
Upload	Uploads a file from your server or PC to the OECB. The LRT, SPL, and backup configuration upload process provides the option of dynamically applying these files to the OECB.
Download All (Log files, only.)	Downloads all Log files from the OECB to your local server or PC (typically to the download directory on your system).
Backup	Creates a file that contains a backup of the device software configuration. You can apply this file to restore a previous configuration.
Restore (Backup configuration files, only.)	Restores and applies a Backup configuration file to the OECB.
Delete	Deletes the file type from the OECB.
Delete All (Log files, only.)	Deletes all Log files from the OECB.

Upload a File

You can upload the following file types from your local server or PC to the Oracle Enterprise Communications Broker (OECB).

 **Note:**

The Log and Audit Log files do not support uploading.

File Type	File Format	Directory
Backup Configuration	.gz	/code/bkups
Configuration CSV	.csv	
Local Subscriber Table (LST)	.xml.gz	/code/gzConfig
Software image	.bz, .tar or no extension specified	/code/images
SPL Plug-on (SPL)	.lua, .spl	/code/spl

The file extension must be applicable to the file type you select. For example, an SPL Plug-in file requires the .lua extension

You can dynamically activate the Local route table and SPL Plug-in during the upload process.

You can immediately restore a backup configuration file after an upload is complete.

 **Note:**

You cannot upload log files.

1. Access File Management: **System, File Management**.
2. On the File management page, select the type of file you want to upload.
3. In the Name column, select the file you want to upload.
4. Click **Upload**.
5. In the Upload file dialog, do the following:
 - a. Click **Browse**.
 - b. Select the file that you want to upload.
 - c. Optional. For the Backup configuration file, select Restore the configuration after upload to apply a previous backed up configuration file immediately to the after the upload is complete.
 - d. Optional. For the Local Subscriber Table file type, select Activate the LST file after upload to apply the LST upon upload.
 - e. Optional. For the SPL Plug-in file type, select Activate the SPL file after upload to apply the SPL file upon upload.
 - f. Click **Upload**.

Download a File

Use this procedure to download a file from the Oracle Enterprise Communications Broker (OECB).

You can download any of the following file types from your local server or PC to the OECB:

- Backup Configuration
 - CSV File
 - Local Subscriber Table (LRT)
 - Log
 - Software Image
 - SPL Plug-in (SPL)
1. Access File Management: **System, File Management.**
 2. On the File Management page, select the type of file you want to download from the File type list.
 3. Place your cursor on the row of the file that you want to download.

 **Note:**

For Log file types, you can select multiple log files to download, or place a checkmark in the box to the left of the Name column heading to select all log files to download. When downloading multiple log files, the File Management GUI compresses the files into one .tar file and downloads that file to your local server or PC.

4. Right-click the selected row, and click **Download.**
5. Do one of the following:
 - Click **Open With** and select the application to open the file.
 - Click **Save File** to save the file to your local server or PC.
6. Click **OK.**

The system downloads the file to the folder on your local server or PC where your Browser sends all downloads (typically your “Download” folder) or opens (decompresses) the file type on your local server or PC (typically in the “Download” folder).

Delete a File

The following information describes the procedure and conditions for deleting a file from the Oracle Enterprise Communications Broker (OECB).

You can delete any of the following file types from your local server or PC to the OECB:

- Local Subscriber Table (LST)

- SPL Plug-in (SPL)
- Backup Configuration

 **Note:**

You can select a single or multiple files to delete.

1. Access File Management: **System, File Management**.
2. From the File Management list, select the type of file that you want to delete.
3. Put your cursor in the row of the file you want to delete, and right-click.
Note: If the GUI does not display a menu upon right-click, use the **Delete All** control located above the list of files.
4. Click **Delete**. The system displays following message.
Are you sure you want to delete the file?
5. Click **Delete**.

Back Up a Configuration File

You can back up a configuration file from the Oracle Enterprise Communications Broker (OECB) to your local server or PC. Back up allows you to save configurations that you can restore to the OECB at a later time.

1. Access File Management: **System, File Management**.
2. Select a file from the list in the table.
3. Click **Backup**.
4. Click **OK**.
The system adds the file to the Backup Configuration table.

Restore a Configuration File

You can restore a backed up configuration file to the Oracle Enterprise Communications Broker (OECB).

1. Access File Management: **System, File Management**.
2. On the File Management page, select Backup Configuration.
3. Select a back up, and right-click.

 **Note:**

Restore activates only when you select a back up file.

4. Click **Restore**.
The GUI displays a confirmation dialog.
5. Click **Restore**.

The system downloads the back up file to the OECB. The OECB re-boots and restores the configuration from the back up file.

CSV Configuration File Creation

The Comma Separated Values (CSV) file is a text format file supported by spreadsheet applications. You can import a CSV file into the Oracle Enterprise Communications Broker (OECB) that contains its configuration, or you can export the current configuration on the OECB to the CSV file. You can also upload parts of your OECB configuration separately, such as users, dial plans, and routes. You can perform the upload manually or set the system to perform a periodic upload, automatically.

In the CSV file format, each row is defined on its own line and each column is separated by a comma.

You can create your own CSV configuration files, but be aware of the following rules for proper formatting.

- The OECB ignores empty lines.
- If an entry contains a comma, enclose it in quotes to prevent it from being treated as a separator.
- The first non-empty line must be the keyword “object:”, followed by the configuration object name that is being configured (shown below as “sip-interface”).

```
object:sip-interface
```

- The second non-empty line must state the operation to perform, which can be ADD, MODIFY, or DELETE.

```
operation: ADD
```

- The third non-empty line must state the parameter name of the object to be configured, and each parameter name must be in its own column. This row defines the “labels” for each column for the subsequent rows. Only the attributes you want defined need to be present. You can specify the parameter names in any order, but the data in subsequent rows must be consistent with the “labels” that you define in this row.

```
state,realm-id,description
```

- The fourth non-empty rows define instances (values) for the configuration object, each instance in its own column. In the following example, the third line defines a new sip-interface with state “enabled”, realm-id “public”, and description “public SIP interface”. These values are based on the “labels” defined in the second row.

```
enabled,public,public SIP interface
```

- On all subsequent rows, you can define any number of instances.
- The next row with an “object” keyword selects a new configuration object that is based on the previous object. You continue to input the data for this object

according to the rules stated above. The following example shows a “sip-port” object added that is related to the sip-interface object.

```
object:sip-port
operation:ADD
address,port,transport-protocol
192.168.1.1,5060,UDP
192.168.1.1,5061,TCP
```

- In the preceding example, “sip-port” is a sub-object of “sip-interface” that creates new sip-ports from the last sip-interface instance (of realm-id “public”).
- Note that the Description field displays all text as one continuous line, unless you insert line breaks. When you want to insert line breaks in the Description field, for example between sentences that you want displayed on separate lines, do the following:
 - From the GUI, in the Description field of a Configuration object, add Line1 to the end of the line where you want the first break to occur. Add Line2 to the end of the next line where you want a break to occur, and so on.
 - In a CSV configuration file, add \010Line1 to the end of the line where you want the first break to occur. Add \010Line2 to the end of the next line where you want a break to occur, and so on.

 **Note:**

After you create the initial CSV configuration file, you can set the OECB to automatically upload updated versions of the file. See "Automatically Upload Updated CSV Configuration Files" and "Configure Automatic CSV File Uploads."

Caveats

- Files are written to the volatile directory of the file system on the system. For the Acme Packet 4500, this is the “ramdrv/” directory. For the appliance and virtual machines, it is the “/var/” directory.
- Import and export occurs to and from the editing configuration.
- All error messages are printed to the screen, where the command was issued. Line numbers are provided with the error when possible.
- Objects and attributes cannot be set to instances (values) that are not allowed. For example, you cannot set an IP address to “enabled”. Parsing continues normally after this error.
- If an object cannot be written (i.e. key field is missing), then that object is discarded and parsing continues as normal.
- The import is additive. Each object that is imported is expected to be new to the configuration. If there is already an object with the same key present, it generates an error 409 and is discarded. Parsing continues as normal after the error.

Create a CSV Configuration File

You can create a Comma Separated Values (CSV) file of the agents, dial plan, users, and routing configurations to make maintaining and updating such information easier. After you create the file, you upload it to the Oracle Enterprise Communications Broker (OECB).

In the CSV file format, each row is defined on its own line and each column is separated by a comma. When preparing the CSV file, you must specify each object and operation that you want the OECB to act on. For each object in the .csv file, the first three lines of the file format include the following information:

- **object**—Specify the object you want the OECB to act on. Valid values: Agents, Dial Plan, Users, and Routing.
- **operation**—Specify the operation to perform. Valid values: ADD, MODIFY, or DELETE. If you do not specify an operation, the system defaults to ADD-MODIFY.
- **parameters**—A system provided list of all of the parameters for the object. The system can perform the specified operation on any of the parameters listed.

To create a CSV file that contains system configuration, do the following:

1. Open an application that supports a CSV file.
 2. In the first row, first column, enter "object:" followed by a configuration object you want to import.`object:sip-interface`
 3. In the second row, enter the operation to perform. `operation:ADD`.
 4. In the third row, and each in its own column, enter the parameter names of the objects to be configured. `state,realm-id,descriptionstate,realm-id,description`.
 5. In the fourth row, and each in its own column, enter the instances (values) for the configuration objects.`enabled,public,public SIP interface`.
 6. In subsequent rows, define additional instances (values), as needed.
 7. In the next empty row, first column, enter another object if needed, related to the first object (sip-interface).`object:sip-port`.
 8. Repeat steps 3 through 6 for this object.
 9. Save the file as a .csv.
 10. Upload the configuration file using the upload button from the applicable dialog. (For example, upload a .csv file of users from the User database.)
- After you create the initial CSV configuration file, you can set the OECB to automatically upload updated versions of the file. See "Automatically Upload Updated CSV Configuration Files" and "Configure Automatic CSV File Uploads."

Automatically Upload Updated CSV Configuration Files

When you want to automatically update selected configuration objects on the Oracle Enterprise Communications Broker (OECB), you can enable the system to periodically check for updates and automatically upload the CSV configuration file. Within the CSV file, you can specify the operation that you want the OECB to perform upon upload, such as add, modify, and delete data. The supported configuration objects include Agents, Dial Plan, Users, and Routing.

When enabled, the OECB checks /code/csv every two minutes for a new file and uploads it to the OECB. (The time interval is not configurable.)

When preparing the CSV file for upload, you specify each object and operation that you want the OECB to act on.

For each object in the CSV file, the first three lines of the file format include the following information:

- **object**—Specify the object you want the OECB to act on. Valid values: Agents, Dial Plan, Users, and Routing.
- **operation**—Specify the operation to perform. Valid values: ADD, MODIFY, or DELETE. If you do not specify an operation, the system defaults to ADD-MODIFY.
- **parameters**—A system provided list of all of the parameters for the object. The system can perform the specified operation on any of the parameters listed.

The following example shows the first three lines of the CSV file format:

```
object:user-number
operation:ADD/MODIFY/DELETE
AOR,number-or-pattern,description,dialing-context,agent,policy,tags
```

In the fourth line, you begin listing the parameter data to ADD, MODIFY, or DELETE. For example:

```
123@oracle.com,555[2000-3999],APAC,Server,
```

The OECB uses only one CSV file at a time as the source file, but the file can contain multiple objects and multiple operations. The following example shows a CSV file with multiple objects, operations, and parameter data:

```
object:user-number
operation:MODIFY
AOR,number-or-pattern,description,Dialing-context,agent,policy,tags
123@oracle.com,555[2000-3999],APAC,Server,
```

```
object:dialing-context
operation:ADD
Name, geographic location, description, country code, outside line prefix,
Dial patterns, remove prefix, pattern, description, country code,
replacement prefix, replacement uri, go to context
APAC.Nepal,CALA.Uruguay,EMEA.Andorra
```

```
object:policy
operation:DELETE
Name, description, name, contains codecs, codec condition, time condition,
next hop compare condition, contains codecs, missing codecs, name, days,
start time, end time, name, next hop compare mode, name, routing mode,
name redirect to agent, hairpin signaling, name, egress number translation
mode, number of digits for n digit dialing, prepend calls on egress, name,
ignore constraints, name, header name, dialing ccontext, result store, new
value, name, display name
<policy name>, next hop compare condition, time condition
```


 **Note:**

The OECB does not allow you to manually launch an upload during the automatic upload operation, and the system does not provide a way to queue a manual upload to launch when the automatic upload finishes. You must wait until the automatic upload finishes to launch a manual upload.

To enable automatic CSV file uploads, you enable the service in **system config**. See "Configure Automatic CSV Configuration File Uploads."

Enable Automatic CSV Configuration File Uploads

When you want the Oracle Enterprise Communications Broker (OECB) to automatically update the .csv configuration file, you must enable the service.

- Prepare the .csv file for upload with any additions, modifications, or deletions.
1. Access the System Config configuration object.
Configuration, System Administration, General Settings, System Config.
 2. On the System Config page, enable CSV Upload.
 3. Click **OK**.
 4. Save the configuration.

Upgrade Software

You can upgrade the Oracle Enterprise Communications Broker (OECB) software from the System tab. The system requires a reboot after the upgrade.

1. Access Upgrade Software: **System, Upgrade Software**.
2. In the Upgrade Software dialog, click **Verification**, and do the following:
 - View the health score.
 - Click **View Health Information**, and confirm that the system components are synchronized.
 - Click **View Configuration Version**, and note the Current Version and Running Version.
 - Click **View Disk Usage**, and confirm that the system has enough free space.
3. Select one of the following upload methods.

Local	Select a file from your system, and proceed to Step 4
Flash	Select a file already on the OECB, and proceed to Step 4.
Network	Do the following: <ul style="list-style-type: none"> • Boot file. (Network) Enter the complete name of the boot file. • Host IP. (Network) Enter the IP address of the FTP server.

- FTP username. (Network) Enter the user name to log onto the FTP server.
- FTP password. (Network) Enter the password to log onto the FTP server.
- Optional. Select `Reboot after upload`.

Proceed to Step 6.

4. Select a file to upload.
5. (Optional) Select `Reboot after upload`.
6. Click `Complete`.
 - If you did not select `Reboot after upload`, the system displays a message stating that a reboot is required for the changes to take effect.
 - If you selected `Reboot after upload`, the system displays a message stating that it is about to reboot.
7. Click `OK`.

If you selected `Reboot after upload`, the system reboots.

System Reboot

You can manually reboot the Oracle Enterprise Communications Broker (OECB) from the Web GUI. If you have a High Availability (HA) deployment, connectivity to the standby OECB stops until the reboot completes.

When the reboot completes, the active and standby systems each display the log on screen and you must manually log on to each system.



Note:

When you reboot the system from the Web GUI, the Web GUI is unavailable until the reboot completes.

When you perform a reboot from the Web GUI	The system behaves
and no boot is in process and the system is not switching over to the standby system of an HA pair,	the GUI session closes and the system displays the Log On screen. You cannot log on to the Web GUI until the reboot completes on the OECB.
and a reboot is already in progress,	the system displays a message stating that a reboot cannot occur. The first reboot must complete before another reboot is initiated.
and the active system is currently switching over to the standby system in an HA environment,	the system displays a message stating that a reboot cannot occur because the HA switch over is underway. The standby OECB is updating and getting its configuration from the active OECB.

Display Log Files

The Oracle Enterprise Communications Broker (OECB) allows you to view log files without needing to download them.

1. Access the **File Management** page. Click **System, File Management**.
2. On the **File Management** page, select a Log file from list.
3. Expand a log file category and select a log file by selecting the check box by the file name.

The OECB enables the **View** control.

4. Click **View**.

The OECB displays the **Viewing log:[filename]** dialog with the log file's contents.

Display System Health

The Oracle Enterprise Communications Broker (OECB) provides a widget that allows you to see the current health score and state of the OECB.

- Access the **System Health** page. Click **Monitoring, Widgets, System, System Health**.

The GUI displays the **System Health Table**, where you can see the health score and state of the OECB.

Obtain Support Information

You can manually generate a file by way of the Web GUI that contains troubleshooting information. You can save the file and send it to Oracle Customer Support.

1. Access Support Information: **System, Support Information**.
2. In the **Confirm** dialog, click **Confirm**.
The system generates the file.
3. The browser asks you to do one of the following:
 - Open the file—Select the browser and click **OK**, select the tool and click **OK**. The GUI displays the file.
 - Save the file—Click **OK**. The GUI saves the file.
4. Click **OK**.

The system downloads the file to the specified location.

Active Directory Modifications

When using the Oracle Enterprise Communications Broker LDAP configuration to access authentication and routing information from Active Directory (AD), you must prepare AD to serve those functions. For authentication, you can add an Oracle-supplied DLL to the system to capture password hashes during password changes and store them for authentication.

The Oracle-supplied DLL, **oecbpwdcn.dll**, is an OSD DLL that provides the Windows-specific password hash capture function. When a user changes their password, the DLL intercepts the hash of the password and stores it for SIP authentication. The user's password is never visible in clear-text.

Related AD changes consist of the following, which can be done manually or by way of Oracle-provided scripts:

1. Create `orclDigestRealmAttribute` attribute (to store digest realm name) and associate it users.
2. Create `orclDigestPwdAttribute` attribute (to store hashed password) and associate with users.
3. Create `orclAgentNameAttribute` and associate it with users.

You can refer to <http://www.microsoft.com/technet/prodtechnol/windows2000serv/technologies/activedirectory/howto/adschema.mspix> for instructions on managing AD. You can manually add the following entries into AD:

```
dn:
cn=orcldigestrealmattribute,cn=schema,cn=configuration,dc=example,dc=com
changetype: add
objectClass: top
objectClass: attributeSchema
cn: orclDigestRealmAttribute
instanceType: 4
attributeID:
1.2.840.113556.1.8000.2554.54362.52699.4250.17878.46369.10622351.7266019.1
attributeSyntax: 2.5.5.4
isSingleValued: TRUE
showInAdvancedViewOnly: TRUE
adminDisplayName: orclDigestRealmAttribute
adminDescription: Oracle ECB Digest Realm
oMSyntax: 20
LDAPDisplayName: orclDigestRealmAttribute
name: orclDigestRealmAttribute
```

This creates the attribute to which **oecbpwdcn.dll** stores password hashes.

```
dn: cn=orcldigestpwdattribute,cn=schema,cn=configuration,dc=example,dc=com
changetype: add
objectClass: top
```

```

objectClass: attributeSchema
cn: orclDigestPwdAttribute
instanceType: 4
attributeID:
1.2.840.113556.1.8000.2554.54362.52699.4250.17878.46369.10622351.7266019.2
attributeSyntax: 2.5.5.4
isSingleValued: TRUE
showInAdvancedViewOnly: TRUE
adminDisplayName: orclDigestPwdAttribute
adminDescription: Oracle ECB Digest Password
oMSyntax: 20
LDAPDisplayName: orclDigestPwdAttribute
name: orclDigestPwdAttribute

```

This creates an attribute that can be used for routing, specifically by providing a field for storing the users' Agent.

```

dn: cn=orclagentnameattribute,cn=schema,cn=configuration,dc=example,dc=com
changetype: add
objectClass: top
objectClass: attributeSchema
cn: orclAgentNameAttribute
instanceType: 4
attributeID:
1.2.840.113556.1.8000.2554.54362.52699.4250.17878.46369.10622351.7266019.3
attributeSyntax: 2.5.5.4
isSingleValued: TRUE
showInAdvancedViewOnly: TRUE
adminDisplayName: orclAgentNameAttribute
adminDescription: Oracle ECB Agent Name
oMSyntax: 20
LDAPDisplayName: orclAgentNameAttribute
name: orclAgentNameAttribute

```

 **Note:**

You must replace %AD_DOMAN_NAME% with your AD domain name, such as dc=acme,dc=com.

For convenience, two LDIF files are provided to facilitate adding these two attributes. They are "addOrclECBAttribute.ldif" and "addUserObjClass.ldif". To add the two attributes automatically:

1. Make sure the Active Directory Schema Snap-In is installed by following the directions from:
 - <http://social.technet.microsoft.com/wiki/contents/articles/20319.how-to-create-a-custom-attribute-in-active-directory.aspx> or
 - [http://technet.microsoft.com/en-us/library/cc759633\(v=ws.10\).aspx](http://technet.microsoft.com/en-us/library/cc759633(v=ws.10).aspx)
2. Open the two files and replace %AD_DOMAN_NAME% with your actual AD domain name, such as dc=acme,dc=com.

3. Run the command "**ldifde -i -f addOrcIECBAAttribute.ldif -v**" to create the three attributes.
4. Then run the command "**ldifde -i -f addUserObjClass.ldif -v**" to associate the attributes to AD users.
5. Reload the AD schema or reboot AD.
6. Verify that the two attributes are present by checking users to see that attributes are available to them.

In addition to AD schema modification, follow the steps below to install **oecbpwdcn.dll**.

1. Install OID Password Change Notification (oecbpwdcn) DLL, by simply copying the oecbpwdcn.dll to your AD WINDOWS\system32
2. Using regedt32 to change the registry and enable the DLL. Invoke regedt32 and modify the registry setting HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\Notification Packages. Add "oecbpwdcn" to the end of this list. Example registry entries, including oecbowdcn, could now include:
 - RASSFM
 - KDCSVC
 - WDIGEST
 - scecli
 - oecbpwdcn
3. Reboot AD.

Test your deployment as follows:

1. Assign a digest realm name to user's orclDigestRealmAttribute in AD. You can use script modifyUsersDigestRealmName.vbs to modify this attribute for all users. Right click on modifyUsersDigestRealmName.vbs and select "Run with Command Prompt"
2. Modify user password for any AD user (or reset the password)
3. Search against AD and look up the AD user and orclDigestPwdAttribute should have the generated hash value.

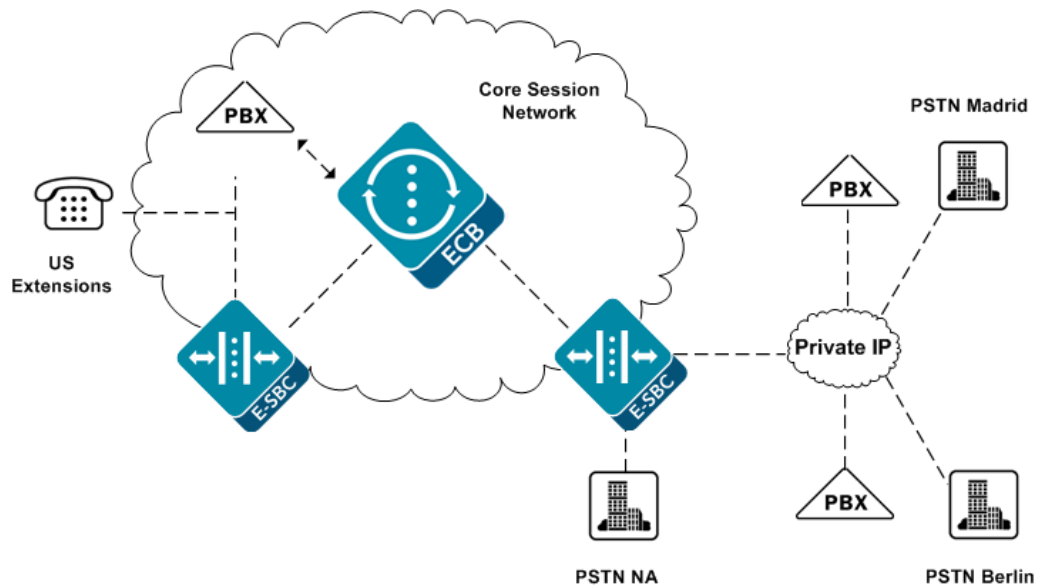
You can use a script named **displayUsersDigestRealmPassword.vbs** to display the values from all users. To do this, right-click **displayUsersDigestRealmPassword.vbs** and select "Run with Command Prompt."

18

Configuration Examples

The Oracle Enterprise Communications Broker (OECB) is a flexible tool that provides a variety of configuration options that can achieve similar results.

In the following diagram an OECB is deployed at the center of the session network to manage SIP signaling traffic for an enterprise headquartered in the U.S., with branch offices in the U.S., Spain, and Germany.



There are multiple E-SBCs, PBXs, and tie lines handling session services. The OECB must normalize multiple signaling formats, integrate multiple vendor processes, provision varied session services, and handle an enterprise user database.

The following topics provide more examples for configuring the OECB using sample configurations targeting specific environmental models. The information may help you to identify configuration options that track closely to the needs of your deployment.

[Configuration Sequence](#)

[Small Enterprise Model](#)

[Large Enterprise Model - v2](#)

[Emergency Dial Configurations](#)

[Alternate Translation Modes](#)

[ENUM Example Configuration](#)

Configuration Sequence

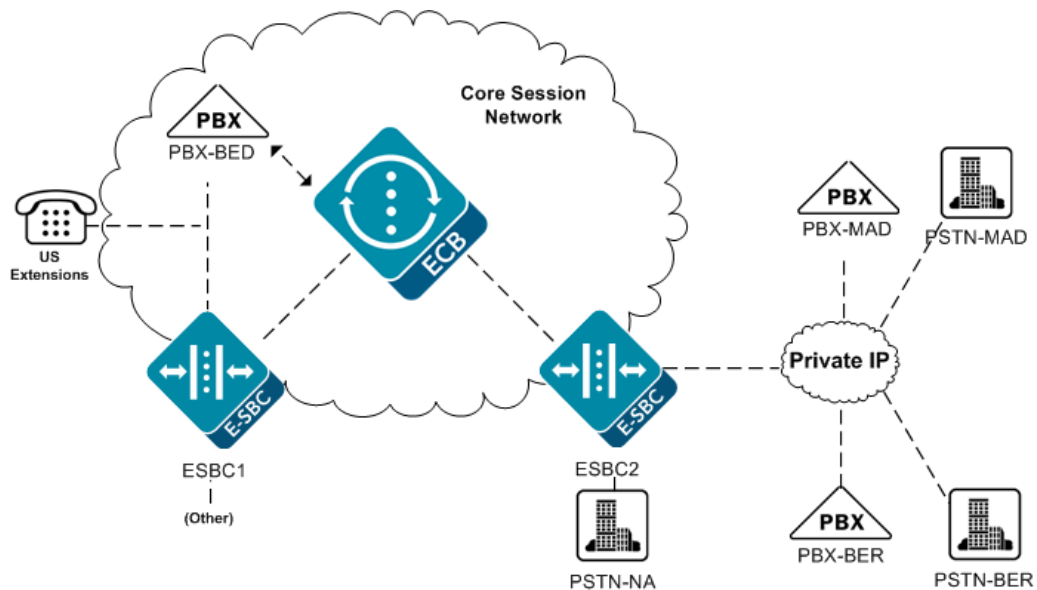
When first configuring the Oracle Enterprise Communications Broker for service, follow the sequence below to establish objects that you need in ensuing object configurations.

1. Agents - Agents help to segregate your network logically and geographically, providing Agent configuration also specifies the hops available to you on your network for routes. In addition Users often use Agents to help specify their contexts.
2. Dial Plans - Dial Plans specify all the contexts that you use in routes. Users may also be configured with contexts.
3. Users - It is often best to configure Users with existing Agents and Dial Plans to specify their locations and applicable policies.
4. Routes - Routes often use Agents to specify source and destinations.

You will find this sequence is of decreasing value over time. In addition, it is not required that you follow this sequence at any point. The flexibility of the Oracle Enterprise Communications Broker's configuration options allow you perform steps such as creating new contexts as you create users. The sequence is most useful for understanding how to piece together the elements of the Oracle Enterprise Communications Broker.

Initial Agent Configuration

The following diagram expands upon the network diagram in "Configuration Examples" by adding specific names for the agents required for this Oracle Enterprise Communications Broker deployment.



The following table shows agent configurations that correspond to the agents in the preceding diagram. This table provides examples from which you can learn the guidelines for configuring your own agents. If you were to perform the configuration for this example, you would configure each agent in the following table with IP address, Port, and Transport mode. Note that in this example, all Agents take the default

Number Translation Mode, which is e.164. No other configuration is necessary to support the configuration in this example.

Hostname	Source Control	# Translation Mode	# of Digits	Prepend Prefix
PBX-BED		e.164		
PBX-MAD		e.164		
PBX-BER		e.164		
ESBC1		e.164		
ESBC2		e.164		
PSTN-NA		e.164		
PSTN-MAD		e.164		
PSTN-BER		e.164		

 **Note:**

In an actual deployment, you can add more agent configurations, as needed.

Dial Plan Strategies

Oracle Enterprise Communications Broker configuration design provides you with the flexibility to make the same settings, such as country code, on multiple elements. To the extent that child elements inherit properties from parent elements, endeavor to elegantly cover the basic requirements of your deployment with your initial configurations while preserving configuration options in child objects to meet the needs of exceptions and expansion.

The simplest way to approach dial plan configuration is to base your corporate contexts on your enterprise's branch offices. You configure dials plan and patterns comes within those contexts. The parent context establishes rules that you need enforced across the enterprise. Each branch office gets a child context that inherits from both the corporate parent and the rules associated with the country in which branch resides (geographic context).

Recall that the Oracle Enterprise Communications Broker comes pre-configured with the vast majority of geographic contexts you need. These contexts include the country code. By setting a geographic location to your branch office, you inherit country code. These contexts also include a description (of their location), which has no relevance to signaling processing.

The following corporate context configurations apply to the early enterprise configuration models presented in the ensuing sections.

Name	Geographic location	Country Code	Outside line prefix
acme			
acme.bedford	NA		9
acme.madrid	EU.Spain		
acme.berlin	EU.Germany		

This appendix uses dial plans and dial patterns to establish differentiation between configuration models, with all using the same routes. For this reason, this appendix strays from the suggested configuration sequence and sets up routes next.

Route Strategies

Route configuration consists of mapping out an extensible strategy according to your deployment model and connecting agents together. Configure all agents as simply as possible and create only as many routes as are necessary. Careful planning allows you to create routes that serve multiple purposes simultaneously.

You may recall that initial configuration procedures has user configuration preceding route configuration. This configurations design, however, covers all expected users without specific user configuration.

Routes Configured to Support both Small and Medium/Large Enterprise Models

Route #	Source Agent	Calling #	Called #	Dest Agent	Route	Cost
#1	*	*	34*	*	PSTN-MAD	20
#2	*	*	*	PSTN-MAD	ESBC2	0
#3	*	*	*	PBX-MAD	ESBC2	0
#4	*	*	49*	*	PSTN-BER	20
#5	*	*	*	PSTN-BER	ESBC2	0
#6	*	*	*	PBX-BER	ESBC2	0
#7	*	*	1*	*	PSTN-NA	20
#8	*	*	*	*	PSTN-NA	70

The table below explains the purpose of each route in the table above.

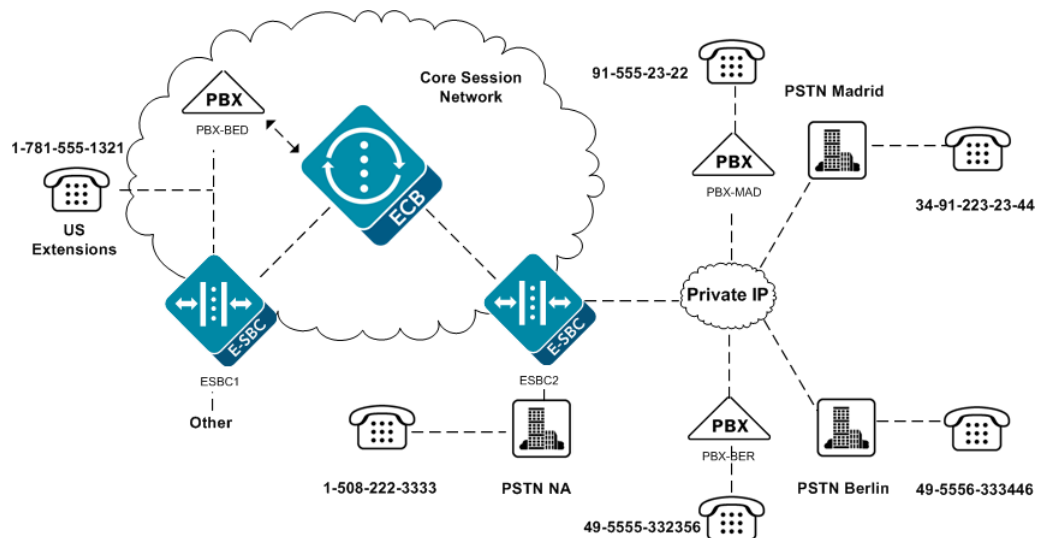
Route #	Description
#1	For traffic destined to Spain preceded with a "34" (Spain Country Code) and sourced anywhere, send to the PSTN agent in Madrid.
#2	For traffic destined to the PSTN in Madrid, use ESBC2.
#3	For traffic destined within the enterprise in Madrid, use ESBC2.
#4	For traffic destined to Germany preceded with a "49" (German Country Code) and sourced anywhere, send to the PSTN agent in Berlin.
#5	For traffic destined to the PSTN in Berlin, use ESBC2.
#6	For traffic destined within the enterprise in Berlin, use ESBC2.
#7	For traffic destined anywhere preceded with a "1" (US Country Code) and sourced anywhere, send to the North American PSTN agent.
#8	Default Route - If unable to determine any preferable route, use the most expensive route, which offloads traffic to the North American PSTN agent. Note the cost of 70, which is the highest cumulative cost of any other route set.

 **Note:**

Routes 1, 4 and 7 are examples of tail hop routing, which keeps traffic within the enterprise network for as long as possible before issuing a call to Germany locally in Berlin.

Small Enterprise Model

The following diagram shows example extension numbers for the small enterprise model that must be reachable. The key characteristic for this model is the absence of overlap of dial patterns, called contexts, across branch locations.



The following tables define an example of the dial plan called `acme` for a small enterprise.

The following table shows dial patterns configured for the `acme` context.

Prefix	Pattern	Country Code	Replace Prefix	Replace URI	Go To Context
	1xxx	1	781555		
	2xxx	34	91555		
	3xxxxx	49	5555		

Applicable user database entries.

Number	Dialing Context	Agent
+17815551xxx		PBX-BED
+34915552xxx		PBX-MAD
+4955553xxxxx		PBX-BER

Child contexts inherit the rules of the corporate context, `acme`, and are also configured to inherit the geographic location contexts for the countries in which they reside.

Regarding the configured routes, numbers preceded with the appropriate country code that do not match these dial patterns go to the PSTN of their respective countries.

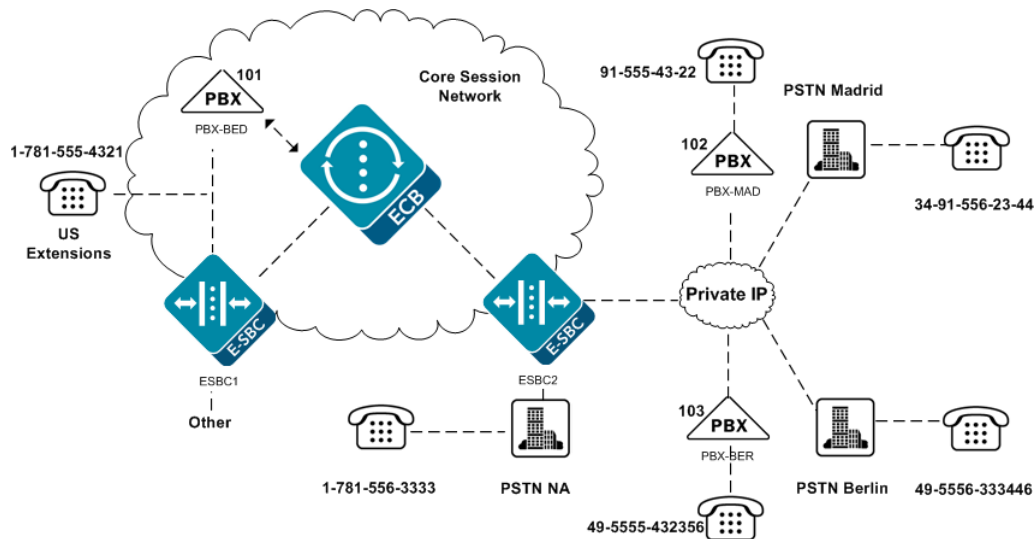
Reviewing the fields that do not require configuration is equally instructive. Based on the inheritance of rules from parent contexts, a means of basic access is available throughout the enterprise. In addition, based on the higher precedence of child context configurations, these fields are available to you to configure for special requirements at the separate branches.

This configuration generates the results shown in the following table.

From	Dial String	Transformation	Result
acme	1321	+17815551321	Call is directed to PBX-BED as e.164
acme	2322	+345552322	Call is directed to PBX-MAD as e.164
acme	332356	+495555332356	Call is directed to PBX-BER as e.164

Large Enterprise Model - v2

The following diagram shows example extension numbers for the medium to large enterprise model that must be reachable. The key characteristic for this model is the overlap of dial patterns, called contexts, across branch locations.



In this example, specific dial patterns are configured for the `acme` context that each child context inherits. These dial patterns provide the Oracle Enterprise Communications Broker with the means of distinguishing between branches, even though the extension patterns still begin with the same number.

Dial Patterns Configured for the `acme` Context

Prefix	Pattern	Country Code	Replace Prefix	Replace URI	Go To Context
101					acme.bedford
102					acme.madrid

Prefix	Pattern	Country Code	Replace Prefix	Replace URI	Go To Context
103					acme.madrid
555				helpdesk@acme.com	

Each child context inherits these dial patterns from the acme context, providing you with a means of hierarchically assigning context.

In contrast to the small enterprise model, note the use of dial patterns specific to each branch. These contexts work in conjunction with the parent context's parents, delivering signaling with the enterprise's own prefixes to the branch's PBX.

Dial patterns configured for the `acme.bedford` context

Prefix	Pattern	Country Code	Replace Prefix	Replace URI	Go To Context
	4xxx		781555		

Dial Patterns Configured for the `acme.madrid` Context

Prefix	Pattern	Country Code	Replace Prefix	Replace URI	Go To Context
	4xxx		91555		

Dial Patterns Configured for the `acme.berlin` Context

Prefix	Pattern	Country Code	Replace Prefix	Replace URI	Go To Context
111	4xxxxx		555 berlinhelpdesk@acme.com		

Applicable user database entries

Number	Dialing Context	Agent
+17815551xxx		PBX-BED
+34915552xxx		PBX-MAD
+4955553xxxxx		PBX-BER

In contrast to the small enterprise model, note the use of dial patterns specific to each branch.

This configuration provides the results shown in the following table.

From	Dial String	Transformation	Result
acme.bedford	4321	+17815554321	Call is directed to PBX-BED as e.164
acme.bedford	1024322	+34915554322	Call is directed to PBX-MAD as e.164

From	Dial String	Transformation	Result
acme.bedford	103432356	+495555432356	Call is directed to PBX-BER as e.164
acme.madrid	4322	+34915554322	Call is directed to PBX-MAD as e.164
acme.madrid	1014321	+17815554321	Call is directed to PBX-BED as e.164
acme.madrid	103432356	+495555432356	Call is directed to PBX-BER as e.164
acme.berlin	432356	+495555432356	Call is directed to PBX-BER as e.164
acme.berlin	1014321	+17815554321	Call is directed to PBX-BED as e.164
acme.berlin	1024322	+34915554322	Call is directed to PBX-MAD as e.164

Emergency Dial Configurations

To dial emergency numbers, you add dial patterns to the corporate context child context for the emergency numbers. Configure these dial patterns with a Replace URI transformation that inserts the RFC 5031 compliant URN for emergency services. The following contexts and routes show examples of such a configuration. The last table in this topic shows the results of this configuration.

Dial Patterns Configured for the `acme.bedford` context

Prefix	Pattern	Country Code	Replace Prefix	Replace URI	Go To Context
	911			URN:service:sos	

Dial Patterns Configured for the `acme.madrid` context

Prefix	Pattern	Country Code	Replace Prefix	Replace URI	Go To Context
	112			URN:service:sos	

Dial Patterns Configured for the `acme.berlin` context

Prefix	Pattern	Country Code	Replace Prefix	Replace URI	Go To Context
	112			URN:service:sos	

In addition to the preceding dial patterns, the system needs three new routes.

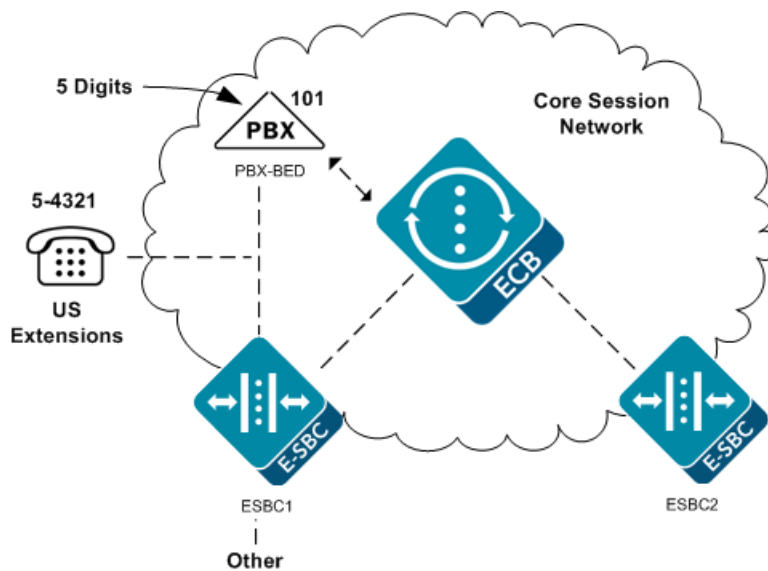
Route #	Source Agent	Calling #	Called #	Dest Agent	Route	Cost
#1	*	1*	"service:sos "	*	PSTN-NA	0
#2	*	34*	"service:sos "	*	PSTN-MAD	0
#3	*	49*	"service:sos "	*	PSTN-BER	0

This configuration produces the following results.

From	Dial String	Transformations	Result
acme.bedford	911	URN:service:sos	Call is directed to PSTN-NA as emergency URN
acme.madrid	112	URN:service:sos	Call is directed to PSTN-MAD as emergency URN
acme.berlin	112	URN:service:sos	Call is directed to PSTN-BER as emergency URN

Alternate Translation Modes

As described in agent configuration instructions, a translation mode specifies the format required by that agent. The configuration normally applies to a PBX when it is the last agent in the path to the user.



This example implements a different configuration for PBX-BED, as follows.

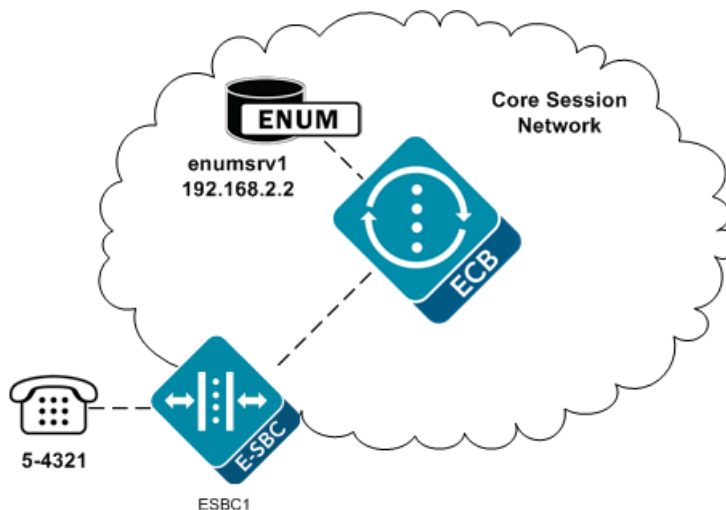
Hostname	Source Context	Number Translation Mode	Number of Digits	Prepend Prefix
PBX-BED		n-digit-dialing	5	

This configuration produces the following results.

From	Dial String	Transformation	Result
acme.bedford	54321	+17815554321	Call is directed to PBX-BED as a 5 digit number (54321)
acme.madrid	10154321	+17815554321	Call is directed to PBX-BED as a 5 digit number (54321)
acme.berlin	10154321	+17815554321	Call is directed to PBX-BED as a 5 digit number (54321)

ENUM Example Configuration

This example provides for the requirement that the signaling make an ENUM dip to resolve some element of the overall signaling path. The configuration must be able to provide ENUM resolutions to provide for the following deployment.



This configuration assumes the agent ESBC1 does not have a configured IP address. In addition, all numbers beginning with 5 are routed to ESBC1. To provide resolution for ESBC1, create an ENUM configuration as follows. In this example, the ENUM configuration includes only one server.

Hostname	Domain	Servers	Number Trans Mode	Number of Digits
enumsrv1	acme.com	192.168.2.2		

Configure a route for all 5-digit extensions beginning with a 5 to the agent named ENUM.

Route #	Source Agent	Calling number	Called number	Dest Agent	Route	Cost
#1		*	5xxxx	*	enum:enum srv1	

The following table explains the purpose of the route in the preceding table.

Route #	Description
#1	For traffic destined to a five-digit number beginning with 5, go to ENUM server to resolve the address of ESBC1.

19

Format of Exported Text Files

This section provides a sample and format of each type of exported file from the Web-based GUI. Sample information in these files are provided as a reference for your convenience.

Exported file examples include:

- Session Summary exported file (text format)
- Session Details exported file (text format)
- Ladder Diagram exported file (HTML format)

Note:

Oracle recommends you open an exported text file using an application that provides advanced text formatting to make it easier to read.

Exporting Files

The Web-based GUI allows you to export Monitor and Trace information to a text file from the Sessions, Registrations, Subscriptions and Notable Events Reports, as well as from a specific ladder diagram, or from a page containing the results of a search. The data exports to a file that you can open and view as required.

You can export any of the following to a file:

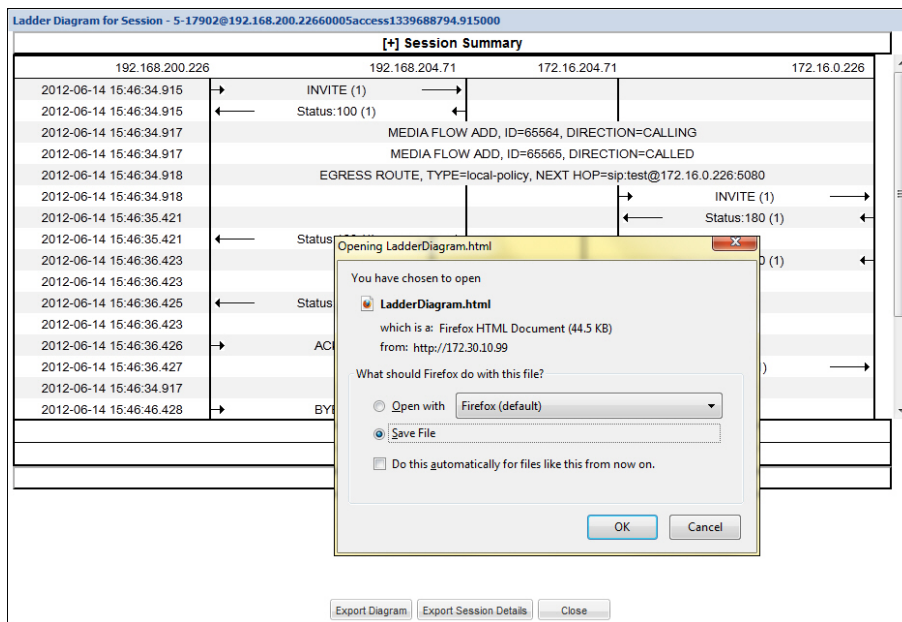
From the Sessions, Registrations, Subscriptions, and Notable Events Reports:

- **Export session details** - Exports the SIP messages and media events associated with the selected session, to a text file.
- **Export summary** - Exports all logged session summary records, to a text file. (Exports ALL call session summary records or the records that matched a search criteria).

From the Ladder Diagram:

- **Export diagram** - Exports all of the information in the Ladder Diagram to an HTML file (Session Summary, SIP Message Details, and QoS statistics).
- **Export session details** - Exports detailed information about the SIP messages and media events in the Ladder Diagram associated with the selected session, to a text file.

The following example shows the export of a Ladder Diagram to a file called LadderDiagram.html.



Session Summary Exported Text File

The following example shows a Session Summary text file exported from the GUI.

```
-----Session Summary-----
```

```
Startup Time: 2011-09-20 12:58:44.375
```

```
State: TERMINATED-200
```

```
Duration: 5
```

```
From URI: sipp < sip:sipp@172.16.34.16:5060 >;tag=1
```

```
To URI: sut < sip:service@172.16.34.225:5060 >;tag=13451
```

```
Ingress Src Address: 172.16.34.16
```

```
Igress Src Port: 5060
```

```
Igress Dest Address: 172.16.34.225
```

```
Igress Dest Port: 5060
```

```
Egress Source Address: 192.168.34.225
```

```
Egress Source Port: 5060
```

```
Egress Destination Address: 192.168.34.17
```

```
Egress Destination Port: 5060
```

```
Igress Realm: access
```

```
Egress Realm: backbone
```

```
Igress NetworkIf: access
```

```
Egress NetworkIf: backbone
```

```
-----Session Summary-----
```

```
Startup Time: 2011-09-20 12:58:05.340
```

```
State: TERMINATED-200
```

```
Duration: 5
```

```
From URI: sipp < sip:sipp@172.16.34.16:5060 >;tag=1
```

```
To URI: sut < sip:service@172.16.34.225:5060 >;tag=13450
```

```
Ingress Src Address: 172.16.34.16
```

```
Igress Src Port: 5060
Igress Dest Address: 172.16.34.225
Igress Dest Port: 5060
Egress Source Address: 192.168.34.225
Egress Source Port: 5060
Egress Destination Address: 192.168.34.17
Egress Destination Port: 5060
Igress Realm: access
Egress Realm: backbone
Igress NetworkIf: access
Egress NetworkIf: backbone
```

Session Details Exported Text File

The following example shows a Session Details text file exported from the GUI.

Session Details:

```
-----
Nov 3 08:50:56.852 On [2:0]172.16.34.225:5060 received from
172.16.34.16:5060

INVITE sip:service@172.16.34.225:5060 SIP/2.0
Via: SIP/2.0/UDP 172.16.34.16:5060;branch=z9hG4bK-1-0
From: sipp <sip:sipp@172.16.34.16:5060>;tag=1
To: sut <sip:service@172.16.34.225:5060>
Call-ID: 1-668@172.16.34.16
CSeq: 1 INVITE
Contact: sip:sipp@172.16.34.16:5060
Max-Forwards: 70
Subject: Performance Test
Content-Type: application/sdp
Content-Length: 135

v=0
o=user1 53655765 2353687637 IN IP4 172.16.34.16
s=-
c=IN IP4 172.16.34.16
t=0 0
m=audio 6000 RTP/AVP 0
a=rtpmap:0 PCMU/8000

-----
Nov 3 08:50:56.855 On [2:0]172.16.34.225:5060 sent to 172.16.34.16:5060

SIP/2.0 100 Trying
Via: SIP/2.0/UDP 172.16.34.16:5060;branch=z9hG4bK-1-0
From: sipp <sip:sipp@172.16.34.16:5060>;tag=1
To: sut <sip:service@172.16.34.225:5060>
Call-ID: 1-668@172.16.34.16
CSeq: 1 INVITE

----MBCD Evt
```

```
Nov 3 08:50:56.862 On 127.0.0.1:2945 sent to 127.0.0.1:2944
mbcdEvent=FLOW ADD
FlowCollapsed=enabled
FlowDirection=CALLING
FlowID=65541
MediaFormat=0
MediaReleased=disabled
MediaType=audio/PCMU/
OtherFlowID=0
TOSBits=0
InputSourcev4Addr=
InputSourcePort=0
InputDestv4Addr=172.16.34.225
InputDestPort=10004
OutputSourcev4Addr=192.168.34.225
OutputDestv4Addr=
OutputDestPort=0
InputRealm=access
OutputRealm=backbone
----MBCD Evt
Nov 3 08:50:56.862 On 127.0.0.1:2945 received from 127.0.0.1:2944
```

```
mbcdEvent=FLOW ADD
FlowCollapsed=enabled
FlowDirection=CALLED
FlowID=65542
MediaFormat=0
MediaReleased=disabled
MediaType=audio/PCMU/
OtherFlowID=0
TOSBits=0
InputSourcev4Addr=
InputSourcePort=0
InputDestv4Addr=192.168.34.225
InputDestPort=20004
OutputSourcev4Addr=172.16.34.225
OutputDestv4Addr=172.16.34.16
OutputDestPort=6000
InputRealm=backbone
OutputRealm=access
```

```
-----
Nov 3 08:50:56.865 On [1:0]192.168.34.225:5060 sent to 192.168.34.17:5060
```

```
INVITE sip:service@192.168.34.17:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.34.225:5060;branch=z9hG4bK4od0io20183g8ssv32f1.1
From: sipp <sip:sipp@172.16.34.16:5060>;tag=1
To: sut <sip:service@172.16.34.225:5060>
Call-ID: 1-668@172.16.34.16
CSeq: 1 INVITE
Contact: <sip:sipp@192.168.34.225:5060;transport=udp>
Max-Forwards: 69
Subject: Performance Test
Content-Type: application/sdp
Content-Length: 140
```

```
v=0
o=user1 53655765 2353687637 IN IP4 192.168.34.225
s=-
c=IN IP4 192.168.34.225
t=0 0
m=audio 20004 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

```
-----
Nov 3 08:50:56.868 On [1:0]192.168.34.225:5060 received from
192.168.34.17:5060
```

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP 192.168.34.225:5060;branch=z9hG4bK4od0io20183g8ssv32f1.1
From: sipp <sip:sipp@172.16.34.16:5060>;tag=1
To: sut <sip:service@172.16.34.225:5060>;tag=2578
Call-ID: 1-668@172.16.34.16
CSeq: 1 INVITE
Contact: <sip:192.168.34.17:5060;transport=UDP>
Content-Length: 0
```

```
-----
Nov 3 08:50:56.872 On [2:0]172.16.34.225:5060 sent to 172.16.34.16:5060
```

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP 172.16.34.16:5060;branch=z9hG4bK-1-0
From: sipp <sip:sipp@172.16.34.16:5060>;tag=1
To: sut <sip:service@172.16.34.225:5060>;tag=2578
Call-ID: 1-668@172.16.34.16
CSeq: 1 INVITE
Contact: <sip:service@172.16.34.225:5060;transport=udp>
Content-Length: 0
```

```
-----
Nov 3 08:50:56.872 On [1:0]192.168.34.225:5060 received from
192.168.34.17:5060
```

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 192.168.34.225:5060;branch=z9hG4bK4od0io20183g8ssv32f1.1
From: sipp <sip:sipp@172.16.34.16:5060>;tag=1
To: sut <sip:service@172.16.34.225:5060>;tag=2578
Call-ID: 1-668@172.16.34.16
CSeq: 1 INVITE
Contact: <sip:192.168.34.17:5060;transport=UDP>
Content-Type: application/sdp
Content-Length: 137
```

```
v=0
o=user1 53655765 2353687637 IN IP4 192.168.34.17
s=-
c=IN IP4 192.168.34.17
t=0 0
m=audio 6000 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

```
----MBCD Evt
```

Nov 3 08:50:56.878 On 127.0.0.1:2945 sent to 127.0.0.1:2944

mbcdEvent=FLOW MODIFY
FlowCollapsed=enabled
FlowDirection=CALLING
FlowID=65541
MediaFormat=0
MediaReleased=disabled
MediaType=audio/PCMU/
OtherFlowID=0
TOSBits=0
InputSourcev4Addr=
InputSourcePort=0
InputDestv4Addr=172.16.34.225
InputDestPort=10004
OutputSourcev4Addr=192.168.34.225
OutputDestv4Addr=192.168.34.17
OutputDestPort=6000
InputRealm=access
OutputRealm=backbone

Nov 3 08:50:56.881 On [2:0]172.16.34.225:5060 sent to 172.16.34.16:5060

SIP/2.0 200 OK
Via: SIP/2.0/UDP 172.16.34.16:5060;branch=z9hG4bK-1-0
From: sipp <sip:sipp@172.16.34.16:5060>;tag=1
To: sut <sip:service@172.16.34.225:5060>;tag=2578
Call-ID: 1-668@172.16.34.16
CSeq: 1 INVITE
Contact: <sip:service@172.16.34.225:5060;transport=udp>
Content-Type: application/sdp
Content-Length: 138
v=0
o=user1 53655765 2353687637 IN IP4 172.16.34.225
s=-
c=IN IP4 172.16.34.225
t=0 0
m=audio 10004 RTP/AVP 0
a=rtpmap:0 PCMU/8000

Nov 3 08:50:56.883 On [2:0]172.16.34.225:5060 received from
172.16.34.16:5060

ACK sip:service@172.16.34.225:5060 SIP/2.0
Via: SIP/2.0/UDP 172.16.34.16:5060;branch=z9hG4bK-1-5
From: sipp <sip:sipp@172.16.34.16:5060>;tag=1
To: sut <sip:service@172.16.34.225:5060>;tag=2578
Call-ID: 1-668@172.16.34.16
CSeq: 1 ACK
Contact: sip:sipp@172.16.34.16:5060
Max-Forwards: 70
Subject: Performance Test
Content-Length: 0

Nov 3 08:50:56.887 On [1:0]192.168.34.225:5060 sent to 192.168.34.17:5060

ACK sip:192.168.34.17:5060;transport=UDP SIP/2.0
Via: SIP/2.0/UDP 192.168.34.225:5060;branch=z9hG4bK48k3k1301ot00ssvf1v0.1
From: sipp <sip:sipp@172.16.34.16:5060>;tag=1
To: sut <sip:service@172.16.34.225:5060>;tag=2578
Call-ID: 1-668@172.16.34.16
CSeq: 1 ACK
Contact: <sip:sipp@192.168.34.225:5060;transport=udp>
Max-Forwards: 69
Subject: Performance Test
Content-Length: 0

Nov 3 08:51:01.883 On [2:0]172.16.34.225:5060 received from
172.16.34.16:5060

BYE sip:service@172.16.34.225:5060 SIP/2.0
Via: SIP/2.0/UDP 172.16.34.16:5060;branch=z9hG4bK-1-7
From: sipp <sip:sipp@172.16.34.16:5060>;tag=1
To: sut <sip:service@172.16.34.225:5060>;tag=2578
Call-ID: 1-668@172.16.34.16
CSeq: 2 BYE
Contact: sip:sipp@172.16.34.16:5060
Max-Forwards: 70
Subject: Performance Test
Content-Length: 0

Nov 3 08:51:01.887 On [1:0]192.168.34.225:5060 sent to 192.168.34.17:5060

BYE sip:192.168.34.17:5060;transport=UDP SIP/2.0
Via: SIP/2.0/UDP 192.168.34.225:5060;branch=z9hG4bK5oq46d301gv0dus227f1.1
From: sipp <sip:sipp@172.16.34.16:5060>;tag=1
To: sut <sip:service@172.16.34.225:5060>;tag=2578
Call-ID: 1-668@172.16.34.16
CSeq: 2 BYE
Contact: <sip:sipp@192.168.34.225:5060;transport=udp>
Max-Forwards: 69
Subject: Performance Test
Content-Length: 0

Nov 3 08:51:01.889 On [1:0]192.168.34.225:5060 received from
192.168.34.17:5060

SIP/2.0 200 OK
Via: SIP/2.0/UDP 192.168.34.225:5060;branch=z9hG4bK5oq46d301gv0dus227f1.1
From: sipp <sip:sipp@172.16.34.16:5060>;tag=1
To: sut <sip:service@172.16.34.225:5060>;tag=2578
Call-ID: 1-668@172.16.34.16
CSeq: 2 BYE
Contact: <sip:192.168.34.17:5060;transport=UDP>
Content-Length: 0

Nov 3 08:51:01.892 On [2:0]172.16.34.225:5060 sent to 172.16.34.16:5060
SIP/2.0 200 OK
Via: SIP/2.0/UDP 172.16.34.16:5060;branch=z9hG4bK-1-7
From: sipp <sip:sipp@172.16.34.16:5060>;tag=1
To: sut <sip:service@172.16.34.225:5060>;tag=2578
Call-ID: 1-668@172.16.34.16
CSeq: 2 BYE
Contact: <sip:service@172.16.34.225:5060;transport=udp>
Content-Length: 0

----MBCD Evt
Nov 3 08:51:01.891 On 127.0.0.1:2945 sent to 127.0.0.1:2944

mbcdEvent=FLOW DELETE
FlowCollapsed=enabled
FlowDirection=CALLING
FlowID=65541
MediaFormat=0
MediaReleased=disabled
MediaType=audio/PCMU/
OtherFlowID=0
TOSBits=0
InputSourcev4Addr=
InputSourcePort=0
InputDestv4Addr=172.16.34.225
InputDestPort=10004
OutputSourcev4Addr=192.168.34.225
OutputDestv4Addr=192.168.34.17
OutputDestPort=6000
InputRealm=access
OutputRealm=backbone

----MBCD Evt
mbcdEvent=FLOW DELETE
FlowCollapsed=enabled
FlowDirection=CALLED
FlowID=65542
MediaFormat=0
MediaReleased=disabled
MediaType=audio/PCMU/
OtherFlowID=65541
TOSBits=0
InputSourcev4Addr=
InputSourcePort=0
InputDestv4Addr=192.168.34.225
InputDestPort=20004
OutputSourcev4Addr=172.16.34.225
OutputDestv4Addr=172.16.34.16
OutputDestPort=6000
InputRealm=backbone
OutputRealm=access

-----Session Summary-----

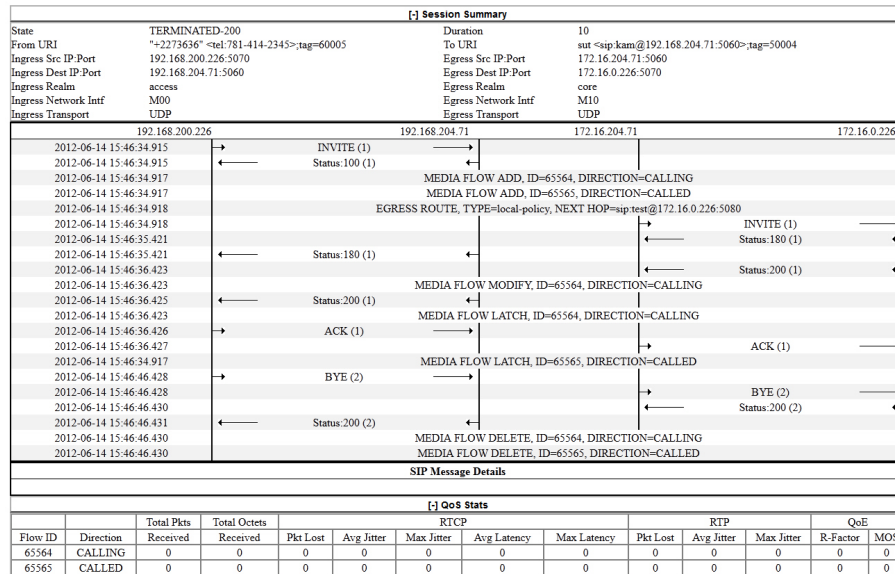
Startup Time: 2012-01-25 10:28:30.394

```

State: TERMINATED-200
Duration: 5
From URI: sipp &lt; ;sip:sipp@172.16.34.16:5060&gt;;tag=1
To URI: sut &lt; ;sip:service@172.16.34.225:5060&gt;;tag=2578
Ingress Src Address: 172.16.34.16
Igress Src Port: 5060
Igress Dest Address: 172.16.34.225
Igress Dest Port: 5060
Egress Source Address: 192.168.34.225
Egress Source Port: 5060
Egress Destination Address: 192.168.34.17
Egress Destination Port: 5060
Igress Realm: access
Egress Realm: backbone
Igress NetworkIf: access
Egress NetworkIf: backbone
    
```

Ladder Diagram Exported HTML File

The following example shows a Ladder Diagram for a session, exported to an HTML file from the GUI.



SIP HMR (Header Manipulation Rules)

SIP header manipulation can also be configured in a way that makes it possible to manipulate the headers in SIP messages both statically and dynamically. Using this feature, you can edit response headers or the Request-URI in a request, and change the status code or reason phrase in SIP responses.

Static SIP Header and Parameter Manipulation allows you to set up rules in your Oracle Enterprise Communications Broker configuration that remove and/or replace designated portions of specified SIP headers. SIP HMR allows you to set up dynamic header manipulation rules, meaning that the Oracle Enterprise Communications Broker has complete control over alterations to the header value. More specifically:

- The Oracle Enterprise Communications Broker can search header for dynamic content or patterns with the header value. It can search, for example, for all User parts of a URI that begin with 617 and end with 5555 (e.g., 617...5555).
- The Oracle Enterprise Communications Broker can manipulate any part of a patterns match with any part of a SIP header. For example, 617 123 5555 can become 617 231 5555 or 508 123 0000, or any combination of those.

To provide dynamic header manipulation, the Oracle Enterprise Communications Broker uses regular expressions to provide a high degree of flexibility for this feature. This allows you to search a specific URI when you do not know that value of the parameter, but want to use the matched parameter value as the header value. It also allows you to preserve matched sections of a pattern, and change what you want to change.

You can apply header manipulation to session agents, SIP interfaces, and realms. You do so by first setting up header manipulations rules, and then applying them in the configurations where they are needed. Within the header manipulation rules, there are sets of element rules that designate the actions that need to be performed on a given header.

Each header rule and each element rule (HMR) have a set of parameters that you configure to identify the header parts to be manipulated, and in what way the Oracle Enterprise Communications Broker is to manipulate them. These parameters are explained in detail, but the parameter that can take regular expression values is **match-value**. This is where you set groupings that you want to store, match against, and manipulate.

Generally, you set a header rule that will store what you want to match, and then you create subsequent rules that operate on this stored value. Because header rules and element rules are applied sequentially, it is key to note that a given rule performs its operations on the results of all the rules that you have entered before it. For example, if you want to delete a portion of a SIP header, you would create Rule 1 that stores the value for the purpose of matching, and then create Rule 2 that would delete the portion of the header you want removed. This prevents removing data that might be used in the other header rules.

Given that you are using regular expression in this type of configuration, this tightly sequential application of rules means that you must be aware of the results to be

yielded from the application of the regular expressions you enter. When you set a regular expression match value for the first rule that you enter, the Oracle Enterprise Communications Broker takes the results of that match, and then a second rule might exist that tells the Oracle Enterprise Communications Broker to use a new value if it the second rule's match value finds a hit (and only 10 matches, 0-9, are permitted) for the results (yield) from applying the first rule.

Consider the example of the following regular expression entry made for a **match-value** parameter: 'Trunk(.+)', which might be set as that match value in the first rule you configure. Given a SIP element rule called uri-param and the param-name tgid, it can yield two values:

- Grouping 0—The entire matching string (Trunk1)
- Grouping 1—The grouping (1)

In turn, these groupings can be referenced in an element rule by using this syntax:

```
<header rule name >.$<element rule name.>.<value>
```

Additional syntax options that can be used with this feature are:

- `$headerName['index']`
- `$headerName['index'][$index]`
- `$headerName['index'].$elementName`
- `$headerName['index'].$elementName[$index]`

Guidelines for Header and Element Rules

Header rules and element rules share these guidelines:

- References to groupings that do not exist result in an empty string.
- References to element rule names alone result in a Boolean condition of whether the expression matched or not.
- A maximum of ten matches are allowed for a regular expression. Match 0 (grouping 0) is always the match of the entire matching string; subsequent numbers are the results for other groups that match.

Splitting and Joining Headers

To simplify header manipulation processes, the Oracle Enterprise Communications Broker provides a means of combining or breaking apart header strings that actually consist of multiple headers. An example application would be to separate headers that another SIP device joined together into a single string. This would allow header manipulation to work on each distinct header, after which the system could re-combine the headers, making the forwarded output consistent with the initial message.

Some SIP devices combine multiple headers into a single header, with each distinct header separated by a comma. This is not precluded by RFC 3261. The user can configure a header manipulation to separate these headers prior to performing the manipulation using the **split-headers** command. The user can also configure the system to join headers together after a manipulation is complete using the **join-**

headers command. Split and join functions do not have to co-exist within a single header manipulation.

Precedence

The Oracle Enterprise Communications Broker applies SIP header rules in the order you have entered them. This guards against the Oracle Enterprise Communications Broker removing data that might be used in the other header rules.

This ordering also provides you with ways to use manipulations strategically. For example, you might want to use two rules if you want to store the values of a regular expression. The first rule would store the value of a matched regular expression, and the second could delete the matched value.

In addition to taking note of the order in which header rules are configured, you now must also configure a given header rule prior to referencing it. For example, you must create Rule1 with the action store for the Contact header BEFORE you can create Rule2 which uses the stored value from the Contact header.

Duplicate Header Names

If more than one header exists for a configured header-name, the OECB stores each value in an array whose index starts at 0. To reference those values, use the syntax `$(header-name)[<index>]`.

Add a trailing [`<index>`] value after the header-name parameter to represent the specific instance of the header on which to operate. Additional stored header values are indexed in the order in which they appear within the SIP message, and there is no limit to the index. The OECB takes no action if the header does not exist.

In addition to numerical values, possible index values are:

- `~` The OECB references the first matching header.
- `*` The OECB references all headers.
- `^` The OECB references the last stored header in the header rule.

Note that the header instance functionality has no impact on HMR's add action, and you cannot use this feature to insert headers into a specific location. Headers are added to the end of the list, except that Via headers are added to the top.

Performing HMR on a Specific Header

HMR has been enhanced so that you can now operate on a specific instance of a given header. The syntax you use to accomplish this is similar to that you used to refer to a specific header rule stored value instance.

Using the header-name parameter, you can now add a trailing [`<index>`] value after the header name. This [`<index>`] is a numerical value representing the specific instance of the header on which to operate. However, the Oracle Enterprise Communications Broker takes no action if the header does not exist. You can also use the caret (^) to reference the last header of that type (if there are multiple instances)

The count for referencing is zero-based, meaning that the first instance of the header counts as 0.

Note that the header instance functionality has no impact on HMR's add action, and you cannot use this feature to insert headers into a specific location. Headers are added to the end of the list, except that Via headers are added to the top.

Multiple SIP HMR Sets

In general you use SIP HMR by configuring rules and then applying those rules to session agents, realms, or SIP interfaces in the inbound or outbound direction. In addition, the Oracle Enterprise Communications Broker has a set method for how certain manipulation rules take precedence over others. For instance, inbound SIP manipulation rules defined in a session agent take precedence over any configured for a realm, and the rules for a realm take precedence over SIP interface manipulation rules.

The multiple SIP HMR feature gives you the ability to:

- Apply multiple inbound and outbound manipulations rules to a SIP message
- Provision the order in which the Oracle Enterprise Communications Broker applies manipulation rules

The **action** parameter in the header rules configuration now takes the value `sip-manip`. When you set the parameter to `sip-manip`, you then configure the **new-value** parameter with the name of a SIP manipulation rule that you want to invoke. The values for the **match-value**, **comparison-type**, and **methods** parameters for invoked rule are all supported. This means that the manipulation defined by the rules identified in the **new-value** parameter are carried out when the values for the **match-value**, **comparison-type**, and **methods** parameters are true.

The relationship between manipulation rules and manipulation rule sets is created once you load your configuration, meaning that the order in which you enter them does not matter. It also means that the Oracle Enterprise Communications Broker cannot dynamically perform validation as you enter rules, so you should use the ACLI **verify-config** command to confirm your manipulation rules contain neither invalid nor circular references. Invalid references are those that point to SIP manipulation rules that do not exist, and circular references are those that create endless loops of manipulation rules being carried out over and over. If you load a configuration exhibiting either of these issues, the Oracle Enterprise Communications Broker forces the action value for the rule to **none** and the rule will not be used.

MIME Support

Using the SIP HMR feature set, you can manipulate MIME types in SIP message bodies. While you can manipulate the body of SIP messages or a specific content type using other iterations of SIP HMR, this version gives you the power to change the MIME attachment of a specific type within the body by using regular expressions. To achieve this, you use the **find-replace-all** action type, which enables the search for a particular string and the replacement of all matches for that type. Although you use **find-replace-all** to manipulate MIME attachments, it can also be used to achieve other goals in SIP HMR.

Note that using **find-replace-all** might consume more system resources than other HMR types. Therefore this powerful action type should only be used when another type cannot perform the type of manipulation you require.

Manipulating MIME Attachments

Set the action type to `find-replace-all` to modify MIME attachments.

To manipulate a particular portion of the MIME attachment, for example when removing a certain attribute within the Content-Type of `application/sdp`, the OECB needs to search the content multiple times because:

- SDP can have more than one media line
- The SIP message body can contain more than one `application/sdp`.

When the action type is `find-replace-all`, the OECB treats the match-value as a regular expression and binds the comparison-type to `pattern-rule`, even if comparison-type is set to some other value. This type of action is both a comparison and action: for each regular expression match within the supplied string, the OECB substitutes the new value for that match.

Use subgroups to replace portions of the regular expression rather than the entire matched expression. The subgroup replacement syntax is formed by adding the string `[[:n:]]` to the end of the regular expression—where `n` is a number between 0 and 9. For example, setting the following parameters

```

action          find-replace-all
match-value     sip:(user)@host[[ :1: ]]
new-value       bob

```

creates a new rule to replace only the user portion of the URI that searches for the regular expression and replaces all instances of the user subgroup with the value `bob`.

Setting the following parameters

```

action          find-replace-all
match-value     0
new-value       1

```

creates a new rule to recursively replace all the 0 digits in a telephone number with 1. With this rule the user portion of a URI—or for any other string—with a value `1-781-308-4400` would be replaced as `1-781-318-4411`.

If you leave the **new-value** parameter blank for **find-replace-all**, the OECB replaces the matched sub-group with an empty string—an equivalent of deleting the sub-group match. You can also replace empty sub-groups, which is like inserting a value within the second sub-group match. For example, `user()@host.com[[:1:]]` with a configured new-value `_bob` yields `user_bob@host.com`.

Setting **find-replace-all** disables the following **parameter-type** values: **uri-param-name**, **uri-header-name**, and **header-param-name**. These values are unusable because the OECB only uses case-sensitive matches for the match-value to find the parameter name within the URI. Since it can only be found by exact match, the OECB does not support finding and replacing that parameter.

Escaped Characters

SIP HMR's support for escaped characters allows for searches for values you would be unable to enter yourself. Because they are necessary to MIME manipulation, support for escaped characters now includes:

- \f
- \n
- \r
- \t
- \v

New Reserved Word

To allow you to search for carriage returns and new lines, the SIP HMR MIME feature also adds support for the reserved word `$CRLF`. Because you can search for these values and replace them, you also must be able to add them back in when necessary. Configuring `$CRLF` in the **new-value** parameter always resolves to `/r/n`, which you normally cannot otherwise enter through the ACLI.

About the MIME Value Type

To modify the MIME attachment, SIP HMR supports a **mime** value for the **type** parameter in the element rules configuration. Like the **status-code** and **reason-phrase** values, you can only use the **mime** type value against the Content-Type header.

When you set the element rule type to **mime**, you must also configure the **parameter-name** with a value. This step is a requirement because it sets the content-type the Oracle Enterprise Communications Broker (OECB) manipulates in a specific part of the MIME attachment. You cannot leave this parameter blank because the OECB does not let you save the configuration if you do. When you use the **store** action on a multi-part MIME attachment that has different attachment types, the OECB stores the final instance of the content-type because it does not support storing multiple instances of element rule stored values.

If you do not know the specific content-type where the OECB will find the **match-value**, you can use a wildcard in the **parameter-name** by setting with the asterisk (*) as a value. You cannot set partial content-types (for example, `application/*`). So configured, the OECB loops through the MIME attachment's content types.

You can set the additional **action** types listed in the following table with the described result:

Action Type	Description
delete-element	Removes the matched mime-type from the body. If this is the last mime-type within in message body, the Oracle Enterprise Communications Broker removes the Content-Type header.
delete-header	Removes all body content and removes the Content-Type header.
replace	Performs a complete replacement of the matched mime-type with the new-value you configure.

Action Type	Description
find-replace-all	Searches the specifies mime-type's contents and replaces all matching regular expressions with the new-value you configure
store	Stores the final instance of the content-type (if there are multi-part MIME attachments of various attachment types)
add	Not supported

MIME manipulation does not support manipulating headers in the individual MIME attachments. For example, the OECB cannot modify the Content-Type given a portion of a message body such as the following one:

```
--boundary-1
Content-Type: application/sdp
v=0
o=use1 53655765 2353687637 IN IP4 192.168.1.60
s=-
c=IN IP4 192.168.1.60
t=0 0
m=audio 10000 RTP/AVP 8
a=rtpmap:8 PCMA/8000/1
a=sendrecv
a=ptime:20
a=maxptime:200
```

Back Reference Syntax

You can use back reference syntax in the **new-value** parameter for header and element rules configurations. Denoted by the use of \$1, \$2, \$3, etc. (where the number refers to the regular expression's stored value), you can reference the header and header rule's stored value without having to use the header rule's name. It instead refers to the stored value of this rule.

For example, when these settings are in place:

- header-rule=changeHeader
- action=manipulate
- match-value=(.+)([^\;])

you can set the **new-value** as sip:\$2 instead of **sip:\$changeHeader.\$2**.

You can use the back reference syntax for:

- Header rule **actions manipulate** and **find-replace-all**
- Element rule **actions replace** and **find-replace-all**

Using back reference syntax simplifies your configuration steps because you do not need to create a store rule and then manipulate rule; the manipulate rule itself performs the store action if the **comparison-type** is set to **pattern-rule**.

Notes on the Regular Expression Library

In the regular expression library, the dot (.) character does not match new lines or carriage returns. Conversely, the not-dot does match new lines and carriage returns. This provides a safety mechanism preventing egregious backtracking of the entire SIP message body when there are no matches. Thus, the Oracle Enterprise Communications Broker (OECB) reduces backtracking to a single line within the body. The OECB also supports:

Syntax	Description
\s	Whitespace
\S	Non-whitespace
\d	Digits
\D	Non-digits
\R	Any \r, \n, \r\n
\w	Word
\W	Non-word
\A	Beginning of buffer
\Z	End of buffer
\	Any character including newline, in the event that the dot (.) is not

In addition, you can also use:

- Escaped character shortcuts (\w\W\s\d\|\R) operating inside brackets [...]

SIP Message-Body Separator Normalization

The Oracle Enterprise Communications Broker supports SIP with Multipurpose Internet Mail Extension (MIME) attachments — up to a maximum payload size of 64KB — and has the ability to allow more than the required two CRLFs between the SIP message headers and the multipart body's first boundary. The first two CRLFs that appear in all SIP messages signify the end of the SIP header and the separation of the header and body of the message, respectively. Sometimes additional extraneous CRLFs can appear within the preamble before any text.

The Oracle Enterprise Communications Broker works by forwarding received SIP messages regardless of whether they contain two or more CRLFs. Although three or more CRLFs are legal, some SIP devices do not accept more than two.

The solution to ensuring all SIP devices accept messages sent from the Oracle Enterprise Communications Broker is to strip all CRLFs located at the beginning of the preamble before the appearance of any text, ensuring that there are no more than two CRLFs between the end of the last header and the beginning of the body within a SIP message. You enable this feature by adding the new `stripPreambleCrlf` option to the global SIP configuration.

To enable the stripping of CRLFs in the preamble, add the **+stripPreambleCrlf** option to SIP Options.

SIP Header Pre-Processing HMR

By default, the Oracle Enterprise Communications Broker performs in-bound SIP manipulations after it carries out header validation. Adding the **inmanip-before-validate** option in the global SIP configuration allows the Oracle Enterprise Communications Broker to perform HMR on received requests prior to header validation. Because there are occasional issues with other SIP implementations—causing invalid headers to be used in messages they send to the Oracle Enterprise Communications Broker—it can be beneficial to use HMR to remove or repair these faulty headers before the request bearing them are rejected.

When configured to do so, the Oracle Enterprise Communications Broker performs pre-validation header manipulation immediately after it executes the top via check. Inbound SIP manipulations are performed in order of increasing precedence: SIP interface, realm, and session agent.

The fact that the top via check happens right before the Oracle Enterprise Communications Broker carries out pre-validation header manipulations means that you cannot use this capability to repairs the first via header if it is indeed invalid. If pre-validation header manipulation were to take place at another time during processing, it would not be possible to use it for SIP session agents. The system learns of matching session agents after top via checking completes.

For logistical reasons, this capability does not extend to SIP responses. Inbound manipulation for responses cannot be performed any sooner that it does by default, a time already preceding any header validation.

To enable SIP header pre-processing, add the **+inmanip-before-validate** option to SIP Options.

Best Practices

The following list describes practices that Oracle recommends you follow for successful implementation of Header Manipulation Rules.

- Define all storage rules first.
This recommendation is made because each subsequent header rule processes against the same SIP message, so each additional header rules works off of the results from the application of the rule that precedes it.

In general, you want to store values from the original SIP header rather than from the iteratively changed versions.
- Implement rules at the element rule rather than the header rule level.
Header rules should only be a container for element rules.
- When you are creating rules to edit a header, add additional element rules to modify a single header rather than try to create multiple header rules each with one element rule. That is, create multiple element rules within a header rule rather than creating multiple header rules.
- Do not use header or element rule names that are all capital letters (for example, \$IP_ADDRESS). Capital letters refer to predefined rules that are used as macros, and they might conflict with a name that uses capital letters.

About Regular Expressions

Two of the most fundamental ideas you need to know to work with regular expressions and with Header Manipulation Rules include:

- Regular expressions are a way of creating strings to match other string values.
- You can use groupings to create stored values on which you can then operate.

To learn more about regex, you can visit the following Web site, which has information and tutorials that can help to get you started:<http://www.regular-expressions.info/>.

Many of the characters you can type on your keyboard are literal, ordinary characters, which present their actual value in the pattern. Some characters have special meaning, and they instruct the regex function (or engine which interprets the expressions) to treat the characters in designated ways. The following table outlines these “special characters” or metacharacters.

Character	Name	Description
.	dot	Matches any one character, including a space; it will match one character, but there must be one character to match. Literally a . (dot) when bracketed ([.]), or placed next to a \ (backslash).
*	star/asterisk	Matches one or more preceding character (0, 1, or any number), bracketed carrier class, or group in parentheses. Used for quantification. Typically used with a . (dot) in the format .* to indicate that a match for any character, 0 or more times. Literally an * (asterisk) when bracketed ([*]).
+	plus	Matches one or more of the preceding character, bracketed carrier class, or group in parentheses. Used for quantification. Literally a + (plus sign) when bracketed ([+]).
	bar/vertical bar/pipe	Matches anything to the left or to the right; the bar separates the alternatives. Both sides are not always tried; if the left does not match, only then is the right attempted. Used for alternation.
{	left brace	Begins an interval range, ended with } (right brace) to match; identifies how many times the previous singles character or group in parentheses must repeat. Interval ranges are entered as minimum and maximums ({minimum,maximum}) where the character/group must appear a minimum of times up to the maximum. You can also use these character to set magnitude, or exactly the number of times a character must appear; you can set this, for example, as the minimum value without the maximum ({minimum,}).

Character	Name	Description
?	question mark	Signifies that the preceding character or group in parentheses is optional; the character or group can appear not at all or one time.
^	caret	Acts as an anchor to represent the beginning of a string.
\$	dollar sign	Acts as an anchor to represent the end of a string.
[left bracket	Acts as the start of a bracketed character class, ended with the] (right bracket). A character class is a list of character options; one and only one of the characters in the bracketed class must appear for a match. A - (dash) in between two character enclosed by brackets designates a range; for example [a-z] is the character range of the lower case twenty-six letters of the alphabet. Note that the] (right bracket) ends a bracketed character class unless it sits directly next to the [(left bracket) or the ^ (caret); in those two cases, it is the literal character.
(left parenthesis	Creates a grouping when used with the) (right parenthesis). Groupings have two functions: They separate pattern strings so that a whole string can have special characters within it as if it were a single character. They allow the designated pattern to be stored and referenced later (so that other operations can be performed on it).

Expression Building Using Parentheses

You can now use parentheses () when you use HMR to support order of operations and to simplify header manipulation rules that might otherwise prove complex. This means that expressions such as (sip + urp) - (u + rp) can now be evaluated to sip. Previously, the same expression would have evaluated to sipurprp. In addition, you previously would have been required to create several different manipulation rules to perform the same expression.

Configuration Examples

This section shows you several configuration examples for HMR. This section shows the configuration for the various rules that the Oracle Enterprise Communications Broker applied, and sample results of the manipulation. These examples present configurations as an entire list of fields and settings for each ruleset, nested header rules and nested element rules. If a field does not have any operation within the set, the field is shown with the setting at the default or blank.

Example 1 Removing Headers

For this manipulation rule, the Oracle Enterprise Communications Broker removes the Custom header if it matches the pattern rule. It stores the defined pattern rule for the goodBye header. Finally, it removes the goodBye header if the pattern rule from above is a match.

This is a sample of the configuration:

```

sip-manipulation
  name removeHeader
  header-rule
    name removeCustom
    header-name Custom
    action delete
    comparison-type boolean
    match-value ^This is my.*
    msg-type request
    new-value
    methods INVITE
  header-rule
    name goodByeHeader
    header-name Goodbye
    action store
  comparison-type boolean
  match-value ^Remove (.+)
  msg-type request
  new-value
  methods INVITE
  header-rule
  name goodBye
  action delete
  comparison-type pattern-rule
  match-value $goodByeHeader
  msg-type request
  new-value
  methods INVITE

```

This is a sample of the result:

```

Request-Line: INVITE sip:service@192.168.200.60:5060;tgid=123 SIP/2.0
Message Header
  Via: SIP/2.0/UDP
192.168.200.61:5060;branch=z9hG4bK0g639r10fgc0aakk26s1.1
  From: sipp <sip:sipp@192.168.1.60:5060>;tag=SDclrm601-1
  To: sut <sip:service@192.168.1.61:5060>
  Call-ID: SDclrm601-d01673bcacfcc112c053d95971330335-06a3gu0
  CSeq: 1 INVITE
  Contact: <sip:sipp@192.168.200.61:5060;transport=udp>
  Display: sipp <sip:user@192.168.1.60:5060;up=abc>;hp=123
  Params: sipp <sip:sipp1@192.168.1.60:5060>
  Params: sipp <sip:sipp2@192.168.1.60:5060>
  Edit: disp <sip:user@192.168.1.60:5060>

```

```

Max-Forwards: 69
Subject: Performance Test
Content-Type: application/sdp
Content-Length: 140

```

Example 2 Manipulating the Request URI

For this manipulation rules, the Oracle Enterprise Communications Broker stores the URI parameter tgid in the Request URI. Then if the pattern rule matches, it adds a new header (x-customer-profile) with the a new header value tgid to the URI parameter in the request URI.

This is a sample of the configuration:

```

sip-manipulation
  name                               CustomerTgid
  header-rule
    name                               ruriRegex
    header-name                         request-uri
    action                               store
    comparison-type                     pattern-rule
    match-value
    msg-type                             request
  new-value
    methods                             INVITE
  element-rule
    name                               tgidParam
    parameter-name                      tgid
    type                                 uri-param
    action                               store
    match-val-type                      any
    comparison-type                     pattern-rule
    match-value
    new-value
  header-rule
    name                               addCustomer
    header-name                         X-Customer-Profile
    action                               add
    comparison-type                     pattern-rule
    match-value                         $ruriRegex.$tgidParam
    msg-type                             request
    new-value                           $ruriRegex.$tgidParam.$0
    methods                             INVITE
  header-rule
    name                               delTgid
    header-name                         request-uri
    action                               manipulate
    comparison-type                     pattern-rule
    match-value                         $ruriRegex.$tgidParam
    msg-type                             request
    new-value
    methods                             INVITE
  element-rule
    name                               tgidParam

```

	parameter-name	tgid
	type	uri-param
	action	delete-element
	match-val-type	any
	comparison-type	case-sensitive
	match-	
value	\$ruriRegex.\$tgidParam.\$0	
	new-value	

This is a sample of the result:

```
Request-Line: INVITE sip:service@192.168.200.60:5060 SIP/2.0
Message Header
Via: SIP/2.0/UDP 192.168.200.61:5060;branch=z9hG4bK0g6plv3088h03acgh6c1.1
    From: sipp <sip:sipp@192.168.1.60:5060>;tag=SDclrg601-1
    To: sut <sip:service@192.168.1.61:5060>
    Call-ID: SDclrg601-f125d8b0ec7985c378b04cab9f91cc09-06a3gu0
    CSeq: 1 INVITE
    Contact: <sip:sipp@192.168.200.61:5060;transport=udp>
    Goodbye: Remove Me
    Custom: This is my custom header
    Display: sipp <sip:user@192.168.1.60:5060;up=abc;hp=123
Params: sipp <sip:sipp1@192.168.1.60:5060>
Params: sipp <sip:sipp2@192.168.1.60:5060>
Edit: disp <sip:user@192.168.1.60:5060>
Max-Forwards: 69
Subject: Performance Test
Content-Type: application/sdp
Content-Length: 140
X-Customer-Profile: 123
```

Example 3 Manipulating a Header

For this manipulation rule, the Oracle Enterprise Communications Broker stores the pattern matches for the Custom header, and replaces the value of the Custom header with a combination of the stored matches and new content.

This is a sample of the configuration:

```
sip-manipulation
    name                               modCustomHdr
    header-rule
        name                             customSearch
        header-name                       Custom
        action                             store
        comparison-type                   pattern-rule
        match-value                       (This is my )(.) ( header)
        msg-type                           request
        new-value                           INVITE
    header-rule
        name                               customMod
        header-name                       Custom
        action                             manipulate
```

```

        comparison-type      pattern-rule
        match-value          $customSearch
        msg-type             request
        new-value
methods                    INVITE
        element-rule
        name                 hdrVal
        parameter-name       hdrVal
        type                 header-value
        action               replace
        match-val-type       any
        comparison-type      case-sensitive
        match-value
new-value                  $customSearch.$1+edited+$customSearch.$3

```

This is a sample of the result:

```

Request-Line: INVITE sip:service@192.168.200.60:5060;tgid=123 SIP/2.0
  Message Header
    Via: SIP/2.0/UDP
192.168.200.61:5060;branch=z9hG4bK20q2s820boghbacgs6o0.1
  From: sipp <sip:sipp@192.168.1.60:5060>;tag=SDelra601-1
  To: sut <sip:service@192.168.1.61:5060>
  Call-ID: SDelra601-4bb668e7ec9eeb92c783c78fd5b26586-06a3gu0
  CSeq: 1 INVITE
  Contact: <sip:sipp@192.168.200.61:5060;transport=udp>
  Goodbye: Remove Me
  Custom: This is my edited header
  Display: sipp <sip:user@192.168.1.60:5060;up=abc;hp=123
  Params: sipp <sip:sipp1@192.168.1.60:5060>
  Params: sipp <sip:sipp2@192.168.1.60:5060>
  Edit: disp <sip:user@192.168.1.60:5060>
  Max-Forwards: 69
  Subject: Performance Test
  Content-Type: application/sdp
  Content-Length: 140

```

Example 4 Storing and Using URI Parameters

For this manipulation rule, the Oracle Enterprise Communications Broker stores the value of the URI parameter tag from the From header. It also creates a new header FromTag with the header value from the stored information resulting from the first rule.

This is a sample of the configuration:

```

sip-manipulation
  name                 storeElemParam
  header-rule
  name                 Frohmr
  header-name          From
  action               store
  comparison-type      case-sensitive
  match-value
  msg-type             request

```



```

new-value
methods
element-rule
    name
    parameter-name
    type
    action
    match-val-type
    comparison-type
    match-value
    new-value
header-rule
    name
    header-name
    action
    comparison-type
    match-value
    msg-type
    new-value
    methods
    elementRule
    tag
    uri-param
    store
    any
    case-sensitive
    newHeader
    FromTag
    add
    pattern-rule
    $FromHR.$elementRule
    any
    $FromHR.$elementRule.$0

```

This is a sample of the result:

```

Request-Line: INVITE sip:service@192.168.200.60:5060;tgid=123 SIP/2.0
Message Header
Via: SIP/2.0/UDP
192.168.200.61:5060;branch=z9hG4bK4oda2e2050ih7acgh6c1.1
From: sipp <sip:sipp@192.168.1.60:5060>;tag=SDflre601-1
To: sut <sip:service@192.168.1.61:5060>
Call-ID: SDflre601-f85059e74e1b443499587dd2dee504c2-06a3gu0
CSeq: 1 INVITE
Contact: <sip:sipp@192.168.200.61:5060;transport=udp>
Goodbye: Remove Me
Custom: This is my custom header
Display: sipp <sip:user@192.168.1.60:5060;up=abc;hp=123
Params: sipp <sip:sipp1@192.168.1.60:5060>
Params: sipp <sip:sipp2@192.168.1.60:5060>
Edit: disp <sip:user@192.168.1.60:5060>
Max-Forwards: 69
Subject: Performance Test
Content-Type: application/sdp
Content-Length: 140
FromTag: 1

```

Example 5 Manipulating Display Names

For this manipulation rule, the Oracle Enterprise Communications Broker sores the display name from the Display header. It replaces the two middle characters of the original display name with a new string. Then is also replaces the From header's display name with "abc 123" if it matches sipp.

This is a sample of the configuration:

```

sip-manipulation
  name                               modDisplayParam
  header-rule
    name                               storeDisplay
    header-name                         Display
    action                               store
    comparison-type                     case-sensitive
    match-value
    msg-type                             request
    new-value
    methods                              INVITE
    element-rule
      name                               displayName
      parameter-name                     display
      type                                uri-display
      action                              store
      match-val-type                     any
  comparison-type                       pattern-rule
    match-value                         (s)(ip)(p )
    new-value
  header-rule
    name                               modDisplay
    header-name                         Display
    action                               manipulate
    comparison-type                     case-sensitive
    match-value
    msg-type                             request
    new-value
    methods                              INVITE
    element-rule
      name                               modRule
      parameter-name                     display
      type                                uri-display
      action                              replace
      match-val-type                     any
      comparison-type                     pattern-rule
      match-
  value                                  $storeDisplay.$displayName
  new-
  value                                  $storeDisplay.$displayName.$1+lur
+$storeDisplay.$displayName.$3
  header-rule
    name                               modFrom
    header-name                         From
    action                               manipulate
    comparison-type                     pattern-rule
    match-value
    msg-type                             request
    new-value
    methods                              INVITE
    element-rule
      name                               fromDisplay
      parameter-name

```

```

type uri-display
action replace
match-val-type any
comparison-type pattern-rule
match-value sipp
new-value "\"abc 123\" "

```

This is a sample of the result:

```

Request-Line: INVITE sip:service@192.168.200.60:5060;tgid=123 SIP/2.0
Message Header
Via: SIP/2.0/UDP
192.168.200.61:5060;branch=z9hG4bK681kot109gp04acgs6o0.1
From: "abc 123" <sip:sipp@192.168.1.60:5060>;tag=SD79ra601-1
To: sut <sip:service@192.168.1.61:5060>
Call-ID: SD79ra601-a487f1259e2370d3dbb558c742d3f8c4-06a3gu0
CSeq: 1 INVITE
Contact: <sip:sipp@192.168.200.61:5060;transport=udp>
Goodbye: Remove Me
Custom: This is my custom header
Display: slurp <sip:user@192.168.1.60:5060;up=abc>;hp=123
Params: sipp <sip:sipp1@192.168.1.60:5060>
Params: sipp <sip:sipp2@192.168.1.60:5060>
Edit: disp <sip:user@192.168.1.60:5060>
Max-Forwards: 69
Subject: Performance Test
Content-Type: application/sdp
Content-Length: 140

```

Example 6 Manipulating Element Parameters

For this more complex manipulation rule, the Oracle Enterprise Communications Broker:

- From the Display header, stores the display name, user name, URI parameter up, and header parameter hp
- Adds the header parameter display to the Params header, with the stored value of the display name from the first step
- Add the URI parameter user to the Params header, with the stored value of the display name from the first step
- If the URI parameter match succeeds in the first step, replaces the URI parameter up with the Display header with the value def
- If the header parameter match succeeds in the first step, deletes the header parameter hp from the Display header

This is a sample of the configuration:

```

sip-manipulation
  name elemParams
  header-rule
    name StoreDisplay
    header-name Display

```

```

        action
        comparison-type
        match-value
        msg-type
        new-value
        methods
        element-rule
            name
            parameter-name
            type
            action
            match-val-type
            comparison-type
            match-value
            new-value
        element-rule
            name
            parameter-name
            type
            action
            match-val-type
            comparison-type
            match-value
            new-value
        element-rule
            name
            parameter-name
            type
            action
            match-val-type
            comparison-type
            match-value
            new-value
        header-rule
            name
            header-name
            action
            comparison-type
            match-value
            msg-type
            new-value
            methods
            element-rule
                name
                parameter-name
                type
    
```

store
case-sensitive
request
INVITE
displayName
uri-display
store
any
pattern-rule
userName
user
uri-user
store
any
pattern-rule
uriParam
up
uri-param
store
any
pattern-rule
headerParam
hp
header-param
store
any
pattern-rule
EditParams
Params
manipulate
case-sensitive
request
INVITE
addHeaderParam
display
header-param

```

        action          add
match-val-type        any
                      comparison-type    case-sensitive
                      match-value
new-
value                 $StoreDisplay.$displayName.$0
                      element-rule
                      name                addUriParam
                      parameter-name     user
                      type                uri-param
                      action              add
                      match-val-type     any
                      comparison-type     case-sensitive
                      match-value
new-value
$StoreDisplay.$userName.$0
header-rule
name                  EditDisplay
header-name           Display
action                manipulate
comparison-type       case-sensitive
match-value
msg-type              request
new-value
methods               INVITE
                      element-rule
                      name                replaceUriParam
                      parameter-name     up
                      type                uri-param
                      action              replace
                      match-val-type     any
                      comparison-type     pattern-rule
                      match-
value                 $StoreDisplay.$uriParam
                      new-value           def
                      element-rule
                      name                delHeaderParam
                      parameter-name     hp
                      type                header-param
                      action              delete-element
                      match-val-type     any
                      comparison-type     pattern-rule
                      match-value        $StoreDisplay.$headerParam
                      new-value

```

This is a sample of the result:

```

Request-Line: INVITE sip:service@192.168.200.60:5060;tgid=123 SIP/2.0
Message Header
Via: SIP/2.0/UDP
192.168.200.61:5060;branch=z9hG4bK7okvei0028jgdacgh6c1.1
From: sipp <sip:sipp@192.168.1.60:5060>;tag=SD89rm601-1
To: sut <sip:service@192.168.1.61:5060>
Call-ID: SD89rm601-b5b746cef19d0154cb1f342cb04ec3cb-06a3gu0

```

```

CSeq: 1 INVITE
Contact: <sip:sipp@192.168.200.61:5060;transport=udp>
Goodbye: Remove Me
Custom: This is my custom header
Display: sipp <sip:user@192.168.1.60:5060;up=def>
Params: sipp <sip:sipp1@192.168.1.60:5060;user=user>;display=sipp
Params: sipp <sip:sipp2@192.168.1.60:5060;user=user>;display=sipp
Edit: disp <sip:user@192.168.1.60:5060>
Max-Forwards: 69
Subject: Performance Test
Content-Type: application/sdp
Content-Length: 140

```

Example 7 Accessing Data from Multiple Headers of the Same Type

For this manipulation rule, the Oracle Enterprise Communications Broker stores the user name from the Params header. It then adds the URI parameter c1 with the value stored from the first Params header. Finally, it adds the URI parameter c2 with the value stored from the second Params header.

This is a sample of the configuration:

```

sip-manipulation
  name
  header-rule
    name
    header-name
    action
    comparison-type
    match-value
    msg-type
    new-value
    methods
    element-rule
      name
      parameter-name
      type
      action
      match-val-type
      comparison-type
      match-value
      new-value
  header-rule
    name
    header-name
    action
    comparison-type
    match-value
    msg-type
    new-value
  methods
  element-rule
    name
    parameter-name

```

```

                                type                uri-param
                                action              add
                                match-val-type     any
                                comparison-type    case-sensitive
                                match-value
                                new-
value                            $storeParams[0].$storeUserName.$0
                                element-rule
                                name                addParam2
                                parameter-name     c2
                                type                uri-param
                                action              add
                                match-val-type     any
                                comparison-type    case-sensitive
                                match-value
                                new-
value                            $storeParams[1].$storeUserName.$0

```

This is a sample of the result:

```

Request-Line: INVITE sip:service@192.168.200.60:5060;tgid=123 SIP/2.0
  Message Header
    Via: SIP/2.0/UDP
192.168.200.61:5060;branch=z9hG4bK9g855p30cos08acgs6o0.1
  From: sipp <sip:sipp@192.168.1.60:5060>;tag=SD99ri601-1
  To: sut <sip:service@192.168.1.61:5060>
  Call-ID: SD99ri601-6f5691f6461356f607b0737e4039caec-06a3gu0
  CSeq: 1 INVITE
  Contact: <sip:sipp@192.168.200.61:5060;transport=udp>
  Goodbye: Remove Me
  Custom: This is my custom header
  Display: sipp <sip:user@192.168.1.60:5060;up=abc;hp=123
  Params: sipp <sip:sipp1@192.168.1.60:5060>
  Params: sipp <sip:sipp2@192.168.1.60:5060>
  Edit: disp <sip:user@192.168.1.60:5060;c1=sipp1;c2=sipp2>
  Max-Forwards: 69
  Subject: Performance Test
  Content-Type: application/sdp
  Content-Length: 140

```

Example 8 Using Header Rule Special Characters

For this manipulation rule, the Oracle Enterprise Communications Broker:

- Stores the header value of the Params header with the given pattern rule, and stores both the user name of the Params header and the URI parameter abc
- Adds the URI parameter lpu with the value stored from the previous Params header
- If any of the Params headers match the pattern rule defined in the first step, adds the URI parameter apu with the value aup
- If all of the Params headers match the pattern rule defined in the first step, adds the URI parameter apu with the value apu

- If the first Params headers does not match the pattern rule for storing the URI parameter defined in the first step, adds the URI parameter not with the value 123
- If the first Params headers matches the pattern rule for storing the URI parameter defined in the first step, adds the URI parameter yes with the value 456

This is a sample of the configuration:

```
sip-manipulation
  name
  header-rule
    name
    header-name
    action
    comparison-type
    match-value
    msg-type
    new-value
    methods
    element-rule
      name
      parameter-name
      type
      action
      match-val-type
      comparison-type
      match-value
      new-value
  element-rule
    name
    parameter-name
    type
    action
    match-val-type
    comparison-type
    match-value
    new-value
  header-rule
    name
    header-name
    action
    comparison-type
    match-value
    msg-type
    new-value
    methods
    element-rule
      name
      parameter-name
      type
      action
      match-val-type
      comparison-type
      match-value
      new-value $searchParams[^].$userName.$0
  element-rule
    name
    parameter-name
    type
    action
    match-val-type
    comparison-type
    match-value
    new-value
```



```

                                name                anyParamUser
                                parameter-name       apu
                                type                 uri-param
                                action               add
                                match-val-type       any
                                comparison-type      pattern-rule
                                match-value         $searchParams[-]
                                new-value           aup
    element-rule
                                name                allParamUser
                                parameter-name       apu
                                type                 header-param
                                action               add
                                match-val-type       any
                                comparison-type      pattern-rule
                                match-value         $searchParams[*]
                                new-value           apu
    element-rule
                                name                notParamYes
                                parameter-name       not
                                type                 uri-param
                                action               add
                                match-val-type       any
                                comparison-type      pattern-rule
                                match-
value                               !$searchParams.$emptyUriParam
                                new-value           123
    element-rule
                                name                notParamNo
                                parameter-name       yes
                                type                 uri-param
                                action               add
                                match-val-type       any
                                comparison-type      pattern-rule
                                match-
value                               $searchParams.$emptyUriParam
                                new-value           456

```

This is a sample of the result:

```

Request-Line: INVITE sip:service@192.168.200.60:5060;tgid=123 SIP/2.0
  Message Header
    Via: SIP/2.0/UDP
192.168.200.61:5060;branch=z9hG4bK681m9t30e0qh6akgj2s1.1
  From: sipp <sip:sipp@192.168.1.60:5060>;tag=SDchrc601-1
  To: sut <sip:service@192.168.1.61:5060>
  Call-ID: SDchrc601-fcf5660a56e2131fd27f12fcbd169fe8-06a3gu0
  CSeq: 1 INVITE
  Contact: <sip:sipp@192.168.200.61:5060;transport=udp>
  Goodbye: Remove Me
  Custom: This is my custom header
  Display: sipp <sip:user@192.168.1.60:5060;up=abc;hp=123
  Params: sipp <sip:sipp1@192.168.1.60:5060>
  Params: sipp <sip:sipp2@192.168.1.60:5060>

```

```

Edit: disp
<sip:user@192.168.1.60:5060;lpu=sipp2;apu=aup;not=123>;apu=apu
Max-Forwards: 69
Subject: Performance Test
Content-Type: application/sdp
Content-Length: 140

```

Example 9 Status-Line Manipulation

This section shows an HMR configuration set up for status-line manipulation.

Given that the object of this example is to drop the 183 Session Progress response when it does not have SDP, your SIP manipulation configuration needs to:

1. Search for the 183 Session Progress response
2. Determine if the identified 183 Session Progress responses contain SDP; the Oracle Enterprise Communications Broker searches the 183 Session Progress responses where the content length is zero
3. If the 183 Session Progress response does not contain SDP, change its status code to 699
4. Drop all 699 responses

```

sip-manipulation
  name                               manip
  description
  header-rule
    name                               IsContentLength0
    header-name                         Content-Length
    action                               store
    comparison-type                       pattern-rule
    match-value                           0
    msg-type                               reply
    new-value
    methods
  header-rule
    name                               is183
    header-name                         @status-line
    action                               store
    comparison-type                       pattern-rule
    match-value
    msg-type                               reply
    new-value
    methods
  element-rule
    name                               is183Code
    parameter-name
    type                                 status-code
    action                               store
    match-val-type                         any
    comparison-type                       pattern-rule
    match-value                           183
    new-value
  header-rule

```

```

        name                change183
        header-name         @status-line
        action              manipulate
        comparison-type     case-sensitive
        match-value
        msg-type            reply
        new-value
        methods
        element-rule
            name                make199
            parameter-name
            type                status-code
            action              replace
            match-val-type     any
            comparison-type    pattern-rule
            match-value        $IsContentLength0
& $is183.$is183Code
            new-value          199

sip-interface    options dropResponse=699

```

Example 10 Use of SIP HMR Sets

The following example shows the configuration for SIP HMR with one SIP manipulation configuration loading another SIP manipulation configuration. The goals of this configuration are to:

- Add a new header to an INVITE
- Store the user portion of the Request URI
- Remove all Route headers from the message only if the Request URI is from a specific user

```

sip-manipulation
    name                deleteRoute
    description         delete all Route Headers
    header-rule
        name                deleteRoute
        header-name        Route
        action              delete
        comparison-type    case-sensitive
        match-value
        msg-type            request
        new-value
        methods            INVITE
sip-manipulation
    name                addAndDelete
    description         Add a New header and delete
Route headers
    header-rule
        name                addHeader
        header-name        New
        action              add
        comparison-type    case-sensitive

```

```

        match-value
        msg-type                request
        new-value                "Some Value"
        methods                  INVITE
header-rule
    name                        storeRURI
    header-name                 request-uri
    action                       store
    comparison-type             pattern-rule
    match-value
    msg-type                    request
    new-value
    methods                      INVITE
element-rule
    name                         storeUser
    parameter-name
    type                          uri-user
    action                         store
    match-val-type                any
    comparison-type              pattern-rule
    match-value                   305.*
    new-value
header-rule
    name                         deleteHeader
    header-name                  request-uri
    action                       sip-manip
    comparison-type              Boolean
    match-value                  $storeRURI.$storeUser
    msg-type                     request
    new-value                    deleteRoute
    methods                      INVITE

```

Example 11 Use of Remote and Local Port Information

The following example shows the configuration for remote and local port information. The goals of this configuration are to:

- Add LOCAL_PORT as a header parameter to the From header
- Add REMOTE_PORT as a header parameter to the From header

```

sip-manipulation
    name                        addOrigIp
    description
    header-rule
        name                     addIpParam
        header-name              From
        action                    manipulate
        comparison-type           case-sensitive
        match-value
        msg-type                  request
        new-value
        methods                   INVITE
    element-rule
        name                      addIpParam

```

```

        parameter-name      newParam
        type                 header-param
        action               add
        match-val-type      any
        comparison-type     case-sensitive
        match-value
        new-value           $LOCAL_IP
    element-rule
        name                 addLocalPort
        parameter-name      lport
        type                 header-param
        action               add
        match-val-type      any
        comparison-type     case-sensitive
        match-value
        new-value           $LOCAL_PORT
    element-rule
        name                 addRemotePort
        parameter-name      rport
        type                 header-param
        action               add
        match-val-type      any
        comparison-type     case-sensitive
        match-value
        new-value           $REMOTE_PORT

```

Example 12 Response Status Processing

Given that the object of this example is to drop the 183 Session Progress response when it does not have SDP, your SIP manipulation configuration needs to:

1. Search for the 183 Session Progress response
2. Determine if the identified 183 Session Progress responses contain SDP; the Oracle Enterprise Communications Broker searches the 183 Session Progress responses where the content length is zero
3. If the 183 Session Progress response does not contain SDP, change its status code to 699
4. Drop all 699 responses

```

sip-manipulation
    name                    manip
    description
    header-rule
        name                IsContentLength0
        header-name         Content-Length
        action               store
        comparison-type     pattern-rule
        match-value         0
        msg-type            reply
        new-value
        methods
    header-rule
        name                is183

```

```

        header-name          @status-line
        action                store
        comparison-type       pattern-rule
        match-value
        msg-type              reply
        new-value
        methods
        element-rule
            name                is183Code
            parameter-name
            type                status-code
            action              store
            match-val-type     any
            comparison-type     pattern-rule
            match-value        183
            new-value
header-rule
    name                      change183
    header-name                @status-line
    action                     manipulate
    comparison-type            case-sensitive
    match-value
    msg-type                   reply
    new-value
    methods
    element-rule
        name                    make699
        parameter-name
        type                    status-code
        action                  replace
        match-val-type         any
        comparison-type         pattern-rule
        match-value             $IsContentLength0
& $is183.$is183Code
        new-value              699
sip-interface
    options dropResponse=699

```

The following four configuration examples are based on the this sample SIP INVITE:

```

INVITE sip:service@192.168.1.61:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.60:5060;branch=z9hG4bK-1-0
From: sipp <sip:sipp@192.168.1.60:5060>;tag=1
To: sut <sip:service@192.168.1.61:5060>
Call-ID: 1-15554@192.168.1.60
CSeq: 1 INVITE
Contact: <sip:sipp@192.168.1.60:5060;user=phone>
Max-Forwards: 70
Content-Type: multipart/mixed;boundary=boundary
Content-Length: 466
--boundary
Content-Type: application/sdp
v=0
o=user1 53655765 2353687637 IN IP4 192.168.1.60

```

```

s=-
c=IN IP4 192.168.1.60
t=0 0
m=audio 12345 RTP/AVP 18
a=rtpmap:8 G729/8000/1
a=fmtp:18 annexb=no
a=sendrecv
a=ptime:20
a=maxptime:200
--boundary
Content-Type: application/sdp
v=0
o=user1 53655765 2353687637 IN IP4 192.168.1.60
s=-
c=IN IP4 192.168.1.60
t=0 0
m=video 12345 RTP/AVP 34
a=rtpmap:34 H263a/90000
a=ptime:30
--boundary--

```

Example 13 Remove a Line from SDP

In this example, the SIP manipulation is configured to remove all p-time attributes from the SDP.

```

sip-manipulation
    name                removePtimeFromBody
    description         removes ptime attribute from all
bodies
    header-rule
        name            CTypeManp
        header-name     Content-Type
        action          manipulate
        comparison-type case-sensitive
        match-value
        msg-type        request
        new-value
        methods         INVITE
    element-rule
        name            remPtime
        parameter-name  application/sdp
        type            mime
        action          find-replace-all
        match-val-type  any
        comparison-type case-sensitive
        match-value     a=ptime:[0-9]{1,2}
(\n|\r\n)
    new-value

```

The result of manipulating the original SIP INVITE (shown above) with the configured SIP manipulation is:

```
INVITE sip:service@192.168.1.61:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.60:5060;branch=z9hG4bK-1-0
From: sipp <sip:sipp@192.168.1.60:5060>;tag=1
To: sut <sip:service@192.168.1.61:5060>
Call-ID: 1-15554@192.168.1.60
CSeq: 1 INVITE
Contact: <sip:sipp@192.168.1.60:5060;user=phone>
Max-Forwards: 70
Content-Type: multipart/mixed;boundary=boundary
Content-Length: 466
--boundary
Content-Type: application/sdp
v=0
o=user1 53655765 2353687637 IN IP4 192.168.1.60
s=-
c=IN IP4 192.168.1.60
t=0 0
m=audio 12345 RTP/AVP 18
a=rtpmap:18 G729/8000/1
a=fmtp:18 annexb=no
a=sendrecv
a=maxptime:200
--boundary
Content-Type: application/sdp
v=0
o=user1 53655765 2353687637 IN IP4 192.168.1.60
s=-
c=IN IP4 192.168.1.60
t=0 0
m=video 12345 RTP/AVP 34
a=rtpmap:34 H263a/90000
--boundary-
```

Example 14 Back Reference Syntax

In this sample of back-reference syntax use, the goal is to change the To user. The SIP manipulation would be configured like the following:

sip-manipulation		
name		changeToUser
description		change user in the To header
header-rule		
name		ChangeHeader
header-name		To
action		manipulate
comparison-type		case-sensitive
match-value		
msg-type		request
new-value		
methods		INVITE
element-rule		


```

name                               replaceValue
parameter-name                      header-value
type                                 replace
action                               any
match-val-type                       pattern-rule
comparison-type                      (.+)(service)(.+)
match-value                          $1+Bob+$3
new-value

```

The result of manipulating the original SIP INVITE (shown above) with the configured SIP manipulation is:

```

INVITE sip:service@192.168.1.61:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.60:5060;branch=z9hG4bK-1-0
From: sipp <sip:sipp@192.168.1.60:5060>;tag=1
To: sut <sip:Bob@192.168.1.61:5060>
Call-ID: 1-15554@192.168.1.60
CSeq: 1 INVITE
Contact: <sip:sipp@192.168.1.60:5060;user=phone>
Max-Forwards: 70
Content-Type: multipart/mixed;boundary=boundary
Content-Length: 466
...
...
...

```

Example 15 Change and Remove Lines from SDP

In this sample of changing and removing lines from the SDP, the goal is to convert the G.729 codec to G.729a. The SIP manipulation would be configured like the following:

```

sip-manipulation
  name                               std2prop-codec-name
  description                         rule to translate standard to
  proprietary codec name
  header-rule
    name                               CTypeManp
    header-name                       Content-Type
    action                             manipulate
    comparison-type                   case-sensitive
    match-value
    msg-type                           any
    new-value
    methods
  element-rule
    name                               g729-annexb-no-
std2prop
  parameter-name                     application/sdp
  type                                 mime
  action                             find-replace-all
  match-val-type                      any
  comparison-type                    case-sensitive
  match-value                         a=rtpmap:[0-9]{1,3}

```

```
(G729/8000/1\r\na=fmtp:[0-9]{1,3} annexb=no)[[:1:]]
new-value
G729a/8000/1
```

The result of manipulating the original SIP INVITE (shown above) with the configured SIP manipulation is:

```
INVITE sip:service@192.168.1.61:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.60:5060;branch=z9hG4bK-1-0
From: sipp <sip:sipp@192.168.1.60:5060>;tag=1
To: sut <sip:service@192.168.1.61:5060>
Call-ID: 1-15554@192.168.1.60
CSeq: 1 INVITE
Contact: <sip:sipp@192.168.1.60:5060;user=phone>
Max-Forwards: 70
Content-Type: multipart/mixed;boundary=boundary
Content-Length: 466
--boundary
Content-Type: application/sdp
v=0
o=user1 53655765 2353687637 IN IP4 192.168.1.60
s=-
c=IN IP4 192.168.1.60
t=0 0
m=audio 12345 RTP/AVP 8
a=rtpmap:18 G729a/8000/1
a=sendrecv
a=maxptime:200
--boundary
Content-Type: application/sdp
v=0
o=user1 53655765 2353687637 IN IP4 192.168.1.60
s=-
c=IN IP4 192.168.1.60
t=0 0
m=video 12345 RTP/AVP 34
a=rtpmap:34 H263a/90000
--boundary-
```

Example 16 Change and Add New Lines to the SDP

In this sample of changing and adding lines from the SDP, the goal is to convert non-standard codec H.263a to H.263. The SIP manipulation would be configured like the following:

```

sip-manipulation
  name
  description
  standard codec name
  header-rule
    name
    header-name
    action
    comparison-type
  prop2std-codec-name
  rule to translate proprietary to
  CodecManp
  Content-Type
  manipulate
  case-sensitive
```

```

match-value
msg-type
new-value
methods
element-rule
    name
    parameter-name
    type
    action
    match-val-type
    comparison-type
    match-value
{1,3}) H263a/.*\r\n
    new-value
H263/90000"+$CRLF+a=fmtp:+$1+" QCIF=4"+$CRLF
    a=rtpmap:+$1+"

```

The result of manipulating the original SIP INVITE (shown above) with the configured SIP manipulation is:

```

INVITE sip:service@192.168.1.61:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.60:5060;branch=z9hG4bK-1-0
From: sipp <sip:sipp@192.168.1.60:5060>;tag=1
To: sut <sip:service@192.168.1.61:5060>
Call-ID: 1-15554@192.168.1.60
CSeq: 1 INVITE
Contact: <sip:sipp@192.168.1.60:5060;user=phone>
Max-Forwards: 70
Content-Type: multipart/mixed;boundary=boundary
Content-Length: 466
--boundary
Content-Type: application/sdp
v=0
o=user1 53655765 2353687637 IN IP4 192.168.1.60
s=-
c=IN IP4 192.168.1.60
t=0 0
m=audio 12345 RTP/AVP 8
a=rtpmap:18 G729/8000/1
a=fmtp:18 annexb=no
a=sendrecv
a=maxptime:200
--boundary
Content-Type: application/sdp
v=0
o=user1 53655765 2353687637 IN IP4 192.168.1.60
s=-
c=IN IP4 192.168.1.60
t=0 0
m=video 12345 RTP/AVP 34
a=rtpmap:34 H263/90000
a=fmtp:34 QCIF=4
--boundary-

```

Dialog-Matching Header Manipulation

The most common headers to manipulate using HMR are the To-URI and From-URI. Along with the to-tag, from-tag, and Call-ID values, these are also all headers that represent dialog-specific information that must match the UAC and UAS to be considered part of the same dialog. If these parameters are modified through HMR, the results can be that the UAC or UAS rejects messages.

While it is possible to ensure that dialog parameters match correctly using regular HMR, this feature offers a simpler and less error-prone method of doing so.

In addition, this section describes the addition of built-in SIP manipulations defined by Oracle best practices, and a new method of testing your SIP manipulations.

About Dialog-Matching Header Manipulations

The goal of this feature is to maintain proper dialog-matching through manipulation of dialog-specific information using HMR. Two fundamental challenges arise when looking at the issue of correctly parameters manipulating dialog-matching:

- Inbound HMR
- Outbound HMR

The new setting **out-of-dialog** (for the **msg-type** parameter) addresses these challenges by offering an intelligent more of dialog matching of messages for inbound and outbound HMR requests. This is a **msg-type** parameter, meaning that it becomes matching criteria for operations performed against a message. If you also specify methods (such as REGISTER) as matching criteria, then the rule is further limited to the designated method.

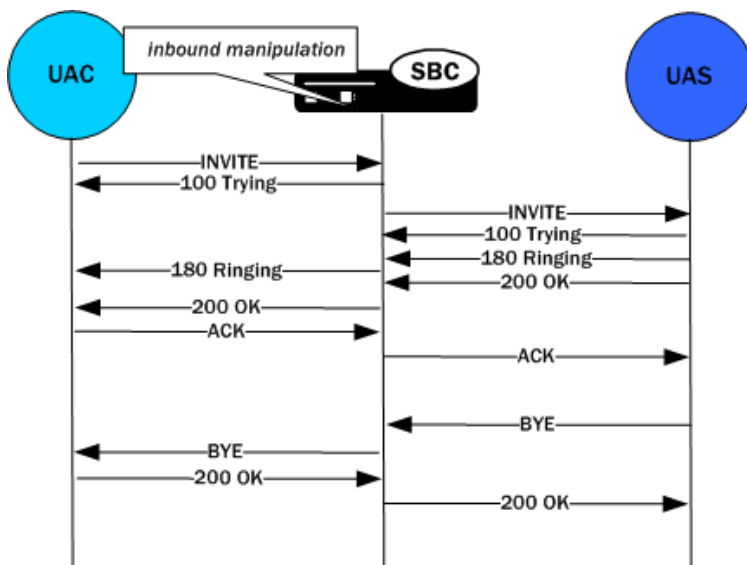
For both inbound and outbound manipulations, using the **out-of-dialog** setting means the message must be a request without a to-tag in order to perform the manipulation.

Inbound HMR Challenge

Because inbound manipulations take place before the message reaches the core of Oracle Enterprise Communications Broker (OECB) SIP processing, the SIP proxy takes the manipulated header as directly received from the client. This can cause problems for requests leaving the OECB for the UAC because the dialog does not match the initial request sent.

The unmodified header must be cached because for any subsequent request (For example, a BYE originating from the terminator. See the following diagram.) the OECB might need to restore the original value, enabling the UAC to identify the message correctly as being part of the same dialog. For out-of-dialog requests (when the To, From, or Call-ID headers are modified) the original header is stored in the dialog when the **msg-type out-of-dialog** is used.

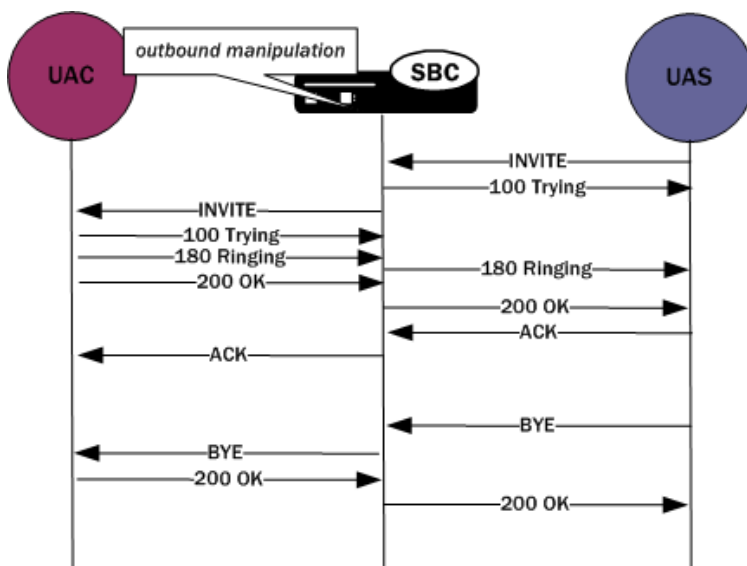
The OECB performs the restoration of original headers outside of SIP manipulations. There are no manipulation rules to configure for restore the header to their original context. The OECB recognizes that the headers are modified, and restores them to their original state prior to sending the message out. Restoration takes place prior to outbound manipulations so that any outbound manipulation can those headers after they are restored.



Outbound HMR Challenge

When you use the **out-of-dialog** setting for an outbound manipulation, the Oracle Enterprise Communications Broker executes this specific SIP header rule only if the same SIP header rule was executed against the initial dialog-creating request.

For example, if the INVITE's To header was not manipulated, it would not be correct to manipulate the To header in the BYE request. To do so would render the UAC unable to properly match the dialog. And this also means that the outbound manipulation should be carried out against a To, From, or Call-ID header in the BYE request if it was manipulated in the INVITE.



Built-In SIP Manipulations

In the course of HMR use, certain rules have become commonly used. Lengthy and complex, these rules do not include any customer-specific information and do they can be used widely. To make using them easier, they have been turned into built-in rules that you can reference in the **in-manipulationid** and **out-manipulationid** parameters that are part of the realm, session agent, and SIP interfaces configurations.

Built-in rules start with the prefix `ACME_`, so Oracle recommends you name your own rules in a different manner to avoid conflict.

While the number of built-in manipulation rules is expected to grow, one is supported at the present time: `ACME_NAT_TO_FROM_IP`. When performed outbound, this rule changes:

- The To-URI hostname to the logical `$TARGET_IP` and port to `$TARGET_PORT`
- The From-URI to the logical `$REPLY_IP` and port to be `$REPLY_PORT`

Unique HMR Regex Patterns and Other Changes

In addition to the HMR support it offers, the Oracle Enterprise Communications Broker can now be provisioned with unique regex patterns for each logical remote entity. This supplement to pre-existing HMR functionality saves you provisioning time and saves Oracle Enterprise Communications Broker resources in instances when it was previously necessary to define a unique SIP manipulation per PBX for a small number of customer-specific rules.

Manipulation Pattern Per Remote Entity

On the Oracle Enterprise Communications Broker, you can configure logical remote entities (session agents, realms, and SIP interfaces) with a manipulation pattern string that the system uses as a regular expression. Then the SIP manipulation references this regular expression using the reserved word `$MANIP_PATTERN`. At runtime, the Oracle Enterprise Communications Broker looks for the logical entity configured with a manipulation pattern string in this order of preference: session agent, realm, and finally SIP interface.

On finding the logical entity configured with the manipulation string, the Oracle Enterprise Communications Broker dynamically determines the expression. When there is an invalid reference to a manipulation pattern, the pattern-rule expression that results will turn out to be the default expression (which is `\,+`).

When the `$MANIP_PATTERN` is used in a manipulation rule's **new-value** parameter, it resolves to an empty string, equivalent of no value. Even though this process ends with no value, it still consumes system resources. And so Oracle recommends you do not use `$MANIP_PATTERN` as a **new-value** value.

In the following example, the SIP manipulation references the regular expression from a realm configuration:

```
realm-config
  identifier                net200
  description
  addr-prefix                0.0.0.0
```

```

network-interfaces          public:0
...
manipulation-pattern       Lorem(+)
sip-manipulation
  name                      manip
  description
  header-rules
    name                    headerRule
    header-name             Subject
    action                  manipulate
    match-value             $MANIP_PATTERN
    msg-type                request
    comparison-type        pattern-rule
    new-value               Math
    methods                 INVITE

```

Reject Action

When you use this action type and a condition matching the manipulation rule arises, the Oracle Enterprise Communications Broker rejects the request (though does not drop responses) and increments a counter.

- If the **msg-type** parameter is set to **any** and the message is a response, the Oracle Enterprise Communications Broker increments a counter to show the intention to reject the message—but the message will continue to be processed.
- If the **msg-type** parameter is set to **any** and the message is a request, the Oracle Enterprise Communications Broker performs the rejection and increments the counter.

The **new-value** parameter is designed to supply the status code and reason phrase corresponding to the reject. You can use the following syntax to supply this information: `status-code[:reason-phrase]`. You do not have to supply the status code and reason phrase information; by default, the system uses 400:Bad Request.

If you do supply this information, then the status code must be a positive integer between 300 and 699. The Oracle Enterprise Communications Broker then provides the reason phrase corresponding to the status code. And if there is no reason phrase, the system uses the one for the applicable reason class.

You can also customize a reason phrase. To do so, you enter the status code followed by a colon (:), being sure to enclose the entire entry in quotation marks (") if your reason code includes spaces.

When the Oracle Enterprise Communications Broker performs the **reject** action, the current SIP manipulation stops processing and does not act on any of the rules following the **reject** rule. This course of action is true for nested SIP manipulations that might have been constructed using the **sip-manip** action type.

SNMP Support

The Oracle Enterprise Communications Broker provides SNMP support for the Rejected Messages data, so you can access this information externally. The new MIB objects are:

```

apSysRejectedMessages    OBJECT-TYPE
    SYNTAX                Counter32
    MAX-ACCESS            read-only
    STATUS                 current
    DESCRIPTION
        "Number of messages rejected by the SD due to matching
        criteria."
    ::= { apSysMgmtMIBGeneralObjects 18 }
apSysMgmtRejectedMessagesThresholdExceededTrap    NOTIFICATION-TYPE
    OBJECTS                { apSysRejectedMessages }
    STATUS                 current
    DESCRIPTION
        " The trap will be generated when the number of rejected messages
        exceed the configured threshold within the configured window."
    ::= { apSystemManagementMonitors 57 }
apSysMgmtRejectedMessagesGroup    OBJECT-GROUP
    OBJECTS {
        apSysRejectedMessages
    }
    STATUS                 current
    DESCRIPTION
        "Objects to track the number of messages rejected by the
        SD."
    ::= { apSystemManagementGroups 18 }
apSysMgmtRejectedMessagesNotificationsGroup    NOTIFICATION-GROUP
    NOTIFICATIONS {
        apSysMgmtRejectedMessagesThresholdExceededTrap
    }
    STATUS                 current
    DESCRIPTION
        "Traps used for notification of rejected messages"
    ::= { apSystemManagementNotificationsGroups 26 }
apSmgmtRejectedMessagesCap
    AGENT-CAPABILITIES
    PRODUCT-RELEASE       "Acme Packet SD"
    STATUS                 current
    DESCRIPTION           "Acme Packet Agent Capability for
    enterprise
                                system management MIB."
    SUPPORTS               APSYSMGMT-MIB
    INCLUDES               {
        apSysMgmtRejectedMessagesGroup,

apSysMgmtRejectedMessagesNotificationsGroup
    }
    ::= { apSmgmtMibCapabilities 37 }
  
```


Log Action

When you use this action type and a condition matching the manipulation rule arises, the Oracle Enterprise Communications Broker logs information about the current message to a separate log file. This log files will be located on the same core in which the SIP manipulation occurred. On the core where sipt runs, a logfile called `matched.log` will appear when this action type is executed.

The `matched.log` file contains a timestamp, received and sent Oracle Enterprise Communications Broker network interface, sent or received IP address:port information, and the peer IP address:port information. It also specifies the rule that triggered the log action in this syntax: `rule-type[rule:name]`. The request URI, Contact header, To Header, and From header are also present.

```
-----  
Apr 17 14:17:54.526 On [0:0]192.168.1.84:5060 sent to 192.168.1.60:5060  
element-rule[checkRURIPort]  
INVITE sip:service@192.168.1.84:5060 SIP/2.0  
From: sipp <sip:+2125551212@192.168.1.60:5060>;tag=3035SIPpTag001  
To: sut <sip:service@192.168.1.84>  
Contact: sip:sipp@192.168.1.60:5060
```

Name Restrictions for Manipulation Rules

Historically, you have been allowed to configure any value for the name parameter within a manipulation rule. This method of naming caused confusion when referencing rules, so now manipulation rules name must follow a specific syntax. They must match the expression `^[[alpha:]][:alnum:~_]+$` and contain at least one lower case letter.

In other words, the name must:

- Start with a letter, and then it can contain any number of letters, numbers, or underscores
- Contain at least one lower case letter

All pre-existing configurations will continue to function normally. If you want to change a manipulation rule, however, you are required to change its name if it does not follow the new format.

The ACLI **verify-config** command warns you if the system has loaded a configuration containing illegal naming syntax.

Please note that the software allows you to make changes to HMRs, including configuring new functionality to existing rules, as long as you do not change the rule name. This results in an important consideration surrounding HMRs with hyphens in previously configured rule names.

- You can reference stored values in new value names. (Recall that stored values may be rule names.)
- You can perform subtraction in new value names.

If you use a rule names with hyphens within the REGEX of new value names, the system cannot determine whether the hyphen is part of the rule name or is intended to

invoke subtraction within the REGEX. For this reason, you need to use great care with legacy HMR naming that includes hyphens.

As a general rule, create new rule names that follow the new rule naming guidelines if you intend to use new functionality in those rules.

New Value Restrictions

To simplify configuration and remove possible ambiguity, the use of boolean and equality operators (`==`, `<=`, `<`, etc.) for **new-value** parameter values has been banned. Since there was no specific functionality tied to their use, their ceasing to be use will have no impact to normal SIP manipulation operations.

Header Manipulation Rules for SDP

The Oracle Enterprise Communications Broker supports SIP header and parameter manipulation rules for four types of SIP message contents:

- headers
- elements within headers
- ASCII-encoded Multipurpose Internet Mail Extensions (MIME) bodies
- binary-encoded MIME ISDN User Part (ISUP) bodies

While Session Description Protocol (SDP) offers and answers can be manipulated in a fashion similar to ASCII-encoded MIME, such manipulation is primitive in that it lacks the ability to operate at the SDP session- and media-levels.

In addition, the system supports a variant of Header Manipulation Rules (HMR) operating on ASCII-encoded SDP bodies, with specific element types for descriptors at both the session-level and media-level, and the ability to apply similar logic to SDP message parts as is done for SIP header elements.

The configuration object, `mime-sdp-rules`, under `sip-manipulation` specifically addresses the manipulation of SDP parts in SIP messages. Just as existing header-rules are used to manipulate specific headers of a SIP message, `mime-sdp-rules` will be used to manipulate the SDP specific mime-attachment of a SIP message.

SDP Manipulation

`mime-sdp-rules` function in a similar fashion as `header-rules`. They provide

- parameters used to match against specific SIP methods and/or message types
- parameters used to match and manipulate all or specified parts of an SDP offer or answer
- a means of comparing search strings or expressions against the entire SDP
- different action types to allow varying forms of manipulation

Since only a single SDP can exist within a SIP message, users need not specify a content-type parameter as is necessary for a mime-rule. A `mime-sdp-rule` operates on the single SDP within the SIP message. If no SDP exists with the message, one can be added. If the message already contains a mime attachment, adding SDP results in a multipart message.

All manipulations performed against all or parts of the SDP are treated as UTF-8 ASCII encoded text. At the parent-level (mime-sdp-rule) the **match-value** and **new-value** parameters execute against the entire SDP as a single string.

An add action only succeeds in the absence of SDP because a message is allowed only a single SDP offer or answer. A delete operation at the mime-sdp-rule level will remove the SDP entirely.

Note that on an inbound sip-manipulation, SDP manipulations interact with the Oracle Enterprise Communications Broker codec-policy. SDP manipulations also interact with codec reordering and media setup. It is very possible to make changes to the SDP such that the call can not be setup due to invalid media parameters, or settings that will affect the ability to transcode the call. Consequently, user manipulation of the SDP can prove risky, and should be approached with appropriate caution.

Three configuration-objects, sdp-session-rule, sdp-media-rule, and mime-header-rule, exist under the mime-sdp-rule. These objects provide finer grained control of manipulating parts of the SDP.

sdp-session-rule

An sdp-session-rule groups all SDP descriptors, up until the first media line, into a single entity, thus allowing the user to perform manipulation operations on a session-specific portion of the SDP.

Like the mime-sdp-rule, all match-value and new-value operations performed at this level are executed against the entire session group as a complete string. Given the sample SDP below, if an sdp-session-rule is configured, the match-value and new-values operate only on the designated portion.

```
v=0
o=mhandley 2890844526 2890842807 IN IP4 126.16.64.4
s=SDP Seminar
i=A Seminar on the session description protocol
u=http://www.cs.ucl.ac.uk/staff/M.Handley/sdp.03.ps
e=mjh@isi.edu (Mark Handley)
c=IN IP4 224.2.17.12/127
t=2873397496 2873404696
a=recvonly
m=audio 49170 RTP/AVP 0
m=video 51372 RTP/AVP 31
m=application 32416 udp wb
a=orient:portrait
```

Nested under the sdp-session-rule configuration object is an sdp-line-rule object, the object that identifies individual descriptors within the SDP. The types of descriptors used at the sdp-session-rule level are v, o, s, i, u, e, p, c, b, t, r, z, k, and a, the descriptors specific to the entire session description.

This level of granularity affords the user a very simple way to making subtle changes to the session portion of the SDP. For instance, it is very common to have to change the connection line at the session level.

The add and delete actions perform no operation at the sdp-session-rule level.

sdp-media-rule

An sdp-media-rule groups all of the descriptors that are associated with a specific media-type into single entity, thus allowing the user to perform manipulation operations on a media-specific portion of the SDP. For example, a user can construct an sdp-media-rule to change an attribute of the audio media type.

Like a mime-sdp-rule, all match-value and new-value operations performed at this level are executed against the entire media-group as a complete string. Given the sample SDP below, if a media-level-descriptor is configured to operate against the application group, the match-value and new-values would operate only on designated portion.

```
v=0
o=mhandley 2890844526 2890842807 IN IP4 126.16.64.4
s=SDP Seminar
i=A Seminar on the session description protocol
u=http://www.cs.ucl.ac.uk/staff/M.Handley/sdp.03.ps
e=mjh@isi.edu (Mark Handley)
c=IN IP4 224.2.17.12/127
t=2873397496 2873404696
a=recvonly
m=audio 49170 RTP/AVP 0
m=video 51372 RTP/AVP 31
m=application 32416 udp wb
a=orient:portrait
```

A configuration parameter **media-type** is used to specify the media group on which to operate. It contains all of the descriptors including the m-line up to the next m-line. This parameter is a string field and must match the media-type exactly as it appears within the SDP. The special media-type media can be used to refer to all media types. This is particularly useful when performing an add operation, when the user wants to add a media section between the first and second medias, but does not know what media type they are. Otherwise, during an add operation, the media section would be added before the specified media-type (if no index parameter was provided).

The types of descriptors used at the sdp-media-rule level are m, i, c, b, k, and a, the descriptors specific to the media description.

This level of granularity affords the user a very simple way to making subtle changes to the media portion of the SDP. For instance, it is very common to have to change the name of an audio format (for example G729 converted to g729b), or to add attributes specific to a certain media-type.

The index operator is supported for the media-type parameter (for example, media-type audio[1]). Like header rules, if no index is supplied, this means operate on all media-types that match the given name. For specifying specific media-types, the non-discrete indices are also supported (for example, ^ - last). Adding a media-type, without any index supplied indicates that the media should be added at the beginning. The special media-type media uses the index as an absolute index to all media sections, while a specific media-type will index relative to that given media type.

For sdp-media-rules set to an action of add where the media-type is set to media, the actual media type is obtained from the new-value, or more specifically, the string after m= and before the first space.

Given the following SDP:

```
v=0
o=mhandley 2890844526 2890842807 IN IP4 126.16.64.4
c=IN IP4 224.2.17.12/127
t=2873397496 2873404696
m=audio 49170 RTP/AVP 0
m=audio 48324 RTP/AVP 8
m=video 51372 RTP/AVP 31
```

With the `sdp-media-rule`:

```
sdp-media-rule
  name                smr
  media-type          audio[1]
  action              manipulate
  comparison-type     case-sensitive
  match-value
  new-value           "m=audio 1234 RTP/AVP 8 16"
```

This rule operates on the 2nd audio line, changing the port and adding another codec, resulting in the SDP:

```
v=0
o=mhandley 2890844526 2890842807 IN IP4 126.16.64.4
c=IN IP4 224.2.17.12/127
t=2873397496 2873404696
m=audio 49170 RTP/AVP 0
m=audio 1234 RTP/AVP 8 16
m=video 51372 RTP/AVP 31
```

The following rule, however:

```
sdp-media-rule
  name                smr
  media-type          media[1]
  action              add
  comparison-type     case-sensitive
  match-value
  new-value           "m=video 1234 RTP/AVP 45"
```

adds a new video media-type at the 2nd position of all media-lines, resulting in the SDP:

```
v=0
o=mhandley 2890844526 2890842807 IN IP4 126.16.64.4
c=IN IP4 224.2.17.12/127
t=2873397496 2873404696
m=audio 49170 RTP/AVP 0
m=video 1234 RTP/AVP 45
m=audio 48324 RTP/AVP 8
m=video 51372 RTP/AVP 31
```

sdp-line-rule

Unlike header-rules, sdp descriptors are not added in the order in which they are configured. Instead they are added to the SDP adhering to the grammar defined by RFC 4566 (as is shown below).

```

Session description
  v= (protocol version)
  o= (originator and session identifier)
  s= (session name)
  i=* (session information)
  u=* (URI of description)
  e=* (email address)
  p=* (phone number)
  c=* (connection information -- not required if included in
      all media)
  b=* (zero or more bandwidth information lines)
  One or more time descriptions ("t=" and "r=" lines; see
      below)
  z=* (time zone adjustments)
  k=* (encryption key)
  a=* (zero or more session attribute lines)
  Zero or more media descriptions (see below)

Time description
  t= (time the session is active)
  r=* (zero or more repeat times)

Media description, if present
  m= (media name and transport address)
  i=* (media title)
  c=* (connection information -- optional if included at
      session level)
  b=* (zero or more bandwidth information lines)
  k=* (encryption key)
  a=* (zero or more media attribute lines)

```

* after the equal sign denotes an optional descriptor.

This hierarchy is enforced meaning that if you configure a rule which adds a session name descriptor followed by a rule which adds a version descriptor, the SDP will be created with the version descriptor first, followed by the session name.

The only validation that will occur is the prevention of adding duplicate values. In much the same way that header-rules prevents the user from adding multiple To headers, the descriptor rule will not allow the user to add multiple descriptors; unless multiple descriptors are allowed, as is in the case of b, t, r and a.

There exists a parameter **type** under the sdp-line-rule object that allows the user to specify the specific line on which to perform the operation. For example: v, o, s, i, u, e, p, c, b, t, r, z, k, a, and m. Details on these types can be found in RFC 4566.

For those descriptors, of which there may exist zero or more (b, t, r, and a) entries, indexing grammar may be used to reference the specific instance of that attribute. This

indexing grammar is consistent with that of header-rules for referring to multiple headers of the same type.

Given the example SDP below:

```
v=0
o=mhandley 2890844526 2890842807 IN IP4 126.16.64.4
s=SDP Seminar
i=A Seminar on the session description protocol
u=http://www.cs.ucl.ac.uk/staff/M.Handley/sdp.03.ps
e=mjh@isi.edu (Mark Handley)
c=IN IP4 224.2.17.12/127
t=2873397496 2873404696
r=604800 3600 0 90000
r=7d 1h 0 25h
a=recvonly
m=audio 49170 RTP/AVP 0
m=video 51372 RTP/AVP 31
m=application 32416 udp wb
a=orient:portrait
```

and the following sdp-line-rule:

```
sdp-line-rule
  name          removeRepeatInterval
  type          r[1]
  action        delete
```

The rule `removeRepeatInterval` removes the second repeat interval descriptor within the SDP.

The behavior of all SDP rules follow the same behavior of all manipulation rules in that they are executed in the order in which they are configured and that each rule executes on the resultant of the previous rule.

Each descriptor follows its own grammar and rules depending on the type specified. The values of the descriptor are evaluated at runtime since the new-values themselves are evaluated at runtime. At this time no validation of the grammar for each of the types is performed. The user is responsible for properly formatting each of the descriptors according to their specifications.

For instance, the version (`v`) descriptor can be removed from the SDP but leaving all descriptors for that SDP, causing the SDP to become invalid. This is consistent with the way header-rules operate, in that there is no validation for the specific headers once they have been manipulated through HMR.

Regular Expression Interpolation

An interpolated regular expression is a regular expression that is compiled and evaluated at runtime. Today all regular expressions are compiled at configuration time in order to improve performance. There are cases where a regular expression is determined dynamically from data within a SIP message. In these circumstances the regular expression is unknown until the time of execution.

In order to have a regular expression be interpolated at runtime, it must be contained within a set of `{}`. An interpolated expression can have any number of regular expressions and strings appended together. Any characters to the left or right of the curly braces will be appended to the value within the curly braces. The curly braces are effectively two operators treated as one (interpolate the value contained within and then concatenate the values to the left and right of the curly braces). If the `comparison-type` is set to `pattern-rule` and the `match-value` contains a value that matches the grammar below, then it will be treated as an interpolated expression.

```
([^\]|^){\${^0-9]+[^\]*}
```

The example below demonstrates using a user defined variable within a regular expression of another rule at runtime.

```
element-rule
```

name	someRule
type	header-value
action	replace
comparison-type	pattern-rule
match-value	^sip:{\$rule1.\$0}@(.+)\$
new-value	sip:bob@company.com

If the value of `$rule1.$0` evaluates to `alice` then it will successfully match against the string `sip:alice@comcast.net`. An interpolated expression can be as simple as `"{$rule1.$0}"` or as complex as `^sip:{rule1.$0}@{$rule2[1].$2}$`. It is not possible to interpolate a normal regular expression since the grammar will not allow the user to enter such an expression. Only variables can be contained with the curly braces.

The resultant of interpolated expressions can be stored in user defined variables. Given the same example from above, if the rule `someRule` was referenced by another rule, the value of `sip:alice@comcast.net` would be stored within that rule.

Interpolation only makes a single pass at interpolation, but does so every time the Rule executes. In other words, if the Rule is applied to the Route header, it will interpolate again for each Route header instance. What this means is that the value within the curly braces will only be evaluated once. For instance, if the value `{$someRule.$1}` evaluates to `{$foobar.$2}` the Oracle Enterprise Communications Broker (OECB) will treat `$foobar.$2` as a literal string which it will compile as a regular expression. The OECB will not recursively attempt to evaluate `$foobar.$2`, even if it was a valid user defined variable.

Interpolated regular expressions will evaluate to `TRUE` if and only if both the regular expression itself can be compiled and it successfully matches against the compared string.

Regular Expressions as Boolean Expressions

Regular expressions can be used as boolean expressions today if they are the only value being compared against a string, as is shown in the case below.

```
mime-rule
```

name	someMimeRule
content-type	application/text
action	replace


```

comparison-type      pattern-rule
match-value          ^every good boy .*
new-value            every good girl does fine

```

However, regular expressions can not be used in conjunction with other boolean expressions to form more complex boolean expressions, as is shown below.

```

mime-rule
  name                someMimeRule
  content-type        application/text
  action              replace
  comparison-type     boolean
  match-value         $someRule & ^every good boy .*
  new-value           every good girl does fine

```

There are many cases where the user has the need to compare some value as a regular expression in conjunction with another stored value. It is possible to perform this behavior today, however it requires an extra step in first storing the value with the regular expression, followed by another Manipulation Rule which compares the two boolean expressions together (e.g. \$someRule & \$someMimeRule).

In order to simplify the configuration of some sip-manipulations and to make them more efficient this functionality is being added.

Unfortunately, it is not possible to just use the example as is shown above. The problem is there are many characters that are commonly used in regular expressions that would confuse the HMR expression parser (such as \$, and +). Therefore delimiting characters need to be used to separate the regular expression from the other parts of the expression.

To treat a regular expression as a boolean expression, it needs to be enclosed within the value \$REGEX(<expression>,<compare_string>=\$ORIGINAL); where <expression> is the regular expression to be evaluated. <compare_string> is the string to compare against the regular expression. This second argument to the function is defaulted to \$ORIGINAL which is the value of the of the specific Manipulation Rule object. It can be overridden to be any other value the user desires.

The proper configuration for the example above to use regular expressions as boolean expressions is

```

mime-rule
  name                someMimeRule
  content-type        application/text
  action              replace
  comparison-type     boolean
  match-value         $someRule & $REGEX("^every good boy .*")
  new-value           every good girl does fine

```

It is also possible to use expressions as arguments to the \$REGEX function. These expressions will in turn be evaluated prior to executing the \$REGEX function. A more complex example is illustrated below.

```

header-rule
  name                checkPAU
  header-name         request-uri

```

```

action          reject
comparison-type boolean
match-value     (!$REGEX($rule1[0],$FROM_USER))&
                (!$REGEX($rule2[0],$PAI_USER))

msg-type        request
new-value       403:Forbidden
methods         INVITE,SUBSCRIBE,MESSAGE,PUBLISH,
                OPTIONS, REFER

```

It should be noted that when using `$REGEX()` in a boolean expression, the result of that expression is not stored in the user variable. The comparison-type must be set to pattern-rule in order to store the result of a regular expression.

The arguments to the `$REGEX()` function are interpolated by default. This is the case since the arguments themselves must be evaluated at runtime. The following example is also valid.

```

mime-rule
  name          someMimeRule
  content-type   application/text
  action         replace
  comparison-type boolean
  match-value    $someRule & $REGEX("^every good
                {$rule1[0].$0} .*")

```

Moving Manipulation Rules

You can move rules within any manipulation-rule container. Any manipulation rule that contains sub-rules offers the ACLI command **move** <from index> <to index>. For example, given the order and list of rules below:

1. rule1
2. rule2
3. rule3
4. rule4

You can move rule3 to position 1 by executing **move 3 1**. The resulting order is: rule3, rule1, rule2, rule4. A move operation causes a shift (or insert before) for all other rules. When you move a rule from the top or middle to the bottom, the system shifts all rules above the bottom up to the position of the rule that you moved. When you move a rule from the bottom or middle to the top, the system shifts all rules below down to the position of the rule that you moved. Positions start from 1.

A valid from-index and to-index are required to be supplied as arguments to the move action. If you enter a range that is out of bounds for either the from-index or to-index, the ACLI informs you that the command did not execute and the reason.

If you create an invalid sip-manipulation by incorrectly ordering the manipulation rules, the Oracle Enterprise Communications Broker validates the rules at configuration time and treats them as invalid prior to runtime. This may or may not affect the outcome of the sip-manipulation as a configured rule may not perform any operation if it refers to a rule that has yet to be executed. It is your responsibility to reorder the remaining rules in order to make the sip-manipulation valid again.

Note that rules of a different type at the same level are all part of the same list. Header-rules, mime-rules, mime-isup-rules, and mime-sdp-rules all share the same configuration level under sip-manipulation. When selecting a move from-index and to-index for a header-rule, you must take into consideration the location of all other rules at the same level because the move is relative to all rules at that level. The move is not relative to the particular rule you selected (for example, the header-rule).

Because the list of rules at any one level can be lengthy, you can issue the **move** command one argument at a time, providing you with the ability to select indices. For example, typing **move** without any arguments displays the list of all the rules at that level. After selecting an appropriate index, the system prompts you with a to-index location based on the same list provided.

For Example:

```
ORACLE(sip-mime-sdp-rules)# move
select a rule to move

1: msr sdp-type=any; action=none; match-value=; msg-type=any
2: addFoo header-name=Foo; action=none; match-value=; msg-type=any
3: addBar header-name=Bar; action=none; match-value=; msg-type=any

selection: 2
destination: 1
Rule moved from position 2 to position 1
ACMEPACKET(sip-mime-sdp-rules)#
```

Rule Nesting and Management

There will be cases where the user wants to take a stored value from the SDP and place it in a SIP header, and vice-versa. All header-rules, element-rules, mime-rules, mime-isup-rules, isup-param-rules, mime-header-rules and mime-sdp-rules are inherited from a Manipulation Rule. A Sip Manipulation is of type Manipulation which contains a list of Manipulation Rules. Each Manipulation Rule can itself contain a list of Manipulation Rules. Therefore when configuring manipulation rules, they will be saved in the order which they have been configured. This is different from the way other configuration objects are configured. Essentially, the user has the option of configuring which type of object they want and when they are done, it gets added to the end of the sip-manipulation, such that order is preserved. This will mean that any Manipulation Rule at the same level can not share the same name. For example, names of header-rules can't be the same as any of the mime-sdp-rule ones or mime-isup-rule. This allows the user to reference stored values from one rule type in another at the same level.

ACLI Configuration Examples

The following eight sections provide sample SDP manipulations.

Remove SDP

```

sip-manipulation
  name          stripSdp
  description   remove SDP from SIP message
  mime-sdp-rule
    name        sdpStrip
    msg-type    request
    methods     INVITE
    action      delete
    comparison-type case-sensitive
    match-value
    new-value

```

Remove Video from SDP

```

sip-manipulation
  name          stripVideo
  description   strip video codecs from SIP
                message
  mime-sdp-rule
    name        stripVideo
    msg-type    request
    methods     INVITE
    action      manipulate
    comparison-type case-sensitive
    match-value
    new-value
    sdp-media-rule
      name      removeVideo
      media-type video
      action    delete
      comparison-type case-sensitive
  match-value
  new-value

```

Add SDP

```

sip-manipulation
  name          addSdp
  description   add an entire SDP if one does
                not exist
  mime-sdp-rule
    name        addSdp
    msg-type    request
    methods     INVITE
    action      add
    comparison-type case-sensitive
    match-value
    new-value   "v=0\r\no=mhandley
                2890844526 2890842807 IN IP4 "+$LOCAL_IP+"\r\ns=SDP Seminar\r\ni=A

```

```
Seminar on the session description protocol\r\nu=http:
//www.cs.ucl.ac.uk/staff/M.Handley/sdp.03.ps\r\nne=mjh@isi.edu
(Mark Handley)\r\nnc=IN IP4 "+$LOCAL_IP+"\r\nnt=2873397496
2873404696\r\nna=recvonly\r\nnm=audio 49170 RTP/AVP 0\r\n"
```

Manipulate Contacts

This rule changes the contact in the SDP to the value contained in the Contact header.

```
sip-manipulation
  name          changeSdpContact
  description   changes the contact in the SDP to
the value of the contact header
  header-rule
    name          storeContact
    header-name   Contact
    action        store
    comparison-type pattern-rule
    msg-type      request
    methods       INVITE
    match-value
    new-value
  element-rule
    name          storeHost
    parameter-name
    type          uri-host
    action        store
    match-val-type ip
    comparison-type pattern-rule
    match-value
    new-value
  mime-sdp-rule
    name          changeConnection
    msg-type      request
    methods       INVITE
    action        manipulate
    comparison-type case-sensitive
    match-value
    new-value
  sdp-session-rule
    name          changeCLine
    action        manipulate
    comparison-type case-sensitive
    match-value
    new-value
  sdp-line-rule
    name          updateConnection
    type          c
    action        replace
    comparison-type case-sensitive
    match-value   $storeContact.$storeHost
    new-value     $storeContact.$storeHost.$0
```

Remove a Codec

This rule changes the contact in the SDP to the value contained in the Contact header.

```

sip-manipulation
  name                removeCodec
  description          remove G711 codec if it exists
  mime-sdp-rule
    name              removeCodec
    msg-type          request
    methods            INVITE
    action             manipulate
    comparison-type   case-sensitive
    match-value
    new-value
  sdp-media-rule
    name              removeG711
    media-type        audio
    action             manipulate
    comparison-type   case-sensitive
    match-value
    new-value
  sdp-line-rule
    name              remove711
    type              m
    action             replace
    comparison-type   pattern-rule
    match-value       ^(audio [0-9]
                      {1,5} RTP.*)( [07]
                      \b)(.*)$
    new-value         $1+$3
  sdp-line-rule
    name              stripAttr
    type              a
    action             delete
    comparison-type   pattern-rule
    match-value       ^(rtptime|fmt):
                      [07]\b$
    new-value

```

Change Codec

```

sip-manipulation
  name                convertCodec
  description          changeG711toG729
  mime-sdp-rule
    name              changeCodec
    msg-type          request
    methods            INVITE
    action             manipulate
    comparison-type   case-sensitive
    match-value

```

```

new-value
sdp-media-rule
    name                change711to729
    media-type          audio
    action              manipulate
    comparison-type     case-sensitive
    match-value
    new-value
    sdp-line-rule
        name            change711
        type            m
        action          replace
        comparison-type pattern-rule
        match-value     ^(audio [0-9]{4,5}
                        RTP/AVP.*)( 0)(.*)$
new-value              $1+" 18"+$3
    sdp-line-rule
        name            stripAttr
        type            a
        action          delete
        comparison-type pattern-rule
        match-value     ^rtptime:0 PCMU/
                        .+$
    new-value
    sdp-line-rule
        name            addAttr
        type            a
        action          add
        comparison-type boolean
        match-value     $change711to729.
                        $stripAttr
    new-value          rtpmap:18 G729/8000

```

Remove Last Codec and Change Port

```

sip-manipulation
    name                removeLastCodec
    description         remove the last codec
    mime-sdp-rule
        name            removeLastCodec
        msg-type        request
        methods         INVITE
        action          manipulate
        comparison-type case-sensitive
        match-value
        new-value
    sdp-media-rule
        name            removeLast
        media-type      audio
        action          manipulate
        comparison-type case-sensitive
        match-value
        new-value
    sdp-line-rule

```

```

                                name                isLastCodec
                                type                m
                                action              store
                                comparison-type     pattern-rule
                                match-value        ^(audio )([0-9]{4,
new-value                        5})( RTP/AVP
                                [0-9]{1-3}))$
                                sdp-line-rule
                                name                changePort
                                type                m
                                action              replace
                                comparison-type     boolean
                                match-value        $removeLastCodec.
                                new-value          $removeLastCodec.
                                $removeLast.$isLastCodec.$1+0+$removeLastCodec.$removeLast.
                                $isLastCodec.$3

```

Remove Codec with Dynamic Payload

```

sip-manipulation
    name                removeAMR
    description         remove the AMR and AMR-WB dynamic
codecs
    mime-sdp-rule
        name            sdpAMR
        msg-type        request
        methods          INVITE
        action           manipulate
        comparison-type case-sensitive
        match-value
        new-value
    sdp-media-rule
name
    mediaAMR
    media-type          audio
    action              manipulate
    comparison-type     case-sensitive
    match-value
    new-value
    sdp-line-rule
        name            isAMR
        type            a
        action           delete
        comparison-type pattern-rule
        match-value     ^rtptime:([0-9]
                        {2,3}) AMR
        new-value
    sdp-media-rule
        name            mediaIsAMR
        media-type      audio
        action           manipulate
        comparison-type boolean
        match-value     $sdpAMR.$media

```



```

new-value
sdp-line-rule
    name          delFmtpAMR
    type          a
    action        delete
    comparison-type pattern-rule
    match-value   ^fmp:{$sdpAMR.
                $mediaAMR.
                $isAMR.$1}\b

new-value
sdp-line-rule
    name          delAMRcodec
    type          m
    action        find-replace-all
    comparison-type pattern-rule
    match-value   ^audio [0-9]+
                RTP.*( {$sdpAMR.
                $mediaAMR.$isAMR.

```

HMR Import-Export

Due to the complexity of SIP manipulations rules and the deep understanding of system syntax they require, it is often difficult to configure reliable rules. This feature provides support for importing and exporting pieces of SIP manipulation configuration in a reliable way so that they can be reused.

To Import HMRs, use the **Upload** link on the sip-manipulation list dialog, which is the first dialog displayed after clicking the HMR icon. To export, use **Download**.

Exporting

The SIP manipulation configuration contains an **export** command which sends the previously selected configuration to the designated file. The syntax is **export [FILENAME]**. The system compresses the file with gzip and writes it to the `/code/imports` directory.



Note:

SIP manipulation configurations can only be exported one at a time.

Exported data will look like this:

```

<?xml version='1.0' standalone='yes'?>
<sipManipulation
  name='manip'
  description=''
  lastModifiedBy='admin@console'
  lastModifiedDate='2009-10-16 14:16:29'>
  <headerRule
    headerName='Foo'

```

```
        msgType='any'  
        name='headerRule'  
        action='manipulate'  
        cmpType='boolean'  
        matchValue='$REGEX("[bB][A-Za-z]{2}")'  
        newValue='foo'  
        methods='INVITE'  
    </headerRule>  
</sipManipulation>
```

To avoid conflicts when importing, the key and object ID are not included as part of the exported XML.

Importing

The **import** command imports data from a previously exported file into the currently-selected configuration. If no configuration was selected, a new one is created. The syntax is **import [FILENAME]**. Include the .gz extension in the filename. After importing, type **done** to save the configuration.

Importing a configuration with the same key as one that already exists returns an error. In this case:

- Delete the object with the same key and re-import.
- Select the object with the same key and perform an import that will overwrite it with new data.

Using SFTP to Move Files

After exporting a configuration, use SFTP to copy the file to other Oracle Enterprise Communications Brokers. Place the file in the `/code/imports` directory before using the **import** command on the second OECB.