

Oracle® Communications

Diameter Signaling Router

Zero Balance Application User's Guide

Release 8.2

E88988-01

January 2018

ORACLE®

Oracle Communications Diameter Signaling Router Zero Balance Application User's Guide, Release 8.2

Copyright © 2018 Oracle and/or its affiliates. All rights reserved.

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 installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. 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 Xeon 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, Opteron, the AMD logo, and the AMD Opteron 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.

My Oracle Support (MOS) (<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 MOS 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>.

See more information on My Oracle Support (MOS) in Appendix A.

Table of Contents

1. Introduction	4
1.1 Audience	4
1.2 References.....	4
1.3 Acronyms/Terms	5
2. Understanding ZBA Functionality and ZBA Logic	6
2.1 Understand ZBA Functionality.....	6
2.2 ZBA Logic Process.....	7
3. ZBA Procedures	7
3.1 Verify ZBA Activation	7
3.2 Import ZBA JSON File.....	8
3.3 Configure ZBA Config_Data, Egress_ResultSet, and Ingress_ResultSet Tables	8
3.4 Disable ZBA	9
4. Customize ZBA Behavior.....	9
5. ZBA Functions.....	9
5.1 Custom MEALS.....	10
5.2 General Options	10
5.3 Trial MPs Assignment	10
5.4 Application Control	10
5.5 System Options (SO Only):	11
6. Use Configured ZBA Database Tables	11
Appendix A. My Oracle Support (MOS).....	11
Appendix B. Locate Product Documentation on the Oracle Help Center Site	12

List of Tables

Table 1. Acronyms/Terms	5
Table 2. ZBA Database Table Fields.....	11

1. Introduction

The Zero Balance Application (ZBA) allows network operators to reduce the load (offload) sent to the Online Charging Servers (OCS) by blocking those subscribers whose accounts have exceeded their credit limit (that is, Zero Balance). To achieve this, the ZBA detects the Credit Control Request (CCR) and (Credit Control Answer (CCA) exchanged with the OCS and prevents subscribers whose account balances have gone down to zero from repeatedly querying the OCS. The ZBA maintains records of subscribers, identified by their MSISDN, whose account balance has gone down to zero.

This document provides information about the ZBA and its functionalities. The ZBA menu options allow you to work with:

- Custom MEALS
- General Options
- Trial MPs Assignment
- Application Control
- System Options (SO only)

ZBA is available as a Diameter Custom Application (DCA) framework application. Like other DCA framework applications, you can use ZBA to work with the DCA framework functions. If ZBA is in the DCA framework GUI menu, the application has already been activated.

The document provides the following types of information

- ZBA tasks:
- ZBA application logic
- Procedures to configure and manage ZBA components including configuration parameters and ZBA profile tables
- ZBA components and GUI elements

1.1 Audience

This content is intended for personnel using the ZBA within the DCA framework. The content includes procedures for performing tasks using the product GUI, but does not describe how to install or replace software or hardware.

The ZBA software interacts with U-SBR (Universal Session Binding Repository). For this reason, this content includes references to the shared applications, and may describe GUI options that are not visible or applicable to ZBA.

1.2 References

The following references capture the source material used to create this document. These documents are included in the Oracle Communications Diameter Signaling Router (DSR) documentation set. See Appendix B Locate Product Documentation on the Oracle Help Center Site.

- [1] DCA Programmer's Guide
- [2] DCA Provisioning Interface
- [3] DCA Feature Activation
- [4] Diameter User's Guide

1.3 Acronyms/Terms

An alphabetized list of acronyms/terms used in the document.

Table 1. Acronyms/Terms

Acronym/Term	Definition
API	
ART	Application Routing Table
CCA	Credit Control Answer
CCA-I/E	Credit Control Answer (Initial/Event)
CCR	Credit Control Request
CCR-I/E	Credit Control Request (Initial/Event)
CTF	Charging Triggering Function
DCA	Diameter Custom Application
DET	Zero Balance Detection Interval (configured)
DRA	Diameter Routing Agent
DSR	Diameter Signaling Router
ECC	Extended Command Code
GUI	Graphic User Interface
MEALS	
MMI	Machine to Machine Interface
MP	
OC-DRA	Online Charging DRA
OCS	Online Charging Server
PCEF	Policy and Charging Enforcement Function
SBR	Session Binding Repository
SO	System Options
U-SBR	Universal SBR
ZBA	Zero Balance Application
ZBC	Zero Balance Counter
ZB-DB	Zero Balance Application Database
ZBT	Zero Balance Blocking Timer
ZDT	Zero Balance Detection Timer

2. Understanding ZBA Functionality and ZBA Logic

Zero Balance Application (ZBA) is a business logic application that function within the DCA framework. The DCA frame must first be active before activating the ZBA to access the ZBA GUI menu functionality.

Note: The DCA framework is a set of APIs and services made available to DCA developers (internal or external) who need to development application that run in the Diameter application layer of the DSR.

When a mobile data service subscriber initiates a data session, a network Element (PCEF or CTF) may attempt to set up a data session for the subscriber and query the OCS for service units. In the case where the subscriber has no credit in their account, the OCS rejects the request, usually with an indication that the subscriber has no credit. However, in most cases the PCEF, or other CTFs (Charging Trigger Functions), continues to attempt to query the OCS. This can create an overload on the OCS to handle requests for subscribers that are always denied during the period in which the subscriber has no credit.

The purpose of the ZBA is to detect when a subscriber, identified by a MSISDN, has a zero balance in his/her account, the ZBA under certain configurable conditions, may respond on behalf of the OCS and reduce the number of queries the OCS must handle for subscribers who temporarily have no credit.

The ZBA is configured as the owner of a U-SBR database. To avoid overloading the ZBA, the Application Routing Table (ART) is configured in such a way so as to route only CCR- I/E type messages to the ZBA.

The ZBA can be enabled and disabled as a DCA framework. Disabling ZBA on a specific site is possible only if ZBA has been disabled on all the DA-MPs on that specific site.

If ZBA is in the DCA framework GUI menu, this means the application is already enabled, but does not guarantee that it is provisioned. After ZBA is installed, a manual enablement is necessary on at least one DA-MP server.

Note: You can enable/disable ZBA from the **Diameter > Maintenance > Applications** screen on the SOAM GUI. When the application is initially installed, it is disabled and it must be manually enabled.

ZBA functionality varies between the SOAM and NOAM. On the NOAM, you have four menus available:

- The **Custom MEALs** menu where you provision the measurements linked to ZBA application
- The **General Options** to set the options of the application.
- The **Trial MP Assignment** to choose on which MP the application will be running.
- The **Application Control** where you can create new development environment and modify them.

The SOAM has the same menu in read only and a fifth one called **System Options** where the behavior of the application in case of errors can be configured.

2.1 Understand ZBA Functionality

The ZBA is able to inspect ingress Gy/Ro Request and Answer messages to verify if subscribers who may have zero balance accounts should be blocked (based on answer messages). It uses MSISDN to check the subscriber's status in SBR.

It forwards the ingress Gy/Ro Request messages to the OCSs if the subscriber's status is not in "blocking" state.

Subscribers may have zero balance, but their messages may still be routed to OCS if the number of Zero Balance events detected is less than a configured threshold during a configured detection interval.

The ZBA blocks subscribers whose number of Zero Balance events has exceeded the configured threshold within the configured detection interval and generates a response message (CCA-I/E) on behalf of the OCS when a subscriber is blocked.

2.2 ZBA Logic Process

ZBA logic follows this process:

1. The ZBA logic is triggered with the reception of a CCR-I/E message.
2. The ZB reads the MSISDN from the Subscription-ID Grouped or the User-Name AVPs in the CCR-I/E message.
3. ZBA checks if the MSISDN extracted from the AVP already has a record entry in the SBR.
4. If no entry exists for that user, the application forwards the CCR-I/E to OCS using DRL and creates a new entry in the SBR. If a qualified ZB event has been received in the CCA-I/E, that subscriber is then monitored. If the OCS answer contains a result code, which does indicate zero balance, then the Zero Balance counter is incremented in the SBR and detection state is set to ON. ZBA forwards the CCA-I/E received from the OCS using DRL.
5. If an entry for that user exists, the application reads the status of the blocking and detection timers from the SBR to determine whether the subscriber is blocked or being monitored.
6. If the subscriber is not blocked, or blocked with expired blocking timer, ZBA forwards the CCR-I/E to OCS using DRL and resets the record to its initial state (for example, not monitored, not blocked).
7. If a request comes in when the subscriber is blocked and not expired (blocking timer is ON, but not expired), the request is not forwarded to OCS, and instead the ZBA sends an answer message with the configurable egress blocking result code to CTF using DRL (the OCS is not involved).
8. ZBA continues to monitor the subscriber and answer on behalf of the OCS until the blocking timer expires.
9. ZBA does not process the Diameter messages in the following cases:
 - missing MSISDN in the CCR-I/E
 - ZBA U-SBR Database is unavailable
 - Stack event errors have been generated during processing

3. ZBA Procedures

This section describes the procedures used to configure the ZBA application

Before you can configure ZBA as a DCA framework, DCA framework and ZBA must be activated.

Note: Following activation, individual applications are in the disabled state. While disabled, no diameter traffic is delivered to an application.

To enable ZBA as an independent application, enable ZBA from the **Diameter > Maintenance > Applications** screen on the SO on the desired DA-MPs. See [4] Diameter User's Guide for details.

3.1 Verify ZBA Activation

To verify if ZBA is activated, navigate to the DCA Framework menu. The Zero Balance Application should display on the screen. You can then use **DCA Framework > Zero Balance Application > Application Control** screen to enable business logic and provision configuration data.

3.2 Import ZBA JSON File

Once ZBA is activated, import the business logic, configuration tables, and MEALs from a previously exported json file.

1. From the NOAM, navigate to **DCA Framework > Zero Balance Application > Application Control**.
2. Click **Import Business Logic** to import the json file.
3. Select the file.
4. Mark the **import also config data** checkbox
5. Click **Import**.

This creates the configuration data, the custom MEAL templates, and maps the logical and physical U-SBR database (if a physical database with the same name already exists). To verify or create the mapping, navigate to **DCA Framework > Zero Balance Application > Application Control > Zero Balance > SBR DB Name Mapping**.

Once imported, the application still needs to be syntax checked and compiled before it can go to the Production cycle.

3.3 Configure ZBA Config_Data, Egress_ResultCode, and Ingress_ResultCode Tables

ZBA uses three tables to collect configuration values;

- Config_Data
- Egress_ResultCode
- Ingress_ResultCode

See Chapter 6 Use Configured ZBA Database Tables.

This procedure configures the Config_Data, Egress_ResultCode, and Ingress_ResultCode tables. The Config_Data table contains the configurable parameters that determine the detection period, the Blocking period, and the Max Zero Balance allowable Events. The Egress_ResultCode table contains the configurable result code used when ZBA generates an answer during a blocking period. The Ingress_ResultCode table contains the configurable result codes used to determine a Zero Balance event when an answer message contains one of the configured result codes.

Note: The available GUI choices differ between the SO and the NO. See [1] DCA Programmer's Guide for GUI details and functionality.

The fields are described in the associated database tables elements.

1. On the SOAM, navigate to **DCA Framework > Zero Balance Application > Application Control**.
2. Select a **Version Name**.
3. Click **Config Data**.
4. Select a Table Name.
5. Select an action, for example, View to view data, Provision Table to populate it.
6. Enter the fields to define or edit the selected table.

3.4 Disable ZBA

This procedure disable ZBA from the SO.

Note: The available GUI choices differ between the SO and the NO. See [1] DCA Programmer's Guide for GUI details and functionality.

1. Navigate to **Diameter > Maintenance > Applications**.
2. Select the DCA_ZBA on the MP where you want to disable the application.
3. Click **Disable**.

4. Customize ZBA Behavior

ZBA can be customize using the **DCA Framework > Zero Balance Application** screen or as an independent application on these DSR menus:

- Custom MEALS
- General Options
- Trial MPs Assignment
- Application Control
- System Options (SO only)

Note: See [1] DCA Programmer's Guide and [2] DCA Provisioning Interface for details. See Chapter 6 Use Configured ZBA Database Tables for specific information about configuration activities.

The ZBA configuration database schema uses the following options to customize the application behavior:

- Detection Interval:** The time interval during which Zero Balance events are Monitored. Range: 1-30. 30. Default is 5 (values are in seconds).
- Blocking Interval:** The time duration during which a subscriber is blocked after exceeding the number of allowed zero balance events. Range: 60-1800. Default is 60 (values in seconds).
- Maximum ZB Events:** The number of allowed zero events during a detection interval before blocking a subscriber. Range: 1-5. Default is 1 (value is integer).
- Egress Result Code:** The configurable egress result code used when ZBA generates an answer during a blocking period (value is string).
- Ingress Result Code:** One to four ingress result codes can be configured to determine a Zero Balance event when an answer message contains one of these result codes (values are string).

5. ZBA Functions

Like other DCA framework applications, you can use ZBA to work with the DCA framework functions. If ZBA is on the DCA framework screen, the application is already activated, but does not guarantee that it is provisioned. After ZBA is installed, enable it manually and complete the configuration items, such as DB association, Config_Data parameters, Diameter connections, ART, etc.

See Chapter 6 Use Configured ZBA Database Tables for specific information about configuration activities.

ZBA lets you work with these functions:

5.1 Custom MEALS

A list of any previously configured ZBA custom defined measurements and events (NO view GUI page). Use this screen to insert, edit, and delete custom MEALS.

5.2 General Options

Available from the NO and SO (view only), general options allows you to configure ZBA behavior.

From the NO, this screen enables specifying the Perl Subroutine for Diameter Request and Answer.

From the SO, this screen allows you to view the Perl Subroutine specified for Diameter Request and Answer in read-only mode.

5.3 Trial MPs Assignment

From the NO, this screen allows specifying which MPs run the trial version of an application. If there is a production version, all MPs enabled for the ZBA application run the application regardless of which is the trial MP. If there is a trial application version configured in the system, but no trial MPs are specified, a warning message is generated.

From the SO, this screen allows you to view which MPs run the trial version of an application. The page is accessible in read-only mode.

5.4 Application Control

From the NO, this page allows:

- Listing all application versions configured in the system
- Inserting a new application version
- Copying and modifying an existing application version
- Exporting an application version entirely (business logic + provisioned data from the NO)
- Exporting only the NO provisioned data of an application version
- Importing a previously exported application version (business logic + NO provisioned data)
- Importing only the NO provisioned data to an existing application version
- Accessing the application version configuration tables
- Accessing a flowchart of an application version
- Deleting an existing application version
- Changing the status of an application version (Development, Trial, Production, Archived)

From the SO, this page allows:

- Listing all application versions configured in the system
- Exporting only the SO provisioned data of an application version
- Importing only the SO provisioned data to an existing application version
- Accessing the application version configuration tables
- Accessing a flowchart of an application version (read-only)

You can control a variety of ZBA functions from the Application Control screen, such as creating new application versions; copying and editing existing versions; importing and exporting business logic; and

making trials. You can also copy an entire application with the business logic and the NO provisioned data and assign a new version name to it.

5.5 System Options (SO Only):

Use this task to work with ZBA Systems Options, which includes the following categories:

- **Run-time error configuration:** Behavior of the DSR application when a Run-time error occurs
Resource exhaustion configuration
- **Resource exhaustion configuration:** Behavior of the DSR application when internal resources are being exhausted
- **Application unavailable configuration:** Action to be taken when the application is unavailable to process messages.
- **Configuration for the DCA generated Answer:** To specify a realm and FQDN different from the default one.

6. Use Configured ZBA Database Tables

This section describes ZBA application database table content and functionality.

Navigate to **DCA Framework > Zero Balance Application > Application Control**, select the current version of the ZBA, and click **Config Data** to access the configuration table page.

Table 2. ZBA Database Table Fields

Field	Description	Single Row Setting	Level
Config_Data	Contains configuration parameters for the ZBA	YES	SO
Egress_StatusCode	Contains the configurable result code used when ZBA generates an answer during a blocking period	YES	SO
Ingress_StatusCode	Contains one to four ingress result codes that can be configured to determine a Zero Balance event when an answer message contains one of these result codes	YES	SO

Appendix A. My Oracle Support (MOS)

MOS (<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 MOS 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 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 MOS, select **2**.

You are connected to a live agent who can assist you with MOS registration and opening a support ticket. MOS is available 24 hours a day, 7 days a week, 365 days a year.

Appendix B. 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 subheading, click the **Oracle Communications documentation** link.

The Communications Documentation page displays. Most products covered by these documentation sets display under the headings **Network Session Delivery and Control Infrastructure** or **Platforms**.

4. Click on your product and then the release number.

A list of the entire documentation set for the selected product and release displays.

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.