Oracle Financial Services Analytical Applications

Configuration for High Availability (HA) Best Practices Guide

Release 8.1.x

August 2021

ORACLE
Financial Services

ORACLE

OFS Analytical Applications Configuration for High Availability (HA) Best Practices Guide

Document Control

Version Number	Revision Date	Change Log
1.2	November 2022	Updated the OFSAA deployment architecture diagram.
1.1	August 2021	Updated the Configure the OFSAA Instance section (Doc 31820287).
1.0	May 2020	Created the document for best practices in the OFSAA 8.1.x release HA process.

Table of Contents

1 Preface

This preface provides information for the Oracle Financial Services Analytical Applications (OFSAA) Configurations for High Availability Best Practices Guide.

Topics:

- Access to Oracle Support
- Audience
- Additional Resources
- Conventions Used

1.1 Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit:

- http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info
- http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

1.2 Audience

This document is intended for the system administrators and users using Oracle Financial Services Analytical Applications (OFSAA) Configurations for High Availability Best Practices.

1.3 Additional Resources

This section identifies additional resources for the Oracle Financial Services Analytical Applications (OFSAA) Configurations for High Availability Best Practices. You can access the online documentation for the OFSAA 8.1.0.0.0 from the Oracle Help Center (OHC).

- OFS Advanced Analytical Applications Infrastructure (OFS AAAI) Application Pack Installation and Configuration Guide
- OFS Analytical Applications Infrastructure Administration Guide

To find additional information about how Oracle Financial Services solves real business problems, see our website at www.oracle.com/financialservices.

1.4 Conventions Used

The following table lists the conventions used in this guide.

Table 1: Conventions Used in this Guide

Convention	Meaning
Italics	• Names of books, chapters, and sections as references • Emphasis
Bold	• The object of an action (menu names, field names, options, button names) in a step-by-step procedure • Commands typed at a prompt • User input
`Monospace`	• Directories and subdirectories • File names and extensions • Process names • Code sample, including keywords and variables within the text and as separate paragraphs, and user-defined program elements within the text
<Variable>	Substitute input value

2 Introduction

A High Availability (HA) architecture is one of the key requirements for any Enterprise Deployment. It refers to the ability of users to access a system without loss of service. Deploying a High Availability system minimizes the time when the system is down or unavailable and maximizes the time when it is running or available. This section provides an overview of high availability from a problem-solution perspective.

High Availability (HA) preparation is an integral part of contingency planning. This document serves as a reference document for the preparation of specific High Availability (HA) architecture. It explains how a standard OFSAA deployment should be architected to protect its applications from unplanned downtime and minimize planned downtime.

Topics:

- Objective
- Assumptions
- Exclusions or Limitations
- Approach

2.1 Objective

The objective of this document is to establish a process to configure OFSAA instance deployment for High Availability (HA).

NOTE This document does not apply to set up a Disaster Recovery (DR) instance. It should be used to ensure service continuity through the maintenance of an additional instance.

2.2 Assumptions

This document is prepared after considering the below assumptions:

1. A Load Balancer (software or hardware) is identified and installed.
2. An appropriate backup strategy for OFSAA File System (`$FIC_HOME` and `FTPSHARE`) and Oracle Database (or Databases) is (or are) already installed and configured.
3. Installation of the OFSAA platform and applications on the primary node is completed and set up is working.
4. A secondary instance (node) for OFSAA is identified and configured with appropriate prerequisite software. No installation of OFSAA products is required at this stage.
5. Hardware configurations (in terms of RAM, CPU, and CORE) do not vary between the OFSAA primary and secondary nodes.

6. It is also mandatory that the file system references such as the OS mount and directories, web application server profiles, domains, deployed paths, and so on are the same between the primary and secondary nodes.

2.3 Exclusions or Limitations

1. The OFSAA instance (or instances) configuration is in ACTIVE-PASSIVE mode. Due to the architectural limitations of the OFSAA platform, the OFSAA components (processing layer) cannot be configured for ACTIVE-ACTIVE mode. However, the web and database tiers can be configured for ACTIVE-ACTIVE mode.

> **NOTE** Though OFSAA instance (or instances) configuration is in ACTIVE-PASSIVE mode, OFSAA allows Distributed Activation Manager (AM) based Batch Processing from v8.0.5.0.0 onwards, to configure AM engines to run on multiple OFSAA nodes. For more information, see Distributed Activation Manager (AM) based Batch Processing section in the OFS Analytical Applications Infrastructure Administration Guide.

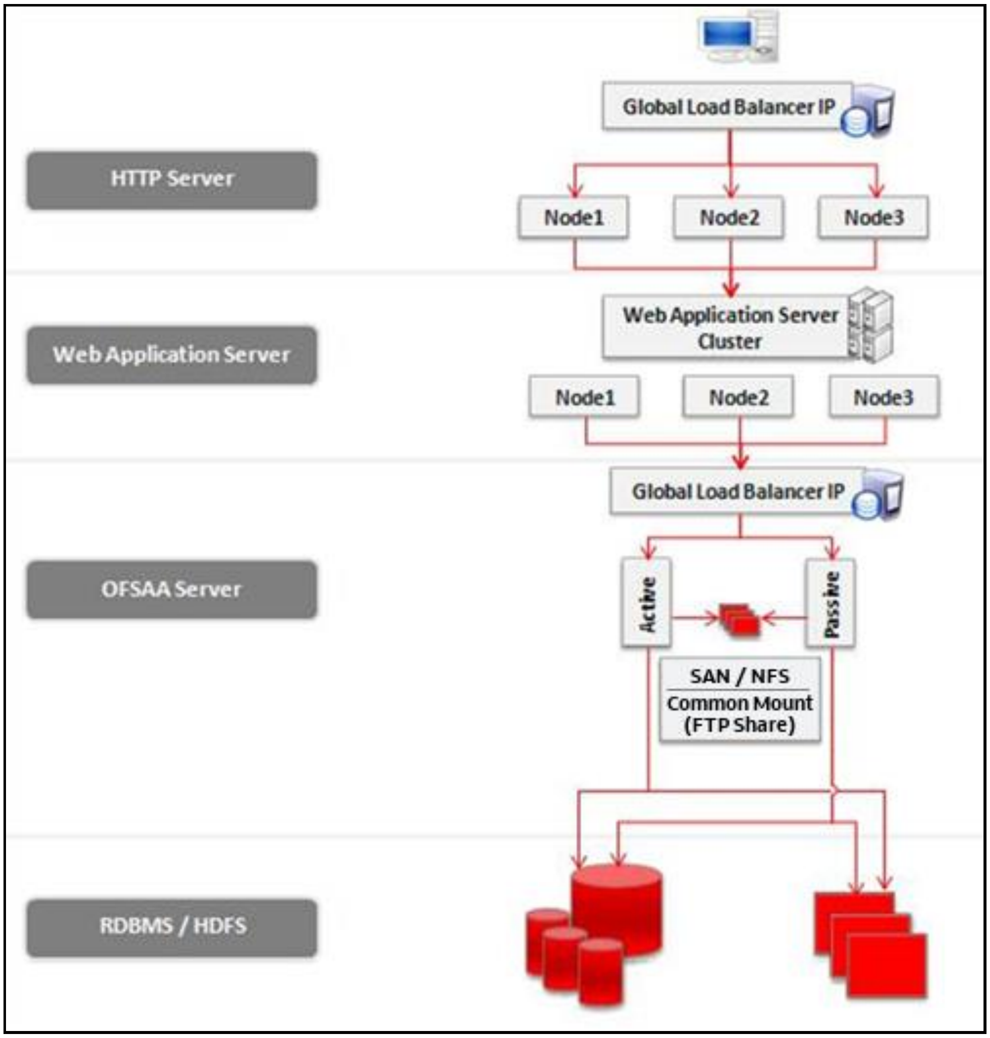
2. This document does not consider any particular OFSAA Application specific configuration. It documents the generic configuration across the platform that is generally applicable for the application stack deployed on top of it.
3. This document does not consider the reporting layer HA configuration. For example, the OBIEE server.
4. This document considers HA configuration only against Oracle WebLogic Server and (or) IBM WebSphere Application Server.

2.4 Approach

There are many ways to devise the HA architecture based on the requirements, but the following is the recommended approach (to be used as reference) to devise any further changes or modifications to the architecture as per the use cases.

Let us consider the following OFSAA deployment architecture for HA configuration as the end state.

Figure 1: OFSAA deployment architecture for HA configuration as the end state



In this illustration, the HA setup is proposed to be ACTIVE-ACTIVE configuration at the HTTP Server, Web Application Server, and Database or HDFS layers. The OFSAA layer is configured for ACTIVE-PASSIVE configuration.

NOTE

Access to OFSAA applications is provided using the Global Load Balancer IP or hostname (Virtual IP). In the event of a primary node failure, the access to the secondary node is seamless, requiring no changes to the configuration information across all tiers.

Session Affinity or Sticky Session is configured at the HTTP Server level.

At any time, OFSAA patch installations should be performed only on an active node. Promotions of patches to a passive node are taken care of as part of the sync-up process for File System components.

HA configuration for HDFS should be created after referring to the HDFS vendor-specific documentation. This document does not describe any details about HA configuration for HDFS.

3 Steps for HA Configuration

The following topics provide the instructions for HA configuration in OFSAA.

Topics:

- Assumptions
- Configure the OFSAA Instance
- Configuring HTTP Load Balancer
- Cloning the OFSAA instance

3.1 Assumptions

The following are the assumptions for HA configuration in OFSAA.

- The Global Load Balancer (or Balancers) is (or are) installed and any post-installation configuration (hardening) is completed before beginning with the following steps. If no load balancer is installed, you can install and configure it on any host at this time.
- The OFSAA primary node installation was not performed keeping in mind the HA architecture. That is, multiple HTTP Servers, Web Application Server cluster nodes, DB RAC cluster nodes, common file storage (FTPSHARE), and so on are not set up.
- The OFSAA primary node installation was performed using the local IP address or Hostname.

3.2 Configure the OFSAA Instance

To configure the OFSAA instance (or instances) for HA configuration, perform the following steps:

1. Ensure that the OFSAA primary node is up and running. You can access the OFSAA applications by entering the URL in the browser and login is successful.
2. Configure HA architecture.
   a. Install at minimum one additional HTTP Server, if only one HTTP Server is installed or configured. If no HTTP Server is installed, you must install at minimum two HTTP Servers. For information on HTTP Servers, see OFS AAAI Application Pack Installation and Configuration Guide.
   b. Configure the Global Load Balancer (at OFSAA Server level, that is, processing layer).
   See the Configuring OFSAA Load Balancer section as an example for configuration of software load balancer at OFSAA Server level.
   c. Configure the Global Load Balancer (at HTTP Server level).
   See the Configuring HTTP Load Balancer section as an example for configuration of software load balancer at HTTP Server level. If this is already configured, skip and move to the next step.
   d. If the web application server is already installed as a cluster, skip and proceed with the next steps.
   Or
   Install or upgrade the Web Application Server as a cluster of nodes. Create the WebLogic Domain or WebSphere Profile as appropriate. (Make a note of the paths). Update the HTTP

Server configuration to use all web application server nodes. For more details, see Configuration for Apache HTTP Server.

e. Archive and restore the existing DB schemas to a DB RAC installation. Ensure to retain the same schema names. (Make a note of the DB RAC URL). If the database is not installed in RAC mode, you can do it now. Otherwise, skip and proceed with the next steps.

f. Create a folder (`FTPSHARE`) on the common file storage (NAS or NFS) and create a local mount point on the OFSAA server to access this folder.

g. Copy the folder contents of the current `FTPSHARE` to the newly created folder as part of Step 2.f.

h. Perform an FTP or SFTP login on to the OFSAA server from the command prompt and ensure you can access this folder's contents.

3. Log in to the OFSAA primary node and stop the OFSAA services. For information about starting or stopping OFSAA services, see OFS AAAI Application Pack Installation and Configuration Guide.

4. Perform Hostname or IP address change by following the steps documented in the *Changing IP Address or Hostname, Ports, Deployed Paths of the OFSAA Instance* section in the OFSAAI Administration Guide.

   At this time, provide the Hostname or IP address for OFSAA node as OFSAA GLIP in the `OFSAA_Server_IP_Address` property. Do not change the ports. Retain the ports to the same as setup during installation.

   In the `Web Server IP address` or `Hostname` and `Port` properties, enter the HTTP Layer GLIP and port configured (HTTP Server level).

   Additionally, update the other parameters in the file to reflect the change of parameter values for changes made (if any) as part of Steps 2.c, 2.d, 2.e, and 2.f.

   a. Edit the `/etc/hosts` file (on the OFSAA primary node) and make an entry by adding the OFSAA GLIP alias as given:

      ```
      192.0.2.1 ofss12345 glip1
      ```

   b. Edit the `web.xml` file in the `$FIC_HOME/ficweb/webroot/WEB-INF` directory (on the OFSAA node) and add an entry for **AllowHosts**.

      For more information about AllowHosts, see the OFSAA Security Guide 8.1.x.

5. Navigate to `$FIC_WEB_HOME` and execute the command:

   ```
   ./ant.sh
   ```

   This generates the OFSAA web archive (.ear or .war) file (or files). For information on generating application archives, see OFS AAAI Application Pack Installation and Configuration Guide.

6. Navigate to `$FIC_HOME/ficapp/common/FICServer/bin/` and start the OFSAA services. For information about starting or stopping of OFSAA services, see OFS AAAI Application Pack Installation and Configuration Guide.

7. Start the Web Server or Web Application Server services. Access the Admin or Deployment Console and deploy the archive (or archives) generated in Step 5.
   For information on deploying application archives, see OFS AAAI Application Pack Installation and Configuration Guide.

8. Access the OFSAA application from a browser by entering the new URL in the following format:

   ```
   <scheme>://<host>:<port>/<ofsaa-context-name>/login.jsp
   ```

   The host and port entered in the URL must be of the Global Load Balancer (at HTTP Server level).

9. Enter the user name and password and ensure you can log in and access the applications. At this point, the OFSAA primary node is ACTIVE and the secondary node is PASSIVE.

10. Stop the OFSAA services on the primary node.
    For information about starting or stopping OFSAA services, see OFS AAAI Application Pack Installation and Configuration Guide.

11. Perform the OFSAA instance cloning on the secondary node.
    For more details, see Cloning the OFSAA instance section.

12. Edit the `/etc/hosts` file (on the OFSAA secondary node) and make an entry by adding the OFSAA GLIP alias as follows:

    ```
    192.0.2.2 ofss54321 glip1
    ```

13. Start the OFSAA services on the secondary node, web servers, and web application server.
    For information on starting or stopping OFSAA services, see OFS AAAI Application Pack Installation and Configuration Guide.

14. Configure the Global Load Balancer (at OFSAA Server level) to forward requests to OFSAA secondary node only if Load Balancer used does not do this automatically.

15. Access the OFSAA application from browser by entering the URL in the following format:

    ```
    <scheme>://<host>:<port>/<ofsaa-context-name>/login.jsp
    ```

    The host and port entered in the URL must be of the Global Load Balancer (at HTTP Server level).

16. Enter the user name and password and ensure you can log in and access the applications.

    At this point, the OFSAA primary node is PASSIVE and the secondary node is ACTIVE.

    At any point in time, only one OFSAA node service should be running. If both the node services are running at the same time, routing OFSAA requests results in incorrect results.

> **NOTE**
>
> If either of the OFSAA instance (primary or secondary) goes down, ensure the folder contents for `$FIC_HOME` are synced up on the other node using a utility such as Remote Sync (rsync) before bringing up the OFSAA services.
>
> The sync-up should be a scheduled activity at regular intervals. If you wait for the sync-up until the primary node goes down, you may not be able to sync-up at a later stage.

See the OS-specific documentation on configuring Rsync.

Example of rsync command:

```
rsync -uavzP /scratch/ofsaaapp/ofsaa
ofsaauser@drsecondaryserver:/scratch/ofsaaapp/ofsaa
```

```
-u   skip files that are newer on the receiver
-a   archive mode; equals -rlptgoD (no -H,-A,-X)
   included with "-a"
   -r  recurse into directories
   -l   copy symlinks as symlinks
   -p   preserve permissions
   -t   preserve modification times
   -g   preserve group
   -o   preserve owner (super-user only)
   -D   same as --devices --specials
-v   increase verbosity
-z   compress file data during the transfer
-P   show progress during transfer
```

3.3 Configuring HTTP Load Balancer

Configure the Global Load Balancer to forward requests to the HTTP Servers using any preferred routing algorithm such as round robin. See the following configuration done using the HAProxy tool.

The following configuration was performed on HAProxy version 1.6.4.

Configure the following setting in `haproxy.cfg` file:

```
frontend ft_web
  bind <hostname>:80
  default_backend bk_web

backend bk_web
  balance roundrobin
  cookie JSESSIONID prefix nocache
  server s1 <server1>:80 check cookie s1
  server s2 <server2>:80 check cookie s2
```

3.3.1 Configuration for Oracle HTTP Server

In the load balancer, there is no requirement to enable sticky sessions (insert cookie)when the Oracle HTTP Server is front-ending Oracle WebLogic Server. You require sticky sessions if you are going directly from the load balancer to Oracle WebLogic Server, which is not the case in the topology described in this document.

For details, see the documentation available at http://docs.oracle.com/cd/E23943_01/core.1111/e12037/web_tier_config.htm#WCEDG577.

3.3.2 Configuration for Apache HTTP Server

Configure the following setting in the `httpd.conf` file:

```
ProxyPass /test balancer://mycluster stickysession=JSESSIONID
<Proxy balancer://mycluster>
BalancerMember http://<server1>:80 route=1
BalancerMember http://<server2>:80 route=2
</Proxy>
```

Alternatively, it can be set within balancer configuration using `ProxySet stickysession=JSESSIONID`:

```
<Proxy balancer://mycluster>
     BalancerMember http://<server1>:80 route=1
     BalancerMember http://<server2>:80 route=2
     ProxySet stickysession=JSESSIONID
</Proxy>
```

For details, see documentation available at http://httpd.apache.org/docs/2.2/mod/mod_proxy.html.

3.3.3 Configuration for IBM HTTP Server

Configure the following setting in the `plugin-cfg.xml` file:

```
IgnoreAffinityRequests="false"
```

For details, see the documentation available at http://www-01.ibm.com/support/knowledgecenter/SSAW57_6.1.0/com.ibm.websphere.nd.multiplatform.doc/info/ae/ae/rwsv_plugincfg.html.

3.3.4 Configuring OFSAA Load Balancer

Configure the Global Load Balancer (for OFSAA server) to forward requests to the OFSAA nodes. Modify the `haproxy.cfg` file using HAProxy tool as follows:

```
#---------------------------------------------------------------------
# Example configuration for a possible web application.  See the
# full configuration options online.
#
#   http://haproxy.1wt.eu/download/1.4/doc/configuration.txt
#
#---------------------------------------------------------------------

#---------------------------------------------------------------------
# Global settings
#---------------------------------------------------------------------
```

```
global
    # to have these messages end up in /var/log/haproxy.log you will
    # need to:
    #
    # 1) configure syslog to accept network log events.  This is done
    #    by adding the '-r' option to the SYSLOGD_OPTIONS in
    #    /etc/sysconfig/syslog
    #
    # 2) configure local2 events to go to the /var/log/haproxy.log
    #   file. A line like the following can be added to
    #   /etc/sysconfig/syslog
    #
    #    local2.*                       /var/log/haproxy.log
    #
    log         192.0.2.1 local2

    chroot      /var/lib/haproxy
    pidfile     /var/run/haproxy.pid
    maxconn     4000
    user        haproxy
    group       haproxy
    daemon

    # turn on stats unix socket
    stats socket /var/lib/haproxy/stats

#---------------------------------------------------------------------
# common defaults that all the 'listen' and 'backend' sections will
# use if not designated in their block
#---------------------------------------------------------------------
defaults
    mode                    http
    log                     global
    option                  httplog
    option                  dontlognull
    option http-server-close
#    option forwardfor       except 192.0.2.1/8
```

```
    option                  redispatch
    retries                 3
    timeout http-request    10s
    timeout queue           1m
    timeout connect         10s
    timeout client          1m
    timeout server          1m
    timeout http-keep-alive 10s
    timeout check           10s
    maxconn                 3000

## Start Entries for OFSAA JAVA port and native port. ##

frontend haproxy_in
                                mode tcp
                                option tcplog
                                bind *:9999
                                default_backend haproxy_backend1

backend haproxy_backend1
                                balance roundrobin
                                mode tcp
                                option tcplog
                                server web1 ofsaaserver1:9999 check
                                server web2 ofsaaserver2:9999 check

frontend haproxy_in1
                                mode tcp
                                option tcplog
                                bind *:6666
                                default_backend haproxy_backend2

backend haproxy_backend2
                                balance roundrobin
                                mode tcp
                                option tcplog
                                server web3 ofsaaserver1:6666 check
                                server web4 ofsaaserver2:6666 check

## End Entries for OFSAA JAVA port and native port. ##

## Start Entries for OFSAA ICC port. ##

frontend haproxy_in2
                                mode tcp
                                option tcplog
                                bind *:6507
                                default_backend haproxy_backend3

backend haproxy_backend3
                                balance roundrobin
                                mode tcp
                                option tcplog
                                server web5 ofsaaserver1:6507 check
                                server web6 ofsaaserver2:6507 check

## End Entries for OFSAA ICC port. ##
```

3.3.5 Configuring Backend Servers to Enable Distribution of Batch Tasks on Multiple AM Nodes

Append the `haproxy.cfg` file using HAProxy tool with the following configuration:

```
## Start Entries for OFSAA Router port. ##
      frontend haproxy_in3
          mode tcp
          option tcplog
          bind *:6500
          default_backend haproxy_backend4

      backend haproxy_backend4
          balance roundrobin
          mode tcp
          option tcplog
          server web7 <<routerhostname:port>> check
          #server web8 <<routerhostname:port>>  check
## End Entries for OFSAA Router port. ##

## Start Entries for OFSAA AM port. ##
      frontend haproxy_in4
          mode tcp
          option tcplog
          bind *:6505
          default_backend haproxy_backend5

      backend haproxy_backend5
          balance roundrobin
          mode tcp
          option tcplog
          server web9 <<AMhostname:port>> check
          server web10 <<AMhostname:port>> check
## End Entries for OFSAA AM port. ##

## Start Entries for OFSAA MessageServer port. ##
      frontend haproxy_in5
          mode tcp
          option tcplog
          bind *:6507
          default_backend haproxy_backend6

      backend haproxy_backend6
          balance roundrobin
          mode tcp
          option tcplog
          server web11 <<Messageserverhostname:port>> check
          #server web12 <<Messageserverhostname:port>> check
## End Entries for OFSAA MessageServer port. ##
```

Message Server should be running in all the nodes where AM servers are configured.

3.4 Cloning the OFSAA instance

To perform a short clone of the OFSAA instance, follow these steps:

1. Log in to the OFSAA primary node as a non-root user.

2. Archive the `$FIC_HOME` directory along with its sub-directories or files using the following command:

   ```
   tar –zcvf FIC_HOME.tar.gz ./FIC_HOME
   ```

3. Copy the archive in binary mode on to the OFSAA secondary node.
4. Log in to the OFSAA secondary node as a non-root user.
5. Extract the archive at appropriate locations on the OFSAA secondary node using the following command:

   ```
   tar –zxvf FIC_HOME.tar.gz
   ```

6. Grant permission 750 recursively on these directories and their contents.

   ```
   chmod -R 750 <folder name>
   ```

7. Copy the user `.profile` file contents (section added by OFSAA installation only) to the user `.profile` on secondary OFSAA instance.
8. Modify the `FIC_HOME, PATH, LIBPATH,` and any other environment variable values as appropriate.
9. Create the `FTPSHARE` directory (using the same path as in the primary node).
10. Save and execute the `.profile` file.

OFSAA Support

Raise a Service Request (SR) in My Oracle Support (MOS) for queries related to the OFSAA applications.

Send Us Your Comments

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, indicate the title and part number of the documentation along with the chapter/section/page number (if available) and contact the My Oracle Support.

Before sending us your comments, you might like to ensure that you have the latest version of the document wherein any of your concerns have already been addressed. You can access My Oracle Support site that has all the revised or recently released documents.