# Oracle® Healthcare Data Repository
# Installation Guide

Release 8.1.3

ORACLE®

# Contents

# Preface

This preface contains the following sections:

## Documentation 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.

## Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

# 1

# Software Requirements

The following software is required for Oracle Healthcare Data Repository 8.1:

- **Oracle Enterprise Linux 7.x (64 bit) or later**

- **Oracle Database 12c (12.2.0.1) and Oracle Database 19c (19.3)**
  Download from the Oracle Software Delivery Cloud at `https://edelivery.oracle.com`.

- **WebLogic Server 12.2.1.3 or 12.2.1.4 with the Coherence option**
  Download from the Oracle Software Delivery Cloud at `https://edelivery.oracle.com`.

- **JDK (Java Development Kit) 8u121 and later**
  Download from My Oracle Support. See Download and install Oracle Java Development Kit.

Download Oracle Healthcare Data Repository 8.1 from the Oracle Software Delivery Cloud at `https://edelivery.oracle.com`.

# 2
# Install Oracle Database

- Get the Oracle Database 12.2.0.1.0 and 19.3 installation guide
- Download and extract the Oracle Database software
- Set up HDR 8.1 database on a new machine
- Install Oracle Database 12.2.0.1.0 or 19.3

## Get the Oracle Database 12.2.0.1.0 and 19.3 installation guide

- For the Oracle Database 12.2.0.1.0 download the Linux installation guide at `https://docs.oracle.com/database/122/LADBI/toc.htm`.
- For the Oracle Database 19.3 download the Linux installation guide at `https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/index.html`..

## Download and extract the Oracle Database software

Download from the Oracle Software Delivery Cloud at `https://edelivery.oracle.com`.

Oracle Healthcare Data Repository 8.1 requires:

- Oracle Database Enterprise Edition
- Oracle Database Client
- Oracle Database Global Service Manager
- Oracle Database Grid Infrastructure (Required for RAC)

## Set up HDR 8.1 database on a new machine

To set up HDR 8.1 on a new database machine, you need to do the following:

1. Perform a backup of your existing HDR 7.x schemas for CTB and HCT. Export the CTB and HCT schemas to a dump.

2. Copy the dump files to new database machine.

3. Create the following tablespaces on the HDR 8.1 target database by connecting as the system user:
   - Tablespace for CTB user(CTB_TBS).
   - Tablespace for HCT user(HCT_TBS).
   - Tablespace for Context (CTX_TBS).
   - Tablespace for Indexes (IDX_TBS).

4. Create the HCT and CTB schema users on the HDR 8.1 target database by connecting as the system user.

5. Run `CREATE OR REPLACE DIRECTORY HDR_DMP_DIR as '<path>'` to create the HTB_DUMP_DIR directory on the target HDR 8.1 Oracle database where the HDR 7.x dump file is located.

6. Import the HCT objects from HDR 7.x to the target HCT schema.

7. Import the CTB objects from HDR7.x to the target CTB schema. Run the example script above with `HTB` replaced by `CTB` where it applies.

**Example 2-1    Schema dump**

```
expdp system/<password>@<SID> schemas=CTB,HCT
exclude=GRANT,USER,STATISTICS,TABLESPACE_QUOTA,DEFAULT_ROLE
directory=hdr_dmp_dir dumpfile=<dump file name>.dmp logfile=hdr_dump.log
```

**Example 2-2    Create migration tablespaces**

```
TABLESPACE <tablespace name>

        LOGGING

        DATAFILE '<data file path>/hdr_hcttb.dbf'

        SIZE <100M>

        AUTOEXTEND ON

        NEXT <100M> MAXSIZE UNLIMITED

        EXTENT MANAGEMENT LOCAL;
```

**Example 2-3    Create HCT user on HCT_TBS**

```
  ALTER USER HCT QUOTA unlimited ON IDX_TBS;

        ALTER USER HCT QUOTA unlimited ON CTX_TBS;

        GRANT UNLIMITED TABLESPACE TO HCT;

        grant CREATE INDEXTYPE to HCT;

        grant CREATE PROCEDURE to HCT;

        grant CREATE SEQUENCE to HCT;

        grant CREATE SESSION to HCT;

        grant CREATE TABLE to HCT;

        grant CREATE TYPE to HCT;

        grant CREATE VIEW to HCT;

        grant ALTER SESSION to HCT;

        grant ANALYZE ANY to HCT;

        grant EXECUTE on CTXSYS.ctx_ddl to HCT;

        grant Create Any Job to HCT;

        grant Create External Job to HCT;
```

```
            grant Create Job to HCT;

            grant Execute Any Class to HCT;

            grant Execute Any Program to HCT;

            grant Manage Scheduler to HCT;

            GRANT CREATE ANY DIRECTORY TO HCT;
```

**Example 2-4    Create CTB user on CTB_TBS**

```
 ALTER USER CTB QUOTA unlimited ON IDX_TBS;        GRANT UNLIMITED TABLESPACE TO
CTB;          grant CREATE INDEXTYPE to CTB;         grant CREATE PROCEDURE to
CTB;          grant CREATE SEQUENCE to CTB;        grant CREATE SESSION to CTB;
grant CREATE TABLE to CTB;        grant CREATE TYPE to CTB;         grant CREATE VIEW
to CTB;         grant CREATE SYNONYM to CTB;        grant ALTER SESSION to CTB;
grant ANALYZE ANY to CTB;         GRANT EXECUTE ON SYS.DBMS_AQIN TO CTB;         GRANT
EXECUTE ON SYS.DBMS_AQADM TO CTB;         grant Create Any Job to CTB;         grant
Create External Job to CTB;         grant Create Job to CTB;         grant Execute Any
Class to CTB;         grant Execute Any Program to CTB;         grant Manage Scheduler
to CTB;         grant create any directory to CTB;         GRANT CREATE ANY DIRECTORY TO
CTB;
```

**Example 2-5    Import HCT objectsto the target schema**

```
impdp system/<password> exclude=GRANT,USER,STATISTICS,TABLESPACE_QUOTA,DEFAULT_ROLE
schemas=hct directory=HDR_DMP_DIR  dumpfile=< HDR7.x dump file name >.dmp
logfile=hdr_hct_import.log.
```

# Install Oracle Database 12.2.0.1.0 or 19.3

Follow instructions in the *Oracle Database Installation Guide*, making selections appropriate for Oracle Healthcare Data Repository 8.1 as noted in the following sections.

You can configure the database as part of the database software installation or after, using the Database Configuration Assistant (DBCA). Oracle Healthcare Data Repository 8.1 supports installation on either:

- a Container Database (CDB) containing a Pluggable Database (PDB)
- a non-CDB database

For an explanation of which options require an additional license, see the *Database Licensing Information User Manual* at https://docs.oracle.com/database/122/DBLIC/toc.htm or https://docs.oracle.com/en/database/oracle/oracle-data-access-components/19.3/livsc/index.html.

## Database software options

During installation of the database software (called "binaries" in the installation guide) the following selections are supported or required with Oracle Healthcare Data Repository 8.1:

- Oracle Partitioning (Required)
- Oracle Real Application Clusters (RAC) (Recommended)
- Oracle Advanced Compression (Recommended)

> **Note:**
>
> Refer to the *Oracle Database Installation Guide* to learn how to configure advanced options that are applicable to the HDR installation process.

# Database configuration options

When you configure the database, select the following options:

- Character Set
  - ALT32UTF8 is recommended.
- Automatic Memory Management
- Oracle JVM
- Oracle Text
- Oracle Database Examples

> **Note:**
>
> Oracle Text, Java Virtual Machine (JVM), and Oracle XML DB are configured automatically if you select a general purpose template.

# 3

# Install the middle tier technology stack

- Download and install Oracle Java Development Kit
- Install WebLogic Server and Coherence
- Configure WebLogic Server

Oracle recommends installing the middle tier on a separate server from Oracle Database in a production environment.

## Download and install Oracle Java Development Kit

1. Go to `http://www.oracle.com/technetwork/java/javase/downloads/index-jsp-138363.html#javasejd`.

2. Scroll down to Java SE 8u*x*.

3. Click the **Download** button for **JDK**.

4. Click the link for Linux x64.

5. Install JDK following instructions in the readme.

## Install WebLogic Server and Coherence

Install WebLogic Server following instructions in the *Fusion Middleware Installing and Configuring Oracle WebLogic Server and Coherence* at

`https://docs.oracle.com/en/middleware/lifecycle/12.2.1.3/wlsig/toc.htm` or `https://docs.oracle.com/en/middleware/fusion-middleware/12.2.1.4/wlsig/toc.htm`.

## Configure WebLogic Server

To support HDR components, do the tasks in the following sections.

### Deselect Wrap Data Types for the HDR data source

Once the HDR installation is complete, perform the following steps from the WebLogic Server admin console:

1. Navigate to **DataSources**, then **HdrDataSource**, then **Connection Pool**.

2. In the Connection Pool page, scroll down, click on **Advanced** and deselect the **Wrap Data Types** checkbox.

3. Click **Save**.

# 4

# Installing Oracle Healthcare Data Repository

- Set environment variables
- Install the HDR database and middle tier using the Oracle Universal Installer
- Start WebLogic servers
- How to uninstall HDR (if required)

## Set environment variables

Set the following environment variables on the machine where you will run the HDR installer:

- **JAVA_HOME** - JDK install directory (for example, `C:/Java/jdk1.8.0_121`)
- **ORACLE_HOME** - Oracle database home directory
- **ORACLE_HOME_LISTENER** - Oracle database listener home directory
- **PATH** - `$ORACLE_HOME/bin :$PATH`
- **HDR_HOME** - Directory where HDR will be installed.
- **WL_HOME** - WebLogic home directory
- **ETS_HOME**- It should be `<HDR_HOME>/hdr_db/ets` if you run it on the database tier and `<HDR_HOME>/weblogic/hdr/ets` if you run it on the middle tier.

## Install the HDR database and middle tier using the Oracle Universal Installer

This section explains how to install the HDR application using the Oracle Universal Installer found at HDR_Installer_Linux64.zip.

- "Create oraInventory directory and file (optional)"
- "Run the HDR database tier Installer"
- "Run the HDR middle tier Installer"

## Create oraInventory directory and file (optional)

If the HDR Installer cannot reuse the default oraInventory location or if you prefer to have a custom oraInventory location, you can:

1. Create an `oraInventory` directory in your preferred location.

2. Create a file named oraInst.loc and put it in the oraInventory directory.
   The contents of the oraInst.loc file must be:

   ```
   inventory_loc=<full_path_of_the_new_oraInventory_location>
   inst_group=<OS_user_group_name_that_owns_the_new_oraInventory_location>
   ```

3. When you run the HDR Installer, use the following command instead of the one specified in Run the HDR database tier Installer.

```
Disk1$ ./install/runInstaller -invPtrLoc <full_path_to_the_new_ oraInst.loc
file>
```

# Run the HDR database tier Installer

The Installer creates the HDR, ETS and HDR_CONFIG schemas and database objects in the HDR, ETS and HDR_CONFIG schemas.

1. Copy HDR_Installer_Linux64.zip folders to the database server machine.

2. Extract the files from HDR_Installer_Linux64.zip to a directory such as /home/hdrinstaller. This location name is not required but it is used in the following step.

3. Navigate to the /hdrinstaller/Disk1 directory using the following command:
   `cd /hdrinstaller/Disk1`

4. Execute the following script:
   `./runInstaller.sh`

   Enter values in the Installer screens. See the following table for details.

**Table 4-1    HDR Oracle Universal Installer screens for the database tier**

| Screen | Details |
| --- | --- |
| Welcome | Click **Next**. |
| Select a Product to Install | Select the database tier and click **Next**. |
| Specify Home Details | **a.** **Name**: Enter the name of the HDR Home. |
| | **b.** **Path**: Browse to the HDR home. This is the location where the HDR product artifacts are copied. For example: `/u01/hdr/product`. |
| | By default, the Oracle Database home directory path is displayed. Be sure to change the values to the HDR home and its path. |
| Database Details | Enter values as follows: |
| | • **Database Host Name**: Enter the host name of the database server. |
| | • **Port**: Enter the Oracle database port number. |
| | • **Database Service Name**: Enter the Oracle database service name. |
| | • **DBA user name**: Enter the Oracle system user name. |
| | • **Password**: Enter the system user password. |
| Choose if want to configure per domain tablespaces | Select an option for the number of tablespaces to configure: |
| | • **Yes**. This setting is recommended. It creates different tablespaces for different clinical domains so that storage management becomes easier as the system accumulates large volumes of data over time. |
| | • **No**. This option is suitable for HDR installations where the data volume is expecte to be small to medium (a few terabytes). Fewer tablespaces are created. |

**Table 4-1    (Cont.) HDR Oracle Universal Installer screens for the database tier**

| Screen | Details |
| --- | --- |
| HDR Current Data Tablespace | If you are upgrading from an earlier version of HTB/HDR, enter the names of the tablespaces where the latest versions of records are stored for the following types of data. The default tablespace names are displayed. Change them if you use different names:<br><br>• **Clinical data tablespace**: `hdr_cln_tbs`<br>• **Administrative data tablespace**: `hdr_admin_tbs`<br>• **Financial data tablespace**: `hdr_fin_tbs`<br>• **Infrastructure data tablespace**: `hdr_infra_tbs`<br>• **Identification data tablespace**: `hdr_ident_tbs` |
| HDR Current Index Tablespace | If you are upgrading from an earlier version of HTB/HDR, enter the names of the tablespaces where indexes for the latest versions of records are stored. The default tablespace names are displayed. Change them if you use different names:<br><br>• **Clinical index tablespace**: `hdr_cln_idx_tbs`<br>• **Administrative index tablespace**: `hdr_admin_idx_tbs`<br>• **Financial index tablespace**: `hdr_fin_idx_tbs`<br>• **Infrastructure index tablespace**: `hdr_infra_idx_tbs`<br>• **Identification index tablespace**: `hdr_ident_idx_tbs` |
| HDR Historical Data Tablespace | If you are upgrading from an earlier version of HTB/HDR, enter the names of the tablespaces where indexes for the historical versions of records are stored. The default tablespace names are displayed. Change them if you use different names:<br><br>• **Clinical index tablespace**: `hdr_cln_hist_idx_tbs`<br>• **Administrative index tablespace**: `hdr_admin_hist_idx_tbs`<br>• **Financial index tablespace**: `hdr_fin_hist_idx_tbs`<br>• **Infrastructure index tablespace**: `hdr_infra_hist_idx_tbs`<br>• **Identification index tablespace**: `hdr_ident_hist_idx_tbs` |
| Enter Password for HDR schema | Enter and confirm the password for the HDR database user account. |
| Choose Password for config schema | Enter and confirm the password for the HDR_CONFIG user account. All static configuration data and seed data of the HDR product are stored in this schema. |

**Table 4-1    (Cont.) HDR Oracle Universal Installer screens for the database tier**

| Screen | Details |
| --- | --- |
| ETS Schema Details | Enter Enterprise Terminology Services details. Default values are displayed for some fields.<br><br>• **ETS user password**. Enter a password.<br>• **Confirm password**. Reenter the password.<br>• **ETS default tablespace**: ets_data_tbs. Enter the tablespace name where terminology data is stored.<br>• **ETS context tablespace**: ets_ctx_tbs<br>• **ETS index tablespace**: ets_idx_tbs. Enter the tablespace name where terminology indexes are stored.<br>• **ETS Language**: ENUS. Enter the ETS base language code in the form of <Languagecode><CountryCode>. |
| Choose Directory for Java Home | Enter or browse to the Java Home location for the ETS Java Scheduler Job programs to run in. |
| Summary | Review the settings, then click **Install**.<br><br>Any errors during the installation are logged in the files under oraInventory/logs folder. Absolute path of the file will be displayed on the OUI screen. |
| End of Installation | Read the message and click **Exit**. |

# Run the HDR middle tier Installer

To install the HDR middle tier, perform the following:

1. Copy HDR_Installer_Linux64.zip to the WebLogic server machine.

2. Extract the files from HDR_Installer_Linux64.zip to a directory such as /home/hdrinstaller. This location name is not required but it is used in the following step.

3. Navigate to the hdrinstaller directory using the following command:
   ```
   cd hdrinstaller/Disk1
   ```

4. Execute the following script:
   ```
   ./runInstaller.sh
   ```

   Enter values as required in the Installer screens.

**Table 4-2    HDR Oracle Universal Installer screens for the middle tier**

| Screen | Details |
| --- | --- |
| Welcome | Click **Next**. |
| Select a Product to Install | Select the WebLogic tier and click **Next**. |

**Table 4-2    (Cont.) HDR Oracle Universal Installer screens for the middle tier**

| Screen | Details |
| --- | --- |
| Specify Home Details | 1. **Name**: Enter the name of the HDR Home. |
| | 2. **Path**: Browse to the HDR home. This is the location where the HDR product artifacts are copied. For example: `/u01/hdr/product`. By default, the Oracle Database home directory path is displayed. Be sure to change the values to the HDR home and its path. |
| | You can choose the same machine for both the middle tier and the database tier, but Oracle recommends installing them on two different servers in a production environment. |
| Choose WebLogic Home Directory | **WebLogic home folder**: Enter the path to the WebLogic home. |
| HDR Domain Properties | Enter values as follows: |
| | • **Domain name**: Enter the WebLogic domain name under which the HDR application has to be deployed. By default this is `hdr_domain`. |
| | • **Domain admin user**: Enter the WebLogic domain admin user name.By default this is `weblogic`. |
| | • **Domain admin password**: Enter the WebLogic domain admin password. |
| | • **Confirm domain admin password**: Re-enter the WebLogic domain admin password. |
| | • **Admin server listen port**: Enter the admin port number of the HDR domain. |
| | • **Admin server SSL listen port**: Enter the SSL port number. |
| HDR Managed Server Properties | Enter values as follows: |
| | • **Server name**: Enter the HDR Managed Server name. |
| | • **Listen port**: Enter the port number on which the HDR Managed Server can listen. |
| | • **SSL Listen port**: Enter the SSL listen port number. |
| | • **HDR Deployment name**: Enter the HDR application name (usually HDR). |
| | • FHIR Deployment name: Enter the FHIR application name (usually FHIR). |
| IHE XDS User Password | Enter the IHE XDS user password, then re-enter it to confirm. |
| Data Source Properties | Enter values as follows: |
| | • **Database host**: Enter the host name or IP address of the Oracle database server where HDR is installed. |
| | • **Database port**: Enter the database port number. |
| | • **Database service name**: Enter the database service name. |
| | • **HDR user's password**: Enter the HDR schema user password. |
| | • **ETS user's password**: Enter the ETS schema user password. |

**Table 4-2    (Cont.) HDR Oracle Universal Installer screens for the middle tier**

| Screen | Details |
| --- | --- |
| OIDs Configuration | Would you like to configure OIDS?<br>• Select **Yes** if this is a fresh installation of the HDR database. In that case, the OID Values screen appears and you must enter values for **Internal root OID**, **CDA MMID**, and **Default IDENT row root ID**.<br>• Select **No** if the HDR database is being upgraded from HDR 7.0.1 or 6.x. |
| Oracle Wallet Details | • Enter the Wallet user password, then reenter it to confirm.<br>• Enter the Wallet output folder location, for example, `/home/holuser/HDRB_MT_Home/Wallet`. |
| IHE Profile Options | To access the IHE web services, select **Yes**.<br>If you are upgrading from HDR 6.1.1 or 7.0.1, select **No**. |
| IHE Profile Option Detail | Enter values as follows:<br>• **Audit log server host**: The host name or IP address of the IHE Audit log server.<br>• **Audit log server port**: The TCP/UDP port of the IHE audit log server.<br>• **Server transfer protocol**: Select from the list.<br>• **Repository unique ID**: The HDR XDSb Respository unique ID.<br>• **Registry sync URL**: The SOAP web service URL of the IHE XDSb Registry server's register document set-b transaction.<br>• **Registry async URL**: The async SOAP web service URL of the IHE XDSb Registry server's register document set-b transaction.<br>For more information on IHE Profile options, see the *Oracle Healthcare Data Repository Implementation Guide*. |
| Summary | Review the changes, then click **Install**.<br>Any errors during the installation are logged in the files under the *<user_home_dir>*/oraInventory/logs folder. |
| End of Installation | Read the message and click **Done**. |

# Start WebLogic servers

1. Start the WebLogic hdr_domain admin server. For example:
   ```
   nohup ./startWebLogic.sh &
   ```

2. Start the hdr_server managed server. For example (all in one line):
   ```
   nohup ./startManagedWebLogic.sh <hdr-nodemanger>http://<ip-
   address>:<port -number> hdr_server.log &
   ```

# How to uninstall HDR (if required)

• "Uninstall database objects"

• "Uninstall the HDR middle tier "

# Uninstall database objects

To uninstall the database objects (HDR, ETS and HDR_CONFIG user schemas), execute the following SQL scripts from sqlplus/sqldeveloper as the sys user:

> **✎ Note:**
>
> If the tablespace names used during HDR install were different from the standard names then adjust the scripts accordingly.

```
alter session set "_oracle_script"=true; set serveroutput on; declare
    type tbs_name_type is table of varchar2(100);
    tbs_names tbs_name_type;
begin
    begin
        dbms_scheduler.drop_job('HDR_ETS_MAINTENANCE', TRUE);
    exception when others then
        dbms_output.put_line(SQLERRM);
    end;

    begin
        dbms_scheduler.drop_job('HDR_ETS_LOADER', TRUE);
    exception when others then
        dbms_output.put_line(SQLERRM);
    end;

    begin
        dbms_scheduler.drop_job('HDR_ETS_IMPORTER', TRUE);
    exception when others then
        dbms_output.put_line(SQLERRM);
    end;

    begin
dbms_scheduler.drop_program('hdr_ets_maintenance_program');
    exception when others then
        dbms_output.put_line(SQLERRM);
    end;

    begin
        dbms_scheduler.drop_program('hdr_ets_loader_program');
    exception when others then
        dbms_output.put_line(SQLERRM);
    end;

    begin
        dbms_scheduler.drop_program('hdr_ets_importer_program');
    exception when others then
        dbms_output.put_line(SQLERRM);
    end;

    select tablespace_name bulk collect into tbs_names from DBA_TS_QUOTAS where
username in ('HDR', 'ETS', 'HDR_CONFIG');

    begin
        execute immediate 'drop user HDR cascade';
    exception when others then
```

```
        dbms_output.put_line(SQLERRM);
    end;

    begin
        execute immediate 'drop user ETS cascade';
    exception when others then
        dbms_output.put_line(SQLERRM);
    end;

    begin
        execute immediate 'drop user HDR_CONFIG cascade';
    exception when others then
        dbms_output.put_line(SQLERRM);
    end;

    for i in 1..tbs_names.count loop
        begin
            execute immediate 'drop tablespace '||tbs_names(i)||'
including contents and datafiles';
        exception when others then
            dbms_output.put_line(SQLERRM);
        end;
    end loop;
end;
/
```

## Uninstall the HDR middle tier

To uninstall the HDR middle tier:

1. Stop the hdr_domain and hdr_server node mangers.

2. Delete the hdr_domain folder completely.

3. Edit WebLogic Oracle_Home/domain-registry.xml and remove the entry for hdr_domain.

4. Delete the HDR home directory (that you provided during installation) completely.

# 5

# Install HDR on a WebLogic Cluster

You can install HDR on a WebLogic cluster with preconfigured managed servers or a dynamic cluster where managed servers can be added on demand.

- Install HDR Using the HDR Installer
- Create and configure the HDR WebLogic machine and node manager
- Create and configure the HDR cluster
- Update the HDR deployment and dependent resources to target the HDR cluster
- Change client-side jndi.properties

## Install HDR Using the HDR Installer

Install HDR using the HDR installer, following the instructions given in Installing Oracle Healthcare Data Repository.

## Create and configure the HDR WebLogic machine and node manager

To create and configure WebLogic machine and node manager:

1. Log in to the WebLogic administration console.

2. In the left pane, click **hdr_domain** and expand **Environment**.

3. Click **Machines** and then click **Lock & Edit**.

4. Click **New**.
   The Create a New Machine screen is displayed.

**Figure 5-1    Create a New Machine**
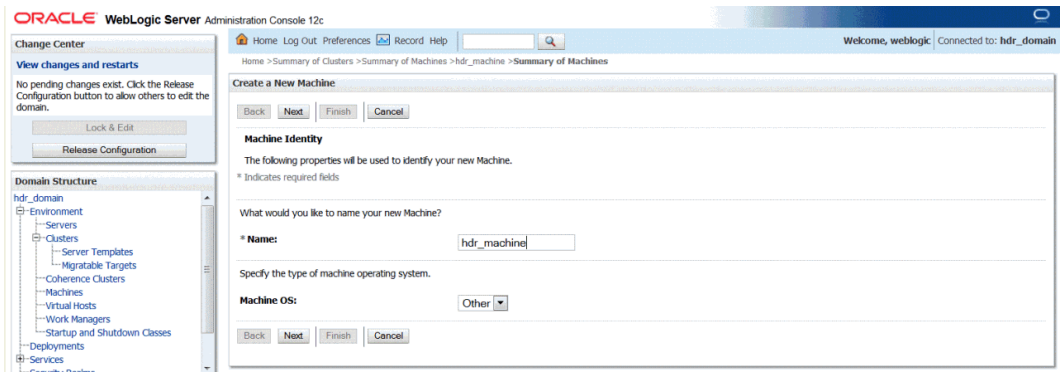
5. Enter `hdr_machine` (or any other preferred name) in the **Name** field and click **Next**.

6. Specify the node manager properties.
   Typically, node manager runs on localhost with SSL port 5556.

7. Click **Finish**.

# Create and configure the HDR cluster

To create and configure the HDR cluster:

1. Log in to the WebLogic administration console.

2. In the left pane, click **hdr_domain** and expand **Environment**.

3. Click **Clusters** and then click **Lock & Edit**.

4. Click **New** and then click **Dynamic Cluster**.
   The Create a New Dynamic Cluster screen is displayed.

**Figure 5-2 Create a New Dynamic Cluster**
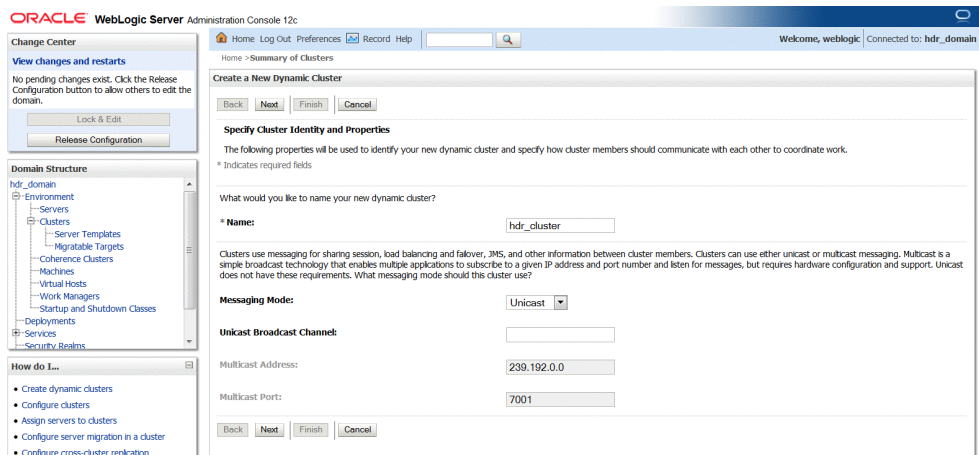


5. Enter `hdr_cluster` in the **Name** field.

6. Select **Unicast** from the **Messaging Mode** drop-down list.

7. Click **Next**.

**Figure 5-3    Specify Dynamic Server Properties**



8. Enter the number of dynamic servers required during the peak load in the **Number of Dynamic Servers** field.

> **Note:**
>
> The number of dynamic servers configured depends on the available system resources and the scalability requirements.

9. Enter a prefix for the dynamic server in the **Server Name Prefix** field. For example, `hdr_server-`.
   Based on the number of dynamic servers configured, the number of corresponding managed servers are created. For example, if you have configured three dynamic servers, three corresponding managed servers (hdr_server-1, hdr_server-2, and hdr_server-3) are created.

10. Select **Create a new server template using domain defaults**.
    The server template contains configurations such as Protocol, Services, SSL, Tuning, and so on, which are configured at each managed server level.

11. Click **Next**.

**Figure 5-4    Specify Machine Bindings**
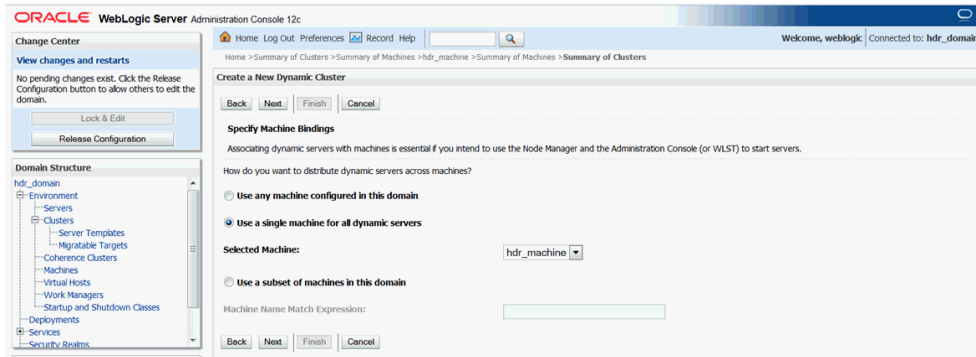


12. Select **Use a single machine for all dynamic servers**.

13. Select **hdr_machine** from the **Selected Machine** drop-down list and click **Next**.

**Figure 5-5    Specify Listen Port Bindings**
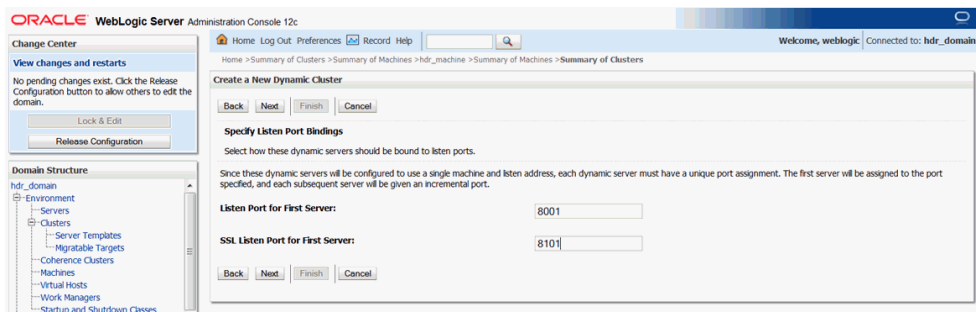


14. Enter a listen port in the **Listen Port for First Server** field. For example, `8001`. The first server is assigned to the port specified, and each subsequent server is assigned to an incremental port.

15. Enter an SSL listen port in the **SSL Listen Port for First Server** field.

16. Click **Finish**.

# Update the HDR deployment and dependent resources to target the HDR cluster

The existing hdr_server managed server created by the HDR installer can be targeted to hdr_cluster or can be deleted so that HDR can be deployed only on the dynamic cluster servers.

1. Stop the hdr_server managed server.

2. Delete HDR deployment.

3. Select the HDR EAR deployment from the `$HDRHome/hdr_exploded_app` folder and select deployment as application.

4. Target HDR deployment by selecting the **HDR_Cluster** and **All servers in the cluster** options.

**Figure 5-6    Install Application Assistant**
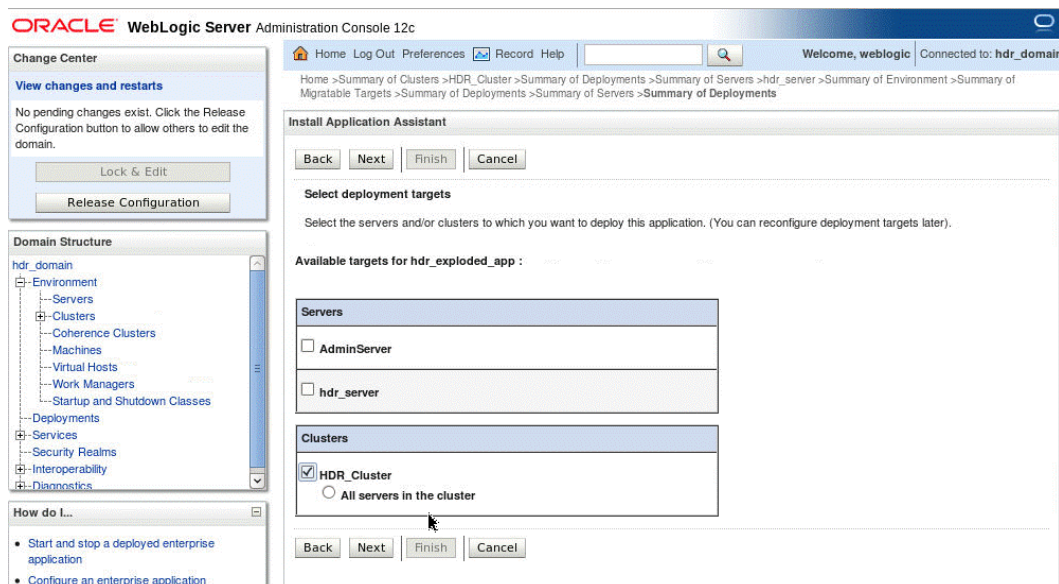


5. Provide appropriate name for HDR and click **Finish**.

6. Click **hdr_domain** and expand **Services**.
The Summary of JDBC Data Sources screen is displayed.

**Figure 5-7    Summary of JDBC Data Sources**
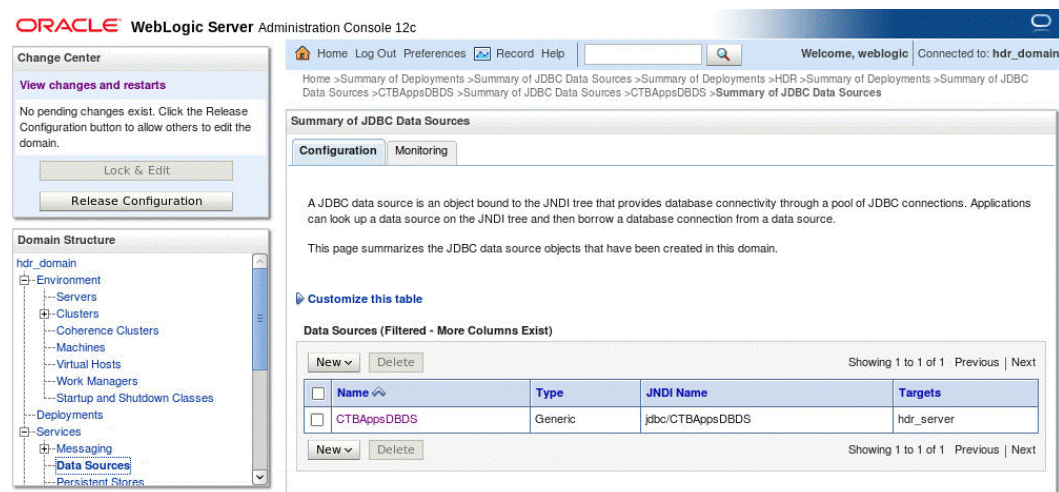


7. Click **Configuration** and then select **HdrDataSource**.

8. Change the targets of datasource HdrDataSource by selecting the **HDR_Cluster** and the **All servers in the cluster** options.

**Figure 5-8    Settings for HdrDataSource**
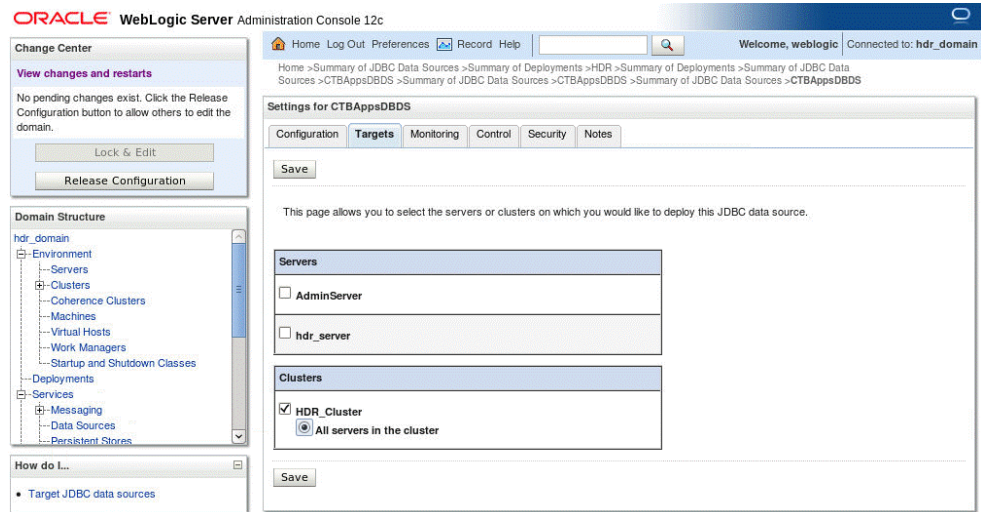


9. Click **Save**.

10. Click **hdr_domain** and expand **Services**.

11. Expand **Messaging** and select **JMS Modules** and note all the JMS queue names and jndi names in the JMS module HDRJMSSystemModule.

12. Recreate HDRConnectionFactory.

13. Delete all the JMS queues created in the JMS module HDRJMSSystemModule.

14. Recreate the queues again with same jndi names by selecting the **Distributed Queue** option.

**Figure 5-9    Create a New JMS System Module Resource**
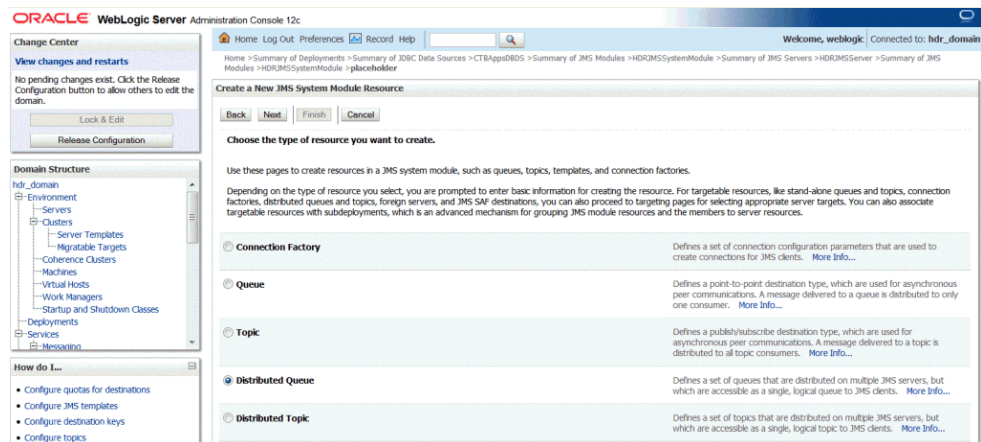


15. Change the targets of HDRJMSSystemModule by selecting the **HDR_Cluster** and the **All servers in the cluster** options.

16. Click **hdr_domain** and expand **Services**.

17. Expand **Messaging** and select **JMS Servers**.

18. Change the targets of HDRJMSServer by selecting the **HDR_Cluster** and **All servers in the cluster** options.

19. Copy the HDR-specific JVM arguments configured in startManagedWebLogic.sh to setDomainEnv.sh.

**Figure 5-10    startManagedWebLogic.sh**

```
        JAVA_OPTIONS="${JAVA_OPTIONS} -DLogFile=hdr.log -Djava.util.logging.config.file=logging.properties -Djava.security.au
th.login.config=/opt/oracle/oraem/Oracle/Middleware/Oracle_Home/user_projects/domains/hdr_domain/config/weblogic.security -Dt
angosol.coherence.mode=prod -DClientMode=local -Dweblogic.security.SSL.trustedCAKeyStore=/opt/oracle/oraem/Oracle/Middleware/
Oracle_Home/wlserver/server/lib/cacerts"
```

**Figure 5-11    setDomainEnv.sh**

```
# If you want to override the default Patch Classpath, Library Path and Path for this domain,
# Please uncomment the following lines and add a valid value for the environment variables
# set PATCH_CLASSPATH=[myPatchClasspath] (windows)
# set PATCH_LIBPATH=[myPatchLibpath] (windows)
# set PATCH_PATH=[myPatchPath] (windows)
# PATCH_CLASSPATH=[myPatchClasspath] (unix)
# PATCH_LIBPATH=[myPatchLibpath] (unix)
# PATCH_PATH=[myPatchPath] (unix)

. ${WL_HOME}/common/bin/commEnv.sh

if [ "${SERVER_NAME:0:10}" = "hdr_server" ] ; then

        JAVA_OPTIONS="${JAVA_OPTIONS} -DLogFile=${SERVER_NAME}_hdr.log -Djava.util.logging.config.file=logging.properties -Dj
ava.security.auth.login.config=/opt/oracle/oraem/Oracle/Middleware/Oracle_Home/user_projects/domains/hdr_domain/config/weblog
ic.security -Dtangosol.coherence.mode=prod -DClientMode=local -Dweblogic.security.SSL.trustedCAKeyStore=/opt/oracle/oraem/Ora
cle/Middleware/Oracle_Home/wlserver/server/lib/cacerts"

fi
```

20. Start the node manages using hdr_domain/bin/startNodeManager.sh.

21. Start the dynamic servers in hdr_cluster using the WebLogic admin console.

# Change client-side jndi.properties

To change client-side jndi.properties:

1. Update the HDR client application jndi.properties to use the new cluster provider URL.

2. List all host and port numbers for all managed servers in the cluster as a comma separated value.
   For example:

   ```
   java.naming.provider.url=t3://<host>:8001,<host>:8003,<host>:8003
   ```

# 6
# Upgrade to HDR 8.1

To upgrade:

1. Upgrade from Release 6.1.1 or 7.0.x to Release 8.1
2. Migrate Data (see Migrate Data )

> **Note:**
>
> The product name was changed from Oracle Healthcare Transactional Database (HTB) to Oracle Healthcare Data Repository (HDR) in Release 7.0.

## Upgrade from Release 6.1.1 or 7.0.x to Release 8.1

Use the following instructions to upgrade from 7.0.x to 8.1.

## Upgrade Oracle Database to Release 12.2.1.2.0 or 19.3

Follow instructions in the *Oracle Database Upgrade Guide* at `https://docs.oracle.com/database/122/UPGRD/toc.htm` or `https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/index.html`.

## Upgrade JDK

Upgrade to **JDK 8_171** or later. See Download and install Oracle Java Development Kit.

## Upgrade WebLogic Server to Release 12.2.1.3.0 or 12.2.1.4.0

Follow instructions in Fusion Middleware Upgrading Oracle WebLogic Server at `https://docs.oracle.com/middleware/12213/wls/WLUPG/toc.htm` or `https://docs.oracle.com/en/middleware/fusion-middleware/12.2.1.4/wlsig/index.htm`.

## Upgrade HDR to Release 8.1

Follow instructions in Install the HDR database and middle tier using the Oracle Universal Installer except choose to upgrade instead of install.

## Copy files from Release 6.x and 7.x locations to Release 8.x locations

1. Copy all the MTK custom schema/MIFs from your HTB 6.X or HDR 7.0.1 to your HDR 8.1 instance by running one of the following commands:
    - If you are upgrading from HTB 6.x:

```
copy $JAVA_TOP/oracle/apps/ctb/message/defs/customSchema/*
to $HDR_DOMAIN/config/hdr/message/defs/customSchema
```

- If you are upgrading from HDR 7.0.1:

```
copy $HDR_HOME/hdr_exploded_app/oracle/apps/ctb/message/defs/
customSchema/* to $HDR_DOMAIN/config/hdr/message/defs/customSchema
```

## Start WebLogic servers

1. Start the WebLogic hdr_domain admin server. For example:
   ```
   nohup ./startWebLogic.sh &
   ```

2. Start the hdr_server managed server. For example (all in one line):
   ```
   nohup ./startManagedWebLogic.sh <hdr-nodemanger>http://<ip-
   address>:<port -number> hdr_server.log &
   ```

## How to uninstall HDR (if required)

See How to uninstall HDR (if required).

# 7

# Migrate Data

The internal HDR data model is changed significantly in this release. To migrate your data, run scripts *in the order given below*. If you want to use any of the class subtypes supported in HDR 8.1, edit one of the migration scripts; see Categorize Observations, Medications, Supplies, and Control Acts (Optional).

- Prerequisites
- Extract the migration scripts
- Categorize Observations, Medications, Supplies, and Control Acts (Optional)
- To Migrate from HDR 7.x
- To Migrate from HDR 6.x

## Prerequisites

Software Requirements

- Oracle WebLogic Server 12c (12.2.1.2.0)
- Oracle Database 12c (12.2.0.1.0)
- JDK 8u121 and later

Operating System Requirements

- Oracle Linux 7(x64)

Installation Prerequisites

- Install JDK.
- Install Oracle WebLogic Server.
- Install Oracle Database.
- Set the JAVA_HOME environment variable.
- Install Cygwin 10.0 or above open source tool if you work on Windows.
- Download the `hdr-8.1.0-SNAPSHOT.zip` file and unzip it to the `<hdr-package-path>` directory. The topmost folder of this directory will be `hdr-8.1.0-SNAPSHOT`.
- Launch the Cygwin Terminal if you work on Windows.

## Extract the migration scripts

1. Copy hdr-8.1.0-SNAPSHOT.zip from the HDR software you downloaded to a working directory on the database server.
2. Unzip the file.

Migration scripts will prompt you to enter your credentials.

# Categorize Observations, Medications, Supplies, and Control Acts (Optional)

HDR 8.1 supports grouping data in certain general classes into subtypes, or categories. These categories follow the HL7 FHIR standard described here: https://www.hl7.org/fhir/codesystem-observation-category.html. You may need all, some, or none of these categories.

HDR 8.1 has a separate table to store data for each category. For more information about the tables, see the HDR *Technical Reference Manual* (eTRM) on the Oracle Help Center at https://docs.oracle.com/health-sciences/health-hdr-81/hdr-docs.htm.

To use categories, it must be possible to identify them in your HDR data. For example, to identify lab observations you may use ETS classifications or an attribute on the observation itself, like title.

You must edit the hdr8_acts_migration.sql migration script with appropriate filter conditions to separate the data into category tables.

1. In the tables below, find the table name that stores data for a category you want to use. See Observation categories, Medication (Substance Administration) categories, Supply categories, or Control Act categories.

2. Open hdr8_acts_migration.sql in a text editor.

3. Search for: `AND 1 = 0 --modify the logic`.

4. Replace `1 = 0 --modify the logic` with the filter condition and logic to write data to the appropriate table.
   For example, for extracting only laboratory observations into OHF_HDR_LAB_OBSEVN_RES table, customers who currently use ETS Classifications to main laboratory terminology codes can replace the condition 1=0 with the below condition with appropriate changes.

```
AND ACT_CODE_ETS_ID IN (
        SELECT CONCEPT_ID FROM HCT_ET_CLSSFCTN_DCLRNS
        WHERE CLASSIFICATION_ID = (
                SELECT CONCEPT_ID FROM HCT_ET_CONCEPTS_V
                WHERE CODINGSCHEME_NAME = 'ETSClassifications'
                    AND CONCEPT_CODE = 'LAB_RESULT'
            )
    );
```

Any data that is not categorized will be classified under the general category. Once the data is migrated to the category-specific tables, the RIM APIs support retrieval and persistence of data into the same tables using a special category attribute.

## Observation categories

There are Observation categories for mood codes EVN, RQO, and DEF.

**Table 7-1    Categories for mood code EVN**

| Category (Subtype) | Table name |
| --- | --- |
| General Observation Event | OHF_HDR_OBS_EVN_RES |

**Table 7-1    (Cont.) Categories for mood code EVN**

| Category (Subtype) | Table name |
| --- | --- |
| Social History | OHF_HDR_SOCHIST_OBSEVN_RES |
| Vital Sign | OHF_HDR_VITAL_OBSEVN_RES |
| Laboratory | OHF_HDR_LAB_OBSEVN_RES |
| Procedure | OHF_HDR_PROC_OBSEVN_RES |
| Survey | OHF_HDR_SURVEY_OBSEVN_RES |
| Exam | OHF_HDR_EXAM_OBSEVN_RES |
| Therapy | OHF_HDR_THERAPY_OBSEVN_RES |
| Allergy | OHF_HDR_ALRGY_OBSEVN_RES |
| Diagnostic Report | OHF_HDR_DIAGRPT_OBSEVN_RES |
| Risk Assessment | OHF_HDR_RISK_OBSEVN_RES |

**Table 7-2    Categories for mood code RQO**

| Category (Subtype) | Table name |
| --- | --- |
| General Observation RQO | OHF_HDR_OBS_RQO_RES |
| Diagnostic Order | OHF_HDR_DIAGORD_OBSRQO_RES |

**Table 7-3    Categories for mood code DEF**

| Category (Subtype) | Table name |
| --- | --- |
| General Observation Def | OHF_HDR_OBS_DEF_RES |
| Data Element | OHF_HDR_DATELE_OBSDEF_RES |

# Medication (Substance Administration) categories

**Table 7-4    Categories for mood code EVN**

| Category (Subtype) | Table name |
| --- | --- |
| General Medication Event | OHF_HDR_SBADM_EVN_RES |
| Medication Statement | OHF_HDR_MEDSTMNT_SBAEVN_RES |
| Immunization Recommendation | OHF_HDR_IMUNIZTN_SBAEVN_RES |

**Table 7-5    Categories for mood code RMD**

| Category (Subtype) | Table name |
| --- | --- |
| General Medication RMD | OHF_HDR_SBADM_RMD_RES |
| Immunization | OHF_HDR_IMUNRCMD_SBARMD_RES |

## Supply categories

**Table 7-6    Categories for mood code EVN**

| Category (Subtype) | Table name |
|---|---|
| General Supply Event | OHF_HDR_SPLY_EVN_RES |
| Medication Dispense | OHF_HDR_MEDDISP_SPLYEVN_RES |

**Table 7-7    Categories for mood code RQO**

| Category (Subtype) | Table name |
|---|---|
| General Supply RQO | OHF_HDR_SPLY_RQO_RES |
| Nutrition Order | OHF_HDR_NUTRORD_SPLYRQO_RES |

## Control Act categories

**Table 7-8    Categories for mood code EVN**

| Category (Subtype) | Table name |
|---|---|
| General Control Act | OHF_HDR_CACT_EVN_RES |
| Audit | OHF_HDR_AUDIT_CACTEVN_RES |
| Provenance | OHF_HDR_PROVENC_CACTEVN_RES |

# To Migrate from HDR 7.x

1. (Optional) Set up the HDR8 Database on a new Database machine
2. HCT/CTB to HDR_CONFIG migration (inter-schema)
3. HCT to ETS migration (inter-schema)
4. CTB to HDR migration (inter-schema)
5. Clean up old objects in the HCT and CTB schemas

## (Optional) Set up the HDR8 Database on a new Database machine

If you want to setup HDR 8.1 on a new Database machine, perform the following:

1. Make a backup of the existing HDR 7.x schemas for CTB and HCT, and export the CTB and HCT schemas to a dump. For example:
   ```
   expdp system/<password>@<SID> schemas=CTB,HCT
   exclude=GRANT,USER,STATISTICS,TABLESPACE_QUOTA,DEFAULT_ROLE
   directory=hdr_dmp_dir dumpfile=<dump file name>.dmp
   logfile=hdr_dump.log
   ```

2. Copy the dump files to the new database machine.

3. Create the following tablespaces on the HDR8 target database by connecting as the sys user:

   - Tablespace for the CTB user (CTB_TBS)

   - Tablespace for the HCT user (HCT_TBS)

   - Tablespace for Context (CTX_TBS)

   - Tablespace for Indexes (IDX_TBS)

   For example:

   ```
   Create TABLESPACE <tablespace name> LOGGING DATAFILE '<data file path>/
   hdr_hcttb.dbf' SIZE <100M> AUTOEXTEND ON NEXT <100M> MAXSIZE UNLIMITED
   EXTENT MANAGEMENT LOCAL;
   ```

4. Create the HCT and CTB schema users on the HDR8 target database by connecting as the sys user:

   - Create user HCT identified by hct default tablespace HCT_TBS quota unlimited on HCT_TBS:
     ```
     ALTER USER HCT QUOTA unlimited ON IDX_TBS; ALTER USER HCT QUOTA unlimited
     ON CTX_TBS; GRANT UNLIMITED TABLESPACE TO HCT; GRANT CREATE INDEXTYPE to
     HCT; GRANT CREATE PROCEDURE to HCT; GRANT CREATE SEQUENCE to HCT; GRANT
     CREATE SESSION to HCT; GRANT CREATE TABLE to HCT; GRANT CREATE TYPE to
     HCT; GRANT CREATE VIEW to HCT; GRANT ALTER SESSION to HCT; GRANT ANALYZE
     ANY to HCT; GRANT EXECUTE on CTXSYS.ctx_ddl to HCT; GRANT Create Any Job
     to HCT; GRANT Create External Job to HCT; GRANT Create Job to HCT; GRANT
     Execute Any Class to HCT; GRANT Execute Any Program to HCT; GRANT Manage
     Scheduler to HCT; GRANT CREATE ANY DIRECTORY TO HCT;
     ```

   - Create user CTB identified by ctb default tablespace CTB_TBS quota unlimited on CTB_TBS:
     ```
     ALTER USER CTB QUOTA unlimited ON IDX_TBS; GRANT UNLIMITED TABLESPACE TO
     CTB; GRANT CREATE INDEXTYPE to CTB; GRANT CREATE PROCEDURE to CTB; GRANT
     CREATE SEQUENCE to CTB; GRANT CREATE SESSION to CTB; GRANT CREATE TABLE
     to CTB; GRANT CREATE TYPE to CTB; GRANT CREATE VIEW to CTB; GRANT CREATE
     SYNONYM to CTB; GRANT ALTER SESSION to CTB; GRANT ANALYZE ANY to CTB;
     GRANT EXECUTE ON SYS.DBMS_AQIN TO CTB; GRANT EXECUTE ON SYS.DBMS_AQADM TO
     CTB; GRANT Create Any Job to CTB; GRANT Create External Job to CTB; GRANT
     Create Job to CTB; GRANT Execute Any Class to CTB; GRANT Execute Any
     Program to CTB; GRANT Manage Scheduler to CTB; GRANT create any directory
     to CTB; GRANT CREATE ANY DIRECTORY TO CTB;
     ```

5. Execute the following script (by connecting as sys user) to create the HTB_DUMP_DIR directory on the target HDR8.1 Oracle database where the HDR 7.x dump file is located:
   ```
   CREATE OR REPLACE DIRECTORY HDR_DMP_DIR as '<path>';
   ```

6. Import the HCT objects from HDR 7.x to the target HCT schema. For example, execute the following script to import the HCT objects:
   ```
   impdp system/<password>
   exclude=GRANT,USER,STATISTICS,TABLESPACE_QUOTA,DEFAULT_ROLE schemas=hct
   directory=HDR_DMP_DIR dumpfile=< HDR7.x dump file name >.dmp
   logfile=hdr_hct_import.log
   ```

7. Import the CTB objects from HDR7.x to the target CTB schema. For example, execute the following script to import the CTB objects:
   ```
   impdp system/<password>
   exclude=GRANT,USER,STATISTICS,TABLESPACE_QUOTA,DEFAULT_ROLE schemas=ctb
   ```

**ORACLE**®

```
directory=HDR_DMP_DIR dumpfile=< HDR7.x dump file name >.dmp
logfile=hdr_ctb_import.log
```

# HCT/CTB to HDR_CONFIG migration (inter-schema)

Preliminary setup:

- Set the **CONFIG_HOME** environment variable (for example: `<hdr-package-path>/hdr-8.1.0-SNAPSHOT/db`).

- Make sure that at least the tablespaces for HDR_CONFIG user are created. You can create these by running `$CONFIG_HOME/config/create_config_tablespace.sh`. For example:
  ```
  sh $CONFIG_HOME/config/create_config_tablespace.sh <ORACLE_HOME>
  <SYS_USER_NAME> <HOST_NAME> <SERVICE_NAME> <DB_PORT>
  <TABLE_SPACE_LOCATION>
  ```

  The script creates the HDR_CONFIG_TBS and HDR_CONFIG_IDX_TBS table spaces for the HDR_CONFIG user.

- (Optional) Create the HDR_CONFIG user by running `$CONFIG_HOME/config/create_config_user.sh`. For example:
  ```
  sh $CONFIG_HOME/config/create_config_user.sh <ORACLE_HOME>
  <SYS_USER_NAME> <HOST_NAME> <SERVICE_NAME> <DB_PORT>
  ```

  The user creation is also handled by the migrate_config.sh script as described below.

Run the scripts by following the steps below (note: if mentioned anywhere, *old* refers to HDR 7.x and *new* refers to HDR 8.1):

1. `cd $CONFIG_HOME/config/HDR7.x_to_HDR8.1`

2. Execute `migrate_config.sh`

3. At the "Have you already created schema/user HDR_CONFIG ? [Y/N] (default=N): " prompt in the terminal window:

   - If a user is already created for the HDR_CONFIG schema, enter Y and continue.

   - If a user is not created for the HDR_CONFIG schema, enter N and continue.

   - In either case, the password for the HDR_CONFIG user will be asked.

4. The console displays the default options set in the script (the password will always be asked by prompt). You can manually customize these parameters by editing `migrate_config.sh`.

5. At the "Use Default Migrate Config Database Options[Y/N] (default=Y): " prompt in the terminal window:

   - If you choose 'Y', the console will ask for the password of sysdba user. The default parameters will be used for running the script.

   - If you choose 'N', you need to enter the values by following the prompts on the terminal window.

Once the script runs successfully, the following changes will be made to support the HDR 8.1 code:

- User: If the option to create HDR_CONFIG user was selected, a new HDR_CONFIG user is created with the provided password.

- Sequences: *New* sequences will be created in the HDR_CONFIG schema and initialized to the last value of the old sequences (from both HCT and CTB schemas, as applicable).

- Tables: *New* tables will be created in the HDR_CONFIG schema by copying the data from *old* tables (from both HCT and CTB schemas, as applicable).

- Indexes: *New* indexes will be created in HDR_CONFIG schema.

- The HDR_CONFIG schema will be compiled to verify if the migration was successful.

All the logs will be found in the log directory

## HCT to ETS migration (inter-schema)

Preliminary setup:

- The `hdr-ets-8.1.0-SNAPSHOT.zip` file is in `<hdr-package-path>/hdr-8.1.0-SNAPSHOT/ets` directory.

- Unzip `hdr-ets-8.1.0-SNAPSHOT.zip` to your work folder `<hdr-package-path>/hdr-8.1.0-SNAPSHOT/ets/hdr-ets-8.1.0-SNAPSHOT`.

- Set the ETS_HOME environment variable. For example:
  `ETS_HOME = <hdr-package-path>/hdr-8.1.0-SNAPSHOT/ets/hdr-ets-8.1.0-SNAPSHOT`

- Make sure that the tablespaces for the ETS user are created. You can create these by running `$ETS_HOME/db/create_tablespace.sh`. For example:
  ```
  $ETS_HOME/db/create_tablespace.sh <ORACLE_HOME> <SYS_USER_NAME> <HOST_NAME>
  <SERVICE_NAME> <DB_PORT> <TABLE_SPACE_LOCATION>
  <ETS_DEFAULT_TABLESPACE_NAME> <ETS_CONTEXT_TABLESPACE_NAME>
  <ETS_INDEX_TABLESPACE_NAME>
  ```

  ```
  $ETS_HOME/db/create_tablespace.sh /scratch/u01/app/oracle/product/12201 sys
  localhost servicename 1521 /scratch/u01/app/oracle/oradata OHF_ETS_TBS
  OHF_ETS_CTX_TBS OHF_ETS_IDX_TBS
  ```

  In the above example, the script creates the OHF_ETS_TBS, OHF_ETS_CTX_TBS, and OHF_ETS_IDX_TBS table spaces for the ETS user.

- Create the ETS user by running `$ETS_HOME/db/create_ets_user.sh`. For example:
  ```
  sh $ETS_HOME/db/create_ets_user.sh <ORACLE_HOME> <SYS_USER_NAME> <HOST_NAME>
  <SERVICE_NAME> <DB_PORT> <ETS_DEFAULT_TABLESPACE_NAME>
  <ETS_CONTEXT_TABLESPACE_NAME> <ETS_INDEX_TABLESPACE_NAME>
  ```

  ```
  $ETS_HOME/db/create_ets_user.sh /scratch/u01/app/oracle/product/12201 sys
  localhost servicename 1521 OHF_ETS_TBS OHF_ETS_CTX_TBS OHF_ETS_IDX_TBS
  ```

Run the scripts by following the steps below (note: if mentioned anywhere, *old* refers to HDR 7.x and *new* refers to HDR 8.1):

1. `cd $ETS_HOME/db/HDR7.x_to_HDR8.1`

2. Execute `migrate_ets.sh`.

3. The console will display the default options set in the script (the password will always be asked by prompt). You can manually customize these parameters by editing `migrate_ets.sh`.

4. At the "Use Default HCT To ETS Migration Options[Y/N] (default=Y): " prompt in terminal window:

- • If you choose 'Y', the console will ask for the password of the sysdba user and ets schema user. Default parameters will be used for running the script.

- • If you choose 'N', you must enter the values by following the prompts on the terminal window.

Once the script runs successfully, the following changes will be made to support the HDR 8.1 code:

- • Sequences: *New* sequences will be created and initialized to the last values of the *old* sequences.

- • Tables: *New* tables will be created.

- • Types: *New* types will be created.

- • Packages/Package Bodies: *New* packages/package bodies will be created.

- • Views: *New* views will be created.

- • Indexes: *New* indexes will be created.

- • Synonyms: Synonyms of the required objects from the HDR_CONFIG schema will be created in the schema.

- • Jobs/Programs: *Old* jobs/programs will be dropped and *new* jobs/programs will be created.

- • The ETS schema will be compiled to verify if the migration was successful.

All the logs will be found in the log directory.

# CTB to HDR migration (inter-schema)

**Pre-migration scripts**

Before running the migration scripts, it is advisable to run the pre-migration scripts to check any data inconsistency or incorrectness. Such incorrect data shall not be handled by the migration scripts. You can correct the data based on the guidelines below.

1. Set environment variable HDR_HOME. For example:
   `<hdr-package-path>/hdr-8.1.0-SNAPSHOT/db`

2. `cd $HDR_HOME/hdr-core/HDR7.x_to_HDR8.1`

3. execute `pre_migrate_hdr.sh`

4. The console will display the default options set in the script (the password will always be asked by prompt). You can manually customize these parameters by editing `pre_migrate_hdr.sh`.

5. At the "Use Default Options[Y/N] (default=Y): " prompt in the terminal window:

   - • If you choose 'Y', the console will ask for the password of the CTB schema user. Default parameters will be used for running the script.

   - • If you choose 'N', you need to enter the values by following the prompts on the terminal window.

The script logs and cleans up "possibly incorrect data". Such incorrect data will be logged in the `log/*incorrect_data.log` files.

The the reasons for incorrect data are:

- entities_incorrect_data.log: NAME_PART_TYPE_CODE of the CTB_CORE_ENTY_NAME_PARTS is NULL

- roles_incorrect_data.log: NAME_PART_TYPE_CODE of the CTB_CORE_ROLE_NAME_PARTS is NULL

All the logs will be found in the log directory.

**Migration scripts**

Preliminary setup

- Set environment variable HDR_HOME. For example:
  ```
  <hdr-package-path>/hdr-8.1.0-SNAPSHOT/db
  ```

- Make sure that the tablespaces for the HDR user are created. You can create these by running `HDR_HOME/hdr-core/create_tablespace.sh`.

  1. If you want to create tablespaces per domain, run the script in the form:
     ```
     sh $HDR_HOME/hdr-core/create_tablespace.sh <ORACLE_HOME> <SYS_USER_NAME>
     <HOST_NAME> <SERVICE_NAME> <DB_PORT> <TABLE_SPACE_LOCATION>
     <NEED_TABLESPACE_PER_DOMAIN> <DEFINE_20_TABLESPACES_NAMES>.....
     ```

     For example:
     ```
     $HDR_HOME/hdr-core/create_tablespace.sh /scratch/u01/app/oracle/product/
     12201 sys localhost servicename 1521 /scratch/u01/app/oracle/oradata Y
     OHF_HDR_CLIN_CUR_TBS OHF_HDR_CLIN_HIS_TBS OHF_HDR_FIN_CUR_TBS
     OHF_HDR_FIN_HIS_TBS OHF_HDR_ADM_CUR_TBS OHF_HDR_ADM_HIS_TBS
     OHF_HDR_INF_CUR_TBS OHF_HDR_INF_HIS_TBS OHF_HDR_IDN_CUR_TBS
     OHF_HDR_IDN_HIS_TBS OHF_HDR_CLIN_CUR_IDX_TBS OHF_HDR_CLIN_HIS_IDX_TBS
     OHF_HDR_FIN_CUR_IDX_TBS OHF_HDR_FIN_HIS_IDX_TBS OHF_HDR_ADM_CUR_IDX_TBS
     OHF_HDR_ADM_HIS_IDX_TBS OHF_HDR_INF_CUR_IDX_TBS OHF_HDR_INF_HIS_IDX_TBS
     OHF_HDR_IDN_CUR_IDX_TBS OHF_HDR_IDN_HIS_IDX_TBS
     ```

  2. If you don't want to create tablespaces per domain, run the script in the form:
     ```
     sh $HDR_HOME/hdr-core/create_tablespace.sh <ORACLE_HOME> <SYS_USER_NAME>
     <HOST_NAME> <SERVICE_NAME> <DB_PORT> <TABLE_SPACE_LOCATION>
     <NEED_TABLESPACE_PER_DOMAIN> <CURRENT_DATA_TABLESPACE_NAME>
     <HISTORY_DATA_TABLESPACE_NAME> <CURRENT_INDEX_TABLESPACE_NAME>
     <HISTORY_INDEX_TABLESPACE_NAME>
     ```

     For example:
     ```
     $HDR_HOME/hdr-core/create_tablespace.sh /scratch/u01/app/oracle/product/
     12201 sys localhost servicename 1521 /scratch/u01/app/oracle/oradata N
     OHF_HDR_CUR_TBS OHF_HDR_HIS_TBS OHF_HDR_CUR_IDX_TBS OHF_HDR_HIS_IDX_TBS
     ```

- Create the HDR user by running `$HDR_HOME/hdr-core/create_hdr_user.sh`:
  ```
  sh $HDR_HOME/hdr-core/create_hdr_user.sh <ORACLE_HOME> <SYS_USER_NAME>
  <HOST_NAME> <SERVICE_NAME> <DB_PORT>
  ```

  For example:
  ```
  sh $HDR_HOME/hdr-core/create_hdr_user.sh /scratch/u01/app/oracle/product/
  12201 sys localhost servicename 1521
  ```

Follow the steps below to migrate the data from the HDR7 (CTB) schema to the HDR8 (HDR) schema:

1. `cd $HDR_HOME/hdr-core/HDR7.x_to_HDR8.1`

2. Execute `migrate_hdr.sh`.

3. At the "Are the tablespaces created per domain?[Y/N] (default=N):" console prompt:

   • Enter 'Y' if the tablespaces are available domain wise. Otherwise enter 'N'.

   • Based the response, the appropriate number of tablespace names will be prompted.

4. The console will display the default options set in the script (password will always be asked by prompt). You also can manually customize these parameters by editing `migrate_hdr.sh`.

5. At the "Use Default CTB To HDR Migration Options[Y/N] (default=Y): " prompt in the terminal window:

   • If you choose 'Y', the console will ask for the password of the sysdba user and HDR schema user. Default parameters will be used for running the script.

   • If you choose 'N', you need to enter the values by following the prompts on the terminal window.

Once the script runs successfully, the following changes will be made to support the HDR 8.1 code:

• Synonyms: Synonyms will be created in the HDR schema for the objects in the HCT/ETS schema and HDR_CONFIG schema.

• Sequences: New sequences will be created in the HDR schema.

• Types: New types will be created in the HDR schema.

• Tables and Indexes: Tables and indexes will be created in the HDR schema by pulling the data from the CTB schema. The tables will be created for acts, roles, entities, act relationships, and participation objects.

• Views: New views will be created in the HDR schema.

• The HDR schema will be compiled.

All the logs will be saved in the log directory. Any error data will be output in the log files, so that users can check and take appropriate action.

**Validate the migration scripts**

Once the migration script is run, you can verify if the migration was successful. This can be done by following below steps:

1. Set the `HDR_HOME` environment variable, for example:
   `<hdr-package-path>/hdr-8.1.0-SNAPSHOT/db`

2. `cd $HDR_HOME/hdr-core/HDR7.x_to_HDR8.1`

3. Execute `validate_migrate_hdr.sh`.

4. The console will display the default options set in the script (the password will always be asked by prompt). You can manually customize these parameters by editing `validate_migrate_hdr.sh`.

5. At the "Use Default Options[Y/N] (default=Y): " prompt in terminal window:

   • If you choose 'Y', the console will ask for the password of the HDR schema user. Default parameters will be used for running the script.

   • If you choose 'N', you need to enter the values by following the prompts on the terminal window.

After the script is run, the following will be validated:

- Count: The count of the total rows for each table in the HDR schema will be validated against the count of rows of corresponding table in CTB. Any issue will be logged to the `log/count_validation.log` file.

- Length: The length of some records in the HDR schema tables will be validated against the count of rows of corresponding table in CTB. Any issue will be logged to the `log/*_row_validation.log` files (where * can be act, entity, role, etc).

The consolidated log of above will be saved in `validate_migrate_to_hdr.sh.log`.

## Clean up old objects in the HCT and CTB schemas

To cleanup the HCT schema objects:

1. `cd $ETS_HOME/db/HDR7.x_to_HDR8.1`

2. Execute `clean_hct.sh`.

To drop the old CTB schema:

1. `cd $HDR_HOME/hdr-core/HDR7.x_to_HDR8.1`

2. Execute `drop_ctb.sh`.

# To Migrate from HDR 6.x

1. Upgrade the HTB 6.x Database Schema to HDR 8.1

2. HCT/CTB to HDR_CONFIG migration (inter-schema)

3. HCT to ETS migration (inter-schema)

4. CTB to HDR migration (inter-schema)

5. Clean up old objects in the HCT and CTB schemas

## Upgrade the HTB 6.x Database Schema to HDR 8.1

1. Make a backup of the existing HTB 6.x schemas for CTB and HCT. Export the following tables from the APPLSYS schema to a dump.

2. Create the following tablespaces on the HDR8.1 target database by connecting as the system user:

   - Tablespace for CTB user(CTB_TBS).

   - Tablespace for HCT user(HCT_TBS).

   - Tablespace for Context(CTX_TBS).

   - Tablespace for Indexes(IDX_TBS).

3. Create the HCT and CTB schemas users by executing the following script:

4. While still connected as a system user, execute the `CREATE OR REPLACE DIRECTORY HTB_DMP_DIR as '<path>'` script to create the `HTB_DUMP_DIR` directory on the Oracle database where the HTB 6.x dump file is located.

5. Import the HCT objects from HTB 6.x to the target HCT schema.

**ORACLE**®

> **Note:**
>
> Make sure that the appropriate tablespace names, dump path, and logfile path are mentioned in the script.

6. Import the CTB objects from HTB 6.X to the target CTB schema. Refer to the HCT migration script from above, and replace `HCT` with `CTB` where this applies.

7. Import the tables from APPLSYS from HTB 6.X to the HCT schema on the target database. Execute the following script to import the tables.

> **Note:**
>
> • Before importing the dumps to the target database, ensure that there is enough tablespace available on the target database.
>
> • Ignore the following errors while importing the dump:
>
> ```
> User or role <role/user> does not exist.
> Object type OBJECT_GRANT failed to create with error.
> The APPLSYS schema does not exist.
> ```

**Example 7-1    Execute a dump**

```
expdp system/<password>@<SID> schemas=CTB,HCT,APPLSYS
exclude=GRANT,USER,STATISTICS,TABLESPACE_QUOTA,DEFAULT_ROLE
directory=htb_dmp_dir dumpfile=<dump file name>.dmp logfile=htb_dump.log.
```

**Example 7-2    Script to create migration tablespaces**

```
CREATE TABLESPACE <tablespace name>

      LOGGING

      DATAFILE '<data file path>/hdr_hcttb.dbf'

      SIZE <100M>

      AUTOEXTEND ON

      NEXT <100M> MAXSIZE UNLIMITED

      EXTENT MANAGEMENT LOCAL;
```

**Example 7-3    Create HCT user on HCT_TBS**

```
      ALTER USER HCT QUOTA unlimited ON IDX_TBS;

      ALTER USER HCT QUOTA unlimited ON CTX_TBS;

      GRANT UNLIMITED TABLESPACE TO HCT;

      grant CREATE INDEXTYPE to HCT;

      grant CREATE PROCEDURE to HCT;
```

```
grant CREATE SEQUENCE to HCT;

grant CREATE SESSION to HCT;

grant CREATE TABLE to HCT;

grant CREATE TYPE to HCT;

grant CREATE VIEW to HCT;

grant ALTER SESSION to HCT;

grant ANALYZE ANY to HCT;

grant EXECUTE on CTXSYS.ctx_ddl to HCT;

grant Create Any Job to HCT;

grant Create External Job to HCT;

grant Create Job to HCT;

grant Execute Any Class to HCT;

grant Execute Any Program to HCT;

grant Manage Scheduler to HCT;
```

### Example 7-4    Create CTB user on CTB_TBS

```
ALTER USER CTB QUOTA unlimited ON IDX_TBS;

GRANT UNLIMITED TABLESPACE TO CTB;

 grant CREATE INDEXTYPE to CTB;

grant CREATE PROCEDURE to CTB;

grant CREATE SEQUENCE to CTB;

grant CREATE SESSION to CTB;

grant CREATE TABLE to CTB;

grant CREATE TYPE to CTB;

grant CREATE VIEW to CTB;

grant CREATE SYNONYM to CTB;

grant ALTER SESSION to CTB;

grant ANALYZE ANY to CTB;

GRANT EXECUTE ON SYS.DBMS_AQIN TO CTB;

GRANT EXECUTE ON SYS.DBMS_AQADM TO CTB;

grant Create Any Job to CTB;
```

```
grant Create External Job to CTB;

grant Create Job to CTB;

grant Execute Any Class to CTB;

grant Execute Any Program to CTB;

grant Manage Scheduler to CTB;

grant create any directory to CTB;

GRANT CREATE ANY DIRECTORY TO CTB;
```

**Example 7-5    Import HCT 6.x objects to 8.1 schema**

```
impdp system/<password> schemas=hct remap_schema=apps:hct remap_

schema=applsys:hct remap_tablespace=APPS_TS_SEED:hct_tbs

remap_tablespace=APPS_TS_TX_DATA:<hct_tbs>

remap_tablespace=APPS_TS_TX_IDX:<idx_tbs>

remap_tablespace=APPS_TS_NOLOGGING:<hct_tbs>

remap_tablespace=APPS_TS_MEDIA:<hct_tbs>

remap_tablespace=APPS_TS_INTERFACE:<hct_tbs>

remap_tablespace=APPS_TS_QUEUES:<hct_tbs>

remap_tablespace=APPS_TS_ARCHIVE:<hct_tbs>

remap_tablespace=APPS_TS_SUMMARY:<hct_tbs>

remap_tablespace=APPS_TS_TOOLS:<hct_tbs> directory=HTB_DMP_DIR

dumpfile=< HTB 6.X dump file name >.dmp logfile=htb_hct_import.log
```

**Example 7-6    Import tables from APPLSYS from HTB 6.X on the target database**

```
impdp system/<password>

tables=APPLSYS.FND_LANGUAGES,APPLSYS.FND_USER,APPLSYS.FND_

APPLICATION,APPLSYS.FND_PROFILE_OPTIONS,APPLSYS.FND_PROFILE_OPTIONS_

TL,APPLSYS.FND_PROFILE_OPTION_VALUES

remap_schema=apps:hct remap_schema=applsys:hct
remap_tablespace=APPS_TS_SEED:<hct_tbs>

remap_tablespace=APPS_TS_TX_DATA:<hct_tbs>

remap_tablespace=APPS_TS_TX_IDX:<idx_tbs>

remap_tablespace=APPS_TS_NOLOGGING:<hct_tbs>

remap_tablespace=APPS_TS_MEDIA:<hct_tbs>
```

```
remap_tablespace=APPS_TS_INTERFACE:<hct_tbs>

remap_tablespace=APPS_TS_QUEUES:<hct_tbs>

remap_tablespace=APPS_TS_ARCHIVE:<hct_tbs>

remap_tablespace=APPS_TS_SUMMARY:<hct_tbs>

remap_tablespace=APPS_TS_TOOLS:<hct_tbs> directory=HTB_DMP_DIR

dumpfile=<HTB 6.X dump file name>.dmp logfile=hct_fnd_import.log
```

# HCT/CTB to HDR_CONFIG migration (inter-schema)

Preliminary setup:

- Set the **CONFIG_HOME** environment variable (example: `<hdr-package-path>/hdr-8.1.0-SNAPSHOT/db`).

- Make sure that at least the tablespaces for HDR_CONFIG user are created. You can create these by running `$CONFIG_HOME/config/create_config_tablespace.sh`. For example:
  `sh $CONFIG_HOME/config/create_config_tablespace.sh <ORACLE_HOME> <SYS_USER_NAME> <HOST_NAME> <SERVICE_NAME> <DB_PORT> <TABLE_SPACE_LOCATION>`

  The script creates the HDR_CONFIG_TBS and HDR_CONFIG_IDX_TBS table spaces for the HDR_CONFIG user.

- (Optional) Create the HDR_CONFIG user by running `$CONFIG_HOME/config/create_config_user.sh`. For example:
  `sh $CONFIG_HOME/config/create_config_user.sh <ORACLE_HOME> <SYS_USER_NAME> <HOST_NAME> <SERVICE_NAME> <DB_PORT>`

  The user creation is also handled by the migrate_config.sh script as described below.

Run the scripts by following the steps below (note: if mentioned anywhere, *old* refers to HDR 6.x and *new* refers to HDR 8.1):

1. `cd $CONFIG_HOME/config/HDR6.x_to_HDR8.1`

2. Execute `migrate_config.sh`

3. At the "Have you already created schema/user HDR_CONFIG ? [Y/N] (default=N): " prompt in the terminal window:

   - If a user is already created for the HDR_CONFIG schema, enter Y and continue.

   - If a user is not created for the HDR_CONFIG schema, enter N and continue.

   - In either case, the password for the HDR_CONFIG user will be asked.

4. The console displays the default options set in the script (the password will always be asked by prompt). You can manually customize these parameters by editing `migrate_config.sh`.

5. At the "Use Default Migrate Config Database Options[Y/N] (default=Y): " prompt in the terminal window:

   - If you choose 'Y', the console will ask for the password of sysdba user. The default parameters will be used for running the script.

   - If you choose 'N', you need to enter the values by following the prompts on the terminal window.

Once the script runs successfully, the following changes will be made to support the HDR 8.1 code:

- User: If the option to create HDR_CONFIG user was selected, a new HDR_CONFIG user is created with the provided password.

- Sequences: *New* sequences will be created in the HDR_CONFIG schema and initialized to the last value of the old sequences (from both HCT and CTB schemas, as applicable).

- Tables: *New* tables will be created in the HDR_CONFIG schema by copying the data from *old* tables (from both HCT and CTB schemas, as applicable).

- Indexes: *New* indexes will be created in HDR_CONFIG schema.

- The HDR_CONFIG schema will be compiled to verify if the migration was successful.

All the logs will be found in the log directory

# HCT to ETS migration (inter-schema)

Preliminary setup:

- The `hdr-ets-8.1.0-SNAPSHOT.zip` file is in `<hdr-package-path>/hdr-8.1.0-SNAPSHOT/ets` directory.

- Unzip `hdr-ets-8.1.0-SNAPSHOT.zip` to your work folder `<hdr-package-path>/hdr-8.1.0-SNAPSHOT/ets/hdr-ets-8.1.0-SNAPSHOT`.

- Set the ETS_HOME environment variable. For example:
  ```
  ETS_HOME = <hdr-package-path>/hdr-8.1.0-SNAPSHOT/ets/hdr-ets-8.1.0-SNAPSHOT
  ```

- Make sure that the tablespaces for the ETS user are created. You can create these by running `$ETS_HOME/db/create_tablespace.sh`. For example:
  ```
  $ETS_HOME/db/create_tablespace.sh <ORACLE_HOME> <SYS_USER_NAME>
  <HOST_NAME> <SERVICE_NAME> <DB_PORT> <TABLE_SPACE_LOCATION>
  <ETS_DEFAULT_TABLESPACE_NAME> <ETS_CONTEXT_TABLESPACE_NAME>
  <ETS_INDEX_TABLESPACE_NAME>

  $ETS_HOME/db/create_tablespace.sh /scratch/u01/app/oracle/product/
  12201 sys localhost servicename 1521 /scratch/u01/app/oracle/oradata
  OHF_ETS_TBS OHF_ETS_CTX_TBS OHF_ETS_IDX_TBS
  ```

  In the above example, the script creates the OHF_ETS_TBS, OHF_ETS_CTX_TBS, and OHF_ETS_IDX_TBS table spaces for the ETS user.

- Create the ETS user by running `$ETS_HOME/db/create_ets_user.sh`. For example:
  ```
  sh $ETS_HOME/db/create_ets_user.sh <ORACLE_HOME> <SYS_USER_NAME>
  <HOST_NAME> <SERVICE_NAME> <DB_PORT> <ETS_DEFAULT_TABLESPACE_NAME>
  <ETS_CONTEXT_TABLESPACE_NAME> <ETS_INDEX_TABLESPACE_NAME>

  $ETS_HOME/db/create_ets_user.sh /scratch/u01/app/oracle/product/12201
  sys localhost servicename 1521 OHF_ETS_TBS OHF_ETS_CTX_TBS
  OHF_ETS_IDX_TBS
  ```

Run the scripts by following the steps below (note: if mentioned anywhere, *old* refers to HDR 6.x and *new* refers to HDR 8.1):

1. `cd $ETS_HOME/db/HDR6.x_to_HDR8.1`

2. Execute `migrate_ets.sh`.

3. The console will display the default options set in the script (the password will always be asked by prompt). You can manually customize these parameters by editing `migrate_ets.sh`.

4. At the "Use Default HCT To ETS Migration Options[Y/N] (default=Y): " prompt in terminal window:

    - If you choose 'Y', the console will ask for the password of the sysdba user and ets schema user. Default parameters will be used for running the script.

    - If you choose 'N', you must enter the values by following the prompts on the terminal window.

Once the script runs successfully, the following changes will be made to support the HDR 8.1 code:

- Sequences: *New* sequences will be created and initialized to the last values of the *old* sequences.

- Tables: *New* tables will be created.

- Types: *New* types will be created.

- Packages/Package Bodies: *New* packages/package bodies will be created.

- Views: *New* views will be created.

- Indexes: *New* indexes will be created.

- Synonyms: Synonyms of the required objects from the HDR_CONFIG schema will be created in the schema.

- Jobs/Programs: *Old* jobs/programs will be dropped and *new* jobs/programs will be created.

- The ETS schema will be compiled to verify if the migration was successful.

All the logs will be found in the log directory.

# CTB to HDR migration (inter-schema)

**Pre-migration scripts**

Before running the migration scripts, it is advisable to run the pre-migration scripts to check any data inconsistency or incorrectness. Such incorrect data shall not be handled by the migration scripts. You can correct the data based on the guidelines below.

1. Set the `HDR_HOME` environment variable. For example:
   `<hdr-package-path>/hdr-8.1.0-SNAPSHOT/db`

2. `cd $HDR_HOME/hdr-core/HDR6.x_to_HDR8.1`

3. execute `pre_migrate_hdr.sh`

4. The console will display the default options set in the script (the password will always be asked by prompt). You can manually customize these parameters by editing `pre_migrate_hdr.sh`.

5. At the "Use Default Options[Y/N] (default=Y): " prompt in the terminal window:

    - If you choose 'Y', the console will ask for the password of the CTB schema user. Default parameters will be used for running the script.

- If you choose 'N', you need to enter the values by following the prompts on the terminal window.

The script logs and cleans up "possibly incorrect data". Such incorrect data will be logged in the `log/*incorrect_data.log` files.

The the reasons for incorrect data are:

- entities_incorrect_data.log: NAME_PART_TYPE_CODE of the CTB_CORE_ENTY_NAME_PARTS is NULL

- roles_incorrect_data.log: NAME_PART_TYPE_CODE of the CTB_CORE_ROLE_NAME_PARTS is NULL

All the logs will be found in the log directory.

**Migration scripts**

Preliminary setup

- Set environment variable HDR_HOME. For example:
  ```
  <hdr-package-path>/hdr-8.1.0-SNAPSHOT/db
  ```

- Make sure that the tablespaces for the HDR user are created. You can create these by running `HDR_HOME/hdr-core/create_tablespace.sh`.

  1. If you want to create tablespaces per domain, run the script in the form:
     ```
     sh $HDR_HOME/hdr-core/create_tablespace.sh <ORACLE_HOME>
     <SYS_USER_NAME> <HOST_NAME> <SERVICE_NAME> <DB_PORT>
     <TABLE_SPACE_LOCATION> <NEED_TABLESPACE_PER_DOMAIN>
     <DEFINE_20_TABLESPACES_NAMES>.....
     ```

     For example:

     ```
     $HDR_HOME/hdr-core/create_tablespace.sh /scratch/u01/app/oracle/
     product/12201 sys localhost servicename 1521 /scratch/u01/app/
     oracle/oradata Y OHF_HDR_CLIN_CUR_TBS OHF_HDR_CLIN_HIS_TBS
     OHF_HDR_FIN_CUR_TBS OHF_HDR_FIN_HIS_TBS OHF_HDR_ADM_CUR_TBS
     OHF_HDR_ADM_HIS_TBS OHF_HDR_INF_CUR_TBS OHF_HDR_INF_HIS_TBS
     OHF_HDR_IDN_CUR_TBS OHF_HDR_IDN_HIS_TBS OHF_HDR_CLIN_CUR_IDX_TBS
     OHF_HDR_CLIN_HIS_IDX_TBS OHF_HDR_FIN_CUR_IDX_TBS
     OHF_HDR_FIN_HIS_IDX_TBS OHF_HDR_ADM_CUR_IDX_TBS
     OHF_HDR_ADM_HIS_IDX_TBS OHF_HDR_INF_CUR_IDX_TBS
     OHF_HDR_INF_HIS_IDX_TBS OHF_HDR_IDN_CUR_IDX_TBS
     OHF_HDR_IDN_HIS_IDX_TBS
     ```

  2. If you don't want to create tablespaces per domain, run the script in the form:
     ```
     sh $HDR_HOME/hdr-core/create_tablespace.sh <ORACLE_HOME>
     <SYS_USER_NAME> <HOST_NAME> <SERVICE_NAME> <DB_PORT>
     <TABLE_SPACE_LOCATION> <NEED_TABLESPACE_PER_DOMAIN>
     <CURRENT_DATA_TABLESPACE_NAME> <HISTORY_DATA_TABLESPACE_NAME>
     <CURRENT_INDEX_TABLESPACE_NAME> <HISTORY_INDEX_TABLESPACE_NAME>
     ```

     For example:

     ```
     $HDR_HOME/hdr-core/create_tablespace.sh /scratch/u01/app/oracle/
     product/12201 sys localhost servicename 1521 /scratch/u01/app/
     oracle/oradata N OHF_HDR_CUR_TBS OHF_HDR_HIS_TBS
     OHF_HDR_CUR_IDX_TBS OHF_HDR_HIS_IDX_TBS
     ```

- **Create the HDR user by running** `$HDR_HOME/hdr-core/create_hdr_user.sh`:

```
sh $HDR_HOME/hdr-core/create_hdr_user.sh <ORACLE_HOME> <SYS_USER_NAME>
<HOST_NAME> <SERVICE_NAME> <DB_PORT>
```

For example:

```
sh $HDR_HOME/hdr-core/create_hdr_user.sh /scratch/u01/app/oracle/product/
12201 sys localhost servicename 1521
```

Follow the steps below to migrate the data from the HDR6 (CTB) schema to the HDR8 (HDR) schema:

1. `cd $HDR_HOME/hdr-core/HDR6.x_to_HDR8.1`

2. Execute `migrate_hdr.sh`.

3. At the "Are the tablespaces created per domain?[Y/N] (default=N):" console prompt:

   • Enter 'Y' if the tablespaces are available domain wise. Otherwise enter 'N'.

   • Based the response, the appropriate number of tablespace names will be prompted.

4. The console will display the default options set in the script (password will always be asked by prompt). You also can manually customize these parameters by editing `migrate_hdr.sh`.

5. At the "Use Default CTB To HDR Migration Options[Y/N] (default=Y): " prompt in the terminal window:

   • If you choose 'Y', the console will ask for the password of the sysdba user and HDR schema user. Default parameters will be used for running the script.

   • If you choose 'N', you need to enter the values by following the prompts on the terminal window.

Once the script runs successfully, the following changes will be made to support the HDR 8.1 code:

• Synonyms: Synonyms will be created in the HDR schema for the objects in the HCT/ETS schema and HDR_CONFIG schema.

• Sequences: New sequences will be created in the HDR schema.

• Types: New types will be created in the HDR schema.

• Tables and Indexes: Tables and indexes will be created in the HDR schema by pulling the data from the CTB schema. The tables will be created for acts, roles, entities, act relationships, and participation objects.

• Views: New views will be created in the HDR schema.

• The HDR schema will be compiled.

All the logs will be saved in the log directory. Any error data will be output in the log files, so that users can check and take appropriate action.

**Validate the migration scripts**

Once the migration script is run, you can verify if the migration was successful. This can be done by following below steps:

1. Set the `HDR_HOME` environment variable. For example:
   `<hdr-package-path>/hdr-8.1.0-SNAPSHOT/db`

2. `cd $HDR_HOME/hdr-core/HDR6.x_to_HDR8.1`

3. Execute `validate_migrate_hdr.sh`.

4. The console will display the default options set in the script (the password will always be asked by prompt). You can manually customize these parameters by editing `validate_migrate_hdr.sh`.

5. At the "Use Default Options[Y/N] (default=Y): " prompt in terminal window:

   - If you choose 'Y', the console will ask for the password of the HDR schema user. Default parameters will be used for running the script.

   - If you choose 'N', you need to enter the values by following the prompts on the terminal window.

After the script is run, the following will be validated:

- Count: The count of the total rows for each table in the HDR schema will be validated against the count of rows of corresponding table in CTB. Any issue will be logged to the `log/count_validation.log` file.

- Length: The length of some records in the HDR schema tables will be validated against the count of rows of corresponding table in CTB. Any issue will be logged to the `log/*_row_validation.log` files (where * can be act, entity, role, etc).

The consolidated log of above will be saved in `validate_migrate_to_hdr.sh.log`.

# Clean up old objects in the HCT and CTB schemas

To cleanup the HCT schema objects:

1. `cd $ETS_HOME/db/HDR6.x_to_HDR8.1`

2. Execute `clean_hct.sh`.

To drop the old CTB schema:

1. `cd $HDR_HOME/hdr-core/HDR6.x_to_HDR8.1`

2. Execute `drop_ctb.sh`.